IBM System x3850 X5

Using

DB2 9.7 Enterprise Edition

and

SUSE Enterprise Linux Server 11 SP1

TPC Benchmark™ C

Full Disclosure Report

Second Edition, Nov. 16, 2010

Second Edition: Nov. 16, 2010

The information contained in this document is distributed on an AS IS basis without any warranty either expressed or implied. The use of this information or the implementation of any of these techniques is a customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item has been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environment do so at their own risk.

In this document, any references made to an IBM licensed program are not intended to state or imply that only IBM's licensed program may be used; any functionally equivalent program may be used.

It is possible that this material may contain references to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such products, programming, or services in your country.

All performance data contained in this publication was obtained in a controlled environment, and therefore the results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data in their specific environment.

Request for additional copies of this document should be sent to the following address:

TPC Benchmark Administrator
IBM Commercial Performance
Mail Stop 9571
11501 Burnet Road
Austin, TX 78758
FAX Number (512) 838-1852

© Copyright International Business Machines Corporation, 2011 All rights reserved.

Permission is hereby granted to reproduce this document in whole or in part, provided the copyright notice printed above is set forth in full text on the title page of each item reproduced.

NOTE: US. Government Users - Documentation related to restricted rights: Use, duplication, or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Trademarks

The following terms used in this publication are trademarks or registered trademarks of **International Business Machines** Corporation in the United States and/or other countries: IBM, System x, System Storage, ServeRAID and DB2.

The following terms used in this publication are trademarks of other companies as follows:

TPC Benchmark, TPC-C, and tpmC are trademarks of the Transaction Processing Performance Council.

Microsoft Windows 2008 server and COM+ are registered trademarks of Microsoft Corporation.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

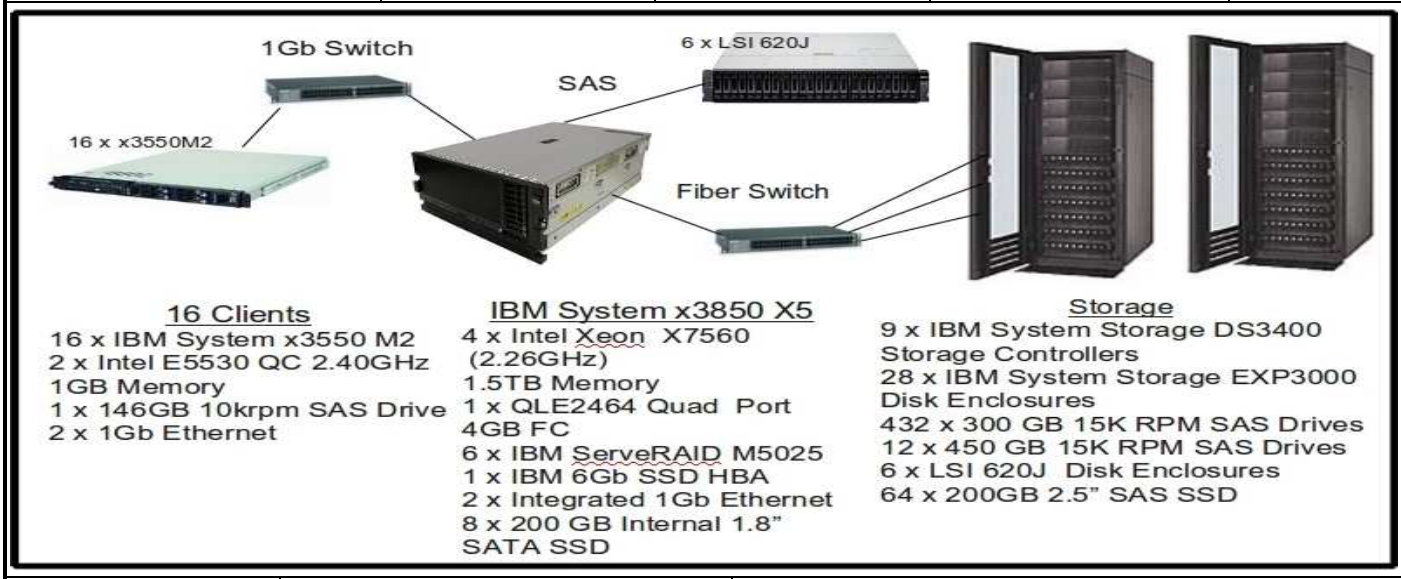
Novell and SUSE are registered trademarks of Novell, Inc.

Intel and Xeon are registered trademarks of Intel Corporation.

	IBM System x3850 X5 DB2® 9.7	TPC-C Rev. 5.11 Report Date: Nov. 16, 2010 Revised May 16, 2011

Total System Cost	TPC-C Throughput	Price/Performance	Availability Date
$1,393,681 USD	**2,308,099**	**$0.60 USD**	**May 16, 2011**

Database Processors/Cores/Threads	Database Manager	Operating System	Other Software	No. Users
4/32/64 2.26 GHz Intel Xeon X7560	DB2 9.7	SLES 11 SP1	Microsoft Visual C++ Microsoft COM+	1,839,360



System Components	Each of the 16 Clients		Server	
	Quantity	Description	Quantity	Description
Processors/Cores/ Threads	2/8/8	2.4GHz Xeon E5530 4 x 256KB L2, 1 x 8MB L2	4/32/64	2.26 GHz Intel Xeon X7560 8 x 256KB L2 1 x 24MB L3
Memory	4	2 GB	96	16 GB
Disk Controllers	1	SAS	1 6 1 1 9	Integrated SAS ServeRAID-M5025 IBM 6Gb SSD HBA 4Gb quad-port FC Adapters IBM DS3400 Controllers for Data
Disk Drives	1	146 GB	432 12 8 64	300 GB 15K RPM SAS 450 GB 15K RPM SAS 1.8" 200GB SATA SSD drives 2.5" 200GB SAS SSD drives
Total Storage		584 GB		136 TB
Terminals	1	System Console	1	System Console

IBM Corporation	IBM x3850 X5	with DB2 9.7	TPC-C Revision 5.11 Report Date: Nov. 16, 2010; Revised May 16, 2011

Description	Part Number	Brand	Price Source	Unit Price	Qty	Extended Price	3-Yr. Maint. Price
Server Hardware							
IBM System x3850 X5/x3950 X5	7145AC1	IBM					
IBM System x3850 X5/x3950 X5 Base	7626	IBM	1	2,079	1	2,079	
Intel Xeon Processor X7560 8c 2.26GHz 24MB Cache 130w	4512	IBM	1	4,999	1	4,999	
Addl. Intel Xeon Processor X7560 8c 2.26GHz 24MB Cache 130w	4513	IBM	1	4,999	3	14,997	
IBM x3850 X5 and x3950 X5 Memory Expansion Card	5102	IBM	1	299	8	2,392	
x3850 X5 System Planar Boards	5695	IBM	1	3,259	1	3,259	
16GB (1x16GB, 4Rx4, 1.5V) PC3-8500 CL7 ECC DDR3 1066MHz LP RDIMM	1707	IBM	1	1,109	64	70,976	
IBM MAX5 for System x	4199	IBM	1	7,495	1	7,495	
IBM MAX5 to x3850 X5 Cable Kit	4192	IBM	1	1,795	1	1,795	
16GB MAX5 1x16G Qrx4 PC3-8500 CL7 ECC DDR3 1066MHz LP RDIMM	2433	IBM	1	1,549	32	49,568	
IBM 6Gb SSD HBA	3876	IBM	1	699	1	699	
IBM eXFlash 8x 1.8" HS SAS SSD Backplane	4191	IBM	1	195	1	195	
IBM 1975W Power Supply	2111	IBM	1	699	2	1,398	
IBM 675W HE Redundant Power Supply	4782	IBM	1	299	1	299	
2.8m, 10A/100-250V, C13 to IEC 320-C14 Rack Power Cable	6311	IBM	1	10	4	40	
QPI Wrap Cord	5104	IBM	1	250	1	250	
Acer A181HLB 18.5" Wide LED Monitor	2146072	CDW	4	98	1	98	
IBM Preferred Pro Keyboard USB – US English 103P	40K9584	IBM	1	29	1	29	
ServeRAID M5025 SAS/SATA Controller	46M0830	IBM	1	649	6	3,894	
ServeRAID M5000 Series Performance Accelerator Key	81Y4426	IBM	1	399	6	2,394	
Qlogic 4Gb Fibre Channel card (+2 spares)	QLE2464	ALHDD	7	523	3	1,569	
3 Year Onsite Repair 24x7 4 Hour Response	67567CG	IBM	1		1		1,250
					Subtotal	168,425	1,250
Server Storage							
IBM System Storage DS3400 FC Dual Controller	1726-42X	IBM	1	9,295	9	83,655	
IBM 1M SAS cable	39R6529	IBM	1	119	62	7,378	
IBM System Storage EXP3000	1727-01X	IBM	1	3,199	28	89,572	
IBM EXP3000 Environmental Services Module (ESM)	39R6515	IBM	1	999	28	27,972	
IBM 300GB 15K 6Gbps SAS 3.5" Hot-Swap HDD	44W2234	IBM	1-s	529	432	228,528	
450GB 15K 3.5" Hot-Swap SAS	44W2239	IBM	1	709	12	8,508	
NetBAY S2 42U Standard Rack	93074RX	IBM	1	1,609	3	4,827	
IBM 200GB High IOPS 1.8" SATA SSD	43W7746	IBM	1	3,199	8	25,592	
LSI 620J JBOD Disk Enclosure (Incl 2 Spares)	LSI00217	LSI	9	2,041	8	16,328	
Smart Modular XceedIOPS 2.5" 200GB eMLC SAS SSD (Incl 7 Spares)	SG9XCA2E200GE10	Smart	8	1,800	71	127,800	
IBM System Storage SAN24B-4 Express	249824E	IBM	1	4,420	1	4,420	
4 pack SW Transcievers for 2109 models F08 or F16	45W0493	IBM	1	150	20	3,000	
8 Port Activation	45W0502	IBM	1	3,470	2	6,940	
ServicePac for 3-Year 24x7x4 Support (DS3400)	44J8073	IBM	1	1,300	9		11,700
ServicePac for 3-Year 24x7x4 Support (EXP3000)	41L2768	IBM	1	760	28		21,280
ServicePac for 3-Year 24x7x4 Support (Rack)	41L2760	IBM	1	300	3		900
					Subtotal	634,520	33,880
Server Software							
DB2 ESE 9.7 SW License and Maintenance 12 Months		IBM	1	405.00	3200	1,296,000	
DB2 ESE 9.7 SW Maintenance Renewal - 1 Year		IBM	1	81.00	6400		518,400
SUSE Linux Enterprise Server 11 SP1 for x86_64 1 Year Priority Support		Novell	3	1,499	3	4,497	
					Subtotal	1,300,497	518,400
Client Hardware							
IBM System x3550 M2 Base	7946AC1 5600	IBM	1	289	16	4,624	
Intel Xeon Processor E5530 4C (2.4GHz 8MB L3 Cache 1066MHz 80w)	3640	IBM	1	859	16	13,744	
Addl. Intel Xeon Processor E5530 4C (2.4GHz 8MB L3 Cache 1066MHz 80w)	4412	IBM	1	849	16	13,584	
2GB DDR3-1333 1Rx4 LP RDIMM x 4	3965	IBM	1	436	64	27,904	
IBM 146GB 10k 6Gbps SAS 2.5" SFF Slim-HS HDD	5537	IBM	1	259	16	4,144	
System Common planar for 1U/2U	5653	IBM	1	749	16	11,984	
PCI-Express (1x16 slot; LP) Riser Card 1	5076	IBM	1	59	16	944	
UltraSlim Enhanced SATA CD-RW / DVD-ROM Combo	4162	IBM	1	129	16	2,064	
PCI-Express (1x16 slot; FH/HL) Riser Card 2	4375	IBM	1	59	16	944	
ServeRAID-BR10i SAS/SATA Controller	3577	IBM	1	299	16	4,784	
2.8m, 10A/100-250V, C13 to IEC 320-C14 Rack Power Cable	6311	IBM	1	10	16	160	
3 Year Onsite Repair 24x7 4 Hour Response	6756298	IBM	1	450	16		7,200
					Subtotal	84,880	7,200
Client Software							
Windows Server Web Edition 2008 R2	LWA-00984	Microsoft	5	469	16	7,504	
Visual Studio Standard 2008	127-00166	Microsoft	5	799	1	799	
Microsoft Problem Resolution Services	NA	Microsoft	5	259	1		259
					Subtotal	8,303	259
Third-Party Components							
IOGEAR 2 port USB KVM Switch (2 spares)	575404	cdw	4	35	3	105	
UPS - powercom KIN-1500AP 1500VA 900Watts (2 spares)	N82E16842106115	newegg	6	135	3	405	
NETGEAR GS748T 48-port Gigabit Smart Switch (2 spares)	GS748TNA	cdw	4	660	5	3,300	
					Subtotal	3,810	0
					Total	2,200,435	560,989
IBM Dollar Volume Discount (See Note 1)			1			Note 1 Discount	1,367,743
						Three-Year Cost of Ownership USD:	1,393,681
						tpmC:	2,308,099
						$ USD/tpmC:	0.60

Pricing: 1 - IBM - 1-800-656-0833 x35304; 2 - Qlogic; 3 - Novell; 4 - CDW.com; 5 – Micosoft
6 - newegg.com; 7 – allhdd.com; 8 – ClearComm 919-859-4400; 9 – Avnet 1-800-332-8638
Note 1: Discount based on IBM Direct guidance applies to all line items where Pricing=1.
Pricing is for this system and software or one of similar size.
S - One or more components of the measured configuration have been substituted in the Priced Configuration. See the FDR for details
* These components are not immediately orderable. See the FDR for more information.

Audited by Francois Raab, InfoSizing, Inc. (www.sizing.com)

Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts reflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark specifications. If you find that stated prices are not available according to these terms, please inform the TPC at pricing @ tpc.org. Thank you.

Numerical Quantities Summary for the IBM System x3850 X5

MQTH, computed Maximum Qualified Throughput: 2,308,099 tpmC

Response Times (in seconds)	90th %	Average	Maximum
New Order	0.660	0.301	13.797
Payment	0.660	0.291	10.812
Order-Status	0.660	0.306	7.734
Delivery (interactive)	0.340	0.165	9.328
Delivery (deferred)	0.32	0.13	2.36
Stock-Level	0.660	0.298	13.297
Menu	0.344	0.167	13.156

Response time delay added for emulated components was 0.1 seconds

Transaction Mix, in percent of total transactions	Percent
New Order	44.950%
Payment	43.021%
Order-Status	4.010%
Delivery	4.008%
Stock-Level	4.010%

Keying/Think Times (in seconds)	Min.	Average	Max.
New Order	18.015/0.00	18.015/12.035	18.047/120.313
Payment	3.015/0.00	3.015/12.036	3.047/120.328
Order-Status	2.015/0.00	2.015/10.036	2.047/100.328
Delivery	2.015/0.00	2.015/5.035	2.047/50.313
Stock-Level	2.015/0.00	2.015/5.035	2.047/50.313

Test Duration

Ramp-up Time	46 minutes
Measurement interval	2 hours 0 minutes
Transactions during measurement interval (all types)	616,176,706

Checkpoints

Number of checkpoints	N/A
Checkpoint interval	N/A

Table of Contents

Preface 9
0 General Items 10
0.1. Application Code Disclosure 10
0.2. Benchmark Sponsor 10
0.3. Parameter Settings 10
0.4. Configuration Diagrams 10
1 Clause 1: Logical Data Base Design Related Items 12
1.1. Table Definitions 12
1.2. Database Organization 12
1.3. Insert and/or Delete Operations 12
1.4. Horizontal or Vertical Partitioning 12
2 Clause 2: Transaction & Terminal Profiles Related Items 13
2.1. Verification for the Random Number Generator 13
2.2. Input/Output Screens 13
2.3. Priced Terminal Features 13
2.4. Presentation Managers 13
2.5. Home and Remote Order-lines 13
2.6. New-Order Rollback Transactions 13
2.7. Number of Items per Order 13
2.8. Home and Remote Payment Transactions 13
2.9. Non-Primary Key Transactions 14
2.10. Skipped Delivery Transactions 14
2.11. Mix of Transaction Types 14
2.12. Queuing Mechanism of Delivery 15
3 Clause 3: Transaction and System Properties 16
3.1. Atomicity Requirements 16
3.2. Consistency Requirements 16
3.3. Isolation Requirements 17
3.4. Durability Requirements 17
4 Clause 4: Scaling and Data Base Population Related Items 19
4.1. Cardinality of Tables 19
4.2. Distribution of Tables and Logs 19
4.3. Data Base Model Implemented 19
4.4. Partitions/Replications Mapping 20
4.5. 60-Day Space Calculations 30
5 Clause 5: Performance Metrics and Response Time Related Items 31
5.1. Response Times 31
5.2. Keying and Think Times 31
5.3. Response Time Frequency Distribution 32
5.4. Performance Curve for Response Time versus Throughput 34
5.5. Think Time Frequency Distribution 35
5.6. Throughput versus Elapsed Time 36
5.7. Steady State Determination 36
5.8. Work Performed During Steady State 36
5.9. Measurement Interval 38
6 Clause 6: SUT, Driver, and Communication Definition Related Items 39
6.1. RTE Availability 39
6.2. Functionality and Performance of Emulated Components 39
6.3. Network Bandwidth 39
6.4. Operator Intervention 39
7 Clause 7: Pricing Related Items 40
7.1. Hardware and Programs Used 40
7.2. Three Year Cost of System Configuration 40
7.3. Availability Dates 40
7.4. Statement of tpmC and Price/Performance 40

8 Clause 9: Audit Related Items 41
9 Appendix A: Client Server Code 45
9.1. Client/Terminal Handler Code 45
9.2. Client Transaction Code 57
10 Appendix B: Tunable Paramaters 92
10.1. Database Parameters 92
10.2. Transaction Monitor Parameters 98
10.3. Linux Parameters 104
11 Appendix C: Database Setup Code 114
11.1. Database Creation Scripts 114
11.2. Data Generation 330
12 Appendix D: Pricing 343

Abstract

This report documents the full disclosure information required by the TPC Benchmark™ C Standard Specification Revision 5.11 dated Feb, 2010, for measurements on the IBM System x3850 X5. The software used on the IBM System x3850 X5 includes SUSE Enterprise Linux Server 11 SP1 operating system and DB2 9.7 data server. Microsoft COM+ is used as the transaction manager. This result was originally published on November 16 2010. This results was repriced on May 16, 2011 due to changing the pricing source and price for 2 of the components. No other changes were made.

IBM System x3850 X5

Company Name	System Name	Database Software	Operating System Software
IBM Corporation	IBM System x3850 X5	DB2 9.7	SUSE Enterprise Linux Server 11 SP1

Total System Cost	TPC-C Throughput	Price/Performance
• Hardware • Software • 3 Years Maintenance	Sustained maximum throughput of system running TPC-C expressed in transactions per minute	Total system cost/tpmC
$1,393,681 USD	2,308,099	$0.60 USD

Preface

TPC Benchmark™ C Standard Specification was developed by the Transaction Processing Performance Council (TPC). It was released on August 13, 1992 and updated with revision 5.11 in Feb. 2010.

This is the full disclosure report for benchmark testing of the IBM System x3850 X5 and DB2 9.7 according to the TPC Benchmark C Standard Specification.

TPC Benchmark C exercises the system components necessary to perform tasks associated with that class of on-line transaction processing (OLTP) environments emphasizing a mixture of read-only and update intensive transactions. This is a complex OLTP application environment exercising a breadth of system components associated by such environments characterized by:

- The simultaneous execution of multiple transaction types that span a breadth of complexity
- On-line and deferred transaction execution modes
- Multiple on-line terminal sessions
- Moderate system and application execution time
- Significant disk input/output
- Transaction integrity (ACID properties)
- Non-uniform distribution of data access through primary and secondary keys
- Data bases consisting of many tables with a wide variety of sizes, attributes, and relationships
- Contention on data access and update

This benchmark defines four on-line transactions and one deferred transaction, intended to emulate functions that are common to many OLTP applications. However, this benchmark does not reflect the entire range of OLTP requirements. The extent to which a customer can achieve the results reported by a vendor is highly dependent on how closely TPC-C approximates the customer application. The relative performance of systems derived from this benchmark does not necessarily hold for other workloads or environments. Extrapolations to any other environment are not recommended.

Benchmark results are highly dependent upon workload, specific application requirements, and systems design and implementation. Relative system performance will vary as a result of these and other factors. Therefore, TPC-C should not be used as a substitute for a specific customer application benchmarks when critical capacity planning and/or product evaluation decisions are contemplated.

The performance metric reported by TPC-C is a “business throughput” measuring the number of orders processed per minute. Multiple transactions are used to simulate the business activity of processing an order, and each transaction is subject to a response time constraint. The performance metric for this benchmark is expressed in transactions-per-minute-C (tpmC). To be compliant with the TPC-C standard, all references to tpmC results must include the tpmC rate, the associated price-per-tpmC, and the availability date of the priced configuration.

0 General Items

0.1. Application Code Disclosure

The application program (as defined in Clause 2.1.7) must be disclosed. This includes, but is not limited to, the code implementing the five transactions and the terminal input and output functions.

Appendix A contains the application code for the five TPC Benchmark™ C transactions.

0.2. Benchmark Sponsor

A statement identifying the benchmark sponsor(s) and other participating companies must be provided.

This benchmark was sponsored by **International Business Machines Corporation.**

0.3. Parameter Settings

Settings must be provided for all customer-tunable parameters and options which have been changed from the defaults found in actual products, including but not limited to:

- *Data Base tuning options*
- *Recovery/commit options*
- *Consistency/locking options*
- *Operating system and application configuration parameters.*

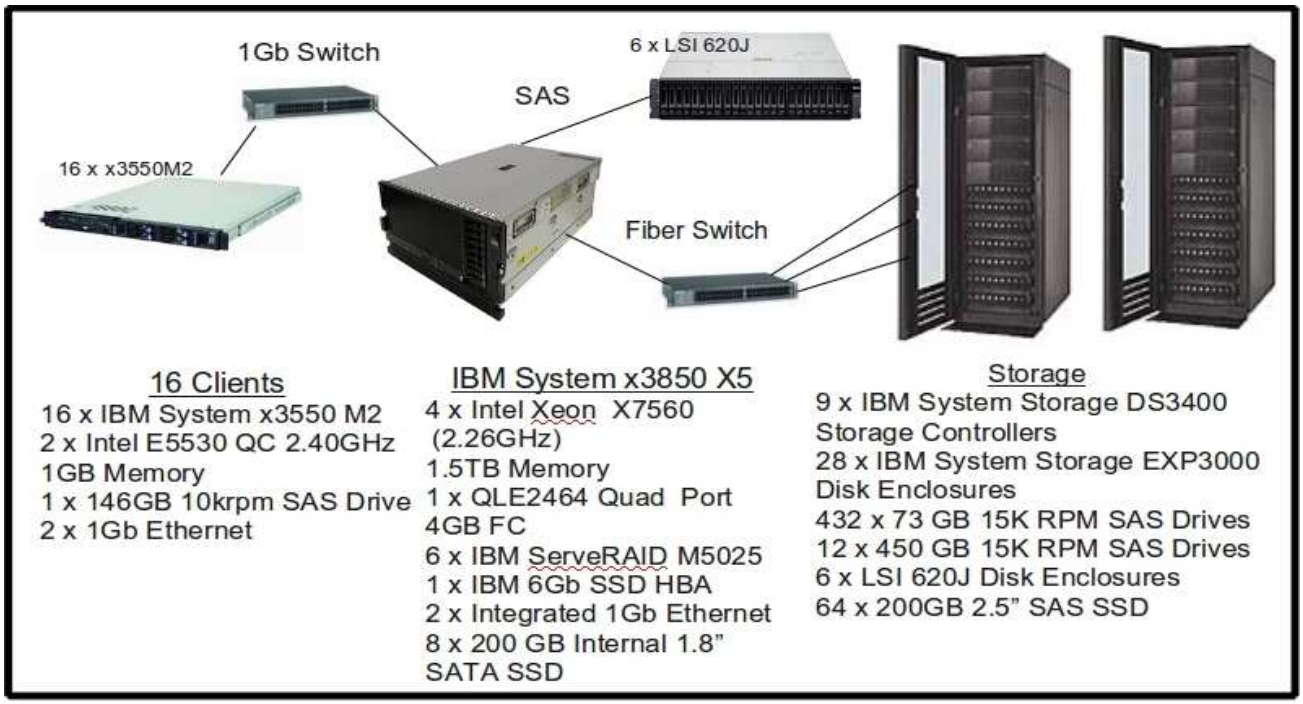
Appendix B contains the system, data base, and application parameters changed from their default values used in these TPC Benchmark™ C tests.

0.4. Configuration Diagrams

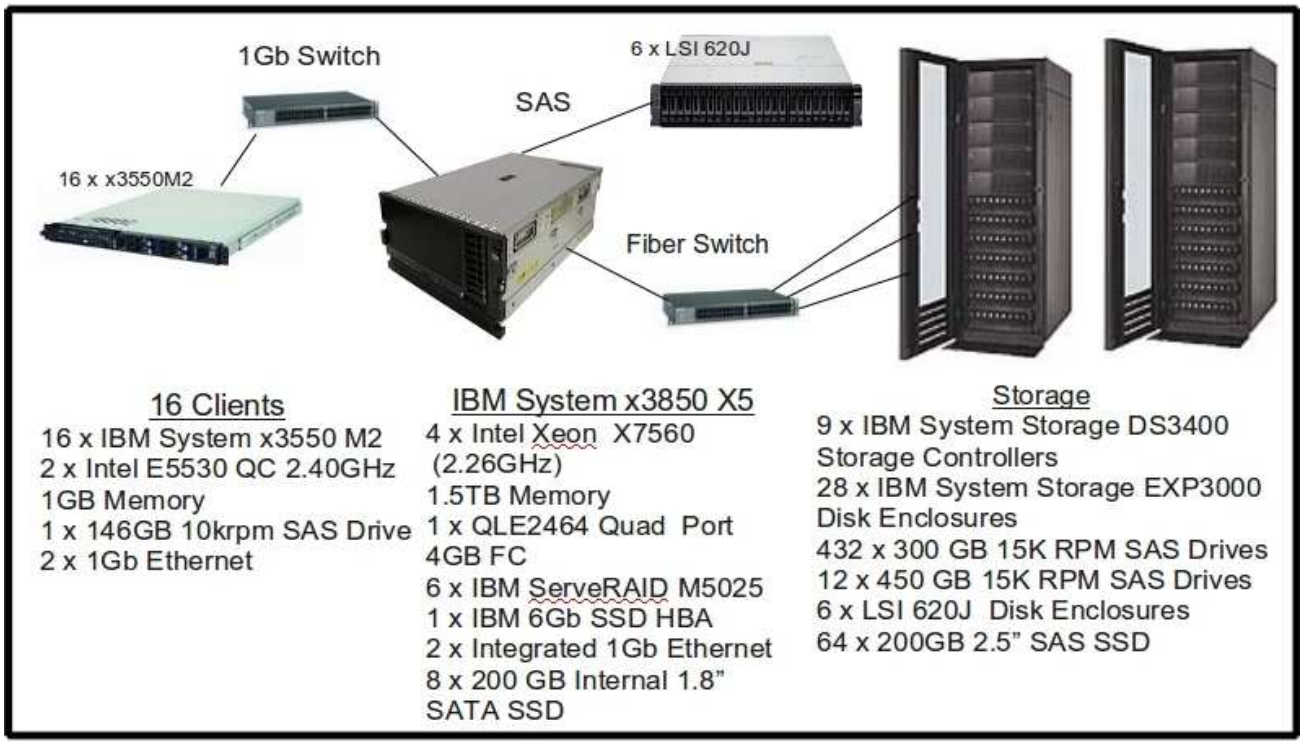
Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences. This includes, but is not limited to:

- *Number and type of processors*
- *Size of allocated memory, and any specific mapping/partitioning of memory unique to the test*
- *Number and type of disk units (and controllers, if applicable)*
- *Number of channels or bus connections to disk units, including the protocol type*
- *Number of LAN (e.g. Ethernet) connections, including routers, work stations, terminals, etc, that were physically used in the test or are incorporated into the pricing structure (see Clause 8.1.8)*
- *Type and run-time execution location of software components (e.g. DBMS, client processes, transaction monitors, software drivers, etc)*

IBM System x3850 X5 Benchmark Configuration



IBM System x3850 X5 Priced Configuration



A total of 432 73GB 15K rpm SAS drives were used in the DS3400s for the benchmark; 432 300GB 15K rpm SAS drives were priced for these drives and substituted on a one for one basis. The auditor reviewed the specifications of the drives and certified that the pricing upgrade was compliant with the TPC-C specification.

1 Clause 1: Logical Data Base Design Related Items

1.1. Table Definitions

Listings must be provided for all table definition statements and all other statements used to setup the data base.

Appendix C contains the table definitions and the database load programs used to build the data base.

1.2. Database Organization

The physical organization of tables and indices, within the data base, must be disclosed.

Physical space was allocated to DB2 on the server disks according to the details provided in Appendix C.

1.3. Insert and/or Delete Operations

It must be ascertained that insert and/or delete operations to any of the tables can occur concurrently with the TPC-C transaction mix. Furthermore, any restriction in the SUT data base implementation that precludes inserts beyond the limits defined in Clause 1.4.11 must be disclosed. This includes the maximum number of rows that can be inserted and the maximum key value for these new rows.

There were no restrictions on insert and/or delete operations to any of the tables. The space required for an additional five percent of the initial table cardinality was allocated to DB2 and priced as static space.

The insert and delete functions were verified by the auditor. In addition, the auditor verified that the primary key for each database table could be updated outside the range of its initial partition.

1.4. Horizontal or Vertical Partitioning

While there are few restrictions placed upon horizontal or vertical partitioning of tables and rows in the TPC-C benchmark, any such partitioning must be disclosed.

WAREHOUSE, DISTRICT, STOCK, CUSTOMER, HISTORY, ORDERS, ORDERLINE, and NEWORDER were horizontally partitioned into multiple tables.

For each partitioned table set, a view was created over all table partitions to provide full transparency of data manipulation.

No tables were replicated.

2 Clause 2: Transaction & Terminal Profiles Related Items

2.1. Verification for the Random Number Generator

The method of verification for the random number generation must be disclosed.

The seeds for each user were captured and verified by the auditor to be unique. In addition, the contents of the database were systematically searched and randomly sampled by the auditor for patterns that would indicate the random number generator had affected any kind of a discernible pattern; none were found.

2.2. Input/Output Screens

The actual layouts of the terminal input/output screens must be disclosed.

The screen layouts are presented in HTML 1.0 web pages. Clauses 2.4.3, 2.5.3, 2.6.3, 2.7.3, and 2.8.3 of the TPC-C specifications were used as guidelines for HTML character placement.

2.3. Priced Terminal Features

The method used to verify that the emulated terminals provide all the features described in Clause 2.2.2.4 must be explained. Although not specifically priced, the type and model of the terminals used for the demonstration in 8.1.3.3 must be disclosed and commercially available (including supporting software and maintenance).

Microsoft Internet Explorer was used to verify the compliance with clause 2.2.2.4.

2.4. Presentation Managers

Any usage of presentation managers or intelligent terminals must be explained.

The workstations did not involve screen presentations, message bundling or local storage of TPC-C rows. All screen processing was handled by the client system. All data manipulation was handled by the server system.

2.5. Home and Remote Order-lines

The percentage of home and remote order-lines in the New-Order transactions must be disclosed.

Table 2-1 shows the percentage of home and remote transactions that occurred during the measurement period for the New-Order transactions.

2.6. New-Order Rollback Transactions

The percentage of New-Order transactions that were rolled back as a result of an illegal item number must be disclosed.

Table 2-1 shows the percentage of New-Order transactions that were rolled back due to an illegal item being entered.

2.7. Number of Items per Order

The number of items per order entered by New-Order transactions must be disclosed.

Table 2-1 show the average number of items ordered per New-Order transaction.

2.8. Home and Remote Payment Transactions

The percentage of home and remote Payment transactions must be disclosed.

Table 2-1 shows the percentage of home and remote transactions that occurred during the measurement period for the Payment transactions.

2.9. Non-Primary Key Transactions

The percentage of Payment and Order-Status transactions that used non-primary key (C_LAST) access to the data base must be disclosed.

Table 2-1 shows the percentage of non-primary key accesses to the data base by the Payment and Order-Status transactions.

2.10. Skipped Delivery Transactions

The percentage of Delivery transactions that were skipped as a result of an insufficient number of rows in the NEW-ORDER table must be disclosed.

Table 2-1 shows the percentage of Delivery transactions missed due to a shortage of supply of rows in the NEW-ORDER table.

2.11. Mix of Transaction Types

The mix (i.e. percentages) of transaction types seen by the SUT must be disclosed.

Table 2-1 shows the mix percentage for each of the transaction types executed by the SUT.

New Order	**IBM System x3850 X5**
Percentage of Home order lines	99.00%
Percentage of Remote order lines	1.00%
Percentage of Rolled Back Transactions	1.000%
Average Number of Items per order	10.00
Payment	
Percentage of Home transactions	85.001%
Percentage of Remote transactions	14.999%
Non-Primary Key Access	
Percentage of Payment using C_LAST	60.002%
Percentage of Order-Status using C_LAST	59.998%
Delivery	
Delivery transactions skipped	0
Transaction Mix	
New-Order	44.950%
Payment	43.021%
Order-Status	4.010%
Delivery	4.008%
Stock-Level	4.010%

Table 2-1: Numerical Quantities for Transaction and Terminal Profiles

2.12. Queuing Mechanism of Delivery

The queuing mechanism used to defer execution of the Delivery transaction must be disclosed.

The Delivery transaction was submitted to an ISAPI queue that is separate from the COM+ queue that the other transactions used. This queue is serviced by a variable amount of threads that are separate from the worker threads inside the web server. Web server threads are able to complete the on-line part of the Delivery transaction and immediately return successful queuing responses to the drivers. The threads servicing the queue are responsible for completing the deferred part of the transaction asynchronously.

3 Clause 3: Transaction and System Properties

The results of the ACID test must be disclosed along with a description of how the ACID requirements were met.

All ACID tests were conducted according to specification.

3.1. Atomicity Requirements

The system under test must guarantee that data base transactions are atomic; the system will either perform all individual operations on the data, or will assure that no partially-completed operations leave any effects on the data.

3.1.1. Atomicity of Completed Transaction

Perform the Payment transaction for a randomly selected warehouse, district, and customer (by customer number) and verify that the records in the CUSTOMER, DISTRICT, and WAREHOUSE tables have been changed appropriately.

The following steps were performed to verify the Atomicity of completed transactions.

1. The balance, BALANCE_1, was retrieved from the CUSTOMER table for a random Customer, District and Warehouse combination.
2. The Payment transaction was executed and committed for the Customer, District, and Warehouse combination used in step 1.
3. The balance, BALANCE_2, was retrieved again for the Customer, District, and Warehouse combination used in step 1 and step 2. It was verified that BALANCE_1 was greater than BALANCE_2 by the amount of the Payment transaction.

3.1.2. Atomicity of Aborted Transactions

Perform the Payment transaction for a randomly selected warehouse, district, and customer (by customer number) and substitute a ROLLBACK of the transaction for the COMMIT of the transaction. Verify that the records in the CUSTOMER, DISTRICT, and WAREHOUSE tables have NOT been changed.

The following steps were performed to verify the Atomicity of the aborted Payment transaction:

1. The Payment application code was implemented with a Perl script that allowed the transaction to be rolled back rather than committed.
2. The balance, BALANCE_3, was retrieved from the Customer table for the same Customer, District, and Warehouse combination used in the completed Payment transaction Atomicity test.
3. The Payment transaction was executed for the Customer, District and Warehouse used in step 2. Rather than commit the transaction, the transaction was rolled back.
4. The balance, BALANCE_4 was retrieved again for the Customer, District, and Warehouse combination used in step 2. It was verified that BALANCE_4 was equal to BALANCE_3, demonstrating that there were no remaining effects of the rolled back Payment transaction.

3.2. Consistency Requirements

Consistency is the property of the application that requires any execution of a data base transaction to take the data base from one consistent state to another, assuming that the data base is initially in a consistent state.

Verify that the data base is initially consistent by verifying that it meets the consistency conditions defined in Clauses 3.3.2.1 to 3.3.2.4. Describe the steps used to do this in sufficient detail so that the steps are independently repeatable.

The specification defines 12 consistency conditions of which the following four are required to be explicitly demonstrated:

1. The sum of balances (d_ytd) for all Districts within a specific Warehouse is equal to the balance (w_ytd) of that Warehouse.
2. For each District within a Warehouse, the next available Order ID (d_next_o_id) minus one is equal to the most recent Order ID [max(o_id)] for the Order table associated with the preceeding District and Warehouse.

Additionally, that same relationship exists for the most recent Order ID [max(o_id)] for the New Order table associated with the same District and Warehouse. Those relationships can be illustrated as follows:

d_next_o_id – 1 = max(o_id) = max(no_o_id)

where (d_w_id = o_w_id = no_w_id) and (d_id = o_d_id = no_d_id)

3. For each District within a Warehouse, the value of the most recent Order ID [max(no_o_id)] minus the first Order ID [min(no_o_id)] plus one, for the New Order table associated with the District and Warehouse equals the number of rows in that New Order table. That relationship can be illustrated as follows:

   max(no_o_id) – min(no_o_id) + 1 = number of rows in New Order for the Warehouse/District

4. For each District within a Warehouse, the sum of Order Line counts [sum(o_ol_cnt)] for the Order table associated with the District equals the number of rows in the Order Line table associated with the same District. That relationship can be illustrated as follows:

   sum(o_ol_cnt) = number of rows in the Order Line table for the Warehouse/District

An RTE driven run was executed against a freshly loaded database. After the run the 4 consistency conditions defined above were tested using a script to issue queries to the database. All queries showed that the database was still in a consistent state.

3.3. Isolation Requirements

Operations of concurrent data base transactions must yield results which are indistinguishable from the results which would be obtained by forcing each transaction to be serially executed to completion in some order.

The benchmark specification defines nine tests to demonstrate the property of transaction isolation. The tests, described in Clauses 3.4.2.1 – 3.4.2.9, were all successfully executed using a series of scripts. Case A was observed during the execution of Isolation Tests 7-9.

3.4. Durability Requirements

The tested system must guarantee durability: the ability to preserve the effects of committed transactions and insure data base consistency after recovery from any one of the failures listed in Clause 3.5.3

- *Permanent irrecoverable failure of any single durable medium containing TPC-C database tables or recovery log data (this test includes failure of all or part of memory)*
- *Instantaneous interruption (system crash/system hang) in processing that requires system reboot to recover*
- *Failure of all or part of memory (loss of contents)*

Failure of Log Disk:

This test was conducted on a 12.5% scaled database. The following steps were successfully performed:

1. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving SUM_1.
2. A 12.5% load test was started and allowed to run and steady state was reached and maintained for over 5 minutes.
3. One of the disks containing the transaction log was removed. Since the log was implemented as a RAID-10 array, DB2 continued to process the transactions successfully.
4. The test continued for at least another 5 minutes.
5. The test was ended and the database was deactivated in a controlled manner.

6. Step 2 was performed returning SUM_2. It was verified that SUM_2 was equal to SUM_1 plus the number of completed New_Order transactions recorded by the RTE 1
7. Consistency condition 3 was verified.

Failure of Durable Medium Containing TPC-C Database Tables:

This test was conducted on a fully-scaled database. Since the tables were split between SSD and HDD drives, this test was run twice, one run failed a HDD and the second run failed an SSD. The following steps were successfully performed both times:

1. The contents of the database were backed up in full.
2. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving SUM_1.
3. A scaled-down test was started with 12.5% of the full load and allowed to run and steady state was reached and maintained for over 5 minutes..
4. A disk containing the TPC-C tables was removed, causing DB2 to report numerous errors.
5. The system was subsequently shutdown.
6. The disk was reinserted.
7. The system was powered back on.
8. The full database was restored from the backup copy in step 1.
9. DB2 was restarted and the transactions in the log were applied to the database.
10. Step 1 was performed returning the value for SUM_2. It was verified that SUM_2 was greater than SUM_1 plus the completed New_Order transactions recorded by the RTE. The additional transactions found in the database were attributed to in-flight activity at the time of the failure.
11. Consistency condition 3 was verified.

Instantaneous Interruption, Memory Failure, and Loss of Power:

This test was conducted on a fully-scaled database. The following steps were successfully performed:

1. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving SUM_1.
2. A full load test was started and allowed to run for over 5 minutes.
3. The database server was powered off, which removed power from all system components, including memory.
4. The database server was powered back on and DB2 was allowed to recover.
5. Step 1 was performed returning the value for SUM_2. It was verified that SUM_2 was greater than SUM_1 plus the completed New_Order transactions recorded by the RTE. The additional transactions found in the database were attributed to in-flight activity at the time of the failure.
6. Consistency condition 3 was verified.

4 Clause 4: Scaling and Data Base Population Related Items

4.1. Cardinality of Tables

The cardinality (e.g., the number of rows) of each table, as it existed at the start of the benchmark run, must be disclosed.

Table 4-1 portrays the TPC Benchmark™ C defined tables and the number of rows for each table as they were built initially.

Table Name	Number of Rows
Warehouse	184,032
District	1,840,320
Customer	5,520,960,000
History	5,520,960,000
Orders	5,520,960,000
New Order	1,656,288,000
Order Line	55,209,587,467
Stock	18,403,200,000
Item	100,000

Table 4-1: Initial Cardinality of Tables

4.2. Distribution of Tables and Logs

The distribution of tables and logs across all media must be explicitly depicted for the tested and priced systems.

The log was configured using one Logical drive attached to one of the ServeRAID-M5025 disk controllers. The log was configured as a RAID-10 disk array consisting of twelve 450GB hot-swap SAS disks housed in one EXP3000 drive enclosure and backed by a UPS.

There are 36 Logical Disks (LD) for the dynamic database tables stored on fiber attached DS3400s. There is one quad port storage adapters for these tables. Each adapter port is assigned 18 LDs. Each LD is configured as a RAID0 disk array, with 12 physical disks. There are a total of 432 data disks for dynamic tables and each physical disk has a capacity of 73.4 GB. Each LD is partitioned identically, see Table 4-2, Table 4-3, and Table 4-4. There are 72 200GB SSD drives attached via SAS or SATA adapters. All of these drives are configured as JBOD and are partitioned identically and hold the static database tables. See Table 4-6,Table 4-7, and Table 4-8.

4.3. Data Base Model Implemented

A statement must be provided that describes the data base model implemented by the DBMS used.

The database manager used for this testing was DB2 9.7. DB2 is a relational DBMS. DB2 remote stored procedures and embedded SQL statements were used. The DB2 stored procedures were invoked via SQL CALL statements. Both the client application and stored procedures were written in embedded C code.

4.4. Partitions/Replications Mapping

The mapping of data base partitions/replications must be explicitly described.

The specifics of the distribution of partitioned and non-partitioned tables across the physical media can be found in tables 4-2.

Fibre Channel Adapter Mappings	
Logical Adapter ID	**Physical Adapter ID**
FC1	2102001b32c1df23
FC2	2100001b3281db23

Table 4-2: IBM System x3850 X5 Fiber Channel Adapter Mapping

Fibre Channel Adapter to Block Device Mapping	
Logical Adapter ID	**Block Device ID**
FC1	/dev/disk/by-id/scsi-3600a0b800034d6ce00001c084bab2ea6
FC1	/dev/disk/by-id/scsi-3600a0b800034d6ce00001c0a4bab2eda
FC1	/dev/disk/by-id/scsi-3600a0b800024589800001a254bab5798
FC1	/dev/disk/by-id/scsi-3600a0b800024589800001a274bab57c8
FC1	/dev/disk/by-id/scsi-3600a0b800036fefd00001d714bab6032
FC1	/dev/disk/by-id/scsi-3600a0b800036fefd00001d734bab6066
FC1	/dev/disk/by-id/scsi-3600a0b80003460b700001d584bab33cc
FC1	/dev/disk/by-id/scsi-3600a0b80003460b700001d5a4bab33fe
FC1	/dev/disk/by-id/scsi-3600a0b800034d5a8000081244bab344a
FC1	/dev/disk/by-id/scsi-3600a0b800034d5a8000081264bab347a
FC1	/dev/disk/by-id/scsi-3600a0b800034d6bb00001ace4bab32b7
FC1	/dev/disk/by-id/scsi-3600a0b800034d6bb00001ad04bab330f
FC1	/dev/disk/by-id/scsi-3600a0b800034d6c500001c4f4bab3421
FC1	/dev/disk/by-id/scsi-3600a0b800034d6c500001c514bab3479
FC1	/dev/disk/by-id/scsi-3600a0b800034608d000026374bab35eb
FC1	/dev/disk/by-id/scsi-3600a0b800034608d000026394bab3643
FC1	/dev/disk/by-id/scsi-3600a0b8000245851000019b34bab587b
FC1	/dev/disk/by-id/scsi-3600a0b8000245851000019b54bab58d3
FC2	/dev/disk/by-id/scsi-3600a0b800034d70b00001bd44bab362d
FC2	/dev/disk/by-id/scsi-3600a0b800034d70b00001bd64bab3687
FC2	/dev/disk/by-id/scsi-3600a0b800036945800001ae54bab60a2
FC2	/dev/disk/by-id/scsi-3600a0b800036945800001ae74bab60da
FC2	/dev/disk/by-id/scsi-3600a0b800034d6cd00001d2b4bab31ee
FC2	/dev/disk/by-id/scsi-3600a0b800034d6cd00001d2d4bab3224
FC2	/dev/disk/by-id/scsi-3600a0b800036985400001aec4bab5930

FC2	/dev/disk/by-id/scsi-3600a0b800036985400001aee4bab5960
FC2	/dev/disk/by-id/scsi-3600a0b800036991b00001b634bab5e24
FC2	/dev/disk/by-id/scsi-3600a0b800036991b00001b654bab5e54
FC2	/dev/disk/by-id/scsi-3600a0b800036969000001ad74bab5f9b
FC2	/dev/disk/by-id/scsi-3600a0b800036969000001ad94bab5ff3
FC2	/dev/disk/by-id/scsi-3600a0b800034d6bd00001bcf4bab37d7
FC2	/dev/disk/by-id/scsi-3600a0b800034d6bd00001bd14bab3831
FC2	/dev/disk/by-id/scsi-3600a0b800036967f00001a2b4bab6191
FC2	/dev/disk/by-id/scsi-3600a0b800036967f00001a2d4bab61eb
FC2	/dev/disk/by-id/scsi-3600a0b800036986400001b084bab5db7
FC2	/dev/disk/by-id/scsi-3600a0b800036986400001b0a4bab5e11

Table 4-3: IBM System x3850 X5 Fiber Channel Disk Mapping

Fiber Channel Disk Partition Number to Tablespace Mapping	
Device Parition Number	**Tablespace ID**
1	ts_W
2	ts_D
3	ts_N
4	<extended partition>
5	is_O2
6	ts_O
7	ts_H
8	ts_OL
9	ts_OL
10	ts_I

Table 4-4: IBM System x3850 X5 Fiber Channel Partition Mapping

Fibre Block Device to Database Tablespace Mapping	
Block Device ID	**Tablespace ID**
/dev/disk/by-id/scsi-3600a0b800034d6ce00001c084bab2ea6	ts_W_001, ts_D_001, ts_N_001, ts_O2_001, ts_O_001, ts_H_001, ts_OS_001, ts_OL_037, ts_I
/dev/disk/by-id/scsi-3600a0b800034d6ce00001c0a4bab2eda	ts_W_002, ts_D_002, ts_N_002, ts_O2_002, ts_O_002, ts_H_002, ts_OS_002, ts_OL_038
/dev/disk/by-id/scsi-3600a0b800024589800001a254bab5798	ts_W_003, ts_D_003, ts_N_003, ts_O2_003, ts_O_003, ts_H_003, ts_OS_003, ts_OL_039
/dev/disk/by-id/scsi-3600a0b800024589800001a274bab57c8	ts_W_004, ts_D_004, ts_N_004, ts_O2_004, ts_O_004, ts_H_004, ts_OS_004, ts_OL_040
/dev/disk/by-id/scsi-3600a0b800036fefd00001d714bab6032	ts_W_005, ts_D_005, ts_N_005, ts_O2_005, ts_O_005, ts_H_005, ts_OS_005, ts_OL_041

/dev/disk/by-id/scsi-3600a0b800036fefd00001d734bab6066	ts_W_006, ts_D_006, ts_N_006, ts_O2_006, ts_O_006, ts_H_006, ts_OS_006, ts_OL_042
/dev/disk/by-id/scsi-3600a0b80003460b700001d584bab33cc	ts_W_007, ts_D_007, ts_N_007, ts_O2_007, ts_O_007, ts_H_007, ts_OS_007, ts_OL_043
/dev/disk/by-id/scsi-3600a0b80003460b700001d5a4bab33fe	ts_W_008, ts_D_008, ts_N_008, ts_O2_008, ts_O_008, ts_H_008, ts_OS_008, ts_OL_044
/dev/disk/by-id/scsi-3600a0b800036945800001ae54bab60a2	ts_W_009, ts_D_009, ts_N_009, ts_O2_009, ts_O_009, ts_H_009, ts_OS_009, ts_OL_045
/dev/disk/by-id/scsi-3600a0b800036945800001ae74bab60da	ts_W_010, ts_D_010, ts_N_010, ts_O2_010, ts_O_010, ts_H_010, ts_OS_010, ts_OL_046
/dev/disk/by-id/scsi-3600a0b800034d6cd00001d2b4bab31ee	ts_W_011, ts_D_011, ts_N_011, ts_O2_011, ts_O_011, ts_H_011, ts_OS_011, ts_OL_047
/dev/disk/by-id/scsi-3600a0b800034d6cd00001d2d4bab3224	ts_W_012, ts_D_012, ts_N_012, ts_O2_012, ts_O_012, ts_H_012, ts_OS_012, ts_OL_048
/dev/disk/by-id/scsi-3600a0b800036985400001aec4bab5930	ts_W_013, ts_D_013, ts_N_013, ts_O2_013, ts_O_013, ts_H_013, ts_OS_013, ts_OL_049
/dev/disk/by-id/scsi-3600a0b800036985400001aee4bab5960	ts_W_014, ts_D_014, ts_N_014, ts_O2_014, ts_O_014, ts_H_014, ts_OS_014, ts_OL_050
/dev/disk/by-id/scsi-3600a0b800034d5a8000081244bab344a	ts_W_015, ts_D_015, ts_N_015, ts_O2_015, ts_O_015, ts_H_015, ts_OS_015, ts_OL_051
/dev/disk/by-id/scsi-3600a0b800034d5a8000081264bab347a	ts_W_016, ts_D_016, ts_N_016, ts_O2_016, ts_O_016, ts_H_016, ts_OS_016, ts_OL_052
/dev/disk/by-id/scsi-3600a0b800036991b00001b634bab5e24	ts_W_017, ts_D_017, ts_N_017, ts_O2_017, ts_O_017, ts_H_017, ts_OS_017, ts_OL_053
/dev/disk/by-id/scsi-3600a0b800036991b00001b654bab5e54	ts_W_018, ts_D_018, ts_N_018, ts_O2_018, ts_O_018, ts_H_018, ts_OS_018, ts_OL_054
/dev/disk/by-id/scsi-3600a0b800034d6bb00001ace4bab32b7	ts_W_019, ts_D_019, ts_N_019, ts_O2_019, ts_O_019, ts_H_019, ts_OS_019, ts_OL_055
/dev/disk/by-id/scsi-3600a0b800034d6bb00001ad04bab330f	ts_W_020, ts_D_020, ts_N_020, ts_O2_020, ts_O_020, ts_H_020, ts_OS_020, ts_OL_056
/dev/disk/by-id/scsi-3600a0b800034d6c500001c4f4bab3421	ts_W_021, ts_D_021, ts_N_021, ts_O2_021, ts_O_021, ts_H_021, ts_OS_021, ts_OL_057
/dev/disk/by-id/scsi-3600a0b800034d6c500001c514bab3479	ts_W_022, ts_D_022, ts_N_022, ts_O2_022, ts_O_022, ts_H_022, ts_OS_022, ts_OL_058
/dev/disk/by-id/scsi-3600a0b800034608d000026374bab35eb	ts_W_023, ts_D_023, ts_N_023, ts_O2_023, ts_O_023, ts_H_023, ts_OS_023, ts_OL_059
/dev/disk/by-id/scsi-3600a0b800034608d000026394bab3643	ts_W_024, ts_D_024, ts_N_024, ts_O2_024, ts_O_024, ts_H_024, ts_OS_024, ts_OL_060
/dev/disk/by-id/scsi-3600a0b800036969000001ad74bab5f9b	ts_W_025, ts_D_025, ts_N_025, ts_O2_025, ts_O_025, ts_H_025, ts_OS_025, ts_OL_061
/dev/disk/by-id/scsi-3600a0b800036969000001ad94bab5ff3	ts_W_026, ts_D_026, ts_N_026, ts_O2_026, ts_O_026, ts_H_026, ts_OS_026, ts_OL_062
/dev/disk/by-id/scsi-3600a0b8000245851000019b34bab587b	ts_W_027, ts_D_027, ts_N_027, ts_O2_027, ts_O_027, ts_H_027, ts_OS_027, ts_OL_063

/dev/disk/by-id/scsi-3600a0b8000245851000019b54bab58d3	ts_W_028, ts_D_028, ts_N_028, ts_O2_028, ts_O_028, ts_H_028, ts_OS_028, ts_OL_064
/dev/disk/by-id/scsi-3600a0b800034d70b00001bd44bab362d	ts_W_029, ts_D_029, ts_N_029, ts_O2_029, ts_O_029, ts_H_029, ts_OS_029, ts_OL_065
/dev/disk/by-id/scsi-3600a0b800034d70b00001bd64bab3687	ts_W_030, ts_D_030, ts_N_030, ts_O2_030, ts_O_030, ts_H_030, ts_OS_030, ts_OL_066
/dev/disk/by-id/scsi-3600a0b800034d6bd00001bcf4bab37d7	ts_W_031, ts_D_031, ts_N_031, ts_O2_031, ts_O_031, ts_H_031, ts_OS_031, ts_OL_067
/dev/disk/by-id/scsi-3600a0b800034d6bd00001bd14bab3831	ts_W_032, ts_D_032, ts_N_032, ts_O2_032, ts_O_032, ts_H_032, ts_OS_032, ts_OL_068
/dev/disk/by-id/scsi-3600a0b800036967f00001a2b4bab6191	ts_W_033, ts_D_033, ts_N_033, ts_O2_033, ts_O_033, ts_H_033, ts_OS_033, ts_OL_069
/dev/disk/by-id/scsi-3600a0b800036967f00001a2d4bab61eb	ts_W_034, ts_D_034, ts_N_034, ts_O2_034, ts_O_034, ts_H_034, ts_OS_034, ts_OL_070
/dev/disk/by-id/scsi-3600a0b800036986400001b084bab5db7	ts_W_035, ts_D_035, ts_N_035, ts_O2_035, ts_O_035, ts_H_035, ts_OS_035, ts_OL_071
/dev/disk/by-id/scsi-3600a0b800036986400001b0a4bab5e1	ts_W_036, ts_D_036, ts_N_036, ts_O2_036, ts_O_036, ts_H_036, ts_OS_036, ts_OL_072

Table 4-5: IBM System x3850 X5 Fiber Channel Table Space Mapping

SSD Adapter Mappings	
Logical Adapter ID	**Physical Adapter ID**
SAS1	SR91701061
SAS2	SR91701173
SAS3	SR91701062
SAS4	SR91700827
SAS5	SR91701132
SAS6	SR91701089
SAS7	500605b001cbd1d0

Table 4-6: IBM System x3850 X5 SSD Adapter Mapping

SSD Adapter to Block Device Mapping	
Logical Adapter ID	**Block Device ID**
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000a9da236e871e
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000a9e123dacc4e
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000a9e7242dfc0c
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000a9ed2490ac9a
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000a9f324e6ca8f

SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000a9fa254e3d54
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000aa07261189e2
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000aa0e267c5859
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a8000146a47bc3d3b8
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000aa1a273cb7f9
SAS1	/dev/disk/by-id/scsi-3600605b000a8d7a80000aa2127a7c531
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a7ea05c9554b
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a7f0062d2d49
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a7f50676e785
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a7fa06bf97f5
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a7ff070f7c29
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a80e07f498f9
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a81508609feb
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a81b08b91020
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a8200909de24
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a826095fbd69
SAS2	/dev/disk/by-id/scsi-3600605b000a8e3f00000a82b09b0ca8d
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac20460dce23
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac28468762df
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac2d46ce7d73
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac32471a0c9e
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000099c963c023d
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac3c47b07f1a
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e0000011c812edfa05
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac47486422c0
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac4e48c86480
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac53491aa9da
SAS3	/dev/disk/by-id/scsi-3600605b000a8e3e00000ac584965108e
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d03f109970
SAS4	/dev/disk/by-id/scsi-3600605b000a8e610000007f97d4c3c3b
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d03f1488ea
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d03f167512
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d13f1876c8
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d13f1a7c7d
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d13f1c71ed

SAS4	/dev/disk/by-id/scsi-360000000000000000000003d13f1e720d
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d13f2067af
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d13f226973
SAS4	/dev/disk/by-id/scsi-360000000000000000000003d13f246713
SAS5	/dev/disk/by-id/scsi-3600605b000a8e85000054eb4ffce095f
SAS5	/dev/disk/by-id/scsi-3600605b000a8e85000010c4901524429
SAS5	/dev/disk/by-id/scsi-3600605b000a8e8500008082691f16d07
SAS5	/dev/disk/by-id/scsi-3600605b000a8e8500004342b289eb1c0
SAS5	/dev/disk/by-id/scsi-3600605b000a8e850000a5f164504ffeb
SAS5	/dev/disk/by-id/scsi-3600605b000a8e8500004342a2892cedd
SAS5	/dev/disk/by-id/scsi-3600605b000a8e8500004342a2894ba3c
SAS5	/dev/disk/by-id/scsi-3600605b000a8e8500004342a2896bf89
SAS5	/dev/disk/by-id/scsi-3600605b000a8e85000002f18d2362b32
SAS5	/dev/disk/by-id/scsi-3600605b000a8e8500004342a289aa9ba
SAS5	/dev/disk/by-id/scsi-3600605b000a8e85000039ed341d30faa
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a00007e5758096745c
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a00000905f9e8fc9d0
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a000000678665b7472
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a00000067866559718
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a000000679665d57ce
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a00000067866578360
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a00000067866598cca
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a00000067966612155
SAS6	/dev/disk/by-id/scsi-3600605b000a8e5a000000679665f3c6a
SAS7	/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000XA
SAS7	/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000QY
SAS7	/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000RJ
SAS7	/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000ZS
SAS7	/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_5030013B
SAS7	/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000ZT
SAS7	/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000Z8
SAS7	/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000XE

Table 4-7: IBM System x3850 X5 SSD Disk Mapping

SSD Partition Number to Tablespace Mapping	
Device Parition Number	**Tablespace ID**
1	ts_S
2	ts_S
3	ts_C
4	<extended partition>
5	ts_C
6	is_C2
7	is_C2

Table 4-8: IBM System x3850 X5 SSD Partition Mapping

SAS Block Device to Database Tablespace Mapping	
Block Device ID	**Tablespace ID**
/dev/disk/by-id/scsi-3600605b000a8d7a80000a9da236e871e	ts_S_001, ts_S_002, ts_C_001, ts_C_002, is_C2_001, is_C2_002
/dev/disk/by-id/scsi-3600605b000a8d7a80000a9e123dacc4e	ts_S_003, ts_S_004, ts_C_003, ts_C_004, is_C2_003, is_C2_004
/dev/disk/by-id/scsi-3600605b000a8d7a80000a9e7242dfc0c	ts_S_005, ts_S_006, ts_C_005, ts_C_006, is_C2_005, is_C2_006
/dev/disk/by-id/scsi-3600605b000a8d7a80000a9ed2490ac9a	ts_S_007, ts_S_008, ts_C_007, ts_C_008, is_C2_007, is_C2_008
/dev/disk/by-id/scsi-3600605b000a8d7a80000a9f324e6ca8f	ts_S_009, ts_S_010, ts_C_009, ts_C_010, is_C2_009, is_C2_010
/dev/disk/by-id/scsi-3600605b000a8d7a80000a9fa254e3d54	ts_S_011, ts_S_012, ts_C_011, ts_C_012, is_C2_011, is_C2_012
/dev/disk/by-id/scsi-3600605b000a8d7a80000aa07261189e2	ts_S_013, ts_S_014, ts_C_013, ts_C_014, is_C2_013, is_C2_014
/dev/disk/by-id/scsi-3600605b000a8d7a80000aa0e267c5859	ts_S_015, ts_S_016, ts_C_015, ts_C_016, is_C2_015, is_C2_016
/dev/disk/by-id/scsi-3600605b000a8d7a8000146a47bc3d3b8	ts_S_017, ts_S_018, ts_C_017, ts_C_018, is_C2_017, is_C2_018
/dev/disk/by-id/scsi-3600605b000a8d7a80000aa1a273cb7f9	ts_S_019, ts_S_020, ts_C_019, ts_C_020, is_C2_019, is_C2_020
/dev/disk/by-id/scsi-3600605b000a8d7a80000aa2127a7c532	ts_S_021, ts_S_022, ts_C_021, ts_C_022, is_C2_021, is_C2_022
/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000XA	ts_S_023, ts_S_024, ts_C_023, ts_C_024, is_C2_023, is_C2_024
/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000QY	ts_S_025, ts_S_026, ts_C_025, ts_C_026, is_C2_025, is_C2_026
/dev/disk/by-id/scsi-3600605b000a8e85000054eb4ffce095f	ts_S_027, ts_S_028, ts_C_027, ts_C_028, is_C2_027, is_C2_028
/dev/disk/by-id/scsi-3600605b000a8e85000010c4901524429	ts_S_029, ts_S_030, ts_C_029, ts_C_030, is_C2_029, is_C2_030

/dev/disk/by-id/scsi-3600605b000a8e8500008082691f16d07	ts_S_031, ts_S_032, ts_C_031, ts_C_032, is_C2_031, is_C2_032
/dev/disk/by-id/scsi-3600605b000a8e5a00007e5758096745c	ts_S_033, ts_S_034, ts_C_033, ts_C_034, is_C2_033, is_C2_034
/dev/disk/by-id/scsi-3600605b000a8e5a00000905f9e8fc9d0	ts_S_035, ts_S_036, ts_C_035, ts_C_036, is_C2_035, is_C2_036
/dev/disk/by-id/scsi-3600605b000a8e3f00000a7ea05c9554b	ts_S_037, ts_S_038, ts_C_037, ts_C_038, is_C2_037, is_C2_038
/dev/disk/by-id/scsi-3600605b000a8e3f00000a7f0062d2d49	ts_S_039, ts_S_040, ts_C_039, ts_C_040, is_C2_039, is_C2_040
/dev/disk/by-id/scsi-3600605b000a8e3f00000a7f50676e785	ts_S_041, ts_S_042, ts_C_041, ts_C_042, is_C2_041, is_C2_042
/dev/disk/by-id/scsi-3600605b000a8e3f00000a7fa06bf97f5	ts_S_043, ts_S_044, ts_C_043, ts_C_044, is_C2_043, is_C2_044
/dev/disk/by-id/scsi-3600605b000a8e3f00000a7ff070f7c29	ts_S_045, ts_S_046, ts_C_045, ts_C_046, is_C2_045, is_C2_046
/dev/disk/by-id/scsi-3600605b000a8e3f00000a80e07f498f9	ts_S_047, ts_S_048, ts_C_047, ts_C_048, is_C2_047, is_C2_048
/dev/disk/by-id/scsi-3600605b000a8e3f00000a81508609feb	ts_S_049, ts_S_050, ts_C_049, ts_C_050, is_C2_049, is_C2_050
/dev/disk/by-id/scsi-3600605b000a8e3f00000a81b08b91020	ts_S_051, ts_S_052, ts_C_051, ts_C_052, is_C2_051, is_C2_052
/dev/disk/by-id/scsi-3600605b000a8e3f00000a8200909de24	ts_S_053, ts_S_054, ts_C_053, ts_C_054, is_C2_053, 27is_C2_054
/dev/disk/by-id/scsi-3600605b000a8e3f00000a826095fbd69	ts_S_055, ts_S_056, ts_C_055, ts_C_056, is_C2_055, is_C2_056
/dev/disk/by-id/scsi-3600605b000a8e3f00000a82b09b0ca8d	ts_S_057, ts_S_058, ts_C_057, ts_C_058, is_C2_057, is_C2_058
/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000RJ	ts_S_059, ts_S_060, ts_C_059, ts_C_060, is_C2_059, is_C2_060
/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000ZS	ts_S_061, ts_S_062, ts_C_061, ts_C_062, is_C2_061, is_C2_062
/dev/disk/by-id/scsi-3600605b000a8e8500004342b289eb1c0	ts_S_063, ts_S_064, ts_C_063, ts_C_064, is_C2_063, is_C2_064
/dev/disk/by-id/scsi-3600605b000a8e850000a5f164504ffeb	ts_S_065, ts_S_066, ts_C_065, ts_C_066, is_C2_065, is_C2_066
/dev/disk/by-id/scsi-3600605b000a8e8500004342a2892cedd	ts_S_067, ts_S_068, ts_C_067, ts_C_068, is_C2_067, is_C2_068
/dev/disk/by-id/scsi-3600605b000a8e5a000000678665b7472	ts_S_069, ts_S_070, ts_C_069, ts_C_070, is_C2_069, is_C2_070
/dev/disk/by-id/scsi-3600605b000a8e5a00000067866559718	ts_S_071, ts_S_072, ts_C_071, ts_C_072, is_C2_071, is_C2_072
/dev/disk/by-id/scsi-3600605b000a8e3e00000ac20460dce23	ts_S_073, ts_S_074, ts_C_073, ts_C_074, is_C2_073, is_C2_074

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac28468762df	ts_S_075, ts_S_076, ts_C_075, ts_C_076, is_C2_075, is_C2_076
/dev/disk/by-id/scsi-3600605b000a8e3e00000ac2d46ce7d73	ts_S_077, ts_S_078, ts_C_077, ts_C_078, is_C2_077, is_C2_078
/dev/disk/by-id/scsi-3600605b000a8e3e00000ac32471a0c9e	ts_S_079, ts_S_080, ts_C_079, ts_C_080, is_C2_079, is_C2_080
/dev/disk/by-id/scsi-3600605b000a8e3e00000099c963c023d	ts_S_081, ts_S_082, ts_C_081, ts_C_082, is_C2_081, is_C2_082
/dev/disk/by-id/scsi-3600605b000a8e3e00000ac3c47b07f1a	ts_S_083, ts_S_084, ts_C_083, ts_C_084, is_C2_083, is_C2_084
/dev/disk/by-id/scsi-3600605b000a8e3e0000011c812edfa05	ts_S_085, ts_S_086, ts_C_085, ts_C_086, is_C2_085, is_C2_086
/dev/disk/by-id/scsi-3600605b000a8e3e00000ac47486422c0	ts_S_087, ts_S_088, ts_C_087, ts_C_088, is_C2_087, is_C2_088
/dev/disk/by-id/scsi-3600605b000a8e3e00000ac4e48c86480	ts_S_089, ts_S_090, ts_C_089, ts_C_090, is_C2_089, is_C2_090
/dev/disk/by-id/scsi-3600605b000a8e3e00000ac53491aa9da	ts_S_091, ts_S_092, ts_C_091, ts_C_092, is_C2_091, is_C2_092
/dev/disk/by-id/scsi-3600605b000a8e3e00000ac584965108e	ts_S_093, ts_S_094, ts_C_093, ts_C_094, is_C2_093, is_C2_094
/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_5030013B	ts_S_095, ts_S_096, ts_C_095, ts_C_096, is_C2_095, is_C2_096
/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000ZT	ts_S_097, ts_S_098, ts_C_097, ts_C_098, is_C2_097, is_C2_098
/dev/disk/by-id/scsi-3600605b000a8e8500004342a2894ba3c	ts_S_099, ts_S_100, ts_C_099, ts_C_100, is_C2_099, is_C2_100
/dev/disk/by-id/scsi-3600605b000a8e8500004342a2896bf89	ts_S_101, ts_S_102, ts_C_101, ts_C_102, is_C2_101, is_C2_102
/dev/disk/by-id/scsi-3600605b000a8e85000002f18d2362b32	ts_S_103, ts_S_104, ts_C_103, ts_C_104, is_C2_103, is_C2_104
/dev/disk/by-id/scsi-3600605b000a8e5a000000679665d57ce	ts_S_105, ts_S_106, ts_C_105, ts_C_106, is_C2_105, is_C2_106
/dev/disk/by-id/scsi-3600605b000a8e5a00000067866578360	ts_S_107, ts_S_108, ts_C_107, ts_C_108, is_C2_107, is_C2_108
/dev/disk/by-id/scsi-360000000000000000000003d03f109970	ts_S_109, ts_S_110, ts_C_109, ts_C_110, is_C2_109, is_C2_110
/dev/disk/by-id/scsi-3600605b000a8e610000007f97d4c3c3b	ts_S_111, ts_S_112, ts_C_111, ts_C_112, is_C2_111, is_C2_112
/dev/disk/by-id/scsi-360000000000000000000003d03f1488ea	ts_S_113, ts_S_114, ts_C_113, ts_C_114, is_C2_113, is_C2_114
/dev/disk/by-id/scsi-360000000000000000000003d03f167512	ts_S_115, ts_S_116, ts_C_115, ts_C_116, is_C2_115, is_C2_116
/dev/disk/by-id/scsi-360000000000000000000003d13f1876c8	ts_S_117, ts_S_118, ts_C_117, ts_C_118, is_C2_117, is_C2_118

/dev/disk/by-id/scsi-360000000000000000000003d13f1a7c7d	ts_S_119, ts_S_120, ts_C_119, ts_C_120, is_C2_119, is_C2_120
/dev/disk/by-id/scsi-360000000000000000000003d13f1c71ed	ts_S_121, ts_S_122, ts_C_121, ts_C_122, is_C2_121, is_C2_122
/dev/disk/by-id/scsi-360000000000000000000003d13f1e720d	ts_S_123, ts_S_124, ts_C_123, ts_C_124, is_C2_123, is_C2_124
/dev/disk/by-id/scsi-360000000000000000000003d13f2067af	ts_S_125, ts_S_126, ts_C_125, ts_C_126, is_C2_125, is_C2_126
/dev/disk/by-id/scsi-360000000000000000000003d13f226973	ts_S_127, ts_S_128, ts_C_127, ts_C_128, is_C2_127, is_C2_128
/dev/disk/by-id/scsi-360000000000000000000003d13f246713	ts_S_129, ts_S_130, ts_C_129, ts_C_130, is_C2_129, is_C2_130
/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000Z8	ts_S_131, ts_S_132, ts_C_131, ts_C_132, is_C2_131, is_C2_132
/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000XE	ts_S_133, ts_S_134, ts_C_133, ts_C_134, is_C2_133, is_C2_134
/dev/disk/by-id/scsi-3600605b000a8e8500004342a289aa9ba	ts_S_135, ts_S_136, ts_C_135, ts_C_136, is_C2_135, is_C2_136
/dev/disk/by-id/scsi-3600605b000a8e85000039ed341d30faa	ts_S_137, ts_S_138, ts_C_137, ts_C_138, is_C2_137, is_C2_138
/dev/disk/by-id/scsi-3600605b000a8e5a00000067866598cca	ts_S_139, ts_S_140, ts_C_139, ts_C_140, is_C2_139, is_C2_140
/dev/disk/by-id/scsi-3600605b000a8e5a00000067966612155	ts_S_141, ts_S_142, ts_C_141, ts_C_142, is_C2_141, is_C2_142
/dev/disk/by-id/scsi-3600605b000a8e5a000000679665f3c6a	ts_S_143, ts_S_144, ts_C_143, ts_C_144, is_C2_143, is_C2_144

Table 4-5: IBM System x3850 X5 SSD Table Space Mapping

4.5. 60-Day Space Calculations

Details of the 60 day space computations along with proof that the database is configured to sustain 8 hours of growth for the dynamic tables (Order, Order-Line, and History) must be disclosed

60-Day Space Computation

All data sizes in MB unless otherwise stated

Warehouses	184,032				
Measured TpmC	2,308,099				
Table	**Rows**	**Table**	**Index**	**5% Space**	**Total Space**
Warehouse	184,032	41	0	2	43
District	1,840,320	225	0	11	236
Item	100,000	10	0	1	11
Stock	18,403,200,000	5,990,904	0	299,545	6,290,449
Customer	5,520,960,000	4,313,466	265,644	228,956	4,808,066
New-Order	1,656,288,000	65,862	0	3,293	69,155
Orders	5,520,960,000	216,129	154,188	0	370,317
Order-Line	55,209,600,000	3,781,816	0	0	3,781,816
History	5,520,960,000	339,750	0	0	339,750
Additional Overhead		3,220,415			3,220,415

Free Space	2,249,923		
Dynamic Space	4,337,695	30 Minute log Computations	
Static Space	14,542,562	Log Written (KB)	155,796,683
Daily Growth	870,443	New-Order Txns	69,242,970
Daily Spread	944,259	Log Written per New-Order (KB)	2.25

Data Storage Requirement

60 Days (MB)	123,424,676
60 Days (GB)	120,532

Log Storage Requirement

8 Hours (GB)	2,377.27

Disk Sizing

	Formatted	SUT		Priced	
Disk Type	**Capacity (GB)**	**# of Disks**	**Capacity (GB)**	**# of Disks**	**Capacity (GB)**
DB DS3400 RAID 0	279.00	432	120,528.00	432	120,528.00
LOG ServeRAID RAID 10	418.16	12	2,508.98	12	2,508.98
DB IBM 2008SAS	186.31	8	1,490.48	8	1,490.48
DB ServeRAID SAS RAID 0	185.33	64	11,861.31	64	11,861.31
Total Capacity (GB)					136,389

5 Clause 5: Performance Metrics and Response Time Related Items

5.1. Response Times

Ninetieth percentile, maximum and average response times must be reported for all transaction types as well as for the Menu response time.

Table 5-2 lists the response times and the ninetieth percentiles for each of the transaction types for the measured system.

Response Times (in seconds)	90th %	Average	Maximum
New Order	0.660	0.301	13.797
Payment	0.660	0.291	10.812
Order-Status	0.660	0.306	7.734
Delivery (interactive)	0.340	0.165	9.328
Delivery (deferred)	0.32	0.13	2.36
Stock-Level	0.660	0.298	7.734
Menu	0.344	0.167	13.156

Response time delay added for emulated components was 0.1 seconds

Table 5-1: Response Times

5.2. Keying and Think Times

The minimum, the average, and the maximum keying and think times must be reported for each transaction type.

Table 5-22 lists the TPC-C keying and think times for the measured system.

Keying/Think Times (in seconds)	Min.	Average	Max.
New Order	18.015/0.00	18.015/12.035	18.047/120.313
Payment	3.015/0.00	3.015/12.036	3.047/120.328
Order-Status	2.015/0.00	2.015/10.036	2.047/100.328
Delivery	2.015/0.00	2.015/5.035	2.047/50.313
Stock-Level	2.015/0.00	2.015/5.035	2.047/50.313

Table 5-2: Think and Keying Times

5.3. Response Time Frequency Distribution

Response time frequency distribution curves must be reported for each transaction type.

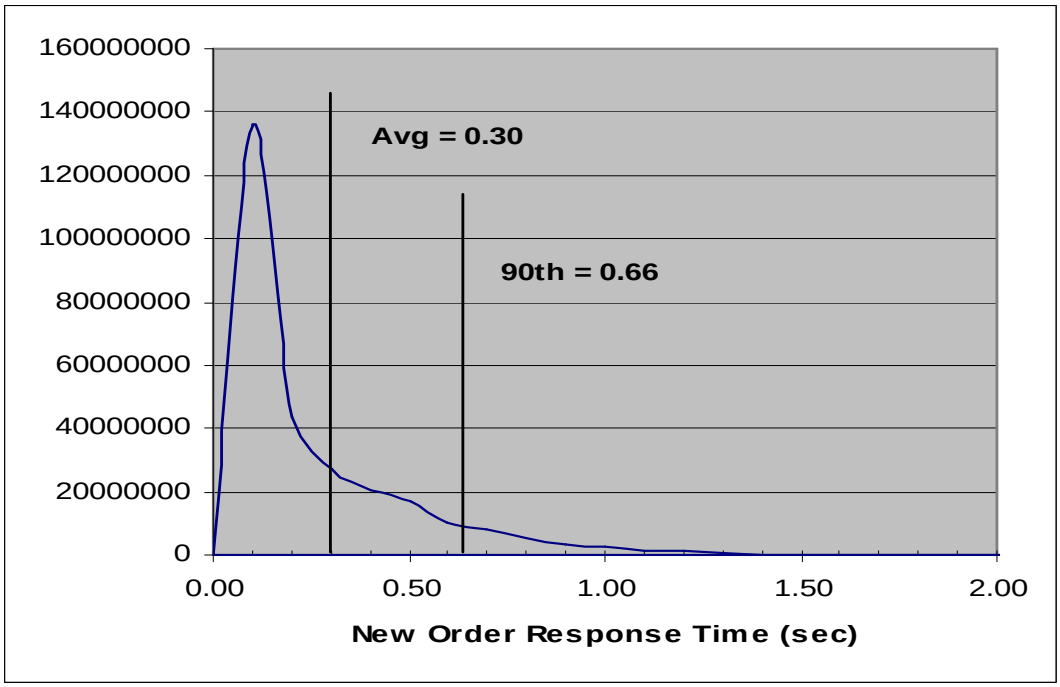


Figure 5-3: New-Order Response Time Distribution

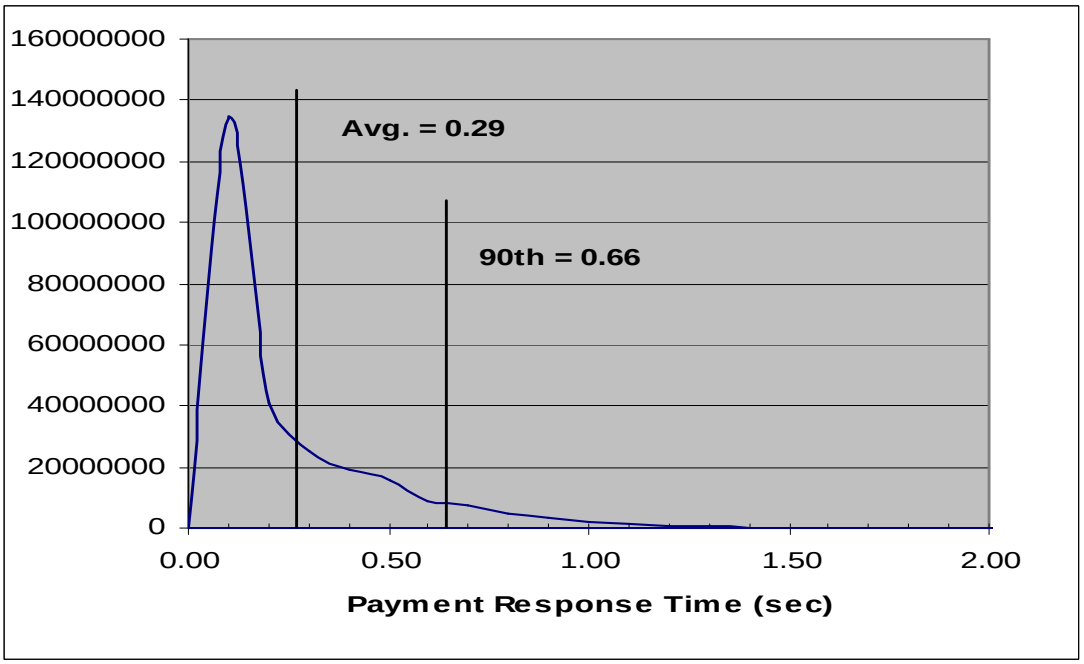


Figure 5-4: Payment Response Time Distribution

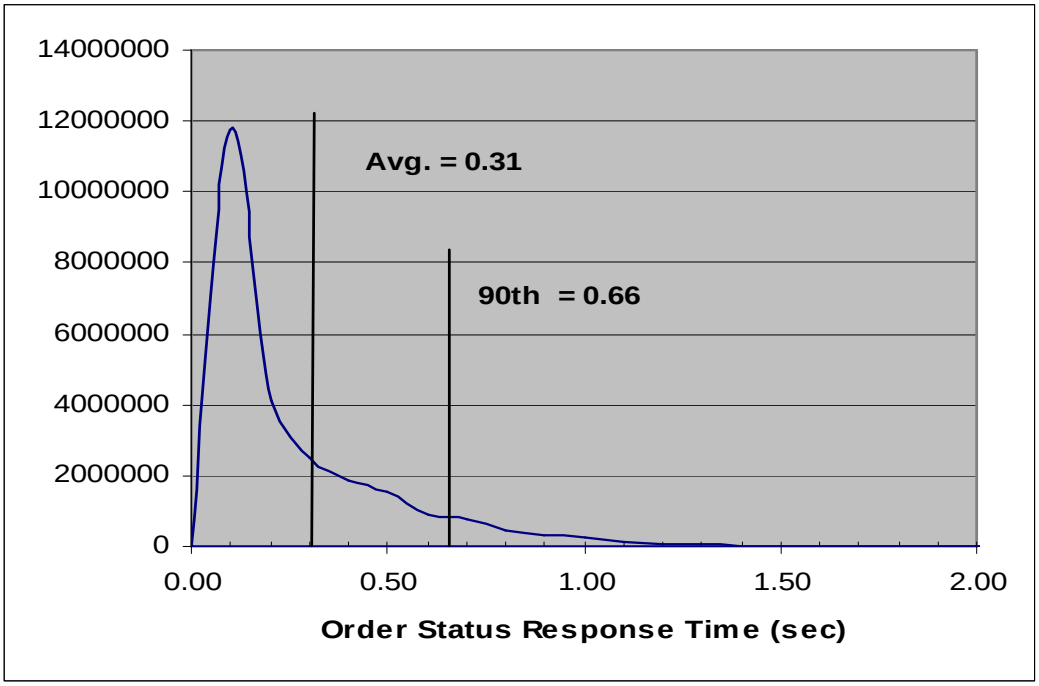


Figure 5-5: Order-Status Response Time Distribution

Figure 5-6: Delivery (Interactive) Response Time Distribution

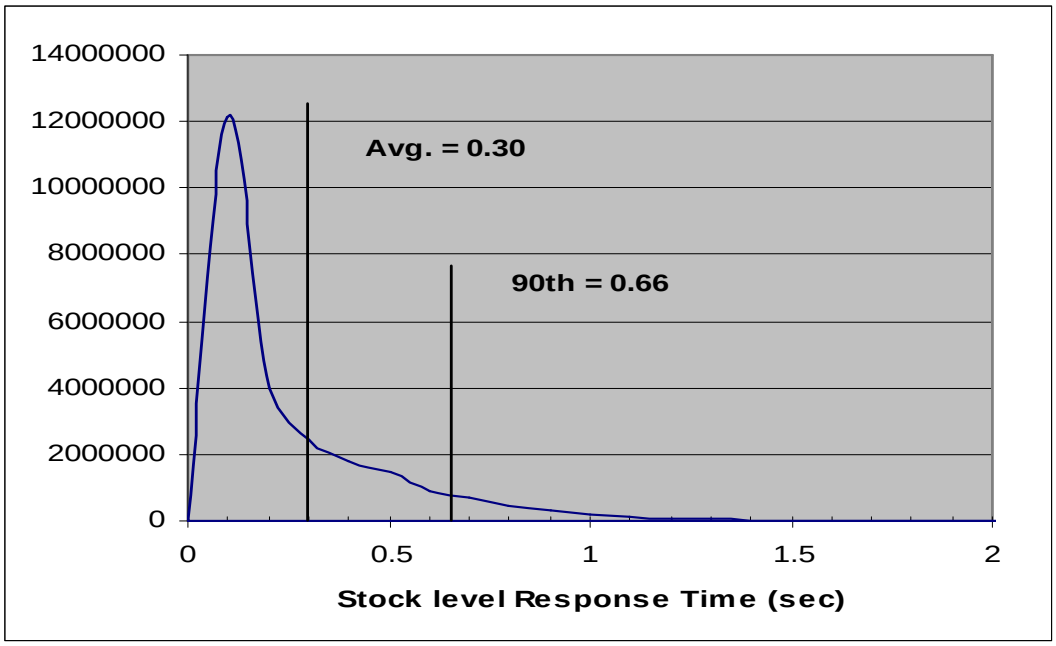


Figure 5-7: Stock Level Response Time Distribution

5.4. Performance Curve for Response Time versus Throughput

The performance curve for response times versus throughput must be reported for the New-Order transaction.

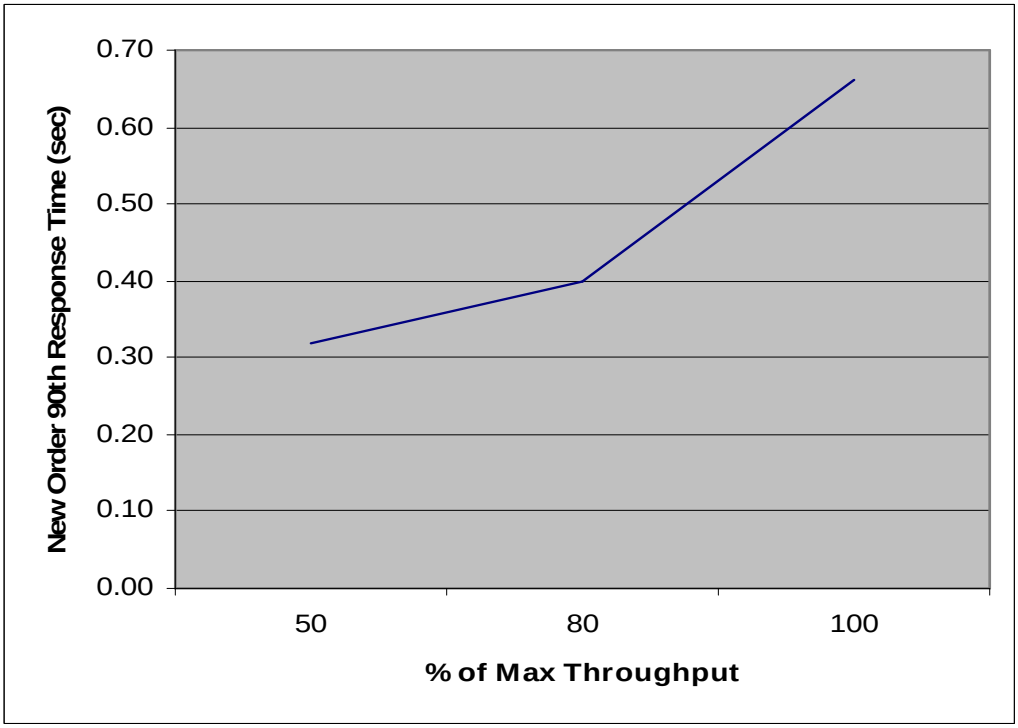


Figure 5-8: New-Order Response Time vs. Throughput

5.5. Think Time Frequency Distribution

A graph of the think time frequency distribution must be reported for the New-Order transaction.

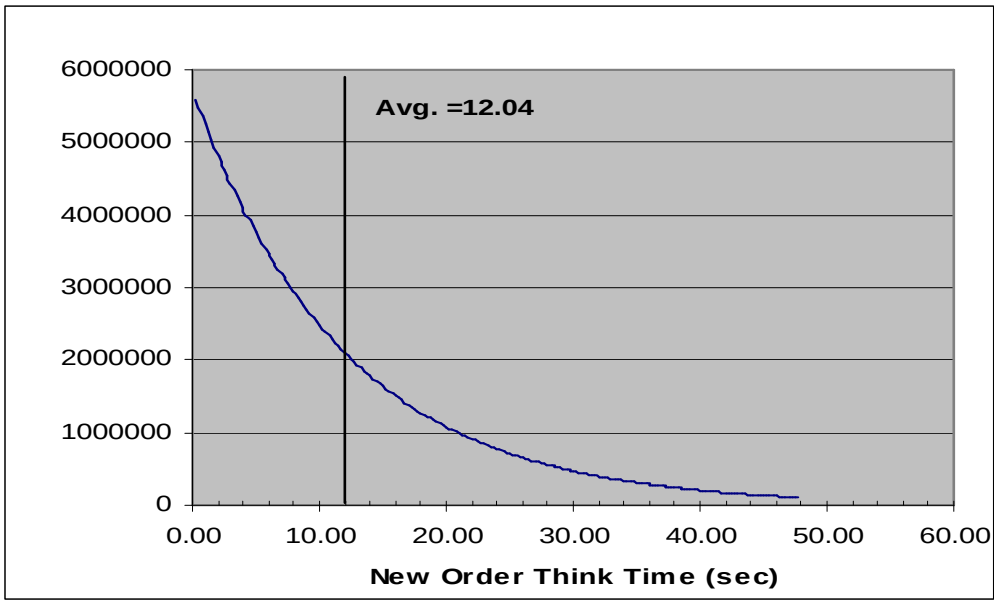


Figure 5-9: New-Order Think Time Distribution

5.6. Throughput versus Elapsed Time

A graph of throughput versus elapsed time must be reported for the New-Order transaction.

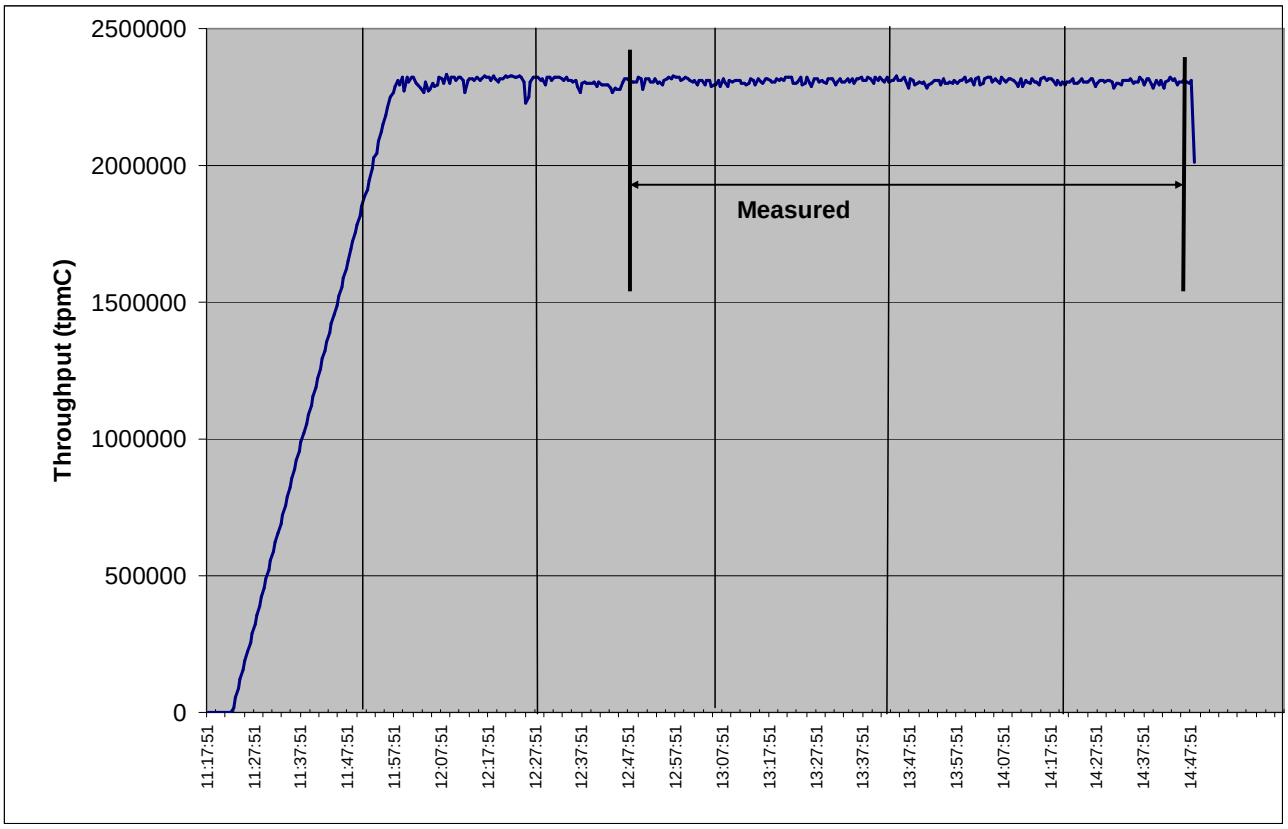


Figure 5-10: New-Order Throughput vs. Elapsed Time

5.7. Steady State Determination

The method used to determine that the SUT had reached a steady state prior to commencing the measurement interval must be described.

All the emulated users were allowed to logon and do transactions. The user ramp-up phase is clearly visible on the graph above. Refer to the Numerical Quantities Summary pages for the rampup time. Figure 5- New-Order throughput versus Elapsed Time graph shows that the system maintained a steady state during the measurement interval

5.8. Work Performed During Steady State

A description of how the work normally performed during a sustained test (for example check pointing, writing redo/undo log records, etc), actually occurred during the measurement interval must be reported.

A 2-hour measurement interval was used to guarantee that all work normally performed during an 8-hour sustained test is included in the reported throughput.

5.8.1. Transaction Flow

Each of the 4 (non-delivery) transactions is serviced by 2 individual programs, Internet Information System 6.0 (IIS) and a Microsoft COM+ 1.0 Queued Component Server, used as the transaction manager (COM+). Both programs are running on the client system:

- The initial HTML 1.0 request is serviced by an ISAPI custom-written handler running on Internet Information System 6.0. IIS is responsible for handling all HTML requests. The web server communicates to the COM+ server through a Microsoft COM+ API interface.
- COM+ communicates with the server system over Ethernet and handles all database operations, using DB2 embedded SQL calls.
  When the COM+ server boots up, it creates a configurable amount of connections to the server (listed in application settings).
  COM+ routes the transaction and balances the load according to the options defined in the Component Services GUI for the COM+ server application and settings in the Windows 2003 registry. The configuration file and registry variables are listed in Appendix B.2.
  At the beginning, each TPC-C user sends a pair of HTML 1.0 requests submitting its unique warehouse and district to the IIS ISAPI handler. Upon successful validation of user's login, IIS the displays an HTML form which encapsulates the TPC-C transaction menu.
  The transaction flow is described below:
  - The TPC-C user requests the transaction type's HTML form and proceeds to generate (fill in) a GET request with the required files for the transaction.
  - IIS accepts the filled in GET request , parses, and validates all values entered by the user.
  - It then proceeds to transmit those values to the COM+ server through an transaction type specific COM+ api interface.
  - The COM+ Pool Manager receives the request and first decides if there is a connection object in the pool available to service it.
    If so, the connection is used to send the transaction request to the Server.
    If no connection is available, the request will enter a COM+ internal queue and will be serviced by the next available connection.
  - Once the connection is available to be used, a COM+ pool thread receives the transaction and calls a TPC-C back end DB2 client api to execute all database operations related to the transaction type. (All the transaction information entered on the HTML form is available in a data structure provided by the ISAPI caller).
  - The transaction is committed and the DB2 back end client returns control back to the COM pool thread.
  - COM pool thread returns control to the ISAPI caller.
    (All transaction results are inside the data structure that the ISAPI caller provided to the COM+ api in the parameter list).
  - ISAPI caller returns control to the "screen application" by doing a PUT request.

5.8.2. Database Transaction

All database operations are performed by the TPC-C back-end programs. The process is described below:

Using embedded SQL calls, the TPC-C back-end program interacts with DB2 Server to perform SQL data manipulations such as update, select, delete and insert, as required by the transaction. After all database operations are performed for a transaction, the transaction is committed.

DB2 Server proceeds to update the database as follows:

When DB2 Server changes a database table with an update, insert, or delete operation, the change is initially made in memory, not on disk. When there is not enough space in the memory buffer to read in or write additional data pages, DB2 Server will make space by flushing some modified pages to disk. Modified pages are also written to disk as part of the “Soft” checkpoint to ensure that no updates remain unflushed for longer than the allowed time. Before a change is made to the database, it is first recorded in the transaction log. This ensures that the database can be recovered completely in the event of a failure. Using the transaction log, transactions that started but did not complete prior to a failure can be undone, and transactions recorded as complete in the transaction log but not yet written to disk can be redone.

5.8.3. Checkpoints

DB2 uses a write-ahead-logging protocol to guarantee recovery. This protocol uses “Soft” checkpoint to write least-recently-used database pages to disk independent of transaction commit. However, enough log information to redo/undo the change to a database pages is committed to disk before the database page itself is written. This protocol therefore renders checkpoint unnecessary for DB2 . For a more detailed description of the general principles of the write-ahead-logging protocol, see the IBM research paper, “ARIES: A Transaction Recovery Method Supporting Fine Granularity

Locking and Partial Rollbacks Using Write-Ahead Logging," by C. Mohan, Database Technology Institute, IBM Almaden Research Center.
(http:// portal.acm.org/citation.cfm?id=128770&coll=portal&dl=ACM&CFID=10343790&CFTOKEN=42047146)

5.9. Measurement Interval

A statement of the duration of the measurement interval for the reported Maximum Qualified Throughput (tpmC) must be included.

A 2-hour measurement interval was used. No connections were lost during the run.

6 Clause 6: SUT, Driver, and Communication Definition Related Items

6.1. RTE Availability

If the RTE is commercially available, then its inputs must be specified. Otherwise, a description must be supplied of what inputs to the RTE had been used.

IBM used an internally developed RTE for these tests. A total of 184,032 warehouses were configured; 183,936 were accessed during the runs. A rampup time of 48 minutes was specified, along with a run time of two hours.

6.2. Functionality and Performance of Emulated Components

It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to that of the priced system.

No components were emulated.

6.3. Network Bandwidth

The bandwidth of the network(s) used in the tested/priced configuration must be disclosed.

The network between the clients and the database server was configured as 1000 MegaBits per second Full Duplex.

6.4. Operator Intervention

If the configuration requires operator intervention, the mechanism and the frequency of this intervention must be disclosed.

No operator intervention is required to sustain the reported throughput during the eight-hour period.

7 Clause 7: Pricing Related Items

7.1. Hardware and Programs Used

A detailed list of the hardware and software used in the priced system must be reported. Each item must have vendor part number, description, and release/revision level, and either general availability status or committed delivery date. If package-pricing is used, contents of the package must be disclosed. Pricing source(s) and effective date(s) must also be reported.

The detailed list of all hardware and programs for the priced configuration is listed in the pricing sheets for each system reported. The prices for all products and features that are provided by IBM are available the same day as product or feature availability.

7.2. Three Year Cost of System Configuration

The total 3-year price of the entire configuration must be reported, including: hardware, software, and maintenance charges. Separate component pricing is recommended. The basis of all discounts used must be disclosed.

The pricing details for this disclosure is contained in the executive summary pages. All 3rd party quotations are included at the end of this report in Appendix D. All prices are based on US list prices.

Discounts are based on US list prices and for similar quantities and configurations. A discount of 53.97% has been applied to specified IBM hardware, software , and services based on the total value and quantities of the components of the configuration.

7.3. Availability Dates

The committed delivery date for general availability (availability date) of products used in the price calculations must be reported. When the priced system includes products with different availability dates, the reported availability date for the priced system must be the date at which all components are committed to be available.

All components of the SUT will be available on May 16, 2011.

7.4. Statement of tpmC and Price/Performance

A statement of the measured tpmC, as well as the respective calculations for 3-year pricing, price/performance (price/tpmC), and the availability date must be disclosed.

System	tpmC	3-year System Cost	$/tpmC	Availability Date
IBM System x3850 X5	2,308,099	$1,393,681 USD	$0.60 USD	May 16, 2011

Please refer to the price list on the Executive Summary page for details.

8 Clause 9: Audit Related Items

If the benchmark has been independently audited, then the auditor's name, address, phone number, and a brief audit summary report indicating compliance must be included in the Full Disclosure Report. A statement should be included, specifying when the complete audit report will become available and who to contact in order to obtain a copy.

The auditor's attestation letter is included in this section of this report:

INFO SIZING

TPC TRANSACTION PROCESSING PERFORMANCE COUNCIL CERTIFIED AUDITOR

Raymond J. Venditti
IBM Linux Performance
11501 Burnet Road
Austin, TX 78758

Berni Schiefer
IBM SB2 Performance
8200 Warden Avenue
Markham, Ontario L6G1C7

May 16, 2011

I verified the TPC Benchmark™ C performance of the following Client Server configuration:

Platform: **IBM System x3850X5** c/s
Operating system: **SLES 11 sp1**
Database Manager: **DB2 9.5**
Transaction Manager: **Microsoft COM+**

The results were:

CPU's Speed	Memory	Disks	New Order 90% Response Time	**tpmC**
Server: IBM System x3850 x5				
4 x Intel Xeon X7560 (2.27GHz)	1,536 GB (8 x 256 MB L2)	432 x 300 GB 15K rpm SAS 12 x 450 GB 15K rpm SAS 8 x 200 GB SATA SSD 64 x 200 GB SAS SSD	0.66 Second	**2,308,099.7**
16 Clients: IBM System x3550 M2 (each with)				
1 x Intel Xeon X3350 QC (2.66 GHz)	1 GB (4 x 6 MB L2)	1 x 73.4 GB SAS	n/a	n/a

In my opinion, these performance results were produced in compliance with the TPC requirements for the benchmark.

125 West Monroe Street • Colorado Springs, CO 80907 • 719-473-7555 • www.sizing.com

The following verification items were given special attention:

- The transactions were correctly implemented
- The database records were the proper size
- The database was properly scaled and populated
- The ACID properties were met
- Input data was generated according to the specified percentages
- The transaction cycle times included the required keying and think times
- The reported response times were correctly measured.
- At least 90% of all delivery transactions met the 80 Second completion time limit
- All 90% response times were under the specified maximums
- The measurement interval was representative of steady state conditions
- The reported measurement interval was 120 minutes
- Write-ahead-logging was active during the measurement interval
- The 60 day storage requirement was correctly computed
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

Disk Substitution - The tested configuration included (432) 73GB 15Krpm SAS drives. These disks were substituted one for one in the priced configuration by (432) 300GB 15Krpm SAS drives. Based on data analysis done for each type of disk, it is my opinion that this substitution is compliant with the benchmark requirements and has no significant effect on performance.

Respectfully Yours,

François Raab, President

125 West Monroe Street • Colorado Springs, CO 80907 • 719-473-7555 • www.sizing.com

9 Appendix A: Client Server Code

9.1. Client/Terminal Handler Code

Makefile.config

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996-2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# Makefile.config - NT/Winx64 Makefile Configuration
#
MAKE=nmake.exe

# Compiler Configuration (MSVC).
# CFLAGS_DEBUG may be set to "-Zi -Od", "-DDEBUGIT" "-Zi -Od -DDEBUGIT" or left
blank
CC=cl.exe
CFLAGS_OS=-DSQLWINT -MD -GS- -DWIN32 -J -Zp8 -DREG_KIT_METHOD
CFLAGS_OUT=/Fo
CFLAGS_DEBUG=

# Linker Configuration (MSVC)
LD_EXEC=link.exe
LD_STORP=link.exe
LDFLAGS_EXEC=
LDFLAGS_SHLIB=/DLL
LDFLAGS_STORP=$(LDFLAGS_SHLIB) /DEF:rpctpcc.def
LDFLAGS_LIB=/LIBPATH:$(TPCC_SQLLIB)\lib /LIBPATH:"C:\MsSDKx64\lib\amd64"
/LIBPATH:"C:\Program Files\Microsoft SDKs\Windows\v6.0A\Lib" /LIBPATH:"C:\Program
Files (x86)\Microsoft Visual Studio 10.0\VC\lib" winmm.lib db2api.lib
LDFLAGS_OUT=/OUT:

# Library Configuration
AR=lib.exe
ARFLAGS=
ARFLAGS_LIB=
ARFLAGS_OUT=/OUT:

# OS Commands
ERASE=del /F
ERASEDIR=rmdir /S
MOVE=MOVE
COPY=COPY

# OS File Extensions & Path Separator
OBJEXT=.obj
LIBEXT=.lib
SHLIBEXT=.dll
BINEXT=.exe
SLASH=\\
CMDSEP=&
```

Src.Cli\Makefile

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# Makefile - Makefile for Src.Cli (RTE/Driver Interface)
#
#
!include $(TPCC_ROOT)/Makefile.config

#
##########################################################################
#
# Preprocessor, Compiler and Linker Flags
#
##########################################################################
#

PRP_OPTS =      PACKAGE \
            ISOLATION RR \
            QUERYOPT 7 \
            EXPLAIN ALL \
            MESSAGES $*.prep.msg \
            LEVEL $(TPCC_VERSION) \
            NOLINEMACRO

INCLUDES =      -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

CFLAGS =        $(CFLAGS_OS) $(INCLUDES) $(CFLAGS_DEBUG) \
            $(UOPTS) -D$(DB2EDITION) -D$(DB2VERSION) -D$(TPCC_SPTYPE)

OBJS =          $(TPCC_ROOT)/Src.Common/tpccdbg$(OBJEXT) \
            $(TPCC_ROOT)/Src.Common/tpccctx$(OBJEXT) \
            tpcccli$(OBJEXT)

LIBS =          tpcccli$(LIBEXT)

#
##########################################################################
#
# User Targets
#
##########################################################################
#

all:    connect $(OBJS) plan $(LIBS) disconnect
        $(AR) $(ARFLAGS) $(ARFLAGS_OUT)tpcccli$(LIBEXT) $(OBJS) $(ARFLAGS_LIB)
        @echo "-----------------------------------------------------"
        @echo "Please copy lval.h, db2tpcc.h, and tpcccli$(LIBEXT) to"
        @echo "a place where they can be #included and linked with the"
        @echo "RTE/driver code."
        @echo "-----------------------------------------------------"

clean:
        - $(ERASE) *.msg *.bnd *.plan *$(OBJEXT) *$(LIBEXT) tpcccli.c

#
##########################################################################
#
# Helper Targets
#
##########################################################################
#

connect:
        - db2 connect to $(TPCC_DBNAME)

disconnect:
        - db2 connect reset
        - db2 terminate

plan:
        - db2exfmt -d $(TPCC_DBNAME) -e $(TPCC_SCHEMA) -s $(TPCC_SCHEMA) -w -
1 -n TPCCCLI -g -# 0 -o TPCCCLI.exfmt.plan
        - db2expln -d $(TPCC_DBNAME) -c $(TPCC_SCHEMA) -p TPCCCLI -s 0 -g -o
TPCCCLI.expln.plan

#
##########################################################################
#
# Build Rules
#
##########################################################################
#

.SUFFIXES:
.SUFFIXES: $(OBJEXT) .c .sqc

tpcccli.c:
        @echo "Prepping $*.sqc"
        db2 prep $*.sqc $(PRP_OPTS)
        db2 grant execute on package TPCCCLI to public

#
##########################################################################
#
# Dependencies
#
##########################################################################
#

# Client Library:
tpcccli$(LIBEXT):       $(OBJS)
```

```
# Source
tpcccli$(OBJEXT):       tpcccli.c

# Headers
tpcccli.c:              $(TPCC_ROOT)/include/db2tpcc.h $(TPCC_ROOT)/include/lval.h
```

Src.Cli\tpcccli.sqc

```
/*******************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*******************************************************************************/

/*
 * tpcccli.sqc - Client/Server code for TPCC
 *
 */

#include <stdlib.h>
#include <errno.h>
#include "db2tpcc.h"
#include "tpccapp.h"
#include "tpccdbg.h"

#include "sqlca.h"
#include "sql.h"

// ----------------------------------------------------------------
// New Order CLIENT
// ----------------------------------------------------------------

static int itemComparison ( const void * a , const void * b )
{
   struct in_items_struct * one = (struct in_items_struct *) a ;
   struct in_items_struct * two = (struct in_items_struct *) b ;

   // Primary comparison key:   I_ID
   // Secondary comparison key: W_ID

   if ( one->s_OL_I_ID != two->s_OL_I_ID )
   {
      return (  one->s_OL_I_ID - two->s_OL_I_ID  ) ;
   }
   else
   {
      return (  one->s_OL_SUPPLY_W_ID - two->s_OL_SUPPLY_W_ID  ) ;
   }
}

int neword_sql (  struct in_neword_struct  * in_neword
                , struct out_neword_struct * neword      )
{
   struct sqlca sqlca ;

   EXEC SQL BEGIN DECLARE SECTION;

      struct vc_new_in
      {
         short len;
         char data[ 262 ] ;
      } * pHostvarInput ;

      struct vc_new_out
      {
         short len;
         char data[ 682 ] ;
      } * pHostvarOutput ;

   EXEC SQL END DECLARE SECTION;

   int clientRc = TRAN_OK ;

   int itemIndex = 0 ;

   // Determine if order is "all-local" or not
   // NOTE: This loop will exit on the iteration *after* finding the last
   // item; this effectively takes care of the 0-based/1-based conversion
   // and we don't have to add one when assigning to s_O_OL_CNT below.
   in_neword->s_all_local  = 1 ;
   for ( itemIndex = 0 ;
         itemIndex < 15 && in_neword->in_item[ itemIndex ].s_OL_I_ID !=
UNUSED_ITEM_ID ;
         itemIndex++
       )
   {
      if ( in_neword->in_item[ itemIndex ].s_OL_SUPPLY_W_ID != in_neword->s_W_ID )
      {
         in_neword->s_all_local = 0 ;
      }
   }

   in_neword->s_O_OL_CNT = itemIndex ;

   // Sort the item list.  Since invalid item IDs = 100001, we will remain
   // compliant with the spec (Section 2.4.2.3 Comment 1.

   qsort( in_neword->in_item, in_neword->s_O_OL_CNT
          ,  sizeof ( in_neword->in_item[ 0 ] )
          , itemComparison
        ) ;

   pHostvarInput      = (struct vc_new_in *) in_neword ;
   pHostvarInput->len = sizeof(struct in_neword_struct) - SPGENERAL_ADJUST ;

   pHostvarOutput      = (struct vc_new_out *) neword;
   pHostvarOutput->len = sizeof(struct out_neword_struct) - SPGENERAL_ADJUST ;

#ifdef DEBUGIT
   new_debug(neword, in_neword, "Client before SP call");
#endif /* DEBUGIT */

#ifdef SWAP_ENDIAN
   for (itemIndex=0; itemIndex<in_neword->s_O_OL_CNT; itemIndex++)
   {
      SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_I_ID);
      SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_SUPPLY_W_ID);
      SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_QUANTITY);
   }
   SWAP_BYTE(in_neword->s_C_ID);
   SWAP_BYTE(in_neword->s_W_ID);
   SWAP_BYTE(in_neword->s_D_ID);
   SWAP_BYTE(in_neword->s_O_OL_CNT);
   SWAP_BYTE(in_neword->s_all_local);
   SWAP_BYTE(in_neword->duplicate_items);
#endif  //SWAP_ENDIAN

   EXEC SQL CALL news ( :*pHostvarInput, :*pHostvarOutput );

#ifdef SWAP_ENDIAN
   SWAP_BYTE(in_neword->s_C_ID);
   SWAP_BYTE(in_neword->s_W_ID);
   SWAP_BYTE(in_neword->s_D_ID);
   SWAP_BYTE(in_neword->s_O_OL_CNT);
   SWAP_BYTE(in_neword->s_all_local);
   SWAP_BYTE(in_neword->duplicate_items);
   for (itemIndex=0; itemIndex<in_neword->s_O_OL_CNT; itemIndex++)
   {
      SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_I_ID);
      SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_SUPPLY_W_ID);
      SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_QUANTITY);
   }

   SWAP_BYTE(neword->s_W_TAX);
   SWAP_BYTE(neword->s_D_TAX);
   SWAP_BYTE(neword->s_C_DISCOUNT);
   SWAP_BYTE(neword->s_total_amount);
   SWAP_BYTE(neword->s_O_ID);
   SWAP_BYTE(neword->s_O_OL_CNT);
   SWAP_BYTE(neword->s_transtatus);
   SWAP_BYTE(neword->deadlocks);
   for (itemIndex=0; itemIndex<in_neword->s_O_OL_CNT; itemIndex++)
   {
      SWAP_BYTE(neword->item[ itemIndex ].s_I_PRICE);
      SWAP_BYTE(neword->item[ itemIndex ].s_OL_AMOUNT);
      SWAP_BYTE(neword->item[ itemIndex ].s_S_QUANTITY);
   }
#endif  //SWAP_ENDIAN

   if ( sqlca.sqlcode == 0 )
   {
      float wtax = neword->s_W_TAX ;
      float dtax = neword->s_D_TAX ;
      float cdisc = neword->s_C_DISCOUNT ;
      float factor = (1.0 - cdisc) * (1.0 + wtax + dtax) ;

      // Compute order total

      neword->s_total_amount = 0 ;

      for ( itemIndex = 0 ;
            itemIndex < in_neword->s_O_OL_CNT ; // from input , not output
            itemIndex++
          )
      {
         if ( neword->item[ itemIndex ].s_I_PRICE > 0 )  // A zero price signifies a bad item
         {
            neword->item[ itemIndex ].s_OL_AMOUNT =  neword-
>item[ itemIndex ].s_I_PRICE *
                                                  in_neword->in_item[ itemIndex ].s_OL_QUANTITY ;  //
reference input value

            neword->s_total_amount += neword->item[ itemIndex ].s_OL_AMOUNT ;

         }
```

```
      }

      neword->s_total_amount *= factor;
    }
    else
    {
      sqlerror( NEWORD_SQL, "NEW", __FILE__, __LINE__, &sqlca) ;
      neword->s_transtatus = FATAL_SQLERROR ;
      clientRc = FATAL_SQLERROR ;
    }

#ifdef DEBUGIT
    new_debug(neword, in_neword, "Client after SP call");
#endif /* DEBUGIT */

    if (neword->s_transtatus <= FATAL_SQLERROR)
    {
      new_debug(neword, in_neword, "NEW failed");
      clientRc = FATAL_SQLERROR ;
    }

    if (neword->s_transtatus == INVALID_ITEM)
    {
      clientRc = INVALID_ITEM ;
    }

    return ( clientRc ) ;
}

// ---------------------------------------------------------------
// Payment CLIENT
// ---------------------------------------------------------------

int payment_sql  (   struct in_payment_struct * in_payment
                   , struct out_payment_struct * payment    )
{
    struct sqlca sqlca ;

    int clientRc = TRAN_OK ;

    EXEC SQL BEGIN DECLARE SECTION;

      // Inputs

      float h_amount ;
      sqlint32 in_c_id ;

      struct s_data_type { short len ; char data[ 16 ] ; } c_last_input ;

      sqlint32 w_id ;
      sqlint32 c_w_id ;
      short    d_id ;
      short    c_d_id ;

      // Outputs

      sqlint32 c_id ;

      double  c_credit_lim ;
      float   c_discount ;
      double  c_balance ;

      char    w_street_1 [ 20 ] , w_street_2 [ 20 ] ;
      char    w_city [ 20 ] ,     w_state [ 2 ] ,      w_zip [ 9 ] ;

      char    d_street_1 [ 20 ] , d_street_2 [ 20 ] ,  d_city [ 20 ] ;
      char    d_state [ 2 ] ,     d_zip [ 9 ] ,        c_first [ 16 ] ;

      char    c_last [ 16 ] ;

      char    c_middle [ 2 ]    , c_street_1 [ 20 ] ;
      char    c_street_2 [ 20 ] , c_city [ 20 ] ,      c_state [ 2 ] ;
      char    c_zip [ 9 ] ,      c_phone [ 16 ] ;

      char    c_credit [ 2 ] ;

      char    c_since [27];

      char    c_data [ 200 ] ;
      short   c_data_indicator = 0 ;

      char    h_date [27];

      struct c_data_prefix_c_last_type { short len ; char data[ 28 ] ; } c_data_prefix_c_last ;
      struct c_data_prefix_c_id_type   { short len ; char data[ 34 ] ; } c_data_prefix_c_id ;


    EXEC SQL END DECLARE SECTION;


    // Input redirects

    #define h_amount          in_payment->s_H_AMOUNT
    #define in_c_id           in_payment->s_C_ID

    #define w_id              in_payment->s_W_ID
    #define d_id              in_payment->s_D_ID

    #define c_d_id            in_payment->s_C_D_ID
    #define c_w_id            in_payment->s_C_W_ID

    // Output redirects

    #define c_credit_lim      payment->s_C_CREDIT_LIM
    #define c_discount        payment->s_C_DISCOUNT
    #define c_balance         payment->s_C_BALANCE

    #define c_id              payment->s_C_ID
    #define c_last            payment->s_C_LAST

    #define c_first           payment->s_C_FIRST
    #define c_middle          payment->s_C_MIDDLE
    #define c_street_1        payment->s_C_STREET_1
    #define c_street_2        payment->s_C_STREET_2
    #define c_city            payment->s_C_CITY
    #define c_state           payment->s_C_STATE
    #define c_zip             payment->s_C_ZIP
    #define c_phone           payment->s_C_PHONE
    #define c_credit          payment->s_C_CREDIT
    #define c_since           payment->s_C_SINCE_time
    #define c_data            payment->s_C_DATA

    #define w_street_1        payment->s_W_STREET_1
    #define w_street_2        payment->s_W_STREET_2
    #define w_city            payment->s_W_CITY
    #define w_state           payment->s_W_STATE
    #define w_zip             payment->s_W_ZIP

    #define d_street_1        payment->s_D_STREET_1
    #define d_street_2        payment->s_D_STREET_2
    #define d_city            payment->s_D_CITY
    #define d_state           payment->s_D_STATE
    #define d_zip             payment->s_D_ZIP

    #define h_date            payment->s_H_DATE_time

    payment->deadlocks = -1 ;
    payment->s_transtatus = TRAN_OK ;

    // Austin's Screen application doesn't fill in C_W_ID and C_D_ID unless
    // it's a remote payment transaction.  Since we expect these to be filled
    // in for all cases, we need to fill them in if they are not already.
    if (c_w_id == 0) { c_w_id = w_id; }
    if (c_d_id == 0) { c_d_id = d_id; }

#ifdef DEBUGIT
    pay_debug(payment, in_payment, "Client before SQL call");
#endif /* DEBUGIT */

    // Create c_data_prefix strings and copy some elements from
    // in -> out struct outside of retry_tran loop

    if ( in_c_id == 0 )
    {

      c_data_prefix_c_last.len = sprintf( c_data_prefix_c_last.data,
" %2.2d %6.6d %2.2d %6.6d %06.2f",  c_d_id , c_w_id , d_id , w_id , h_amount ) ;

      // Setup the input c_last varchar
      c_last_input.len = strlen( in_payment->s_C_LAST ) ;
      memcpy( c_last_input.data , in_payment->s_C_LAST , c_last_input.len ) ;

      // Copy to the output structure
      memcpy( payment->s_C_LAST , in_payment->s_C_LAST, sizeof( payment-
>s_C_LAST ) ) ;

    } else {

      // Copy c_id to the output structure
      c_id = in_c_id ;


      c_data_prefix_c_id.len = sprintf( c_data_prefix_c_id.data,
" %5.5d %2.2d %6.6d %2.2d %6.6d %06.2f", c_id , c_d_id , c_w_id , d_id , w_id ,
h_amount) ;

    }

retry_tran:

    payment->deadlocks ++ ;

    if ( in_c_id == 0 )
    {
        EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

          SELECT   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
                 , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
                 , C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
                 , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT,
C_CREDIT_LIM
                 , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE

            INTO   :w_street_1 , :w_street_2 , :w_city , :w_state , :w_zip
```

```
            , :d_street_1 , :d_street_2 , :d_city , :d_state , :d_zip
            , :c_id , :c_first , :c_middle ,   :c_street_1 , :c_street_2 , :c_city , :c_state
            , :c_zip , :c_phone , :c_since , :c_credit , :c_credit_lim
            , :c_discount , :c_balance, :c_data :c_data_indicator, :h_date
        FROM TABLE ( PAY_C_LAST(   :w_id
                                  , :d_id
                                  , :c_w_id
                                  , :c_d_id
                                  , :c_last_input
                                  , CAST(:h_amount AS DECIMAL(6,2))
                                  , :c_data_prefix_c_last
                                 )

                ) AS T (   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
                         , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
                         , C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
                         , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT,
C_CREDIT_LIM
                         , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE
                        )
        ;

        COMMIT ;

      END COMPOUND ;

  }
  else
  {
      EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

        SELECT   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
            , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
            , C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
            , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT,
C_CREDIT_LIM
            , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE

        INTO   :w_street_1 , :w_street_2 , :w_city , :w_state , :w_zip
            , :d_street_1 , :d_street_2 , :d_city , :d_state , :d_zip
            , :c_last, :c_first , :c_middle , :c_street_1 , :c_street_2 , :c_city , :c_state
            , :c_zip , :c_phone , :c_since , :c_credit , :c_credit_lim
            , :c_discount , :c_balance, :c_data :c_data_indicator, :h_date
        FROM TABLE ( PAY_C_ID(   :w_id
                                , :d_id
                                , :c_w_id
                                , :c_d_id
                                , :in_c_id
                                , CAST(:h_amount AS DECIMAL(6,2))
                                , :c_data_prefix_c_id
                               )

                ) AS T(   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
                        , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
                        , C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
                        , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT,
C_CREDIT_LIM
                        , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE
                       )
        ;

        COMMIT ;

      END COMPOUND ;

  }

#ifdef DEBUGIT
  pay_debug(payment, in_payment, "Client after SQL call");
#endif /* DEBUGIT */

  if ( sqlca.sqlcode != 0 )
  {
    DLCHK( retry_tran ) ;

    sqlerror( PAYMENT_SQL , "PAY" , __FILE__, __LINE__ , &sqlca) ;
    payment->s_transtatus = FATAL_SQLERROR ;
    clientRc = FATAL_SQLERROR ;

    pay_debug( payment, in_payment, "PAY failed" ) ;

    EXEC SQL ROLLBACK WORK ;

    if ( sqlca.sqlcode != 0 )
    {
      sqlerror( PAYMENT_SQL, "ROLLBACK FAILED", __FILE__, __LINE__, &sqlca ) ;
    }
  }

  return ( clientRc ) ;
}

// ---------------------------------------------------------------
// Order Status CLIENT
// ---------------------------------------------------------------

int ordstat_sql  (  struct in_ordstat_struct * in_ordstat
                  , struct out_ordstat_struct * ordstat)
{
  struct sqlca sqlca ;

  EXEC SQL BEGIN DECLARE SECTION;

    struct vc_ord_in
    {
      short len ;
      char data[ 42 ] ;
    } * in_ord ;

    struct vc_ord_out
    {
      short len ;
      char data[ 822 ] ;
    } * out_ord ;

  EXEC SQL END DECLARE SECTION;

  int clientRc = TRAN_OK ;
  int itemIndex = 0 ;

  in_ord      = (struct vc_ord_in *) in_ordstat ;
  in_ord->len = sizeof(struct in_ordstat_struct) - SPGENERAL_ADJUST ;

  out_ord      = (struct vc_ord_out *) ordstat ;
  out_ord->len = sizeof(struct out_ordstat_struct) - SPGENERAL_ADJUST ;

#ifdef DEBUGIT
  ord_debug(ordstat, in_ordstat, "Client before SP call");
#endif /* DEBUGIT */

#ifdef SWAP_ENDIAN
  SWAP_BYTE(in_ordstat->s_C_ID);
  SWAP_BYTE(in_ordstat->s_W_ID);
  SWAP_BYTE(in_ordstat->s_D_ID);
#endif  //SWAP_ENDIAN

  EXEC SQL CALL ords ( :*in_ord, :*out_ord ) ;

#ifdef SWAP_ENDIAN
  SWAP_BYTE(in_ordstat->s_C_ID);
  SWAP_BYTE(in_ordstat->s_W_ID);
  SWAP_BYTE(in_ordstat->s_D_ID);

  SWAP_BYTE(ordstat->s_C_BALANCE);
  SWAP_BYTE(ordstat->s_C_ID);
  SWAP_BYTE(ordstat->s_O_ID);
  SWAP_BYTE(ordstat->s_O_CARRIER_ID);
  SWAP_BYTE(ordstat->s_ol_cnt);
  SWAP_BYTE(ordstat->s_transtatus);
  SWAP_BYTE(ordstat->deadlocks);
  for (itemIndex=0; itemIndex<ordstat->s_ol_cnt; itemIndex++)
  {
    SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_AMOUNT);
    SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_I_ID);
    SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_SUPPLY_W_ID);
    SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_QUANTITY);
  }
#endif  //SWAP_ENDIAN

  if ( sqlca.sqlcode == 0 )
  {
    // Propogate the field we already knew into the output structure
    // 60% of the time, we already new c_last (input c_id is 0)

    if ( in_ordstat->s_C_ID == 0 )
    {
      memcpy( ordstat->s_C_LAST , in_ordstat->s_C_LAST, sizeof( ordstat-
>s_C_LAST ) ) ;
    }
    else
    {
      ordstat->s_C_ID = in_ordstat->s_C_ID ;
    }
  }
  else
  {
    sqlerror( ORDSTAT_SQL, "ORD", __FILE__, __LINE__, &sqlca) ;
    ordstat->s_transtatus = FATAL_SQLERROR ;
    clientRc = FATAL_SQLERROR ;
  }

#ifdef DEBUGIT
  ord_debug(ordstat, in_ordstat, "Client after SP call");
#endif /* DEBUGIT */

  if ( ordstat->s_transtatus <= FATAL_SQLERROR )
  {
    ord_debug(ordstat, in_ordstat, "ORD failed");
    clientRc = FATAL_SQLERROR ;
  }

  return ( clientRc ) ;
}
```

```
// ------------------------------------------------------------
// Delivery CLIENT
// ------------------------------------------------------------

int delivery_sql (   struct in_delivery_struct  * in_delivery
                   , struct out_delivery_struct * delivery      )
{
   struct sqlca sqlca ;

   EXEC SQL BEGIN DECLARE SECTION;

      struct vc_del_in
      {
         short len ;
         char data[ 14 ] ;
      } * in_del ;

      struct vc_del_out
      {
         short len;
         char data[ 50 ] ;
      } * out_del ;

   EXEC SQL END DECLARE SECTION;

   int clientRc = TRAN_OK ;
   int orderIndex = 0 ;

   in_del      = (struct vc_del_in *) in_delivery ;
   in_del->len = sizeof(struct in_delivery_struct) - SPGENERAL_ADJUST;

   out_del      = (struct vc_del_out *) delivery ;
   out_del->len = sizeof(struct out_delivery_struct) - SPGENERAL_ADJUST;

#ifdef DEBUGIT
   del_debug(delivery, in_delivery, "Client before SP call");
#endif /* DEBUGIT */

#ifdef SWAP_ENDIAN
   SWAP_BYTE(in_delivery->s_W_ID);
   SWAP_BYTE(in_delivery->s_O_CARRIER_ID);
#endif  //SWAP_ENDIAN

   EXEC SQL CALL dels ( :*in_del, :*out_del ) ;

#ifdef SWAP_ENDIAN
   SWAP_BYTE(in_delivery->s_W_ID);
   SWAP_BYTE(in_delivery->s_O_CARRIER_ID);

   for (orderIndex=0; orderIndex<10; orderIndex++) {
     SWAP_BYTE(delivery->s_O_ID[ orderIndex ]);
   }
   SWAP_BYTE(delivery->s_transtatus);
   SWAP_BYTE(delivery->deadlocks);
#endif  //SWAP_ENDIAN

#ifdef DEBUGIT
   del_debug(delivery, in_delivery, "Client after SP call");
#endif /* DEBUGIT */

   if ( sqlca.sqlcode != 0 )
   {
      sqlerror( DELIVERY_SQL, "DEL", __FILE__, __LINE__, &sqlca) ;
      delivery->s_transtatus = FATAL_SQLERROR ;
      clientRc = FATAL_SQLERROR ;
   }

   if ( delivery->s_transtatus <= FATAL_SQLERROR )
   {
      del_debug(delivery, in_delivery, "DEL failed");
      clientRc = FATAL_SQLERROR ;
   }

   return ( clientRc ) ;
}

// ------------------------------------------------------------
// Stock CLIENT
// ------------------------------------------------------------

#undef w_id
#undef d_id

int stocklev_sql  (   struct in_stocklev_struct  * in_stocklev
                    , struct out_stocklev_struct * stocklev     )
{
   struct sqlca sqlca ;

   int clientRc = TRAN_OK ;

   EXEC SQL BEGIN DECLARE SECTION;

      // input

      sqlint32       threshold ;

      // output

      sqlint32    low_stock ;

   EXEC SQL END DECLARE SECTION;

   #define w_id      in_stocklev->s_W_ID
   #define d_id      in_stocklev->s_D_ID
   #define threshold in_stocklev->s_threshold
   #define low_stock stocklev->s_low_stock

   stocklev->deadlocks = -1 ;
   stocklev->s_transtatus = TRAN_OK ;

#ifdef DEBUGIT
   stk_debug(stocklev, in_stocklev, "Client before SQL call");
#endif /* DEBUGIT */

retry_tran:

   stocklev->deadlocks ++ ;

   EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

      SELECT COUNT( S_I_ID ) INTO :low_stock

        FROM ( SELECT DISTINCT S_I_ID

                FROM ORDER_LINE , STOCK , DISTRICT

               WHERE D_W_ID = :w_id
                 AND D_ID = :d_id
                 AND OL_O_ID < d_next_o_id
                 AND OL_O_ID >= ( d_next_o_id - 20 )
                 AND OL_W_ID = D_W_ID
                 AND OL_D_ID = D_ID
                 AND S_I_ID = OL_I_ID
                 AND S_W_ID = OL_W_ID
                 AND S_QUANTITY < :threshold

             ) OLS

        WITH CS
      ;

      COMMIT ;

   END COMPOUND ;

#ifdef DEBUGIT
   stk_debug(stocklev, in_stocklev, "Client after SQL call");
#endif /* DEBUGIT */

   if ( sqlca.sqlcode != 0 )
   {
      DLCHK( retry_tran ) ;

      sqlerror( STOCKLEV_SQL , "STK" , __FILE__, __LINE__ , &sqlca);
      stocklev->s_transtatus = FATAL_SQLERROR ;
      clientRc = FATAL_SQLERROR ;

      stk_debug( stocklev, in_stocklev, "STK failed" ) ;

      EXEC SQL ROLLBACK WORK ;

      if ( sqlca.sqlcode != 0 )
      {
         sqlerror( STOCKLEV_SQL, "ROLLBACK FAILED", __FILE__, __LINE__, &sqlca ) ;
      }
   }

   return ( clientRc ) ;
}
```

Src.Common\Makefile

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# Makefile - Makefile for Src.Common
#
#
```

```
!include $(TPCC_ROOT)/Makefile.config

#
##########################################################################
#
# Preprocessor, Compiler and LInker Flags
#
##########################################################################
#

PRP_OPTS =      PACKAGE \
                OPTLEVEL 1 \
                ISOLATION RR \
                MESSAGES $*.prep.msg \
                LEVEL $(TPCC_VERSION) \
                NOLINEMACRO

INCLUDES =      -I$(TPCC_SQLLIB)$(SLASH)include -I$(TPCC_ROOT)$(SLASH)include

CFLAGS =        $(CFLAGS_OS) $(CFLAGS_DEBUG) $(INCLUDES) \
                -DSQLA_NOLINES -D$(DB2EDITION) -D$(DB2VERSION) \
                -D$(TPCC_SPTYPE)

UTIL_OBJ_DBG =  tpccdbg$(OBJEXT)
UTIL_OBJ_GEN =  tpccmisc$(OBJEXT)
UTIL_OBJ_DB2 =  tpccctx$(OBJEXT)

#
##########################################################################
#
# User Targets
#
##########################################################################
#

all:    $(UTIL_OBJ_DBG) $(UTIL_OBJ_GEN) connect $(UTIL_OBJ_DB2) disconnect

dbgen:  $(UTIL_OBJ_GEN)

clean:
        - $(ERASE) *$(OBJEXT) *.bnd *.msg tpccctx.c

#
##########################################################################
#
# Helper Targets
#
##########################################################################
#

connect:
        - db2 connect to $(TPCC_DBNAME)

disconnect:
        - db2 connect reset
        - db2 terminate

#
##########################################################################
#
# Build Rules
#
##########################################################################
#

.SUFFIXES:
.SUFFIXES: $(OBJEXT) .c .sqc

.sqc.c:
        @echo "Prepping $*.sqc"
        db2 prep $*.sqc $(PRP_OPTS)
        db2 grant execute on package TPCCCTX to public

#
##########################################################################
#
# Dependencies
#
##########################################################################
#

# Source
tpccdbg$(OBJEXT):       tpccdbg.c
tpccctx$(OBJEXT):       tpccctx.c
tpccmisc$(OBJEXT):      tpccmisc.c

# Headers
tpccdbg.c:      $(TPCC_ROOT)/include/db2tpcc.h
```

Src.Common\tpccctx.sqc

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 *
 * tpccctx.sqc - TPCC context code
 *
 */

#include <string.h>
#include <sqlutil.h>
#include "db2tpcc.h"
#include "tpccdbg.h"

int connect_to_TM(char *in_dbname);
int connect_to_TM_auth(char *in_dbname, char *in_username, char *in_password);
int disconnect_from_TM(void);
int create_context();
int destroy_context();
int attach_context(void*);
int detach_context(void*);
int get_context(void**);

int connect_to_TM(char *in_dbname)
{
  return connect_to_TM_auth(in_dbname, "", "");
}

int connect_to_TM_auth(char *in_dbname, char *in_username, char *in_password)
{
   SQL_STRUCTURE sqlca  sqlca;
   int ConnectSQLCODE = 0;

   EXEC SQL BEGIN DECLARE SECTION;
   char dbname[9];
   char username[129];
   char password[15];
   EXEC SQL END DECLARE SECTION;

   SQLCODE = create_context();
   if (SQLCODE != 0) { return SQLCODE; }

   /* Copy 9 characters - 8 for dbname, 1 for NULL */
   strncpy(dbname,in_dbname,9);
   if (strcmp(in_username,"") == 0)
   {
      EXEC SQL CONNECT TO :dbname IN SHARE MODE;
   } else {
      strncpy(username,in_username,128);
      strncpy(password,in_password,14);
      EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username
USING :password;
   }

   ConnectSQLCODE = SQLCODE;
   if (ConnectSQLCODE != 0)
   {
      sqlerror( CLIENT_SQL, "CONNECT", __FILE__, __LINE__, &sqlca);

      SQLCODE = destroy_context();
      if (SQLCODE != 0) { return SQLCODE; }

      return ConnectSQLCODE;
   }

   return 0;
}

int disconnect_from_TM(void)
{
   SQL_STRUCTURE sqlca  sqlca;
   int DisconnectSQLCODE = 0;

   EXEC SQL CONNECT RESET;

   DisconnectSQLCODE = SQLCODE;
   if (DisconnectSQLCODE != 0) {
      sqlerror( CLIENT_SQL, "DISCONNECT", __FILE__, __LINE__, &sqlca);
   }

   SQLCODE = destroy_context();
   if (SQLCODE != 0) { return SQLCODE; }

   if (DisconnectSQLCODE) {
      return DisconnectSQLCODE;
   }
   return 0;
}

int create_context(void)
{
```

```
   SQL_STRUCTURE sqlca sqlca;
   void *ctx;

   sqleSetTypeCtx(SQL_CTX_MULTI_MANUAL);
   sqleBeginCtx(&ctx, SQL_CTX_BEGIN_ALL, NULL, &sqlca);

   if (SQLCODE != 0) {
      sqlerror( CLIENT_SQL, "CREATE", __FILE__, __LINE__, &sqlca);
      return SQLCODE;
   }

   return 0;
}

int attach_context(void *ctx)
{
   SQL_STRUCTURE sqlca sqlca;

   sqleAttachToCtx(ctx, NULL, &sqlca);

   if (SQLCODE != 0) {
      sqlerror( CLIENT_SQL, "ATTACH", __FILE__, __LINE__, &sqlca);
      return SQLCODE;
   }

   return 0;
}

int detach_context(void *ctx)
{
   SQL_STRUCTURE sqlca sqlca;

   sqleDetachFromCtx(ctx, NULL, &sqlca);

   if (SQLCODE != 0) {
      sqlerror( CLIENT_SQL, "DETACH", __FILE__, __LINE__, &sqlca);
      return SQLCODE;
   }

   return 0;
}

int destroy_context(void)
{
   SQL_STRUCTURE sqlca sqlca;
   void *ctx;

   SQLCODE = get_context(&ctx);
   if (SQLCODE) { return SQLCODE; }

   sqleEndCtx(&ctx, SQL_CTX_END_ALL, NULL, &sqlca);

   if (SQLCODE != 0) {
      sqlerror( CLIENT_SQL, "DESTROY", __FILE__, __LINE__, &sqlca);
      return SQLCODE;
   }

   return 0;
}

int get_context(void **ctx)
{
   SQL_STRUCTURE sqlca sqlca;

   sqleGetCurrentCtx(ctx, NULL, &sqlca);

   if (SQLCODE != 0) {
      sqlerror( CLIENT_SQL, "GETCTX", __FILE__, __LINE__, &sqlca);
      return SQLCODE;
   }

   return 0;
}
```

Src.Common\tpccdbg.c

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tccdbg.c - Debugging Routines
 *
 */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include <time.h>

#include "sqlca.h"
#include "sql.h"
#include "db2tpcc.h"
#include "tpccdbg.h"

#define DEBUG_FILENAME_SZ 128
#define DEBUG_PATH_SIZE 128

void    del_print();
void    new_print();
void    ord_print();
void    pay_print();
void    stk_print();

void current_tmstmp(char *buf);

static int debugInit = 0;
static char debugPath[DEBUG_PATH_SIZE] = "";

/*-----------------------------------------------------------------*/
/*       InitializeDebug                                           */
/*-----------------------------------------------------------------*/
__inline void InitializeDebug(void) {
   if (debugInit == 0) {
      char *p = getenv("TPCC_DEBUGDIR");
      if (p) {
         strncpy(debugPath, p, DEBUG_PATH_SIZE);
      } else {
         strcpy(debugPath, "C:\\temp");
      }
      strcat(debugPath, "\\");
   }
   debugInit = 1;
}

/*-----------------------------------------------------------------*/
/*       sqlerror                                                  */
/*-----------------------------------------------------------------*/
void sqlerror(int tranType, char *msg, char *file, int line, SQL_STRUCTURE sqlca *psqlca)
{
   FILE *err_fp = NULL;
   char  err_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];
   char  tranName[16];
   int j,k;
   char  timeStamp[27];
   char  errStr[512] = "";

   InitializeDebug();
   strncpy(err_fn, debugPath, DEBUG_PATH_SIZE);
   current_tmstmp(&timeStamp[0]);
   timeStamp[19] = (char)NULL;

   switch(tranType)
   {
      case NEWORD_SQL:
         // sprintf(err_fn, "%d.err.out", getpid());
         strcat(err_fn, "new.err.out");
         strcpy(tranName, "NEW_ORDER");
         break;

      case DELIVERY_SQL:
         // sprintf(err_fn, "%d.err.out", getpid());
         strcat(err_fn, "del.err.out");
         strcpy(tranName, "DELIVERY");
         break;

      case PAYMENT_SQL:
         // sprintf(err_fn, "%d.err.out", getpid());
         strcat(err_fn, "pay.err.out");
         strcpy(tranName, "PAYMENT");
         break;

      case ORDSTAT_SQL:
         // sprintf(err_fn, "%d.err.out", getpid());
         strcat(err_fn, "ord.err.out");
         strcpy(tranName, "ORDER_STAT");
         break;

      case STOCKLEV_SQL:
         //sprintf(err_fn, "%d.err.out", getpid());
         strcat(err_fn, "stk.err.out");
         strcpy(tranName, "STOCK_LVL");
         break;

      case 0:
         strcat(err_fn, "cli.err.out");
         strcpy(tranName, "CLIENT");
         break;

      default:
         return;
```

```
  }

  /* Generate Formatted Error Message */
  sqlaintp(errStr, 512, 78, psqlca);

  if ((err_fp = fopen(err_fn, "a+")) == NULL)
  {
    return;
  }

  fprintf(err_fp, "----------------------------------------\n");
  fprintf(err_fp, "Transaction: %s (%s)\n", tranName, msg);
  fprintf(err_fp, "FILE %s (%u)\n", file, line);
  fprintf(err_fp, "SQLCODE %d ", psqlca->sqlcode);
  fprintf(err_fp, "TIME %s\n", timeStamp);
  fprintf(err_fp, "----------------------------------------\n");
  fprintf(err_fp, "%s", errStr);
  fprintf(err_fp, "----------------------------------------\n");

  if (psqlca->sqlerrmc[0] != ' ' || psqlca->sqlerrmc[1] != ' ')
  {
    fprintf(err_fp, "slerrmc:  ");

    for(j = 0; j < 5; j++)
    {
      for(k = 0; k < 16; k++) {
        int pos = j * 16 + k;
        if (pos < 70) fprintf(err_fp, "%02x ", psqlca->sqlerrmc[pos]);
        else fprintf(err_fp, "   ");
      }
      fprintf(err_fp,"   |");
      for(k = 0; k < 16; k++) {
        int pos = j * 16 + k;
        char c = ' ';
        if (pos < 70) {
          c = psqlca->sqlerrmc[pos];
          if (!isprint(c)) c = ' ';
        }
        fprintf(err_fp,"%c", c);
      }
      fprintf(err_fp,"|\n");
      if (j < 4) fprintf(err_fp,"          ");
    }
  }

  fprintf(err_fp, "sqlerrp:  ");
  for(j = 0; j < 8; j++)
    fprintf(err_fp, "%c", psqlca->sqlerrp[j]);
  fprintf(err_fp,"\n");

  fprintf(err_fp, "sqlerrd: ");
  for(j = 0; j < 6; j++)
    fprintf(err_fp, " %d", psqlca->sqlerrd[j]);
  fprintf(err_fp,"\n");

  if (psqlca->sqlwarn[0] != ' ')
  {
    fprintf(err_fp, "sqlwarn: ");
    for(j = 0; j < 8; j++)
      fprintf(err_fp, "%c ", psqlca->sqlwarn[j]);
    fprintf(err_fp,"\n");
  }

  fprintf(err_fp, "\n");

  fclose(err_fp);
}


/*-----------------------------------------------------------*/
/* del_debug                                                 */
/*-----------------------------------------------------------*/
void del_debug (struct out_delivery_struct *delivery_ptr,
                struct in_delivery_struct  *in_delivery,
                char *msg)
{
  char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

  InitializeDebug();
  strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
  strcat(debug_fn, "del.debug.out");
  del_print(delivery_ptr, in_delivery, debug_fn, msg);
}

/*-----------------------------------------------------------*/
/* del_print                                                 */
/*-----------------------------------------------------------*/
void del_print (struct out_delivery_struct *delivery_ptr,
                struct in_delivery_struct  *in_delivery,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j;
  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;

  if ((debug_fp = fopen(filename, "a+")) == NULL)
  {
    return;
  }

  fprintf(debug_fp,"Delivery debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp,"\n==================================================\n");

  fprintf(debug_fp,"in_delivery_struct {\n");
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
          in_delivery->s_W_ID, in_delivery->s_W_ID);
  fprintf(debug_fp,"\ts_O_CARRIER_ID = %d (%X)\n",
          in_delivery->s_O_CARRIER_ID, in_delivery->s_O_CARRIER_ID);
  fprintf(debug_fp,"}\n\n");

  fprintf(debug_fp,"out_delivery_struct {\n");
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          delivery_ptr->s_transtatus,delivery_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          delivery_ptr->deadlocks,delivery_ptr->deadlocks);

  for (j = 0; j < 10; j++) {
    fprintf(debug_fp,"\t\ts_O_ID[%d]         = %d\n",
            j, delivery_ptr->s_O_ID[j]);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}


/*-----------------------------------------------------------*/
/* new_debug                                                 */
/*-----------------------------------------------------------*/
void new_debug (struct out_neword_struct *neword_ptr,
                struct in_neword_struct *in_neword,
                char *msg)
{
  char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

  InitializeDebug();
  strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
  strcat(debug_fn, "new.debug.out");
  new_print(neword_ptr, in_neword, debug_fn, msg);
}

/*-----------------------------------------------------------*/
/* new_print                                                 */
/*-----------------------------------------------------------*/
void new_print (struct out_neword_struct *neword_ptr,
                struct in_neword_struct *in_neword,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j, items;

  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;

  if ((debug_fp = fopen(filename, "a+")) == NULL)
  {
    return;
  }

  fprintf(debug_fp,"New order debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp,"\n==================================================\n");

  fprintf(debug_fp,"in_neword_struct {\n");

  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
          in_neword->s_C_ID, in_neword->s_C_ID);
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
          in_neword->s_W_ID, in_neword->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
          in_neword->s_D_ID, in_neword->s_D_ID);
  fprintf(debug_fp,"\ts_O_OL_CNT     = %d (%X)\n",
          in_neword->s_O_OL_CNT, in_neword->s_O_OL_CNT);
  fprintf(debug_fp,"\ts_all_local    = %d (%X)\n",
          in_neword->s_all_local, in_neword->s_all_local);
// fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
//         in_neword->s_transtatus, in_neword->s_transtatus);
// fprintf(debug_fp,"\tduplicate_items= %d (%X)\n",
//         in_neword->duplicate_items, in_neword->duplicate_items);

  fprintf(debug_fp,"\titems {\n");
  items = in_neword->s_O_OL_CNT;
  for (j=0; j<items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_OL_I_ID[%d]        = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_I_ID, in_neword->in_item[j].s_OL_I_ID);
    fprintf(debug_fp,"\t\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
```

```
            j, in_neword->in_item[j].s_OL_SUPPLY_W_ID, in_neword-
>in_item[j].s_OL_SUPPLY_W_ID);
      fprintf(debug_fp,"\t\ts_OL_QUANTITY[%d]    = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_QUANTITY, in_neword-
>in_item[j].s_OL_QUANTITY);
   }
   fprintf(debug_fp,"\t}\n}\n\n");

   fprintf(debug_fp,"out_neword_struct {\n");
   fprintf(debug_fp,"\ts_C_LAST       = %s\n",
         neword_ptr->s_C_LAST);
   fprintf(debug_fp,"\ts_C_CREDIT     = %s\n",
         neword_ptr->s_C_CREDIT);
   fprintf(debug_fp,"\ts_W_TAX        = %04.4f \n",
         neword_ptr->s_W_TAX);
   fprintf(debug_fp,"\ts_D_TAX        = %04.4f \n",
         neword_ptr->s_D_TAX);
   fprintf(debug_fp,"\ts_C_DISCOUNT   = %04.4f \n",
         neword_ptr->s_C_DISCOUNT);
   fprintf(debug_fp,"\ts_O_ID         = %d (%X)\n",
         neword_ptr->s_O_ID, neword_ptr->s_O_ID);
   fprintf(debug_fp,"\ts_O_OL_CNT     = %d (%X)\n",
         neword_ptr->s_O_OL_CNT, neword_ptr->s_O_OL_CNT);
   fprintf(debug_fp,"\ts_O_ENTRY_D    = %s \n",
         neword_ptr->s_O_ENTRY_D_time);
   fprintf(debug_fp,"\ts_total_amount = %.2f \n",
         neword_ptr->s_total_amount);
   fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
         neword_ptr->s_transtatus, neword_ptr->s_transtatus);
   fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
         neword_ptr->deadlocks, neword_ptr->deadlocks);

// fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
//         neword_ptr->s_W_ID, neword_ptr->s_W_ID);
// fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
//         neword_ptr->s_D_ID, neword_ptr->s_D_ID);
// fprintf(debug_fp,"\ts_all_local    = %d (%X)\n",
//         neword_ptr->s_all_local, neword_ptr->s_all_local);
// fprintf(debug_fp,"\tduplicate_items= %d (%X)\n",
//         neword_ptr->duplicate_items, neword_ptr->duplicate_items);

   fprintf(debug_fp,"\titems {\n");
   items = neword_ptr->s_O_OL_CNT;
   for (j=0; j<items; j++) {
      if(j != 0)
         fprintf(debug_fp,"\n");
      fprintf(debug_fp,"\t\ts_I_NAME[%d]         = %s\n",
            j, neword_ptr->item[j].s_I_NAME);
      fprintf(debug_fp,"\t\ts_I_PRICE[%d]        = %.2f \n",
            j, neword_ptr->item[j].s_I_PRICE);
      fprintf(debug_fp,"\t\ts_OL_AMOUNT[%d]      = %.2f \n",
            j, neword_ptr->item[j].s_OL_AMOUNT);
      fprintf(debug_fp,"\t\ts_S_QUANTITY[%d]     = %d (%X)\n",
            j, neword_ptr->item[j].s_S_QUANTITY, neword_ptr->item[j].s_S_QUANTITY);
      fprintf(debug_fp,"\t\ts_brand_generic[%d]  = %c\n",
            j, neword_ptr->item[j].s_brand_generic);
   }
   fprintf(debug_fp,"\t}\n}\n\n");
   fclose(debug_fp);
}



/*-----------------------------------------------------------------*/
/* ord_debug                                                       */
/*-----------------------------------------------------------------*/
void ord_debug (struct out_ordstat_struct *ordstat_ptr,
               struct in_ordstat_struct  *in_ordstat,
               char *msg)
{
   char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

   InitializeDebug();
   strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
   strcat(debug_fn, "ord.debug.out");
   ord_print(ordstat_ptr, in_ordstat, debug_fn, msg);
}

/*-----------------------------------------------------------------*/
/* ord_print                                                       */
/*-----------------------------------------------------------------*/
void ord_print (struct out_ordstat_struct *ordstat_ptr,
               struct in_ordstat_struct  *in_ordstat,
               char *filename,
               char *msg)
{
   FILE *debug_fp;
   char  timeStamp[27];
   int   j, items;

   current_tmstmp(&timeStamp[0]);
   timeStamp[19] = (char)NULL;

   if ((debug_fp = fopen(filename, "a+")) == NULL)
   {
      return;
   }

   fprintf(debug_fp,"Order status debug information follows %s (%s)\n", timeStamp, msg);
   fprintf(debug_fp,"\n==================================================\n");

   fprintf(debug_fp,"in_ordstat_struct {\n");
   fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
         in_ordstat->s_W_ID, in_ordstat->s_W_ID);
   fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
         in_ordstat->s_D_ID, in_ordstat->s_D_ID);
   fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
         in_ordstat->s_C_ID, in_ordstat->s_C_ID);
   fprintf(debug_fp,"\ts_C_LAST       = %s\n",
         in_ordstat->s_C_LAST);
   fprintf(debug_fp,"}\n\n");

   fprintf(debug_fp,"out_ordstat_struct {\n");
   fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
         ordstat_ptr->s_C_ID, ordstat_ptr->s_C_ID);
   fprintf(debug_fp,"\ts_C_FIRST      = %s\n",
         ordstat_ptr->s_C_FIRST);
   fprintf(debug_fp,"\ts_C_MIDDLE     = %s\n",
         ordstat_ptr->s_C_MIDDLE);
   fprintf(debug_fp,"\ts_C_LAST       = %s\n",
         ordstat_ptr->s_C_LAST);
   fprintf(debug_fp,"\ts_C_BALANCE    = %.2f\n",
         ordstat_ptr->s_C_BALANCE);
   fprintf(debug_fp,"\ts_O_ID         = %d (%X)\n",
         ordstat_ptr->s_O_ID, ordstat_ptr->s_O_ID);
   fprintf(debug_fp,"\ts_O_ENTRY_D    = %s \n",
         ordstat_ptr->s_O_ENTRY_D_time);
   fprintf(debug_fp,"\ts_O_CARRIER_ID = %d (%X)\n",
         ordstat_ptr->s_O_CARRIER_ID, ordstat_ptr->s_O_CARRIER_ID);
   fprintf(debug_fp,"\ts_ol_cnt       = %d (%X)\n",
         ordstat_ptr->s_ol_cnt, ordstat_ptr->s_ol_cnt);
   fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
         ordstat_ptr->s_transtatus, ordstat_ptr->s_transtatus);
   fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
         ordstat_ptr->deadlocks, ordstat_ptr->deadlocks);

   fprintf(debug_fp,"\titems {\n");
   items = ordstat_ptr->s_ol_cnt;
   for (j = 0; j < items; j++) {
      if(j != 0)
         fprintf(debug_fp,"\n");
      fprintf(debug_fp,"\t\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_SUPPLY_W_ID, ordstat_ptr-
>item[j].s_OL_SUPPLY_W_ID);
      fprintf(debug_fp,"\t\ts_OL_I_ID[%d]        = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_I_ID, ordstat_ptr->item[j].s_OL_I_ID);
      fprintf(debug_fp,"\t\ts_OL_QUANTITY[%d]    = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_QUANTITY, ordstat_ptr->item[j].s_OL_QUANTITY);
      fprintf(debug_fp,"\t\ts_OL_AMOUNT[%d]      = %.2f\n",
            j, ordstat_ptr->item[j].s_OL_AMOUNT);
      fprintf(debug_fp,"\t\ts_OL_DELIVERY_D[%d]  = %s \n",
            j,ordstat_ptr->item[j].s_OL_DELIVERY_D_time);
   }
   fprintf(debug_fp,"\t}\n}\n\n");
   fclose(debug_fp);
}


/*-----------------------------------------------------------------*/
/* pay_debug                                                       */
/*-----------------------------------------------------------------*/
void pay_debug (struct out_payment_struct *payment_ptr,
               struct in_payment_struct  *in_payment,
               char *msg)
{
   char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

   InitializeDebug();
   strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
   strcat(debug_fn, "pay.debug.out");
   pay_print(payment_ptr, in_payment, debug_fn, msg);
}

/*-----------------------------------------------------------------*/
/* pay_print                                                       */
/*-----------------------------------------------------------------*/
void pay_print (struct out_payment_struct *payment_ptr,
               struct in_payment_struct  *in_payment,
               char *filename,
               char *msg)
{
   FILE *debug_fp;
   char  timeStamp[27];

   current_tmstmp(&timeStamp[0]);
   timeStamp[19] = (char)NULL;

   if ((debug_fp = fopen(filename, "a+")) == NULL)
   {
      return;
   }

   fprintf(debug_fp,"Payment debug information follows %s (%s)\n", timeStamp, msg);
```

```
  fprintf(debug_fp,"\n==============================================\n");

  fprintf(debug_fp,"in_payment_struct {\n");
  fprintf(debug_fp,"\ts_H_AMOUNT     = %.2f \n",
        in_payment->s_H_AMOUNT);
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
        in_payment->s_C_ID, in_payment->s_C_ID);
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
        in_payment->s_W_ID, in_payment->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
        in_payment->s_D_ID, in_payment->s_D_ID);
  fprintf(debug_fp,"\ts_C_D_ID       = %d (%X)\n",
        in_payment->s_C_D_ID, in_payment->s_C_D_ID);
  fprintf(debug_fp,"\ts_C_W_ID       = %d (%X)\n",
        in_payment->s_C_W_ID, in_payment->s_C_W_ID);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
        in_payment->s_C_LAST);
  fprintf(debug_fp,"\n}\n\n");

  fprintf(debug_fp,"out_payment_struct {\n");
  fprintf(debug_fp,"\ts_C_CREDIT_LIM = %.2f\n",
        payment_ptr->s_C_CREDIT_LIM);
  fprintf(debug_fp,"\ts_C_DISCOUNT   = %04.4f\n",
        payment_ptr->s_C_DISCOUNT);
  fprintf(debug_fp,"\ts_C_BALANCE    = %.2f\n",
        payment_ptr->s_C_BALANCE);
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
        payment_ptr->s_C_ID, payment_ptr->s_C_ID);
  fprintf(debug_fp,"\ts_W_STREET_1   = %s\n",
        payment_ptr->s_W_STREET_1);
  fprintf(debug_fp,"\ts_W_STREET_2   = %s\n",
        payment_ptr->s_W_STREET_2);
  fprintf(debug_fp,"\ts_W_CITY       = %s\n",
        payment_ptr->s_W_CITY);
  fprintf(debug_fp,"\ts_W_STATE      = %s\n",
        payment_ptr->s_W_STATE);
  fprintf(debug_fp,"\ts_W_ZIP        = %s\n",
        payment_ptr->s_W_ZIP);
  fprintf(debug_fp,"\ts_D_STREET_1   = %s\n",
        payment_ptr->s_D_STREET_1);
  fprintf(debug_fp,"\ts_D_STREET_2   = %s\n",
        payment_ptr->s_D_STREET_2);
  fprintf(debug_fp,"\ts_D_CITY       = %s\n",
        payment_ptr->s_D_CITY);
  fprintf(debug_fp,"\ts_D_STATE      = %s\n",
        payment_ptr->s_D_STATE);
  fprintf(debug_fp,"\ts_D_ZIP        = %s\n",
        payment_ptr->s_D_ZIP);
  fprintf(debug_fp,"\ts_C_FIRST      = %s\n",
        payment_ptr->s_C_FIRST);
  fprintf(debug_fp,"\ts_C_MIDDLE     = %s\n",
        payment_ptr->s_C_MIDDLE);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
        payment_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_STREET_1   = %s\n",
        payment_ptr->s_C_STREET_1);
  fprintf(debug_fp,"\ts_C_STREET_2   = %s\n",
        payment_ptr->s_C_STREET_2);
  fprintf(debug_fp,"\ts_C_CITY       = %s\n",
        payment_ptr->s_C_CITY);
  fprintf(debug_fp,"\ts_C_STATE      = %s\n",
        payment_ptr->s_C_STATE);
  fprintf(debug_fp,"\ts_C_ZIP        = %s\n",
        payment_ptr->s_C_ZIP);
  fprintf(debug_fp,"\ts_C_PHONE      = %s\n",
        payment_ptr->s_C_PHONE);
  fprintf(debug_fp,"\ts_C_SINCE      = %s \n",
        payment_ptr->s_C_SINCE_time);
  fprintf(debug_fp,"\ts_C_CREDIT     = %s\n",
        payment_ptr->s_C_CREDIT);
  fprintf(debug_fp,"\ts_C_DATA       = %s\n",
        payment_ptr->s_C_DATA);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
        payment_ptr->s_transtatus,payment_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
        payment_ptr->deadlocks,payment_ptr->deadlocks);
  fprintf(debug_fp,"\n}\n\n");
  fclose(debug_fp);
}


/*-----------------------------------------------------------*/
/* stk_debug                                                 */
/*-----------------------------------------------------------*/
void stk_debug (struct out_stocklev_struct *stocklev,
            struct in_stocklev_struct  *in_stocklev,
            char *msg)
{
  char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

  InitializeDebug();
  strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
  strcat(debug_fn, "stk.debug.out");
  stk_print(stocklev, in_stocklev, debug_fn, msg);
}

/*-----------------------------------------------------------*/
/* stk_print                                                 */
/*-----------------------------------------------------------*/
void stk_print (struct out_stocklev_struct *stocklev,
            struct in_stocklev_struct  *in_stocklev,
            char *filename,
            char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];

  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;

  if ((debug_fp = fopen(filename, "a+")) == NULL)
  {
    return;
  }

  fprintf(debug_fp,"Stock level debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp,"\n==============================================\n");

  fprintf(debug_fp,"in_stocklev_struct {\n");
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
        in_stocklev->s_W_ID, in_stocklev->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
        in_stocklev->s_D_ID, in_stocklev->s_D_ID);
  fprintf(debug_fp,"\ts_threshold    = %d (%X)\n",
        in_stocklev->s_threshold, in_stocklev->s_threshold);
  fprintf(debug_fp,"}\n\n");

  fprintf(debug_fp,"out_stocklev_struct {\n");
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
        stocklev->s_transtatus, stocklev->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
        stocklev->deadlocks, stocklev->deadlocks);
  fprintf(debug_fp,"\ts_low_stock    = %d (%X)\n",
        stocklev->s_low_stock, stocklev->s_low_stock);
  fprintf(debug_fp,"}\n\n");
  fclose(debug_fp);
}

void current_tmstmp(char *buf)
{
  time_t t = time(NULL);
  strncpy(buf,ctime(&t),19);
}
```

include\db2tpcc.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * db2tpcc.h - Macros and Miscellany
 *
 */

#ifndef __DB2TPCC_H
#define __DB2TPCC_H

#include <sys/types.h>
typedef __int16 int16_t;
typedef __int32 int32_t;
typedef __int64 int64_t;

#include "lval.h"

/* ******************************************************************** */
/* Transaction Return Codes (s_transtatus)                              */
/* ******************************************************************** */

#define INVALID_ITEM          100
#define TRAN_OK               0
#define FATAL_SQLERROR        -1

/* ******************************************************************** */
/* Definition of Unused and Bad Items                                   */
/* ******************************************************************** */
/* Define unused item ID to be 0. This allows the SUT to determine the     */
/* number of items in the order as required by 2.4.1.3 and 2.4.2.2 since   */
/* the assumption that any item with OL_I_ID = 0 is unused will be true.   */
/* This in turn requires that the value used for an invalid item is        */
/* equal to ITEMS + 1.                                                      */
/* ******************************************************************** */
```

```
#define INVALID_ITEM_ID (2 * ITEMS) + 1
#define UNUSED_ITEM_ID 0

#define MIN_WAREHOUSE 1
#define MAX_WAREHOUSE WAREHOUSES

/*****************************************************************************/
/* NURand Constants                                                          */
/* C_C_LAST_RUN and C_C_LAST_LOAD must adhere to clause 2.1.6.               */
/* Analysis indicates that a C_LAST delta of 85 is optimal.                  */
/*****************************************************************************/
#define C_C_LAST_RUN    88
#define C_C_LAST_LOAD   173
#define C_C_ID          319
#define C_OL_I_ID       3849
#define A_C_LAST        255
#define A_C_ID          1023
#define A_OL_I_ID       8191

/*****************************************************************************/
/* Transaction Type Identifiers                                              */
/*****************************************************************************/

#define CLIENT_SQL   0
#define NEWORD_SQL   1
#define PAYMENT_SQL  2
#define ORDSTAT_SQL  3
#define DELIVERY_SQL 4
#define STOCKLEV_SQL 5

/* ************************************************************************** */

#define SPGENERAL_PAD 3
#define SPGENERAL_ADJUST sizeof(int16_t)

struct in_neword_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  struct   in_items_struct {
    int32_t  s_OL_I_ID;
    int32_t  s_OL_SUPPLY_W_ID;
    int16_t  s_OL_QUANTITY;
    int16_t  pad1[3];
  } in_item[15];
  int32_t  s_C_ID;
  int32_t  s_W_ID;
  int16_t  s_D_ID;
  int16_t  s_O_OL_CNT;          /* init by SUT */
  int16_t  s_all_local;
  int16_t  duplicate_items;
};

struct out_neword_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  struct   items_struct {
    float  s_I_PRICE;
    float  s_OL_AMOUNT;
    int16_t  s_S_QUANTITY;
    int16_t  pad2;
    char     s_I_NAME[25];
    char     s_brand_generic;
  } item[15];
  float  s_W_TAX;
  float  s_D_TAX;
  float  s_C_DISCOUNT;
  float  s_total_amount;
  int32_t  s_O_ID;
  int16_t  s_O_OL_CNT;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_LAST[17];
  char     s_C_CREDIT[3];
  char     s_O_ENTRY_D_time[27];

};

struct in_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  float  s_H_AMOUNT;
  int32_t  s_W_ID;
  int32_t  s_C_W_ID;
  int32_t  s_C_ID;
  int16_t  s_C_D_ID;
  int16_t  s_D_ID;
  char     s_C_LAST[17];
};

struct out_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_CREDIT_LIM;
  double  s_C_BALANCE;
  float  s_C_DISCOUNT;
  int32_t  s_C_ID;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_W_STREET_1[21];
  char     s_W_STREET_2[21];
  char     s_W_CITY[21];
  char     s_W_STATE[3];
  char     s_W_ZIP[10];
  char     s_D_STREET_1[21];
  char     s_D_STREET_2[21];
  char     s_D_CITY[21];
  char     s_D_STATE[3];
  char     s_D_ZIP[10];
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_C_STREET_1[21];
  char     s_C_STREET_2[21];
  char     s_C_CITY[21];
  char     s_C_STATE[3];
  char     s_C_ZIP[10];
  char     s_C_PHONE[17];
  char     s_C_CREDIT[3];
  char     s_C_DATA[201];
  char     s_H_DATE_time[27];
  char     s_C_SINCE_time[27];
};

struct in_ordstat_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_C_ID;
  int32_t  s_W_ID;
  int16_t  s_D_ID;
  int16_t  pad1[3];
  char     s_C_LAST[17];
};

struct out_ordstat_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_BALANCE;
  int32_t  s_C_ID;
  int32_t  s_O_ID;
  int16_t  s_O_CARRIER_ID;
  int16_t  s_ol_cnt;
  int16_t  pad1[2];
  struct   oitems_struct {
    double  s_OL_AMOUNT;
    int32_t  s_OL_I_ID;
    int32_t  s_OL_SUPPLY_W_ID;
    int16_t  s_OL_QUANTITY;
    int16_t  pad2;
    char     s_OL_DELIVERY_D_time[27];
  } item[15];
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_O_ENTRY_D_time[27];
  int16_t  pad3[2];
};

struct in_delivery_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_W_ID;
  int16_t  s_O_CARRIER_ID;
};

struct out_delivery_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_O_ID[10];
  int16_t  s_transtatus;
  int16_t  deadlocks;
};

struct in_stocklev_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_threshold;
  int32_t  s_W_ID;
  int16_t  s_D_ID;
};

struct out_stocklev_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_low_stock;
  int16_t  s_transtatus;
  int16_t  deadlocks;
};

/* ************************************************************************** */
/* Transaction Prototypes                                                     */
/* ************************************************************************** */
```

```
#ifdef __cplusplus
extern "C" {
#endif

extern int neword_sql(struct in_neword_struct*, struct out_neword_struct*);
extern int payment_sql(struct in_payment_struct*, struct out_payment_struct*);
extern int ordstat_sql(struct in_ordstat_struct*, struct out_ordstat_struct*);
extern int delivery_sql(struct in_delivery_struct*, struct out_delivery_struct*);
extern int stocklev_sql(struct in_stocklev_struct*, struct out_stocklev_struct*);

#ifdef __cplusplus
}
#endif

/* ******************************************************************** */
/* DB2 Connect/Disconnect & Thread Context Wrappers                     */
/* ******************************************************************** */

#ifdef __cplusplus
extern "C" {
#endif

extern int connect_to_TM(char*);
extern int connect_to_TM_auth(char*, char*, char*);
extern int disconnect_from_TM(void);

extern int create_context(void);
extern int destroy_context(void);
extern int get_context(void**);
extern int attach_context(void*);
extern int detach_context(void*);

#ifdef __cplusplus
}
#endif

#endif // __DB2TPCC_H
```

include\tpccdbg.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tpccdbg.h - Debugging Macros
 *
 */

#ifndef __TPCCDBG_H
#define __TPCCDBG_H

#ifdef __cplusplus
extern "C" {
#endif

extern void sqlerror (int tranType, char *msg, char *file, int line,
                      SQL_STRUCTURE sqlca *psqlca);

extern void new_debug (struct out_neword_struct *neword_ptr,
                       struct in_neword_struct *in_neword_ptr,
                       char *msg);
extern void pay_debug (struct out_payment_struct *payment_ptr,
                       struct in_payment_struct *in_payment_ptr,
                       char *msg);
extern void ord_debug (struct out_ordstat_struct *ordstat_ptr,
                       struct in_ordstat_struct *in_ordstat_ptr,
                       char *msg);
extern void del_debug (struct out_delivery_struct *delivery_ptr,
                       struct in_delivery_struct *in_delivery_ptr,
                       char *msg);
extern void stk_debug (struct out_stocklev_struct *stocklev_ptr,
                       struct in_stocklev_struct *in_stocklev_ptr,
                       char *msg);

extern void new_print (struct out_neword_struct *neword_ptr,
                       struct in_neword_struct *in_neword_ptr,
                       char *filename,
                       char *msg);
extern void pay_print (struct out_payment_struct *payment_ptr,
                       struct in_payment_struct *in_payment_ptr,
                       char *filename,
                       char *msg);
extern void ord_print (struct out_ordstat_struct *ordstat_ptr,
                       struct in_ordstat_struct *in_ordstat_ptr,
                       char *filename,
                       char *msg);
extern void del_print (struct out_delivery_struct *delivery_ptr,
                       struct in_delivery_struct *in_delivery_ptr,
                       char *filename,
                       char *msg);
extern void stk_print (struct out_stocklev_struct *stocklev_ptr,
                       struct in_stocklev_struct *in_stocklev_ptr,
                       char *filename,
                       char *msg);

#ifdef __cplusplus
}
#endif

#endif // __TPCCDBG_H
```

tpccenv.bat

```
@REM ***************************************************************************
@REM  Licensed Materials - Property of IBM
@REM
@REM  Governed under the terms of the International
@REM  License Agreement for Non-Warranted Sample Code.
@REM
@REM  (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
@REM  All Rights Reserved.
@REM
@REM  US Government Users Restricted Rights - Use, duplication or
@REM  disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
@REM ***************************************************************************
@REM
@REM tpccenv.bat - Windows Environment Setup
@REM

@REM The Kit Version
set TPCC_VERSION=CK080131

@REM The DB2 Instance Name (for DB2)
set DB2INSTANCE=DB2

@REM The OS being used (i.e. "WINDOWS")
set PLATFORM=WINDOWS
set SERVER_PLATFORM=UNIX

@REM The type of make command and slash used by the OS
@REM (i.e. UNIX - "/", WINDOWS - "\")
@REM These are referenced all over the kit.
set SLASH=\
set MAKE=nmake

@REM Specifies whether or not to use dari stored proc's for the TPC-C driver.  Set to
either DARIVERSION or NONDARI;
@REM set TPCC_SPTYPE=NOSP
@REM set TPCC_SPTYPE=SPGENERAL2
set TPCC_SPTYPE=SPGENERAL
@REM set TPCC_SPTYPE=DARI2SQLDA

set DB2VERSION=v9

@REM The schema name is typically the SQL autorization ID (or username).
@REM This is required for runstats and EEE.
set TPCC_SCHEMA=tpcc
set SERVER_TPCC_SCHEMA=tpcc

@REM DB2 EE/EEE Configuration
set DB2EDITION=EE
@REM set DB2EDITION=EEE
set DB2NODE=0
@REM set to the number of nodes you have.  Set to 1 for EE.
set DB2NODES=1

@REM TPCC General Configuration
@REM ** IMPORTANT NOTE **
@REM The kit is not guaranteed to work properly if TPCC_ROOT or TPCC_SQLLIB
@REM have spaces in them.  If you absolutely must use paths with spaces,
@REM then the entire path must be surrounded by double quotes.
@REM For example:  HOME="C:\Program Files\IBM"
set HOME=c:\home\tpcc\tpcc_db2_comless
set TPCC_DBNAME=TPCC
set TPCC_ROOT=c:\home\tpcc\tpcc_db2_comless\tpc-c.ibm
set TPCC_SQLLIB=c:\sqllib
set TPCC_RUNDATA=%HOME%\tpccdata

@REM TPCC Debug Configuration
@REM This is the path where all error and debug logs are placed.
@REM To get debugging from within the stored procedures, you must
@REM set DB2ENVLIST="TPCC_DEBUGDIR" in tpcc.config.
set TPCC_DEBUGDIR=c:\temp

@REM Specifies where stored procedures should be placed and if they should
@REM be fenced.
set TPCC_SPDIR=%TPCC_SQLLIB%\function
set TPCC_FENCED=NO
```

9.2. Client Transaction Code

Makefile.config

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# Makefile.config - Linux 64-bit
#
#


# Make Configuration
MAKE=make

# Compiler Configuration.
# CFLAGS_DEBUG may be set to "-g", "-DDEBUGIT" "-g -DDEBUGIT" or left blank
CC=cc
CFLAGS_OS=-DSQLUNIX -DSQLLinux -O2 -fpic -m64
CFLAGS_OUT=-o
CFLAGS_DEBUG=

# Linker Configuration
LD_EXEC=gcc
LD_STORP=gcc
LDFLAGS_EXEC=
LDFLAGS_SHLIB=-shared
LDFLAGS_STORP=$(LDFLAGS_SHLIB)
LDFLAGS_LIB=-L$(TPCC_SQLLIB)/lib -ldb2 -m64
LDFLAGS_OUT=-o

# Library Configuration
AR=ar
ARFLAGS=-rv
ARFLAGS_LIB=
ARFLAGS_OUT=

# OS Commands
ERASE=rm -f
ERASEDIR=$(ERASE) -R
MOVE=mv
COPY=cp

# OS File Extensions & Path Separators
OBJEXT=.o
LIBEXT=.a
SHLIBEXT=.so
BINEXT=
SLASH=/
CMDSEP=;
```

Src.Common/Makefile

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# Makefile - Makefile for Src.Common
#
#

include $(TPCC_ROOT)/Makefile.config

#
##########################################################################
#
# Preprocessor, Compiler and LInker Flags
#
##########################################################################
#

PRP_OPTS =              PACKAGE \
                        OPTLEVEL 1 \
                        ISOLATION RR \
                        MESSAGES $*.prep.msg \
                        LEVEL $(TPCC_VERSION) \
                        NOLINEMACRO

INCLUDE =  -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

CFLAGS =   $(CFLAGS_OS) $(CFLAGS_DEBUG) $(INCLUDE) \
                        -DSQLA_NOLINES -D$(DB2EDITION) -D$(TPCC_SPTYPE)

UTIL_OBJ_DBG =          tpccdbg$(OBJEXT)
UTIL_OBJ_GEN =          tpccmisc$(OBJEXT)
UTIL_OBJ_DB2 =          tpccctx$(OBJEXT)

#
##########################################################################
#
# User Targets
#
##########################################################################
#

all:            $(UTIL_OBJ_DBG) $(UTIL_OBJ_GEN) connect $(UTIL_OBJ_DB2)
disconnect

dbgen:          $(UTIL_OBJ_GEN)

clean:
                - $(ERASE) *$(OBJEXT) *.bnd *.msg tpccctx.c

#
##########################################################################
#
# Helper Targets
#
##########################################################################
#

connect:
                - db2 connect to $(TPCC_DBNAME)

disconnect:
                - db2 connect reset
                - db2 terminate

#
##########################################################################
#
# Build Rules
#
##########################################################################
#

.SUFFIXES:
.SUFFIXES: $(OBJEXT) .c .sqc

.sqc.c:
                @echo "Prepping $*.sqc"
                db2 prep $*.sqc $(PRP_OPTS) bindfile
                db2 grant execute on package TPCCCTX to public

#
##########################################################################
#
# Dependencies
#
##########################################################################
#

# Source
tpccdbg$(OBJEXT):       tpccdbg.c
tpccctx$(OBJEXT):       tpccctx.c
tpccmisc$(OBJEXT):      tpccmisc.c

# Headers
tpccdbg.c:      $(TPCC_ROOT)/include/db2tpcc.h
```

Src.Common/tpccctx.sqc

```
/**************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
```

```
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*******************************************************************************/

/*
 *
 * tpccctx.sqc - TPCC context code
 *
 */

#include <string.h>
#include <sqlutil.h>
#include "db2tpcc.h"
#include "tpccdbg.h"

int connect_to_TM(char *in_dbname);
int connect_to_TM_auth(char *in_dbname, char *in_username, char *in_password);
int disconnect_from_TM(void);

int connect_to_TM(char *in_dbname)
{
  return connect_to_TM_auth(in_dbname, "", "");
}

int connect_to_TM_auth(char *in_dbname, char *in_username, char *in_password)
{
   SQL_STRUCTURE sqlca  sqlca;
   int ConnectSQLCODE = 0;

   EXEC SQL BEGIN DECLARE SECTION;
   char dbname[9];
   char username[129];
   char password[15];
   EXEC SQL END DECLARE SECTION;

   /* Copy 9 characters - 8 for dbname, 1 for NULL */
   strncpy(dbname,in_dbname,9);
   if (strcmp(in_username,"") == 0)
   {
      EXEC SQL CONNECT TO :dbname IN SHARE MODE;
   } else {
      strncpy(username,in_username,128);
      strncpy(password,in_password,14);
      EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username
USING :password;
   }

   ConnectSQLCODE = SQLCODE;
   if (ConnectSQLCODE != 0)
   {
      sqlerror( CLIENT_SQL, "CONNECT", __FILE__, __LINE__, &sqlca);

      return ConnectSQLCODE;
   }

   return 0;
}

int disconnect_from_TM(void)
{
   SQL_STRUCTURE sqlca  sqlca;
   int DisconnectSQLCODE = 0;

   EXEC SQL CONNECT RESET;

   DisconnectSQLCODE = SQLCODE;
   if (DisconnectSQLCODE != 0) {
      sqlerror( CLIENT_SQL, "DISCONNECT", __FILE__, __LINE__, &sqlca);
   }

   if (DisconnectSQLCODE) {
      return DisconnectSQLCODE;
   }
   return 0;
}
```

Src.Common/tpccdbg.c

```
/*******************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*******************************************************************************/

/*
 * tccdbg.c - Debugging Routines
 *
 */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include <time.h>
#include <unistd.h>

#include "sqlca.h"
#include "sql.h"
#include "db2tpcc.h"
#include "tpccdbg.h"

#define DEBUG_FILENAME_SZ 128
#define DEBUG_PATH_SIZE 128

void          del_print();
void          new_print();
void          ord_print();
void          pay_print();
void          stk_print();

void current_tmstmp(char *buf);

static int debugInit = 0;
static char debugPath[DEBUG_PATH_SIZE] = "";

/*-----------------------------------------------------------------*/
/*      InitializeDebug                                            */
/*-----------------------------------------------------------------*/
void InitializeDebug(void) {
  if (debugInit == 0) {
    char *p = getenv("TPCC_DEBUGDIR");
    if (p) {
      strncpy(debugPath, p, DEBUG_PATH_SIZE);
    } else {
      strcpy(debugPath, "/tmp");
    }
    strcat(debugPath, "/");
  }
  debugInit = 1;
}

/*-----------------------------------------------------------------*/
/*      sqlerror                                                   */
/*-----------------------------------------------------------------*/
void sqlerror(int tranType, char *msg, char *file, int line, SQL_STRUCTURE sqlca *psqlca)
{
  FILE *err_fp = NULL;
  char  err_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];
  char  tranName[16];
  int j,k;
  char  timeStamp[27];
  char  errStr[512] = "";

  InitializeDebug();
  strncpy(err_fn, debugPath, DEBUG_PATH_SIZE);
  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;

  switch(tranType)
  {
    case NEWORD_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "new.err.out");
      strcpy(tranName, "NEW_ORDER");
      break;

    case DELIVERY_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "del.err.out");
      strcpy(tranName, "DELIVERY");
      break;

    case PAYMENT_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "pay.err.out");
      strcpy(tranName, "PAYMENT");
      break;

    case ORDSTAT_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "ord.err.out");
      strcpy(tranName, "ORDER_STAT");
      break;

    case STOCKLEV_SQL:
      //sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "stk.err.out");
      strcpy(tranName, "STOCK_LVL");
      break;

    case 0:
      strcat(err_fn, "cli.err.out");
```

```
        strcpy(tranName, "CLIENT");
        break;

      default:
        return;

    }

    /* Generate Formatted Error Message */
    sqlaintp(errStr, 512, 78, psqlca);

    if ((err_fp = fopen(err_fn, "a+")) == NULL)
    {
      return;
    }

    fprintf(err_fp, "----------------------------------------\n");
    fprintf(err_fp, "Transaction: %s (%s)\n", tranName, msg);
    fprintf(err_fp, "FILE %s (%u)\n", file, line);
    fprintf(err_fp, "SQLCODE %d ", psqlca->sqlcode);
    fprintf(err_fp, "PID %d ", getpid());
    fprintf(err_fp, "TIME %s\n", timeStamp);
    fprintf(err_fp, "----------------------------------------\n");
    fprintf(err_fp, "%s", errStr);
    fprintf(err_fp, "----------------------------------------\n");

    if (psqlca->sqlerrmc[0] != ' ' || psqlca->sqlerrmc[1] != ' ')
    {
      fprintf(err_fp, "slerrmc:  ");

      for(j = 0; j < 5; j++)
      {
        for(k = 0; k < 16; k++) {
          int pos = j * 16 + k;
          if (pos < 70) fprintf(err_fp, "%02x ", psqlca->sqlerrmc[pos]);
          else fprintf(err_fp, "   ");
        }
        fprintf(err_fp,"   |");
        for(k = 0; k < 16; k++) {
          int pos = j * 16 + k;
          char c = ' ';
          if (pos < 70) {
            c = psqlca->sqlerrmc[pos];
            if (!isprint(c)) c = ' ';
          }
          fprintf(err_fp,"%c", c);
        }
        fprintf(err_fp,"|\n");
        if (j < 4) fprintf(err_fp,"          ");
      }
    }

    fprintf(err_fp, "sqlerrp:  ");
    for(j = 0; j < 8; j++)
      fprintf(err_fp, "%c", psqlca->sqlerrp[j]);
    fprintf(err_fp,"\n");

    fprintf(err_fp, "sqlerrd: ");
    for(j = 0; j < 6; j++)
      fprintf(err_fp, " %d", psqlca->sqlerrd[j]);
    fprintf(err_fp,"\n");

    if (psqlca->sqlwarn[0] != ' ')
    {
      fprintf(err_fp, "sqlwarn: ");
      for(j = 0; j < 8; j++)
        fprintf(err_fp, "%c ", psqlca->sqlwarn[j]);
      fprintf(err_fp,"\n");
    }

    fprintf(err_fp, "\n");

    fclose(err_fp);
  }


  /*-----------------------------------------------------------*/
  /* del_debug                                                 */
  /*-----------------------------------------------------------*/
  void del_debug (struct out_delivery_struct *delivery_ptr,
                  struct in_delivery_struct  *in_delivery,
                  char *msg)
  {
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

    InitializeDebug();
    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "del.debug.out");
    del_print(delivery_ptr, in_delivery, debug_fn, msg);
  }

  /*-----------------------------------------------------------*/
  /* del_print                                                 */
  /*-----------------------------------------------------------*/
  void del_print (struct out_delivery_struct *delivery_ptr,
                  struct in_delivery_struct  *in_delivery,
                  char *filename,
                  char *msg)
  {
    FILE *debug_fp;
    char  timeStamp[27];
    int   j;

    current_tmstmp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;

    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
      return;
    }

    fprintf(debug_fp,"Delivery debug information follows %s (%s)\n", timeStamp, msg);
    fprintf(debug_fp, " PID %d ", getpid());
    fprintf(debug_fp,"\n==============================================\n");

    fprintf(debug_fp,"in_delivery_struct {\n");
    fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
            in_delivery->s_W_ID, in_delivery->s_W_ID);
    fprintf(debug_fp,"\ts_O_CARRIER_ID = %d (%X)\n",
            in_delivery->s_O_CARRIER_ID, in_delivery->s_O_CARRIER_ID);
    fprintf(debug_fp,"}\n\n");

    fprintf(debug_fp,"out_delivery_struct {\n");
    fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
            delivery_ptr->s_transtatus,delivery_ptr->s_transtatus);
    fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
            delivery_ptr->deadlocks,delivery_ptr->deadlocks);

    for (j = 0; j < 10; j++) {
      fprintf(debug_fp,"\t\ts_O_ID[%d]         = %d\n",
              j, delivery_ptr->s_O_ID[j]);
    }
    fprintf(debug_fp,"\t}\n}\n\n");
    fclose(debug_fp);
  }


  /*-----------------------------------------------------------*/
  /* new_debug                                                 */
  /*-----------------------------------------------------------*/
  void new_debug (struct out_neword_struct *neword_ptr,
                  struct in_neword_struct *in_neword,
                  char *msg)
  {
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

    InitializeDebug();
    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "new.debug.out");
    new_print(neword_ptr, in_neword, debug_fn, msg);
  }

  /*-----------------------------------------------------------*/
  /* new_print                                                 */
  /*-----------------------------------------------------------*/
  void new_print (struct out_neword_struct *neword_ptr,
                  struct in_neword_struct *in_neword,
                  char *filename,
                  char *msg)
  {
    FILE *debug_fp;
    char  timeStamp[27];
    int   j, items;

    current_tmstmp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;

    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
      return;
    }

    fprintf(debug_fp,"New order debug information follows %s (%s)\n", timeStamp, msg);
    fprintf(debug_fp, " PID %d ", getpid());
    fprintf(debug_fp,"\n==============================================\n");

    fprintf(debug_fp,"in_neword_struct {\n");

    fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
            in_neword->s_C_ID, in_neword->s_C_ID);
    fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
            in_neword->s_W_ID, in_neword->s_W_ID);
    fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
            in_neword->s_D_ID, in_neword->s_D_ID);
    fprintf(debug_fp,"\ts_O_OL_CNT     = %d (%X)\n",
            in_neword->s_O_OL_CNT, in_neword->s_O_OL_CNT);
    fprintf(debug_fp,"\ts_all_local    = %d (%X)\n",
            in_neword->s_all_local, in_neword->s_all_local);
// fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
//         in_neword->s_transtatus, in_neword->s_transtatus);
// fprintf(debug_fp,"\tduplicate_items= %d (%X)\n",
//         in_neword->duplicate_items, in_neword->duplicate_items);

    fprintf(debug_fp,"\titems {\n");
```

```
  items = in_neword->s_O_OL_CNT;
  for (j=0; j<items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_OL_I_ID[%d]        = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_I_ID, in_neword->in_item[j].s_OL_I_ID);
    fprintf(debug_fp,"\t\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_SUPPLY_W_ID, in_neword-
>in_item[j].s_OL_SUPPLY_W_ID);
    fprintf(debug_fp,"\t\ts_OL_QUANTITY[%d]    = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_QUANTITY, in_neword-
>in_item[j].s_OL_QUANTITY);
  }
  fprintf(debug_fp,"\t}\n}\n\n");

  fprintf(debug_fp,"out_neword_struct {\n");
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
          neword_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_CREDIT     = %s\n",
          neword_ptr->s_C_CREDIT);
  fprintf(debug_fp,"\ts_W_TAX        = %04.4f \n",
          neword_ptr->s_W_TAX);
  fprintf(debug_fp,"\ts_D_TAX        = %04.4f \n",
          neword_ptr->s_D_TAX);
  fprintf(debug_fp,"\ts_C_DISCOUNT   = %04.4f \n",
          neword_ptr->s_C_DISCOUNT);
  fprintf(debug_fp,"\ts_O_ID         = %d (%X)\n",
          neword_ptr->s_O_ID, neword_ptr->s_O_ID);
  fprintf(debug_fp,"\ts_O_OL_CNT     = %d (%X)\n",
          neword_ptr->s_O_OL_CNT, neword_ptr->s_O_OL_CNT);
  fprintf(debug_fp,"\ts_O_ENTRY_D    = %s \n",
          neword_ptr->s_O_ENTRY_D_time);
  fprintf(debug_fp,"\ts_total_amount = %.2f \n",
          neword_ptr->s_total_amount);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          neword_ptr->s_transtatus, neword_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          neword_ptr->deadlocks, neword_ptr->deadlocks);

// fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
//         neword_ptr->s_W_ID, neword_ptr->s_W_ID);
// fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
//         neword_ptr->s_D_ID, neword_ptr->s_D_ID);
// fprintf(debug_fp,"\ts_all_local    = %d (%X)\n",
//         neword_ptr->s_all_local, neword_ptr->s_all_local);
// fprintf(debug_fp,"\tduplicate_items= %d (%X)\n",
//         neword_ptr->duplicate_items, neword_ptr->duplicate_items);

  fprintf(debug_fp,"\titems {\n");
  items = neword_ptr->s_O_OL_CNT;
  for (j=0; j<items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_I_NAME[%d]         = %s\n",
            j, neword_ptr->item[j].s_I_NAME);
    fprintf(debug_fp,"\t\ts_I_PRICE[%d]        = %.2f \n",
            j, neword_ptr->item[j].s_I_PRICE);
    fprintf(debug_fp,"\t\ts_OL_AMOUNT[%d]      = %.2f \n",
            j, neword_ptr->item[j].s_OL_AMOUNT);
    fprintf(debug_fp,"\t\ts_S_QUANTITY[%d]     = %d (%X)\n",
            j, neword_ptr->item[j].s_S_QUANTITY, neword_ptr->item[j].s_S_QUANTITY);
    fprintf(debug_fp,"\t\ts_brand_generic[%d]  = %c\n",
            j, neword_ptr->item[j].s_brand_generic);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}



/*-----------------------------------------------------------*/
/* ord_debug                                                 */
/*-----------------------------------------------------------*/
void ord_debug (struct out_ordstat_struct *ordstat_ptr,
                struct in_ordstat_struct  *in_ordstat,
                char *msg)
{
  char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

  InitializeDebug();
  strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
  strcat(debug_fn, "ord.debug.out");
  ord_print(ordstat_ptr, in_ordstat, debug_fn, msg);
}

/*-----------------------------------------------------------*/
/* ord_print                                                 */
/*-----------------------------------------------------------*/
void ord_print (struct out_ordstat_struct *ordstat_ptr,
                struct in_ordstat_struct  *in_ordstat,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j, items;

  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;

  if ((debug_fp = fopen(filename, "a+")) == NULL)
  {
    return;
  }

  fprintf(debug_fp,"Order status debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp, " PID %d ", getpid());
  fprintf(debug_fp,"\n==============================================\n");

  fprintf(debug_fp,"in_ordstat_struct {\n");
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
          in_ordstat->s_W_ID, in_ordstat->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
          in_ordstat->s_D_ID, in_ordstat->s_D_ID);
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
          in_ordstat->s_C_ID, in_ordstat->s_C_ID);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
          in_ordstat->s_C_LAST);
  fprintf(debug_fp,"}\n\n");

  fprintf(debug_fp,"out_ordstat_struct {\n");
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
          ordstat_ptr->s_C_ID, ordstat_ptr->s_C_ID);
  fprintf(debug_fp,"\ts_C_FIRST      = %s\n",
          ordstat_ptr->s_C_FIRST);
  fprintf(debug_fp,"\ts_C_MIDDLE     = %s\n",
          ordstat_ptr->s_C_MIDDLE);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
          ordstat_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_BALANCE    = %.2f\n",
          ordstat_ptr->s_C_BALANCE);
  fprintf(debug_fp,"\ts_O_ID         = %d (%X)\n",
          ordstat_ptr->s_O_ID, ordstat_ptr->s_O_ID);
  fprintf(debug_fp,"\ts_O_ENTRY_D    = %s \n",
          ordstat_ptr->s_O_ENTRY_D_time);
  fprintf(debug_fp,"\ts_O_CARRIER_ID = %d (%X)\n",
          ordstat_ptr->s_O_CARRIER_ID, ordstat_ptr->s_O_CARRIER_ID);
  fprintf(debug_fp,"\ts_ol_cnt       = %d (%X)\n",
          ordstat_ptr->s_ol_cnt, ordstat_ptr->s_ol_cnt);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          ordstat_ptr->s_transtatus, ordstat_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          ordstat_ptr->deadlocks, ordstat_ptr->deadlocks);

  fprintf(debug_fp,"\titems {\n");
  items = ordstat_ptr->s_ol_cnt;
  for (j = 0; j < items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_SUPPLY_W_ID, ordstat_ptr-
>item[j].s_OL_SUPPLY_W_ID);
    fprintf(debug_fp,"\t\ts_OL_I_ID[%d]        = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_I_ID, ordstat_ptr->item[j].s_OL_I_ID);
    fprintf(debug_fp,"\t\ts_OL_QUANTITY[%d]    = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_QUANTITY, ordstat_ptr->item[j].s_OL_QUANTITY);
    fprintf(debug_fp,"\t\ts_OL_AMOUNT[%d]      = %.2f\n",
            j, ordstat_ptr->item[j].s_OL_AMOUNT);
    fprintf(debug_fp,"\t\ts_OL_DELIVERY_D[%d]  = %s \n",
            j,ordstat_ptr->item[j].s_OL_DELIVERY_D_time);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}


/*-----------------------------------------------------------*/
/* pay_debug                                                 */
/*-----------------------------------------------------------*/
void pay_debug (struct out_payment_struct *payment_ptr,
                struct in_payment_struct  *in_payment,
                char *msg)
{
  char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

  InitializeDebug();
  strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
  strcat(debug_fn, "pay.debug.out");
  pay_print(payment_ptr, in_payment, debug_fn, msg);
}

/*-----------------------------------------------------------*/
/* pay_print                                                 */
/*-----------------------------------------------------------*/
void pay_print (struct out_payment_struct *payment_ptr,
                struct in_payment_struct  *in_payment,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];

  current_tmstmp(&timeStamp[0]);
```

```
  timeStamp[19] = (char)NULL;

  if ((debug_fp = fopen(filename, "a+")) == NULL)
  {
    return;
  }

  fprintf(debug_fp,"Payment debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp, " PID %d ", getpid());
  fprintf(debug_fp,"\n==================================================\n");

  fprintf(debug_fp,"in_payment_struct {\n");
  fprintf(debug_fp,"\ts_H_AMOUNT     = %.2f \n",
        in_payment->s_H_AMOUNT);
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
        in_payment->s_C_ID, in_payment->s_C_ID);
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
        in_payment->s_W_ID, in_payment->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
        in_payment->s_D_ID, in_payment->s_D_ID);
  fprintf(debug_fp,"\ts_C_D_ID       = %d (%X)\n",
        in_payment->s_C_D_ID, in_payment->s_C_D_ID);
  fprintf(debug_fp,"\ts_C_W_ID       = %d (%X)\n",
        in_payment->s_C_W_ID, in_payment->s_C_W_ID);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
        in_payment->s_C_LAST);
  fprintf(debug_fp,"\n}\n\n");

  fprintf(debug_fp,"out_payment_struct {\n");
  fprintf(debug_fp,"\ts_C_CREDIT_LIM = %.2f\n",
        payment_ptr->s_C_CREDIT_LIM);
  fprintf(debug_fp,"\ts_C_DISCOUNT   = %04.4f\n",
        payment_ptr->s_C_DISCOUNT);
  fprintf(debug_fp,"\ts_C_BALANCE    = %.2f\n",
        payment_ptr->s_C_BALANCE);
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
        payment_ptr->s_C_ID, payment_ptr->s_C_ID);
  fprintf(debug_fp,"\ts_W_STREET_1   = %s\n",
        payment_ptr->s_W_STREET_1);
  fprintf(debug_fp,"\ts_W_STREET_2   = %s\n",
        payment_ptr->s_W_STREET_2);
  fprintf(debug_fp,"\ts_W_CITY       = %s\n",
        payment_ptr->s_W_CITY);
  fprintf(debug_fp,"\ts_W_STATE      = %s\n",
        payment_ptr->s_W_STATE);
  fprintf(debug_fp,"\ts_W_ZIP        = %s\n",
        payment_ptr->s_W_ZIP);
  fprintf(debug_fp,"\ts_D_STREET_1   = %s\n",
        payment_ptr->s_D_STREET_1);
  fprintf(debug_fp,"\ts_D_STREET_2   = %s\n",
        payment_ptr->s_D_STREET_2);
  fprintf(debug_fp,"\ts_D_CITY       = %s\n",
        payment_ptr->s_D_CITY);
  fprintf(debug_fp,"\ts_D_STATE      = %s\n",
        payment_ptr->s_D_STATE);
  fprintf(debug_fp,"\ts_D_ZIP        = %s\n",
        payment_ptr->s_D_ZIP);
  fprintf(debug_fp,"\ts_C_FIRST      = %s\n",
        payment_ptr->s_C_FIRST);
  fprintf(debug_fp,"\ts_C_MIDDLE     = %s\n",
        payment_ptr->s_C_MIDDLE);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
        payment_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_STREET_1   = %s\n",
        payment_ptr->s_C_STREET_1);
  fprintf(debug_fp,"\ts_C_STREET_2   = %s\n",
        payment_ptr->s_C_STREET_2);
  fprintf(debug_fp,"\ts_C_CITY       = %s\n",
        payment_ptr->s_C_CITY);
  fprintf(debug_fp,"\ts_C_STATE      = %s\n",
        payment_ptr->s_C_STATE);
  fprintf(debug_fp,"\ts_C_ZIP        = %s\n",
        payment_ptr->s_C_ZIP);
  fprintf(debug_fp,"\ts_C_PHONE      = %s\n",
        payment_ptr->s_C_PHONE);
  fprintf(debug_fp,"\ts_C_SINCE      = %s \n",
        payment_ptr->s_C_SINCE_time);
  fprintf(debug_fp,"\ts_C_CREDIT     = %s\n",
        payment_ptr->s_C_CREDIT);
  fprintf(debug_fp,"\ts_C_DATA       = %s\n",
        payment_ptr->s_C_DATA);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
        payment_ptr->s_transtatus,payment_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
        payment_ptr->deadlocks,payment_ptr->deadlocks);
  fprintf(debug_fp,"\n}\n\n");
  fclose(debug_fp);
}


/*-----------------------------------------------------------*/
/* stk_debug                                                 */
/*-----------------------------------------------------------*/
void stk_debug (struct out_stocklev_struct *stocklev,
            struct in_stocklev_struct  *in_stocklev,
            char *msg)
{
  char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

  InitializeDebug();
  strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
  strcat(debug_fn, "stk.debug.out");
  stk_print(stocklev, in_stocklev, debug_fn, msg);
}

/*-----------------------------------------------------------*/
/* stk_print                                                 */
/*-----------------------------------------------------------*/
void stk_print (struct out_stocklev_struct *stocklev,
            struct in_stocklev_struct  *in_stocklev,
            char *filename,
            char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];

  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;

  if ((debug_fp = fopen(filename, "a+")) == NULL)
  {
    return;
  }

  fprintf(debug_fp,"Stock level debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp, " PID %d ", getpid());
  fprintf(debug_fp,"\n==================================================\n");

  fprintf(debug_fp,"in_stocklev_struct {\n");
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
        in_stocklev->s_W_ID, in_stocklev->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
        in_stocklev->s_D_ID, in_stocklev->s_D_ID);
  fprintf(debug_fp,"\ts_threshold    = %d (%X)\n",
        in_stocklev->s_threshold, in_stocklev->s_threshold);
  fprintf(debug_fp,"}\n\n");

  fprintf(debug_fp,"out_stocklev_struct {\n");
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
        stocklev->s_transtatus, stocklev->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
        stocklev->deadlocks, stocklev->deadlocks);
  fprintf(debug_fp,"\ts_low_stock    = %d (%X)\n",
        stocklev->s_low_stock, stocklev->s_low_stock);
  fprintf(debug_fp,"}\n\n");
  fclose(debug_fp);
}

void current_tmstmp(char *buf)
{
  time_t t = time(NULL);
  strncpy(buf,ctime(&t),19);
}
```

Src.Common/tpccmisc.c

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 *
 * tpccmisc.c - Miscellaneous routines
 *
 */

#include <stdlib.h>
#include <sys/types.h>
#include <sys/time.h>

double current_time_ms(void);
double current_time(void);


/* Current time in SECONDS, precision SECONDS */
double current_time(void)
{
  /* use time() to get seconds */
  return(time(NULL));
}

/* Current time in SECONDS, precision MILLISECONDS */
double current_time_ms(void)
{
```

```
  /* gettimeofday() returns seconds and microseconds */
  /* convert to fractional seconds */
  struct timeval t;
  gettimeofday(&t,NULL);
  return (t.tv_sec + (double)t.tv_usec/(1000*1000));
}
```

Src.Srv/Makefile

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# Makefile - Makefile for Src.Srv
#
#

include $(TPCC_ROOT)/Makefile.config

#
##########################################################################
#
# Preprocessor, Compiler and Linker Flags
#
##########################################################################
#

PRP_OPTS =              PACKAGE \
                        ISOLATION RR \
                        QUERYOPT 7 \
                        EXPLAIN ALL \
                        MESSAGES $*.prep.msg

INCLUDE =       -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

CFLAGS =    $(CFLAGS_OS) $(INCLUDE) $(CFLAGS_DEBUG) \
                        -D$(DB2EDITION) \
                        -DSQLA_NOLINES -DLINT_ARGS

LDFLAGS =  $(LDFLAGS_STORP) $(LDFLAGS_LIB)

#
##########################################################################
#
# File Collections
#
##########################################################################
#

STORED_PROCS =          new ord del

UTIL_OBJ = $(TPCC_ROOT)/Src.Common/tpccmisc$(OBJEXT) \
                        $(TPCC_ROOT)/Src.Common/tpccdbg$(OBJEXT)

EXE =                   news ords dels

#
##########################################################################
#
# User Targets
#
##########################################################################
#

all:            connect explain catalog $(EXE) install plan disconnect

clean:          connect uncatalog unexplain disconnect
                - $(ERASE) $(TPCC_SPDIR)$(SLASH)news
                - $(ERASE) $(TPCC_SPDIR)$(SLASH)ords
                - $(ERASE) $(TPCC_SPDIR)$(SLASH)dels
                - $(ERASE) *.bnd *.msg *.out *$(OBJEXT) $(EXE) tpcc_all_sql.c
                - $(ERASE) TPCC_ALL.*.plan

#
##########################################################################
#
# Helper Targets
#
##########################################################################
#

catalog:        uncatalog
                - perl $(TPCC_ROOT)$(SLASH)utils$(SLASH)genproc.pl
$(STORED_PROCS)
                - db2 -tvf cat-proc.ddl +o -z cat-proc.out
                - db2 -td% -vf cat-func.ddl +o -z cat-func.out

uncatalog:
                - perl $(TPCC_ROOT)$(SLASH)utils$(SLASH)genproc.pl
$(STORED_PROCS)
                - db2 -td% -vf uncat-func.ddl +o -z uncat-func.out
                - db2 -tvf uncat-proc.ddl +o -z uncat-proc.out

explain:
                - perl $(TPCC_ROOT)$(SLASH)utils$(SLASH)fixup_explain.pl
                - db2 -tvf $(TPCC_ROOT)$(SLASH)utils$(SLASH)EXPLAIN.DDL +o -z
EXPLAIN.out

unexplain:
                - db2 -tvf $(TPCC_ROOT)$(SLASH)utils$(SLASH)UNEXPLAIN.DDL +o -z
UNEXPLAIN.out

connect:
                - db2 connect to $(TPCC_DBNAME)

disconnect:
                - db2 connect reset
                - db2 terminate

plan:
                - db2exfmt -d $(TPCC_DBNAME) -e $(TPCC_SCHEMA) -s
$(TPCC_SCHEMA) -w -1 -n TPCC_ALL -g -# 0 -o TPCC_ALL.exfmt.plan
                - (DB2EXPLN_BUFFER=3000000;export DB2EXPLN_BUFFER; db2expln -
d $(TPCC_DBNAME) -c $(TPCC_SCHEMA) -p TPCC_ALL -s 0 -g -o
TPCC_ALL.expln.plan )

#
##########################################################################
#
# Install Targets
#
##########################################################################
#

install:        $(EXE)
                - mkdir -p $(TPCC_SPDIR)
                $(COPY) ords $(TPCC_SPDIR)
                $(COPY) news $(TPCC_SPDIR)
                $(COPY) dels $(TPCC_SPDIR)

#
##########################################################################
#
# Build Rules
#
##########################################################################
#

.SUFFIXES: $(OBJEXT) .c .sqc

tpcc_all_sql.c:
                @echo "Prepping $*.sqc"
                db2 prep $*.sqc $(PRP_OPTS)
                db2 grant execute on package TPCC_ALL to public

tpcc_all_sql$(OBJEXT):
                $(CC) -c tpcc_all_sql.c $(CFLAGS) -D$(TPCC_SPTYPE)
$(CFLAGS_OUT)$@

$(EXE): $(UTIL_OBJ) tpcc_all_sql.o
                $(LD_STORP) $(LDFLAGS) $(UTIL_OBJ) tpcc_all_sql.o
$(LDFLAGS_OUT)$@

#
##########################################################################
#
# Dependencies
#
##########################################################################
#

# Executables (Stored Procedures)
$(EXE): $(UTIL_OBJ) tpcc_all_sql.o

# Source
tpcc_all_sql$(OBJEXT): tpcc_all_sql.c

# Headers
tpcc_all_sql.c:                 $(TPCC_ROOT)/include/db2tpcc.h
```

Src.Srv/cat-func.ddl

```
--------------------------------------------------------------------------
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
```

```
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----------------------------------------------------------------------------
--
-- cat-func.ddl - Create table functions
--
--

--
-- DELIVERY
--

CREATE FUNCTION DEL(   W_ID          INTEGER
                     , D_ID          SMALLINT
                     , CARRIER_ID     SMALLINT
                   )

RETURNS TABLE ( O_ID INTEGER )

SPECIFIC DELIVERY

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE O_ID   INTEGER ;
  DECLARE C_ID   INTEGER ;
  DECLARE AMOUNT DECIMAL(12,2) ;

  /* Delete the order from new order table */

  SET VAR.O_ID = ( SELECT NO_O_ID

                 FROM OLD TABLE ( DELETE

                              FROM ( SELECT NO_O_ID

                                   FROM NEW_ORDER

                                  WHERE NO_W_ID = DEL.W_ID
                                    AND NO_D_ID = DEL.D_ID

                                  ORDER BY NO_O_ID ASC

                                  FETCH FIRST 1 ROW ONLY
                                ) AS NEW_ORDER
                            ) AS D
              )
  ;

  /* Update the order as delivered and retrieve the customer id */

  SET VAR.C_ID = ( SELECT O_C_ID

                 FROM OLD TABLE ( UPDATE ORDERS

                              SET O_CARRIER_ID = DEL.CARRIER_ID

                            WHERE O_W_ID = DEL.W_ID
                              AND O_D_ID = DEL.D_ID
                              AND O_ID   = VAR.O_ID
                            ) AS U
              )
  ;

  SET VAR.AMOUNT = ( SELECT SUM( OL_AMOUNT )

                   FROM OLD TABLE ( UPDATE ORDER_LINE

                                SET OL_DELIVERY_D = CURRENT TIMESTAMP

                              WHERE OL_W_ID = DEL.W_ID
                                AND OL_D_ID = DEL.D_ID
                                AND OL_O_ID = VAR.O_ID
                            ) AS U
                )
  ;

  /* Charge the customer */

  UPDATE CUSTOMER

    SET   C_BALANCE      = C_BALANCE + VAR.AMOUNT
        , C_DELIVERY_CNT = C_DELIVERY_CNT + SMALLINT( 1 )

   WHERE C_W_ID = DEL.W_ID
     AND C_D_ID = DEL.D_ID
     AND C_ID   = VAR.C_ID
  ;

  /* Return the order id to the caller (or NULL) */

  RETURN VALUES VAR.O_ID ;

END
%


--
-- ORDER STATUS
--

CREATE FUNCTION ORD_C_LAST(   W_ID INTEGER
                            , D_ID SMALLINT
                            , C_LAST VARCHAR(16)
                          )

RETURNS TABLE(   O_ID         INTEGER
               , O_CARRIER_ID SMALLINT
               , O_ENTRY_D    TIMESTAMP
               , C_BALANCE    DECIMAL(12,2)
               , C_FIRST      VARCHAR(16)
               , C_MIDDLE     CHAR(2)
               , C_ID         INTEGER
             )

SPECIFIC ORD_C_LAST

READS SQL DATA NO EXTERNAL ACTION DETERMINISTIC LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE C_BALANCE    DECIMAL(12,2) ;
  DECLARE C_FIRST      VARCHAR(16) ;
  DECLARE C_MIDDLE     CHAR(2) ;
  DECLARE C_ID         INTEGER ;
  DECLARE O_ID         INTEGER;
  DECLARE O_CARRIER_ID SMALLINT;
  DECLARE O_ENTRY_D    TIMESTAMP;

  /* Retrieve the Customer information */

  SET ( C_BALANCE, C_FIRST, C_MIDDLE, C_ID )

    = ( SELECT  C_BALANCE, C_FIRST, C_MIDDLE , C_ID

        FROM ( SELECT   C_ID
                      , C_BALANCE
                      , C_FIRST
                      , C_MIDDLE
                      , COUNT(*) OVER() AS COUNT
                      , ROWNUMBER() OVER (ORDER BY C_FIRST) AS NUM

               FROM CUSTOMER
              WHERE C_W_ID = ORD_C_LAST.W_ID
                AND C_D_ID = ORD_C_LAST.D_ID
                AND C_LAST = ORD_C_LAST.C_LAST

            ) AS V1

       WHERE NUM = (COUNT + BIGINT( 1 ) ) / BIGINT( 2 )
    )
  ;

  SET ( O_ID , O_CARRIER_ID , O_ENTRY_D )
    =    ( SELECT    O_ID
                   , O_CARRIER_ID
                   , O_ENTRY_D

           FROM ORDERS

          WHERE O_W_ID = ORD_C_LAST.W_ID
            AND O_D_ID = ORD_C_LAST.D_ID
            AND O_C_ID = VAR.C_ID

          ORDER BY O_ID DESC
          FETCH FIRST 1 ROW ONLY

       )
  ;


  RETURN VALUES (   VAR.O_ID
                  , VAR.O_CARRIER_ID
                  , VAR.O_ENTRY_D
                  , VAR.C_BALANCE
                  , VAR.C_FIRST
                  , VAR.C_MIDDLE
                  , VAR.C_ID
                )
  ;


END
%

CREATE FUNCTION ORD_C_ID(   W_ID INTEGER
                          , D_ID SMALLINT
                          , C_ID INTEGER
                        )
```

```
RETURNS TABLE(   O_ID         INTEGER
            , O_CARRIER_ID SMALLINT
            , O_ENTRY_D    TIMESTAMP
            , C_BALANCE    DECIMAL(12,2)
            , C_FIRST      VARCHAR(16)
            , C_MIDDLE     CHAR(2)
            , C_LAST       VARCHAR(16)
          )

SPECIFIC ORD_C_ID

READS SQL DATA NO EXTERNAL ACTION DETERMINISTIC LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE C_BALANCE    DECIMAL(12,2) ;
  DECLARE C_FIRST      VARCHAR(16) ;
  DECLARE C_MIDDLE     CHAR(2) ;
  DECLARE C_LAST       VARCHAR(16) ;
  DECLARE O_ID         INTEGER;
  DECLARE O_CARRIER_ID SMALLINT;
  DECLARE O_ENTRY_D    TIMESTAMP;

  /* Retrieve the Customer information */

  SET ( C_BALANCE, C_FIRST, C_MIDDLE, C_LAST )

    = ( SELECT  C_BALANCE, C_FIRST, C_MIDDLE, C_LAST

          FROM CUSTOMER

         WHERE C_ID   = ORD_C_ID.C_ID
           AND C_W_ID = ORD_C_ID.W_ID
           AND C_D_ID = ORD_C_ID.D_ID
      )
  ;

  SET (O_ID, O_CARRIER_ID, O_ENTRY_D)

    =    ( SELECT   O_ID
              , O_CARRIER_ID
              , O_ENTRY_D

            FROM ORDERS

                WHERE O_W_ID = ORD_C_ID.W_ID
                  AND O_D_ID = ORD_C_ID.D_ID
                  AND O_C_ID = ORD_C_ID.C_ID

           ORDER BY O_ID DESC
           FETCH FIRST 1 ROW ONLY
        )
  ;

  RETURN VALUES (   VAR.O_ID
              , VAR.O_CARRIER_ID
              , VAR.O_ENTRY_D
              , VAR.C_BALANCE
              , VAR.C_FIRST
              , VAR.C_MIDDLE
              , VAR.C_LAST
            ) ;

END
%



--
-- PAYMENT
--

CREATE FUNCTION PAY_C_LAST(  W_ID INTEGER
                        , D_ID SMALLINT
                        , C_W_ID INTEGER
                        , C_D_ID SMALLINT
                        , C_LAST VARCHAR(16)
                        , H_AMOUNT DECIMAL(6,2)
                        , BAD_CREDIT_PREFIX VARCHAR(28)
                       )

RETURNS TABLE(   W_STREET_1 CHAR(20)
            , W_STREET_2 CHAR(20)
            , W_CITY CHAR(20)
            , W_STATE CHAR(2)
            , W_ZIP CHAR(9)
            , D_STREET_1 CHAR(20)
            , D_STREET_2 CHAR(20)
            , D_CITY CHAR(20)
            , D_STATE CHAR(2)
            , D_ZIP CHAR(9)
            , C_ID INTEGER
            , C_FIRST VARCHAR(16)
            , C_MIDDLE CHAR(2)
            , C_STREET_1 VARCHAR(20)
            , C_STREET_2 VARCHAR(20)
            , C_CITY VARCHAR(20)
            , C_STATE CHAR(2)
            , C_ZIP CHAR(9)
            , C_PHONE CHAR(16)
            , C_SINCE TIMESTAMP
            , C_CREDIT CHAR(2)
            , C_CREDIT_LIM DECIMAL(12,2)
            , C_DISCOUNT INTEGER
            , C_BALANCE DECIMAL(12,2)
            , C_DATA CHAR(200)
            , H_DATE TIMESTAMP
          )

SPECIFIC PAY_C_LAST

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE  W_NAME CHAR(10) ;
  DECLARE  D_NAME CHAR(10) ;

  DECLARE  W_STREET_1 CHAR(20) ;
  DECLARE  W_STREET_2 CHAR(20) ;
  DECLARE  W_CITY CHAR(20) ;
  DECLARE  W_STATE CHAR(2) ;
  DECLARE  W_ZIP CHAR(9) ;

  DECLARE  D_STREET_1 CHAR(20) ;
  DECLARE  D_STREET_2 CHAR(20) ;
  DECLARE  D_CITY CHAR(20) ;
  DECLARE  D_STATE CHAR(2) ;
  DECLARE  D_ZIP CHAR(9) ;

  DECLARE  C_ID INTEGER ;

  DECLARE  C_FIRST VARCHAR(16) ;
  DECLARE  C_MIDDLE CHAR(2) ;
  DECLARE  C_STREET_1 VARCHAR(20) ;
  DECLARE  C_STREET_2 VARCHAR(20) ;
  DECLARE  C_CITY VARCHAR(20) ;
  DECLARE  C_STATE CHAR(2) ;
  DECLARE  C_ZIP CHAR(9) ;
  DECLARE  C_PHONE CHAR(16) ;
  DECLARE  C_SINCE TIMESTAMP ;
  DECLARE  C_CREDIT CHAR(2) ;
  DECLARE  C_CREDIT_LIM DECIMAL(12,2) ;
  DECLARE  C_DISCOUNT INTEGER ;
  DECLARE  C_BALANCE DECIMAL(12,2) ;
  DECLARE  C_DATA CHAR(200) ;

  DECLARE  H_DATE TIMESTAMP ;

  /* Generate the current date and time for the payment date */
  SET H_DATE = CURRENT TIMESTAMP ;

  /* Update District and retieve its data */

  SET ( D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP)

    =  ( SELECT   D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP

          FROM OLD TABLE ( UPDATE DISTRICT

                        SET D_YTD = D_YTD + PAY_C_LAST.H_AMOUNT

                       WHERE D_W_ID = PAY_C_LAST.W_ID
                         AND D_ID   = PAY_C_LAST.D_ID
                     ) AS U
      )
  ;

  /* Determine the C_ID */

  SET ( C_ID )

    = ( SELECT C_ID
        FROM ( SELECT   C_ID
                  , COUNT(*) OVER() AS COUNT
                  , ROWNUMBER() OVER (ORDER BY C_FIRST) AS NUM

               FROM CUSTOMER
               WHERE C_LAST = PAY_C_LAST.C_LAST
                 AND C_W_ID = PAY_C_LAST.C_W_ID
                 AND C_D_ID = PAY_C_LAST.C_D_ID
             ) AS T

        WHERE NUM = (COUNT + BIGINT( 1 ) ) / BIGINT( 2 )
      )
  ;


  /* Update the middle customer */

  SET (   C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
        , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
        , C_DISCOUNT, C_BALANCE, C_DATA )
```

```
   = ( SELECT   C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
            , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT,
C_CREDIT_LIM
            , C_DISCOUNT, C_BALANCE
            , CASE WHEN C_CREDIT = 'BC' THEN SUBSTR(C_DATA, 1, 200) ELSE
NULL END AS C_DATA

        FROM NEW TABLE ( UPDATE CUSTOMER

                        SET   C_BALANCE     = C_BALANCE - PAY_C_LAST.H_AMOUNT
                            , C_YTD_PAYMENT = C_YTD_PAYMENT +
PAY_C_LAST.H_AMOUNT
                            , C_PAYMENT_CNT = C_PAYMENT_CNT + SMALLINT( 1 )

                            , C_DATA = CASE WHEN C_CREDIT = 'BC'
                                       THEN    CHAR( C_ID )            -- 11 bytes long
                                           || BAD_CREDIT_PREFIX        -- 28 bytes long
                                           || SUBSTR( C_DATA, 1, 461 ) -- 461 + 39 = 500
                                       ELSE C_DATA
                                     END

                      WHERE C_W_ID = PAY_C_LAST.C_W_ID
                        AND C_D_ID = PAY_C_LAST.C_D_ID
                        AND C_ID   = VAR.C_ID
                    ) AS U
    )
  ;

  /* Update the warehouse */

  SET ( W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP )

   = ( SELECT W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP

        FROM OLD TABLE ( UPDATE WAREHOUSE

                        SET W_YTD = W_YTD + PAY_C_LAST.H_AMOUNT

                      WHERE W_ID = PAY_C_LAST.W_ID
                    ) AS U
    )
  ;

  /* Finally insert into the warehouse */

  INSERT

    INTO HISTORY ( H_C_ID, H_C_D_ID, H_C_W_ID, H_D_ID, H_W_ID, H_DATA,
H_DATE, H_AMOUNT )

    VALUES (   VAR.C_ID
            , PAY_C_LAST.C_D_ID
            , PAY_C_LAST.C_W_ID
            , PAY_C_LAST.D_ID
            , PAY_C_LAST.W_ID
            , VAR.W_NAME || CHAR( '    ', 4 )  || VAR.D_NAME
            , VAR.H_DATE
            , PAY_C_LAST.H_AMOUNT
          )
  ;

  /* Done - return the collected data */

  RETURN VALUES (   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
                , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
                , C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
                , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT,
C_CREDIT_LIM
                , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE
              )
  ;

END
%

CREATE FUNCTION PAY_C_ID(  W_ID INTEGER
                  , D_ID SMALLINT
                  , C_W_ID INTEGER
                  , C_D_ID SMALLINT
                  , C_ID INTEGER
                  , H_AMOUNT DECIMAL(6,2)
                  , BAD_CREDIT_PREFIX VARCHAR(34)
                 )

RETURNS TABLE(   W_STREET_1 CHAR(20)
            , W_STREET_2 CHAR(20)
            , W_CITY CHAR(20)
            , W_STATE CHAR(2)
            , W_ZIP CHAR(9)
            , D_STREET_1 CHAR(20)
            , D_STREET_2 CHAR(20)
            , D_CITY CHAR(20)
            , D_STATE CHAR(2)
            , D_ZIP CHAR(9)
            , C_LAST VARCHAR(16)
            , C_FIRST VARCHAR(16)
            , C_MIDDLE CHAR(2)
            , C_STREET_1 VARCHAR(20)
            , C_STREET_2 VARCHAR(20)
            , C_CITY VARCHAR(20)
            , C_STATE CHAR(2)
            , C_ZIP CHAR(9)
            , C_PHONE CHAR(16)
            , C_SINCE TIMESTAMP
            , C_CREDIT CHAR(2)
            , C_CREDIT_LIM DECIMAL(12,2)
            , C_DISCOUNT REAL
            , C_BALANCE DECIMAL(12,2)
            , C_DATA CHAR(200)
            , H_DATE TIMESTAMP
          )
SPECIFIC PAY_C_ID

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE  W_NAME CHAR(10) ;
  DECLARE  D_NAME CHAR(10) ;

  DECLARE  W_STREET_1 CHAR(20) ;
  DECLARE  W_STREET_2 CHAR(20) ;
  DECLARE  W_CITY CHAR(20) ;
  DECLARE  W_STATE CHAR(2) ;
  DECLARE  W_ZIP CHAR(9) ;

  DECLARE  D_STREET_1 CHAR(20) ;
  DECLARE  D_STREET_2 CHAR(20) ;
  DECLARE  D_CITY CHAR(20) ;
  DECLARE  D_STATE CHAR(2) ;
  DECLARE  D_ZIP CHAR(9) ;

  DECLARE  C_LAST VARCHAR(16) ;

  DECLARE  C_FIRST VARCHAR(16) ;
  DECLARE  C_MIDDLE CHAR(2) ;
  DECLARE  C_STREET_1 VARCHAR(20) ;
  DECLARE  C_STREET_2 VARCHAR(20) ;
  DECLARE  C_CITY VARCHAR(20) ;
  DECLARE  C_STATE CHAR(2) ;
  DECLARE  C_ZIP CHAR(9) ;
  DECLARE  C_PHONE CHAR(16) ;
  DECLARE  C_SINCE TIMESTAMP ;
  DECLARE  C_CREDIT CHAR(2) ;
  DECLARE  C_CREDIT_LIM DECIMAL(12,2) ;
  DECLARE  C_DISCOUNT REAL ;
  DECLARE  C_BALANCE DECIMAL(12,2) ;
  DECLARE  C_DATA CHAR(200) ;
  DECLARE  H_DATE TIMESTAMP;

  /* Generate the current date and time for the payment date */
  SET H_DATE = CURRENT TIMESTAMP;

  /* Update District and retieve its data */

  SET ( D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP )

   =  ( SELECT   D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP

        FROM OLD TABLE ( UPDATE DISTRICT

                        SET D_YTD = D_YTD + PAY_C_ID.H_AMOUNT

                      WHERE D_W_ID = PAY_C_ID.W_ID
                        AND D_ID   = PAY_C_ID.D_ID
                    ) AS U
    )
  ;

  /* Update the middle customer */

  SET (   C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
      , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
      , C_DISCOUNT, C_BALANCE, C_DATA )

   = ( SELECT   C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
            , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT,
C_CREDIT_LIM
            , C_DISCOUNT, C_BALANCE
            , CASE WHEN C_CREDIT = 'BC' THEN SUBSTR(C_DATA, 1, 200) ELSE
NULL END AS C_DATA

        FROM NEW TABLE ( UPDATE CUSTOMER

                      SET   C_BALANCE     = C_BALANCE - PAY_C_ID.H_AMOUNT
                          , C_YTD_PAYMENT = C_YTD_PAYMENT +
PAY_C_ID.H_AMOUNT
                          , C_PAYMENT_CNT = C_PAYMENT_CNT + SMALLINT( 1 )

                          , C_DATA = CASE WHEN C_CREDIT = 'BC'
                                     THEN    BAD_CREDIT_PREFIX        -- 34 bytes long
                                         || SUBSTR( C_DATA, 1, 466 ) -- 466 + 34 = 500 bytes
                                     ELSE C_DATA
```

```
                                END

                        WHERE C_W_ID = PAY_C_ID.C_W_ID
                          AND C_D_ID = PAY_C_ID.C_D_ID
                          AND C_ID   = PAY_C_ID.C_ID
                      ) AS U
    )
  ;

  /* Update the warehouse */

  SET ( W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP )

   = ( SELECT W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP

        FROM OLD TABLE ( UPDATE WAREHOUSE

                          SET W_YTD = W_YTD + PAY_C_ID.H_AMOUNT

                        WHERE W_ID = PAY_C_ID.W_ID
                      ) AS U
    )
  ;

  /* Finally insert into the warehouse */

  INSERT

    INTO HISTORY ( H_C_ID, H_C_D_ID, H_C_W_ID, H_D_ID, H_W_ID, H_DATA,
H_DATE, H_AMOUNT )

    VALUES (   PAY_C_ID.C_ID
             , PAY_C_ID.C_D_ID
             , PAY_C_ID.C_W_ID
             , PAY_C_ID.D_ID
             , PAY_C_ID.W_ID
             , VAR.W_NAME || CHAR( '    ', 4 ) || VAR.D_NAME
             , VAR.H_DATE
             , PAY_C_ID.H_AMOUNT
           )
  ;

  /* Done - return the collected data */

  RETURN VALUES (   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
                  , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
                  , C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
                  , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT,
C_CREDIT_LIM
                  , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE
                )
  ;

END
%

--
-- NEW ORDER
--

CREATE FUNCTION NEW_OL_ALL(   I_ID      INT
                            , I_QTY     SMALLINT
                            , W_ID      INT
                            , SUPP_W_ID INT
                            , O_ID      INT
                            , D_ID      SMALLINT
                            )

RETURNS TABLE(   I_PRICE       DECIMAL(5,2)
               , I_NAME        CHAR(24)
               , I_DATA        VARCHAR(50)
               , OL_DIST_INFO  CHAR(24)
               , S_DATA        VARCHAR(50)
               , S_QUANTITY    SMALLINT
               )

SPECIFIC NEW_OL_ALL

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE I_PRICE       DECIMAL(5,2) ;
  DECLARE I_NAME        CHAR(24)  ;
  DECLARE I_DATA        VARCHAR(50)  ;
  DECLARE OL_DIST_INFO  CHAR(24)  ;
  DECLARE S_DATA        VARCHAR(50)  ;
  DECLARE S_QUANTITY    SMALLINT  ;


  SET   ( I_PRICE , I_NAME , I_DATA )

    =  ( SELECT
               I_PRICE
             , I_NAME
             , I_DATA
```

```
        FROM ITEM

       WHERE ITEM.I_ID = NEW_OL_ALL.I_ID
     ) ;

   SET ( OL_DIST_INFO , S_DATA , S_QUANTITY )

    = ( SELECT   OL_DIST_INFO
               , S_DATA
               , S_QUANTITY

        FROM NEW TABLE (    UPDATE STOCK

                      INCLUDE (  OL_DIST_INFO CHAR( 24 )  )

                        SET S_QUANTITY = CASE WHEN S_QUANTITY -
NEW_OL_ALL.I_QTY >= 10
                                    THEN S_QUANTITY - NEW_OL_ALL.I_QTY
                                    ELSE S_QUANTITY - NEW_OL_ALL.I_QTY + 91
                                  END

                                  , S_ORDER_CNT = S_ORDER_CNT + SMALLINT( 1 )

                                  , S_YTD = S_YTD + NEW_OL_ALL.I_QTY

                                  , S_REMOTE_CNT = CASE WHEN
NEW_OL_ALL.SUPP_W_ID = NEW_OL_ALL.W_ID
                                              THEN S_REMOTE_CNT
                                              ELSE S_REMOTE_CNT + SMALLINT( 1 )
                                            END

                                  , OL_DIST_INFO = CASE D_ID WHEN SMALLINT(  1 )
THEN S_DIST_01
                                                 WHEN SMALLINT(  2 )  THEN
S_DIST_02
                                                 WHEN SMALLINT(  3 )  THEN
S_DIST_03
                                                 WHEN SMALLINT(  4 )  THEN
S_DIST_04
                                                 WHEN SMALLINT(  5 )  THEN
S_DIST_05
                                                 WHEN SMALLINT(  6 )  THEN
S_DIST_06
                                                 WHEN SMALLINT(  7 )  THEN
S_DIST_07
                                                 WHEN SMALLINT(  8 )  THEN
S_DIST_08
                                                 WHEN SMALLINT(  9 )  THEN
S_DIST_09
                                                 WHEN SMALLINT( 10 )  THEN
S_DIST_10
                                            END
                        WHERE S_I_ID = NEW_OL_ALL.I_ID
                          AND S_W_ID = NEW_OL_ALL.SUPP_W_ID
                      ) AS U
    )
  ;

  RETURN VALUES(   VAR.I_PRICE
                 , VAR.I_NAME
                 , VAR.I_DATA
                 , VAR.OL_DIST_INFO
                 , VAR.S_DATA
                 , VAR.S_QUANTITY
                )
  ;

END
%

CREATE FUNCTION NEW_OL_LOCAL(   I_ID      INT
                              , I_QTY     SMALLINT
                              , W_ID      INT
                              , O_ID      INT
                              , D_ID      SMALLINT
                              )

RETURNS TABLE(   I_PRICE       DECIMAL(5,2)
               , I_NAME        CHAR(24)
               , I_DATA        VARCHAR(50)
               , OL_DIST_INFO  CHAR(24)
               , S_DATA        VARCHAR(50)
               , S_QUANTITY    SMALLINT
               )

SPECIFIC NEW_OL_LOCAL

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE I_PRICE       DECIMAL(5,2)  ;
  DECLARE I_NAME        CHAR(24)  ;
  DECLARE I_DATA        VARCHAR(50)  ;
  DECLARE OL_DIST_INFO  CHAR(24)  ;
  DECLARE S_DATA        VARCHAR(50)  ;
  DECLARE S_QUANTITY    SMALLINT  ;
```

```
 SET   ( I_PRICE , I_NAME , I_DATA )

  =  ( SELECT
             I_PRICE
           , I_NAME
           , I_DATA

        FROM ITEM

       WHERE ITEM.I_ID = NEW_OL_LOCAL.I_ID
     ) ;

   SET ( OL_DIST_INFO , S_DATA , S_QUANTITY )

    = ( SELECT   OL_DIST_INFO
           , S_DATA
           , S_QUANTITY

         FROM NEW TABLE (    UPDATE STOCK

                       INCLUDE (  OL_DIST_INFO CHAR( 24 )  )

                         SET S_QUANTITY = CASE WHEN S_QUANTITY -
NEW_OL_LOCAL.I_QTY >= 10
                                    THEN S_QUANTITY - NEW_OL_LOCAL.I_QTY
                                    ELSE S_QUANTITY - NEW_OL_LOCAL.I_QTY + 91
                                 END

                                 , S_ORDER_CNT = S_ORDER_CNT + SMALLINT( 1 )

                                 , S_YTD = S_YTD + NEW_OL_LOCAL.I_QTY

                                 , OL_DIST_INFO = CASE D_ID WHEN SMALLINT(  1 )
THEN S_DIST_01
                                              WHEN SMALLINT(  2 )  THEN
S_DIST_02
                                              WHEN SMALLINT(  3 )  THEN
S_DIST_03
                                              WHEN SMALLINT(  4 )  THEN
S_DIST_04
                                              WHEN SMALLINT(  5 )  THEN
S_DIST_05
                                              WHEN SMALLINT(  6 )  THEN
S_DIST_06
                                              WHEN SMALLINT(  7 )  THEN
S_DIST_07
                                              WHEN SMALLINT(  8 )  THEN
S_DIST_08
                                              WHEN SMALLINT(  9 )  THEN
S_DIST_09
                                              WHEN SMALLINT( 10 )  THEN
S_DIST_10
                                         END
                       WHERE S_I_ID = NEW_OL_LOCAL.I_ID
                         AND S_W_ID = NEW_OL_LOCAL.W_ID
                    ) AS U
     )
   ;

   RETURN VALUES(   VAR.I_PRICE
              , VAR.I_NAME
              , VAR.I_DATA
              , VAR.OL_DIST_INFO
              , VAR.S_DATA
              , VAR.S_QUANTITY
            )
   ;

END
%

CREATE FUNCTION NEW_WH (   O_ID        INTEGER
                    , W_ID       INTEGER
                    , D_ID       SMALLINT
                    , C_ID       INTEGER
                    , O_OL_CNT    SMALLINT
                    , O_ALL_LOCAL SMALLINT
                  )

RETURNS TABLE (   W_TAX       REAL
            , C_DISCOUNT  REAL
            , C_LAST      VARCHAR(16)
            , C_CREDIT    CHAR(2)
            , O_ENTRY_D   TIMESTAMP
          )

SPECIFIC NEW_WH

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE C_DISCOUNT  REAL ;
  DECLARE C_LAST      VARCHAR(16) ;
  DECLARE C_CREDIT    CHAR(2) ;
  DECLARE W_TAX       REAL ;
  DECLARE O_ENTRY_D   TIMESTAMP;

  SET O_ENTRY_D = CURRENT TIMESTAMP;

  INSERT

   INTO NEW_ORDER ( NO_O_ID, NO_D_ID, NO_W_ID )

         VALUES (   O_ID
                  , D_ID
                  , W_ID
                )
  ;


  INSERT

   INTO ORDERS ( O_C_ID, O_ENTRY_D, O_CARRIER_ID, O_OL_CNT,
O_ALL_LOCAL, O_ID, O_W_ID, O_D_ID )

       VALUES (   C_ID
                , O_ENTRY_D
                , 0
                , O_OL_CNT
                , O_ALL_LOCAL
                , O_ID
                , W_ID
                , D_ID
              )
  ;

  SET ( C_DISCOUNT, C_LAST, C_CREDIT )

   =  ( SELECT  C_DISCOUNT, C_LAST, C_CREDIT

        FROM  CUSTOMER

       WHERE  C_ID = NEW_WH.C_ID
         AND  C_W_ID = W_ID
         AND  C_D_ID = D_ID
     )
  ;


  SET W_TAX
   =    ( SELECT W_TAX

          FROM WAREHOUSE

         WHERE W_ID = NEW_WH.W_ID
       )
  ;

  RETURN VALUES ( W_TAX , C_DISCOUNT , C_LAST , C_CREDIT, O_ENTRY_D ) ;

END
%
```

Src.Srv/cat-proc.ddl

```
CREATE PROCEDURE news
                 (in new_in   varchar(262) FOR BIT DATA,
                  out new_out varchar(682) FOR BIT DATA)
                 LANGUAGE C
                 PARAMETER STYLE GENERAL
                 EXTERNAL NAME '/autobench/sources/db2_tpcc-
1/users/tpcc/sqllib/function/news!news'
                 not fenced;

CREATE PROCEDURE ords
                 (in ord_in   varchar(42) FOR BIT DATA,
                  out ord_out varchar(822) FOR BIT DATA)
                 LANGUAGE C
                 PARAMETER STYLE GENERAL
                 EXTERNAL NAME '/autobench/sources/db2_tpcc-
1/users/tpcc/sqllib/function/ords!ords'
                 not fenced;

CREATE PROCEDURE dels
                 (in del_in   varchar(14) FOR BIT DATA,
                  out del_out varchar(50) FOR BIT DATA)
                 LANGUAGE C
                 PARAMETER STYLE GENERAL
                 EXTERNAL NAME '/autobench/sources/db2_tpcc-
1/users/tpcc/sqllib/function/dels!dels'
                 not fenced;
```

Src.Srv/tpcc_all_sql.sqc

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
```

```
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*******************************************************************************/

/*
 * tpcc_all_sql.sqc - Client/Server code for TPCC
 *
 */

#include <stdlib.h>
#include <errno.h>
#include "db2tpcc.h"
#include "tpccapp.h"
#include "tpccdbg.h"

#include "sqlca.h"
#include "sql.h"

// ---------------------------------------------------------------
// New Order  SERVER
// ---------------------------------------------------------------

int  static is_ORIGINAL( char *string, short length ) ;

SQL_API_RC new_order_internal( char *pin, char *pout )
{

   struct out_neword_struct *neword;

   struct in_neword_struct  *in_neword;

   struct sqlca sqlca ;

   int     fbadItemDetected = 0 ;

   EXEC SQL BEGIN DECLARE SECTION;

      char    c_last [ 16 ] ;
      char    c_credit [ 2 ] ;
      float  c_discount ;
      float  dist_tax ;
      float  ware_tax ;

      sqlint32   w_id ;
      short      d_id ;
      sqlint32   c_id ;

      sqlint32   next_o_id ;

      short      s_quantity ;

      sqlint32   supply_w_id ;

      short   inputItemCount ;

      char    stockDistrictInformation [ 24 ];
      char    item_name[ 24 ] ;

      char    o_entry_d [27];

      short   allLocal ;

      float    item_price ;

      struct i_data_type { short len ; char data[ 50 ] ; } i_data ;
      struct s_data_type { short len ; char data[ 50 ] ; } s_data ;

      sqlint32    id0, id1, id2,  id3,  id4,  id5,  id6,  id7;
      sqlint32    id8, id9, id10, id11, id12, id13, id14;

      sqlint32   supply_w_id0,  supply_w_id1, supply_w_id2,  supply_w_id3;
      sqlint32   supply_w_id4,  supply_w_id5, supply_w_id6,  supply_w_id7;
      sqlint32   supply_w_id8,  supply_w_id9, supply_w_id10,  supply_w_id11;
      sqlint32   supply_w_id12,  supply_w_id13, supply_w_id14;

      short   ol_quantity0,  ol_quantity1, ol_quantity2,  ol_quantity3;
      short   ol_quantity4,  ol_quantity5, ol_quantity6,  ol_quantity7;
      short   ol_quantity8,  ol_quantity9, ol_quantity10,  ol_quantity11;
      short   ol_quantity12,  ol_quantity13, ol_quantity14;


   EXEC SQL END DECLARE SECTION;

   int      storedProcRc ;
   int      inputItemArrayIndex ;

   char    stockDistrictInformationArray [15][25];

   #define stockDistrictInformation  stockDistrictInformationArray[ inputItemArrayIndex ]

   // Redirected input fields

   #define w_id in_neword->s_W_ID
   #define d_id in_neword->s_D_ID
   #define c_id in_neword->s_C_ID

   #define inputItemCount in_neword->s_O_OL_CNT

   #define allLocal in_neword->s_all_local

   // Redirected output fields

   #define  c_last       neword->s_C_LAST
   #define  c_credit     neword->s_C_CREDIT
   #define  c_discount   neword->s_C_DISCOUNT
   #define  ware_tax     neword->s_W_TAX
   #define  dist_tax     neword->s_D_TAX
   #define  s_quantity   neword->item[ inputItemArrayIndex ].s_S_QUANTITY
   #define o_entry_d     neword->s_O_ENTRY_D_time

   // This output field becomes an input field to order_line

   #define next_o_id neword->s_O_ID

   // item price/name

   #define  item_name    neword->item[ inputItemArrayIndex ].s_I_NAME

   float    i_priceArray[ 15 ] ;

   #define  item_price   i_priceArray[ inputItemArrayIndex ]

   // Handle the generic/brand distinction

   struct  i_data_type i_dataArray[ 15 ] ;
   struct  s_data_type s_dataArray[ 15 ] ;

   #define  i_data       i_dataArray[ inputItemArrayIndex ]
   #define  s_data       s_dataArray[ inputItemArrayIndex ]

   // Redirect hostvars to input structure

   #define  id0  in_neword->in_item[0].s_OL_I_ID
   #define  id1  in_neword->in_item[1].s_OL_I_ID
   #define  id2  in_neword->in_item[2].s_OL_I_ID
   #define  id3  in_neword->in_item[3].s_OL_I_ID
   #define  id4  in_neword->in_item[4].s_OL_I_ID
   #define  id5  in_neword->in_item[5].s_OL_I_ID
   #define  id6  in_neword->in_item[6].s_OL_I_ID
   #define  id7  in_neword->in_item[7].s_OL_I_ID
   #define  id8  in_neword->in_item[8].s_OL_I_ID
   #define  id9  in_neword->in_item[9].s_OL_I_ID
   #define  id10 in_neword->in_item[10].s_OL_I_ID
   #define  id11 in_neword->in_item[11].s_OL_I_ID
   #define  id12 in_neword->in_item[12].s_OL_I_ID
   #define  id13 in_neword->in_item[13].s_OL_I_ID
   #define  id14 in_neword->in_item[14].s_OL_I_ID

   #define ol_quantity0  in_neword->in_item[  0 ].s_OL_QUANTITY
   #define ol_quantity1  in_neword->in_item[  1 ].s_OL_QUANTITY
   #define ol_quantity2  in_neword->in_item[  2 ].s_OL_QUANTITY
   #define ol_quantity3  in_neword->in_item[  3 ].s_OL_QUANTITY
   #define ol_quantity4  in_neword->in_item[  4 ].s_OL_QUANTITY
   #define ol_quantity5  in_neword->in_item[  5 ].s_OL_QUANTITY
   #define ol_quantity6  in_neword->in_item[  6 ].s_OL_QUANTITY
   #define ol_quantity7  in_neword->in_item[  7 ].s_OL_QUANTITY
   #define ol_quantity8  in_neword->in_item[  8 ].s_OL_QUANTITY
   #define ol_quantity9  in_neword->in_item[  9 ].s_OL_QUANTITY
   #define ol_quantity10 in_neword->in_item[ 10 ].s_OL_QUANTITY
   #define ol_quantity11 in_neword->in_item[ 11 ].s_OL_QUANTITY
   #define ol_quantity12 in_neword->in_item[ 12 ].s_OL_QUANTITY
   #define ol_quantity13 in_neword->in_item[ 13 ].s_OL_QUANTITY
   #define ol_quantity14 in_neword->in_item[ 14 ].s_OL_QUANTITY

   #define supply_w_id0  in_neword->in_item[  0 ].s_OL_SUPPLY_W_ID
   #define supply_w_id1  in_neword->in_item[  1 ].s_OL_SUPPLY_W_ID
   #define supply_w_id2  in_neword->in_item[  2 ].s_OL_SUPPLY_W_ID
   #define supply_w_id3  in_neword->in_item[  3 ].s_OL_SUPPLY_W_ID
   #define supply_w_id4  in_neword->in_item[  4 ].s_OL_SUPPLY_W_ID
   #define supply_w_id5  in_neword->in_item[  5 ].s_OL_SUPPLY_W_ID
   #define supply_w_id6  in_neword->in_item[  6 ].s_OL_SUPPLY_W_ID
   #define supply_w_id7  in_neword->in_item[  7 ].s_OL_SUPPLY_W_ID
   #define supply_w_id8  in_neword->in_item[  8 ].s_OL_SUPPLY_W_ID
   #define supply_w_id9  in_neword->in_item[  9 ].s_OL_SUPPLY_W_ID
   #define supply_w_id10 in_neword->in_item[ 10 ].s_OL_SUPPLY_W_ID
   #define supply_w_id11 in_neword->in_item[ 11 ].s_OL_SUPPLY_W_ID
   #define supply_w_id12 in_neword->in_item[ 12 ].s_OL_SUPPLY_W_ID
   #define supply_w_id13 in_neword->in_item[ 13 ].s_OL_SUPPLY_W_ID
   #define supply_w_id14 in_neword->in_item[ 14 ].s_OL_SUPPLY_W_ID

   EXEC SQL DECLARE ISOL_Remote_1 CURSOR FOR

     WITH DATA AS (  SELECT  O_ID
                   , D_ID
                   , W_ID
                   , OL_NUMBER
                   , I_ID
                   , I_SUPPLY_W_ID
                   , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
```

```
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                              , :w_id AS W_ID
                              , :d_id as D_ID
                              , OL_NUMBER
                              , I_ID
                              , I_SUPPLY_W_ID
                              , I_QTY

                        FROM  Table( VALUES


(  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                        , TABLE( NEW_OL_ALL(   I_ID
                                    , I_QTY
                                    , W_ID
                                    , I_SUPPLY_W_ID
                                    , O_ID
                                    , D_ID
                                   )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                            , D_ID
                            , W_ID
                            , OL_NUMBER
                            , I_ID
                            , I_SUPPLY_W_ID
                            , OL_DELIVERY_D
                            , I_QTY
                            , TOTAL_PRICE
                            , OL_DIST_INFO
                            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

                ) AS INS
  ;

  EXEC SQL DECLARE ISOL_Remote_2 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                              , :w_id AS W_ID
                              , :d_id as D_ID
                              , OL_NUMBER
                              , I_ID
                              , I_SUPPLY_W_ID
                              , I_QTY

                        FROM  Table( VALUES


(  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                          ,
(  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                        , TABLE( NEW_OL_ALL(   I_ID
                                    , I_QTY
                                    , W_ID
                                    , I_SUPPLY_W_ID
                                    , O_ID
                                    , D_ID
                                   )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                            , D_ID
                            , W_ID
                            , OL_NUMBER
                            , I_ID
                            , I_SUPPLY_W_ID
                            , OL_DELIVERY_D
                            , I_QTY
                            , TOTAL_PRICE
                            , OL_DIST_INFO
                            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

                ) AS INS
  ;

  EXEC SQL DECLARE ISOL_Remote_3 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                              , :w_id AS W_ID
                              , :d_id as D_ID
                              , OL_NUMBER
                              , I_ID
                              , I_SUPPLY_W_ID
                              , I_QTY

                        FROM  Table( VALUES


(  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                          ,
(  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                                          ,
(  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                        ) AS ITEMLIST
```

```
                    , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                 )
                        ) AS NEW_OL_ALL

                    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                          , D_ID
                          , W_ID
                          , OL_NUMBER
                          , I_ID
                          , I_SUPPLY_W_ID
                          , OL_DELIVERY_D
                          , I_QTY
                          , TOTAL_PRICE
                          , OL_DIST_INFO
                          , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

                ) AS INS
  ;

  EXEC SQL DECLARE ISOL_Remote_4 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
              , D_ID
              , W_ID
              , OL_NUMBER
              , I_ID
              , I_SUPPLY_W_ID
              , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
              , I_QTY
              , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
              , OL_DIST_INFO
              , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM ( SELECT   :next_o_id as O_ID
                                , :w_id AS W_ID
                                , :d_id as D_ID
                                , OL_NUMBER
                                , I_ID
                                , I_SUPPLY_W_ID
                                , I_QTY

                            FROM  Table( VALUES

(  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                ,
(  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                                ,
(  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )
                                ,
(  SMALLINT( 4 )        , :id3  , :ol_quantity3  , :supply_w_id3  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                 )
                        ) AS NEW_OL_ALL

                    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                          , D_ID
                          , W_ID
                          , OL_NUMBER
                          , I_ID
                          , I_SUPPLY_W_ID
                          , OL_DELIVERY_D
                          , I_QTY
                          , TOTAL_PRICE
                          , OL_DIST_INFO
                          , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

                ) AS INS
  ;

  EXEC SQL DECLARE ISOL_Remote_5 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
              , D_ID
              , W_ID
              , OL_NUMBER
              , I_ID
              , I_SUPPLY_W_ID
              , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
              , I_QTY
              , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
              , OL_DIST_INFO
              , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM ( SELECT   :next_o_id as O_ID
                                , :w_id AS W_ID
                                , :d_id as D_ID
                                , OL_NUMBER
                                , I_ID
                                , I_SUPPLY_W_ID
                                , I_QTY

                            FROM  Table( VALUES

(  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                ,
(  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                                ,
(  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )
                                ,
(  SMALLINT( 4 )        , :id3  , :ol_quantity3  , :supply_w_id3  )
                                ,
(  SMALLINT( 5 )        , :id4  , :ol_quantity4  , :supply_w_id4  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                 )
                        ) AS NEW_OL_ALL

                    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY
```

```
    FROM NEW TABLE ( INSERT INTO ORDER_LINE

                ( OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE ( I_PRICE  DECIMAL(5,2)
                        , I_NAME   CHAR(24)
                        , I_DATA   VARCHAR(50)
                        , S_DATA   VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT  O_ID
                          , D_ID
                          , W_ID
                          , OL_NUMBER
                          , I_ID
                          , I_SUPPLY_W_ID
                          , OL_DELIVERY_D
                          , I_QTY
                          , TOTAL_PRICE
                          , OL_DIST_INFO
                          , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_6 CURSOR FOR

  WITH DATA AS ( SELECT  O_ID
              , D_ID
              , W_ID
              , OL_NUMBER
              , I_ID
              , I_SUPPLY_W_ID
              , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
              , I_QTY
              , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
              , OL_DIST_INFO
              , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM ( SELECT  :next_o_id as O_ID
                                  , :w_id AS W_ID
                                  , :d_id as D_ID
                                  , OL_NUMBER
                                  , I_ID
                                  , I_SUPPLY_W_ID
                                  , I_QTY

                              FROM  Table( VALUES


( SMALLINT( 1 )      , :id0  , :ol_quantity0  , :supply_w_id0  )
                                  ,
( SMALLINT( 2 )      , :id1  , :ol_quantity1  , :supply_w_id1  )
                                  ,
( SMALLINT( 3 )      , :id2  , :ol_quantity2  , :supply_w_id2  )
                                  ,
( SMALLINT( 4 )      , :id3  , :ol_quantity3  , :supply_w_id3  )
                                  ,
( SMALLINT( 5 )      , :id4  , :ol_quantity4  , :supply_w_id4  )
                                  ,
( SMALLINT( 6 )      , :id5  , :ol_quantity5  , :supply_w_id5  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY        ,
I_SUPPLY_W_ID  )
                          ) AS ITEMLIST

                        , TABLE( NEW_OL_ALL(  I_ID
                                            , I_QTY
                                            , W_ID
                                            , I_SUPPLY_W_ID
                                            , O_ID
                                            , D_ID
                                            )
                              ) AS NEW_OL_ALL

                      WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

    FROM NEW TABLE ( INSERT INTO ORDER_LINE

                ( OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE ( I_PRICE  DECIMAL(5,2)
                        , I_NAME   CHAR(24)
                        , I_DATA   VARCHAR(50)
                        , S_DATA   VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT  O_ID
                          , D_ID
                          , W_ID
                          , OL_NUMBER
                          , I_ID
                          , I_SUPPLY_W_ID
                          , OL_DELIVERY_D
                          , I_QTY
                          , TOTAL_PRICE
                          , OL_DIST_INFO
                          , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_7 CURSOR FOR

  WITH DATA AS ( SELECT  O_ID
              , D_ID
              , W_ID
              , OL_NUMBER
              , I_ID
              , I_SUPPLY_W_ID
              , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
              , I_QTY
              , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
              , OL_DIST_INFO
              , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM ( SELECT  :next_o_id as O_ID
                                  , :w_id AS W_ID
                                  , :d_id as D_ID
                                  , OL_NUMBER
                                  , I_ID
                                  , I_SUPPLY_W_ID
                                  , I_QTY

                              FROM  Table( VALUES


( SMALLINT( 1 )      , :id0  , :ol_quantity0  , :supply_w_id0  )
                                  ,
( SMALLINT( 2 )      , :id1  , :ol_quantity1  , :supply_w_id1  )
                                  ,
( SMALLINT( 3 )      , :id2  , :ol_quantity2  , :supply_w_id2  )
                                  ,
( SMALLINT( 4 )      , :id3  , :ol_quantity3  , :supply_w_id3  )
                                  ,
( SMALLINT( 5 )      , :id4  , :ol_quantity4  , :supply_w_id4  )
                                  ,
( SMALLINT( 6 )      , :id5  , :ol_quantity5  , :supply_w_id5  )
                                  ,
( SMALLINT( 7 )      , :id6  , :ol_quantity6  , :supply_w_id6  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY        ,
I_SUPPLY_W_ID  )
                          ) AS ITEMLIST

                        , TABLE( NEW_OL_ALL(  I_ID
                                            , I_QTY
                                            , W_ID
                                            , I_SUPPLY_W_ID
                                            , O_ID
                                            , D_ID
                                            )
                              ) AS NEW_OL_ALL

                      WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

    FROM NEW TABLE ( INSERT INTO ORDER_LINE

                ( OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
```

```
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_8 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , I_QTY

                        FROM  Table( VALUES


(  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                                ,
(  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                                ,
(  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                                ,
(  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                                ,
(  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                                ,
(  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                                ,
(  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                                ,
(  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY         ,
I_SUPPLY_W_ID  )
                     ) AS ITEMLIST

                     , TABLE( NEW_OL_ALL(   I_ID
                                 , I_QTY
                                 , W_ID
                                 , I_SUPPLY_W_ID
                                 , O_ID
                                 , D_ID
                                )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
```

```
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_9 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , I_QTY

                        FROM  Table( VALUES


(  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                                ,
(  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                                ,
(  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                                ,
(  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                                ,
(  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                                ,
(  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                                ,
(  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                                ,
(  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                                ,
(  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY         ,
I_SUPPLY_W_ID  )
                     ) AS ITEMLIST

                     , TABLE( NEW_OL_ALL(   I_ID
                                 , I_QTY
                                 , W_ID
                                 , I_SUPPLY_W_ID
                                 , O_ID
                                 , D_ID
                                )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (   OL_O_ID
```

```
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_10 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , I_QTY

                       FROM  Table( VALUES

( SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                                  ,
( SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                                  ,
( SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                                  ,
( SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                                  ,
( SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                                  ,
( SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                                  ,
( SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                                  ,
( SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                                  ,
( SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                                  ,
( SMALLINT( 10 )       , :id9  , :ol_quantity9  , :supply_w_id9   )

                                 ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                 )
                       ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )

   SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY
   FROM NEW TABLE ( INSERT INTO ORDER_LINE

               (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_11 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , I_QTY

                       FROM  Table( VALUES

( SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                                  ,
( SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                                  ,
( SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                                  ,
( SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                                  ,
( SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                                  ,
( SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                                  ,
( SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                                  ,
( SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                                  ,
( SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                                  ,
( SMALLINT( 10 )       , :id9  , :ol_quantity9  , :supply_w_id9   )
                                  ,
( SMALLINT( 11 )       , :id10 , :ol_quantity10 , :supply_w_id10  )

                                 ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                 )
                       ) AS NEW_OL_ALL
```

```
                    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )

    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                ( OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE ( I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                           , D_ID
                           , W_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , OL_DELIVERY_D
                           , I_QTY
                           , TOTAL_PRICE
                           , OL_DIST_INFO
                           , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

              ) AS INS
  ;

  EXEC SQL DECLARE ISOL_Remote_12 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
              , D_ID
              , W_ID
              , OL_NUMBER
              , I_ID
              , I_SUPPLY_W_ID
              , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
              , I_QTY
              , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
              , OL_DIST_INFO
              , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM ( SELECT   :next_o_id as O_ID
                                , :w_id AS W_ID
                                , :d_id as D_ID
                                , OL_NUMBER
                                , I_ID
                                , I_SUPPLY_W_ID
                                , I_QTY

                            FROM  Table( VALUES


(  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                ,
(  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                                ,
(  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )
                                ,
(  SMALLINT( 4 )        , :id3  , :ol_quantity3  , :supply_w_id3  )
                                ,
(  SMALLINT( 5 )        , :id4  , :ol_quantity4  , :supply_w_id4  )
                                ,
(  SMALLINT( 6 )        , :id5  , :ol_quantity5  , :supply_w_id5  )
                                ,
(  SMALLINT( 7 )        , :id6  , :ol_quantity6  , :supply_w_id6  )
                                ,
(  SMALLINT( 8 )        , :id7  , :ol_quantity7  , :supply_w_id7  )
                                ,
(  SMALLINT( 9 )        , :id8  , :ol_quantity8  , :supply_w_id8  )
                                ,
(  SMALLINT( 10 )        , :id9  , :ol_quantity9  , :supply_w_id9   )
                                ,
(  SMALLINT( 11 )        , :id10 , :ol_quantity10 , :supply_w_id10  )
                                ,
(  SMALLINT( 12 )        , :id11 , :ol_quantity11 , :supply_w_id11  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                        ) AS ITEMLIST
```

```
                      , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                )
                        ) AS NEW_OL_ALL

                    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )

    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                ( OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE ( I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                           , D_ID
                           , W_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , OL_DELIVERY_D
                           , I_QTY
                           , TOTAL_PRICE
                           , OL_DIST_INFO
                           , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

              ) AS INS
  ;

  EXEC SQL DECLARE ISOL_Remote_13 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
              , D_ID
              , W_ID
              , OL_NUMBER
              , I_ID
              , I_SUPPLY_W_ID
              , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
              , I_QTY
              , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
              , OL_DIST_INFO
              , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM ( SELECT   :next_o_id as O_ID
                                , :w_id AS W_ID
                                , :d_id as D_ID
                                , OL_NUMBER
                                , I_ID
                                , I_SUPPLY_W_ID
                                , I_QTY

                            FROM  Table( VALUES


(  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                ,
(  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                                ,
(  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )
                                ,
(  SMALLINT( 4 )        , :id3  , :ol_quantity3  , :supply_w_id3  )
                                ,
(  SMALLINT( 5 )        , :id4  , :ol_quantity4  , :supply_w_id4  )
                                ,
(  SMALLINT( 6 )        , :id5  , :ol_quantity5  , :supply_w_id5  )
                                ,
(  SMALLINT( 7 )        , :id6  , :ol_quantity6  , :supply_w_id6  )
                                ,
(  SMALLINT( 8 )        , :id7  , :ol_quantity7  , :supply_w_id7  )
                                ,
(  SMALLINT( 9 )        , :id8  , :ol_quantity8  , :supply_w_id8  )
                                ,
(  SMALLINT( 10 )        , :id9  , :ol_quantity9  , :supply_w_id9   )
```

```
                                                ,
(  SMALLINT( 11 )       , :id10 , :ol_quantity10 , :supply_w_id10  )
                                                ,
(  SMALLINT( 12 )       , :id11 , :ol_quantity11 , :supply_w_id11  )
                                                ,
(  SMALLINT( 13 )       , :id12 , :ol_quantity12 , :supply_w_id12  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                            ) AS ITEMLIST

                           , TABLE( NEW_OL_ALL(   I_ID
                                        , I_QTY
                                        , W_ID
                                        , I_SUPPLY_W_ID
                                        , O_ID
                                        , D_ID
                                        )
                              ) AS NEW_OL_ALL

                        WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
                )


     SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

      FROM NEW TABLE ( INSERT INTO ORDER_LINE

                  (   OL_O_ID
                    , OL_D_ID
                    , OL_W_ID
                    , OL_NUMBER
                    , OL_I_ID
                    , OL_SUPPLY_W_ID
                    , OL_DELIVERY_D
                    , OL_QUANTITY
                    , OL_AMOUNT
                    , OL_DIST_INFO
                  )

                  INCLUDE (   I_PRICE    DECIMAL(5,2)
                          , I_NAME     CHAR(24)
                          , I_DATA     VARCHAR(50)
                          , S_DATA     VARCHAR(50)
                          , S_QUANTITY SMALLINT )

                        SELECT   O_ID
                             , D_ID
                             , W_ID
                             , OL_NUMBER
                             , I_ID
                             , I_SUPPLY_W_ID
                             , OL_DELIVERY_D
                             , I_QTY
                             , TOTAL_PRICE
                             , OL_DIST_INFO
                             , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                        FROM DATA

                  ) AS INS
  ;

  EXEC SQL DECLARE ISOL_Remote_14 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                        FROM ( SELECT   :next_o_id as O_ID
                                     , :w_id AS W_ID
                                     , :d_id as D_ID
                                     , OL_NUMBER
                                     , I_ID
                                     , I_SUPPLY_W_ID
                                     , I_QTY

                              FROM  Table( VALUES


(  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                                ,
(  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                                                ,
(  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )
                                                ,
(  SMALLINT( 4 )        , :id3  , :ol_quantity3  , :supply_w_id3  )
                                                ,
(  SMALLINT( 5 )        , :id4  , :ol_quantity4  , :supply_w_id4  )
                                                ,
(  SMALLINT( 6 )        , :id5  , :ol_quantity5  , :supply_w_id5  )
                                                ,
(  SMALLINT( 7 )        , :id6  , :ol_quantity6  , :supply_w_id6  )
                                                ,
(  SMALLINT( 8 )        , :id7  , :ol_quantity7  , :supply_w_id7  )
                                                ,
(  SMALLINT( 9 )        , :id8  , :ol_quantity8  , :supply_w_id8  )
                                                ,
(  SMALLINT( 10 )       , :id9  , :ol_quantity9  , :supply_w_id9   )
                                                ,
(  SMALLINT( 11 )       , :id10 , :ol_quantity10 , :supply_w_id10  )
                                                ,
(  SMALLINT( 12 )       , :id11 , :ol_quantity11 , :supply_w_id11  )
                                                ,
(  SMALLINT( 13 )       , :id12 , :ol_quantity12 , :supply_w_id12  )
                                                ,
(  SMALLINT( 14 )       , :id13 , :ol_quantity13 , :supply_w_id13  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                            ) AS ITEMLIST

                           , TABLE( NEW_OL_ALL(   I_ID
                                        , I_QTY
                                        , W_ID
                                        , I_SUPPLY_W_ID
                                        , O_ID
                                        , D_ID
                                        )
                              ) AS NEW_OL_ALL

                        WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
                )


     SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

      FROM NEW TABLE ( INSERT INTO ORDER_LINE

                  (   OL_O_ID
                    , OL_D_ID
                    , OL_W_ID
                    , OL_NUMBER
                    , OL_I_ID
                    , OL_SUPPLY_W_ID
                    , OL_DELIVERY_D
                    , OL_QUANTITY
                    , OL_AMOUNT
                    , OL_DIST_INFO
                  )

                  INCLUDE (   I_PRICE    DECIMAL(5,2)
                          , I_NAME     CHAR(24)
                          , I_DATA     VARCHAR(50)
                          , S_DATA     VARCHAR(50)
                          , S_QUANTITY SMALLINT )

                        SELECT   O_ID
                             , D_ID
                             , W_ID
                             , OL_NUMBER
                             , I_ID
                             , I_SUPPLY_W_ID
                             , OL_DELIVERY_D
                             , I_QTY
                             , TOTAL_PRICE
                             , OL_DIST_INFO
                             , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                        FROM DATA

                  ) AS INS
  ;

  EXEC SQL DECLARE ISOL_Remote_15 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                        FROM ( SELECT   :next_o_id as O_ID
                                     , :w_id AS W_ID
                                     , :d_id as D_ID
                                     , OL_NUMBER
                                     , I_ID
                                     , I_SUPPLY_W_ID
                                     , I_QTY
```

```
                    FROM  Table( VALUES

(  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                                ,
(  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                                ,
(  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                                ,
(  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                                ,
(  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                                ,
(  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                                ,
(  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                                ,
(  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                                ,
(  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                                ,
(  SMALLINT( 10 )       , :id9  , :ol_quantity9  , :supply_w_id9   )
                                ,
(  SMALLINT( 11 )       , :id10 , :ol_quantity10 , :supply_w_id10  )
                                ,
(  SMALLINT( 12 )       , :id11 , :ol_quantity11 , :supply_w_id11  )
                                ,
(  SMALLINT( 13 )       , :id12 , :ol_quantity12 , :supply_w_id12  )
                                ,
(  SMALLINT( 14 )       , :id13 , :ol_quantity13 , :supply_w_id13  )
                                ,
(  SMALLINT( 15 )       , :id14 , :ol_quantity14 , :supply_w_id14  )

                        ) AS X ( OL_NUMBER , I_ID  , I_QTY          ,
I_SUPPLY_W_ID  )
                  ) AS ITEMLIST

                  , TABLE( NEW_OL_ALL(   I_ID
                                , I_QTY
                                , W_ID
                                , I_SUPPLY_W_ID
                                , O_ID
                                , D_ID
                                )
                      ) AS NEW_OL_ALL

                WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
          )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
              )

              INCLUDE (   I_PRICE    DECIMAL(5,2)
                    , I_NAME     CHAR(24)
                    , I_DATA     VARCHAR(50)
                    , S_DATA     VARCHAR(50)
                    , S_QUANTITY SMALLINT )

                  SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_1 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
          , D_ID
          , W_ID
          , OL_NUMBER
          , I_ID
          , W_ID AS I_SUPPLY_W_ID
          , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
          , I_QTY
          , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
          , OL_DIST_INFO
          , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_QTY

                    FROM  Table( VALUES

                            (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )

                           ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                  ) AS ITEMLIST

                , TABLE( NEW_OL_LOCAL(   I_ID
                              , I_QTY
                              , W_ID
                              , O_ID
                              , D_ID
                              )
                    ) AS NEW_OL_LOCAL

                WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
          )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
              )

              INCLUDE (   I_PRICE    DECIMAL(5,2)
                    , I_NAME     CHAR(24)
                    , I_DATA     VARCHAR(50)
                    , S_DATA     VARCHAR(50)
                    , S_QUANTITY SMALLINT )

                  SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_2 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
          , D_ID
          , W_ID
          , OL_NUMBER
          , I_ID
          , W_ID AS I_SUPPLY_W_ID
          , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
          , I_QTY
          , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
          , OL_DIST_INFO
          , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_QTY

                    FROM  Table( VALUES

                            (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                           , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
```

```
                    ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                ) AS ITEMLIST

                , TABLE( NEW_OL_LOCAL(   I_ID
                                       , I_QTY
                                       , W_ID
                                       , O_ID
                                       , D_ID
                                       )
                    ) AS NEW_OL_LOCAL

                WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )

  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (   OL_O_ID
              , OL_D_ID
              , OL_W_ID
              , OL_NUMBER
              , OL_I_ID
              , OL_SUPPLY_W_ID
              , OL_DELIVERY_D
              , OL_QUANTITY
              , OL_AMOUNT
              , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                      , I_NAME     CHAR(24)
                      , I_DATA     VARCHAR(50)
                      , S_DATA     VARCHAR(50)
                      , S_QUANTITY SMALLINT )

                SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_3 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , W_ID AS I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM ( SELECT   :next_o_id as O_ID
                                  , :w_id AS W_ID
                                  , :d_id as D_ID
                                  , OL_NUMBER
                                  , I_ID
                                  , I_QTY

                        FROM  Table( VALUES

                                  (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                  , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                  , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                    ) AS ITEMLIST

                    , TABLE( NEW_OL_LOCAL(   I_ID
                                           , I_QTY
                                           , W_ID
                                           , O_ID
                                           , D_ID
                                           )
                    ) AS NEW_OL_LOCAL

                WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )

  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (   OL_O_ID
              , OL_D_ID
              , OL_W_ID
              , OL_NUMBER
              , OL_I_ID
              , OL_SUPPLY_W_ID
              , OL_DELIVERY_D
              , OL_QUANTITY
              , OL_AMOUNT
              , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                      , I_NAME     CHAR(24)
                      , I_DATA     VARCHAR(50)
                      , S_DATA     VARCHAR(50)
                      , S_QUANTITY SMALLINT )

                SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_4 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , W_ID AS I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM ( SELECT   :next_o_id as O_ID
                                  , :w_id AS W_ID
                                  , :d_id as D_ID
                                  , OL_NUMBER
                                  , I_ID
                                  , I_QTY

                        FROM  Table( VALUES

                                  (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                  , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                  , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                  , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )

                                  ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                    ) AS ITEMLIST

                    , TABLE( NEW_OL_LOCAL(   I_ID
                                           , I_QTY
                                           , W_ID
                                           , O_ID
                                           , D_ID
                                           )
                    ) AS NEW_OL_LOCAL

                WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )

  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (   OL_O_ID
              , OL_D_ID
              , OL_W_ID
              , OL_NUMBER
              , OL_I_ID
              , OL_SUPPLY_W_ID
              , OL_DELIVERY_D
              , OL_QUANTITY
              , OL_AMOUNT
              , OL_DIST_INFO
            )
```

```
                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                           , D_ID
                           , W_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , OL_DELIVERY_D
                           , I_QTY
                           , TOTAL_PRICE
                           , OL_DIST_INFO
                           , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

                ) AS INS
;

EXEC SQL DECLARE ISOL_Local_5 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM ( SELECT   :next_o_id as O_ID
                                 , :w_id AS W_ID
                                 , :d_id as D_ID
                                 , OL_NUMBER
                                 , I_ID
                                 , I_QTY

                          FROM  Table( VALUES

                                  (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                 , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                 , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                 , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                                 , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )

                                 ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                          , I_QTY
                                          , W_ID
                                          , O_ID
                                          , D_ID
                                          )
                        ) AS NEW_OL_LOCAL

                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                           , D_ID
                           , W_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , OL_DELIVERY_D
                           , I_QTY
                           , TOTAL_PRICE
                           , OL_DIST_INFO
                           , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

                ) AS INS
;

EXEC SQL DECLARE ISOL_Local_6 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM ( SELECT   :next_o_id as O_ID
                                 , :w_id AS W_ID
                                 , :d_id as D_ID
                                 , OL_NUMBER
                                 , I_ID
                                 , I_QTY

                          FROM  Table( VALUES

                                  (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                 , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                 , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                 , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                                 , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                                 , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )

                                 ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                          , I_QTY
                                          , W_ID
                                          , O_ID
                                          , D_ID
                                          )
                        ) AS NEW_OL_LOCAL

                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                           , D_ID
                           , W_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , OL_DELIVERY_D
                           , I_QTY
                           , TOTAL_PRICE
                           , OL_DIST_INFO
                           , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

                ) AS INS
;

EXEC SQL DECLARE ISOL_Local_7 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
```

```
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_QTY

                        FROM  Table( VALUES

                                 (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                                , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                                , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                                , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )

                                ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                     ) AS ITEMLIST

                    , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                       ) AS NEW_OL_LOCAL

                  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


   SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

    FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
              )

              INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                  SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

             ) AS INS
;

EXEC SQL DECLARE ISOL_Local_8 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_QTY

                        FROM  Table( VALUES

                                 (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                                , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                                , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                                , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                                , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )

                                ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                     ) AS ITEMLIST

                    , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                       ) AS NEW_OL_LOCAL

                  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


   SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

    FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
              )

              INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                  SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

             ) AS INS
;

EXEC SQL DECLARE ISOL_Local_9 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_QTY

                        FROM  Table( VALUES

                                 (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
```

```
                                , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                                , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                                , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                                , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                                , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )

                                ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                  )
                          ) AS NEW_OL_LOCAL

                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
              )

              INCLUDE (   I_PRICE    DECIMAL(5,2)
                      , I_NAME     CHAR(24)
                      , I_DATA     VARCHAR(50)
                      , S_DATA     VARCHAR(50)
                      , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                        , D_ID
                        , W_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , OL_DELIVERY_D
                        , I_QTY
                        , TOTAL_PRICE
                        , OL_DIST_INFO
                        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Local_10 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM ( SELECT   :next_o_id as O_ID
                            , :w_id AS W_ID
                            , :d_id as D_ID
                            , OL_NUMBER
                            , I_ID
                            , I_QTY

                        FROM  Table( VALUES

                                (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                                , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                                , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                                , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                                , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                                , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                                , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )

                                ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                  )
                          ) AS NEW_OL_LOCAL

                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
              )

              INCLUDE (   I_PRICE    DECIMAL(5,2)
                      , I_NAME     CHAR(24)
                      , I_DATA     VARCHAR(50)
                      , S_DATA     VARCHAR(50)
                      , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                        , D_ID
                        , W_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , OL_DELIVERY_D
                        , I_QTY
                        , TOTAL_PRICE
                        , OL_DIST_INFO
                        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Local_11 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM ( SELECT   :next_o_id as O_ID
                            , :w_id AS W_ID
                            , :d_id as D_ID
                            , OL_NUMBER
                            , I_ID
                            , I_QTY

                        FROM  Table( VALUES

                                (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                                , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                                , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                                , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                                , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                                , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                                , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                                , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )

                                ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                  )
                          ) AS NEW_OL_LOCAL
```

```
                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                           , D_ID
                           , W_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , OL_DELIVERY_D
                           , I_QTY
                           , TOTAL_PRICE
                           , OL_DIST_INFO
                           , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Local_12 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM ( SELECT   :next_o_id as O_ID
                              , :w_id AS W_ID
                              , :d_id as D_ID
                              , OL_NUMBER
                              , I_ID
                              , I_QTY

                          FROM  Table( VALUES

                                  (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                                , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                                , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                                , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                                , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                                , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                                , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                                , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )
                                , (  SMALLINT( 12 ) , :id11 , :ol_quantity11 )

                                ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                  )
                          ) AS NEW_OL_LOCAL

                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                      SELECT   O_ID
                           , D_ID
                           , W_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , OL_DELIVERY_D
                           , I_QTY
                           , TOTAL_PRICE
                           , OL_DIST_INFO
                           , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Local_13 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM ( SELECT   :next_o_id as O_ID
                              , :w_id AS W_ID
                              , :d_id as D_ID
                              , OL_NUMBER
                              , I_ID
                              , I_QTY

                          FROM  Table( VALUES

                                  (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                                , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                                , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                                , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                                , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                                , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                                , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                                , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                                , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                                , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                                , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )
                                , (  SMALLINT( 12 ) , :id11 , :ol_quantity11 )
                                , (  SMALLINT( 13 ) , :id12 , :ol_quantity12 )

                                ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                  )
                          ) AS NEW_OL_LOCAL

                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


    SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

     FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
```

```
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Local_14 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                             , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                             , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                             , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                             , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                             , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )
                             , (  SMALLINT( 12 ) , :id11 , :ol_quantity11 )
                             , (  SMALLINT( 13 ) , :id12 , :ol_quantity12 )
                             , (  SMALLINT( 14 ) , :id13 , :ol_quantity13 )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                       ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                       ) AS NEW_OL_LOCAL

                  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

               (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

              ) AS INS
;

EXEC SQL DECLARE ISOL_Local_15 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                             , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                             , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                             , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                             , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                             , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )
                             , (  SMALLINT( 12 ) , :id11 , :ol_quantity11 )
                             , (  SMALLINT( 13 ) , :id12 , :ol_quantity12 )
                             , (  SMALLINT( 14 ) , :id13 , :ol_quantity13 )
                             , (  SMALLINT( 15 ) , :id14 , :ol_quantity14 )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                       ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                       ) AS NEW_OL_LOCAL

                  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

               (   OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
```

```
                    , I_DATA     VARCHAR(50)
                    , S_DATA     VARCHAR(50)
                    , S_QUANTITY SMALLINT )

                  SELECT   O_ID
                        , D_ID
                        , W_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , OL_DELIVERY_D
                        , I_QTY
                        , TOTAL_PRICE
                        , OL_DIST_INFO
                        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

              ) AS INS
  ;

  // Start processing

  in_neword = (struct in_neword_struct *)  pin ;
  neword    = (struct out_neword_struct *) pout ;

#ifdef DEBUGIT
  new_debug( neword, in_neword, "SP upon entry");
#endif

  // Using I_PRICE == 0 as a flag to the client that the ITEM was not fetched (hence bad).

  for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < in_neword->s_O_OL_CNT ;
inputItemArrayIndex++ )
  {
    i_priceArray[ inputItemArrayIndex ] = 0 ;
  }

  neword->deadlocks = -1 ;

 retry_tran:

  neword->deadlocks++ ;

  EXEC SQL

    SELECT D_TAX, D_NEXT_O_ID  INTO :dist_tax , :next_o_id

      FROM OLD TABLE ( UPDATE DISTRICT

                        SET  D_NEXT_O_ID = D_NEXT_O_ID + 1

                      WHERE D_W_ID = :w_id
                        AND D_ID = :d_id
              ) AS OT
  ;

  if ( sqlca.sqlcode != 0 )
  {
    DLCHK( retry_tran );
    sqlerror( NEWORD_SQL, "DISTRICT", __FILE__, __LINE__, &sqlca ) ;
    goto ferror;
  }

  #define NEW_CURSOR_OPEN_ERROR                                          \
  {                                                                      \
    if( sqlca.sqlcode != 0    )                                          \
    {                                                                    \
      goto sql_error ;                                                   \
    }                                                                    \
  }

  #define NEW_CURSOR_ERROR                                               \
  {                                                                      \
    if( sqlca.sqlcode == 0    )                                          \
    {                                                                    \
      neword->s_O_OL_CNT ++ ;                                            \
    }                                                                    \
    else                                                                 \
    if( sqlca.sqlcode == +100 )                                          \
    {                                                                    \
      break ;                                                            \
    }                                                                    \
    else                                                                 \
      goto sql_error ;                                                   \
  }

  if ( allLocal )
  {
    switch( inputItemCount )
    {
      case 1:
        EXEC SQL OPEN ISOL_Local_1 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_1
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 2:
        EXEC SQL OPEN ISOL_Local_2 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_2
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 3:
        EXEC SQL OPEN ISOL_Local_3 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_3
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 4:
        EXEC SQL OPEN ISOL_Local_4 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_4
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 5:
        EXEC SQL OPEN ISOL_Local_5 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_5
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 6:
        EXEC SQL OPEN ISOL_Local_6 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_6
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 7:
        EXEC SQL OPEN ISOL_Local_7 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_7
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 8:
        EXEC SQL OPEN ISOL_Local_8 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_8
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 9:
        EXEC SQL OPEN ISOL_Local_9 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_9
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 10:
        EXEC SQL OPEN ISOL_Local_10 ;
        NEW_CURSOR_OPEN_ERROR
```

```
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Local_10
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 11:
          EXEC SQL OPEN ISOL_Local_11 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Local_11
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 12:
          EXEC SQL OPEN ISOL_Local_12 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Local_12
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 13:
          EXEC SQL OPEN ISOL_Local_13 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Local_13
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 14:
          EXEC SQL OPEN ISOL_Local_14 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Local_14
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 15:
          EXEC SQL OPEN ISOL_Local_15 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Local_15
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;

        default:
          sqlerror(NEWORD_SQL, "Default switch on local  orderline/stock/index",
__FILE__, __LINE__, &sqlca );
          goto ferror;
      }
    }
    else
    {
      switch( inputItemCount )
      {
        case 1:
          EXEC SQL OPEN ISOL_Remote_1 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_1
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 2:
          EXEC SQL OPEN ISOL_Remote_2 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_2
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 3:
          EXEC SQL OPEN ISOL_Remote_3 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_3
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 4:
          EXEC SQL OPEN ISOL_Remote_4 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_4
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 5:
          EXEC SQL OPEN ISOL_Remote_5 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_5
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 6:
          EXEC SQL OPEN ISOL_Remote_6 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_6
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 7:
          EXEC SQL OPEN ISOL_Remote_7 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_7
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 8:
          EXEC SQL OPEN ISOL_Remote_8 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_8
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 9:
          EXEC SQL OPEN ISOL_Remote_9 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_9
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 10:
          EXEC SQL OPEN ISOL_Remote_10 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_10
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 11:
          EXEC SQL OPEN ISOL_Remote_11 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_11
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
```

```
          break ;
        case 12:
          EXEC SQL OPEN ISOL_Remote_12 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_12
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 13:
          EXEC SQL OPEN ISOL_Remote_13 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_13
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 14:
          EXEC SQL OPEN ISOL_Remote_14 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_14
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;
        case 15:
          EXEC SQL OPEN ISOL_Remote_15 ;
          NEW_CURSOR_OPEN_ERROR
          for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
          {
            EXEC SQL FETCH ISOL_Remote_15
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
            NEW_CURSOR_ERROR
          }
          break ;

        default:
          sqlerror(NEWORD_SQL, "Default switch on remote orderline/stock/index",
__FILE__, __LINE__, &sqlca);
          goto ferror;
      }
    }

    for ( inputItemArrayIndex = 0 ;
          inputItemArrayIndex < in_neword->s_O_OL_CNT  // from input
        && i_priceArray[ inputItemArrayIndex ] != 0 ;
        inputItemArrayIndex++ )
    {
      // s_I_NAME, and s_S_QUANTITY already set as output host variables

      neword->item[ inputItemArrayIndex ].s_I_PRICE = i_priceArray[ inputItemArrayIndex ] ;

      if (    is_ORIGINAL( s_dataArray[ inputItemArrayIndex ].data,
s_dataArray[ inputItemArrayIndex ].len )
          && is_ORIGINAL( i_dataArray[ inputItemArrayIndex ].data,
i_dataArray[ inputItemArrayIndex ].len )    )
      {
        neword->item[ inputItemArrayIndex ].s_brand_generic = 'B';
      }
      else
      {
        neword->item[ inputItemArrayIndex ].s_brand_generic = 'G';
      }
    }

    EXEC SQL

      SELECT W_TAX, C_DISCOUNT, C_LAST, C_CREDIT, O_ENTRY_D

        INTO :ware_tax, :c_discount, :c_last, :c_credit, :o_entry_d

        FROM TABLE ( NEW_WH (   :next_o_id
                              , :w_id
                              , :d_id
                              , :c_id
                              , :inputItemCount
                              , :allLocal
                              )
                   ) AS NEW_WH_TABLE
    ;

    if ( sqlca.sqlcode == 0 )
    {
      if ( neword->s_O_OL_CNT == in_neword->s_O_OL_CNT  )
      {
        neword->s_transtatus = TRAN_OK ;

        EXEC SQL COMMIT;

        if( sqlca.sqlcode != 0 )
        {
          sqlerror(NEWORD_SQL, "COMMIT", __FILE__, __LINE__ , &sqlca ) ;
          goto ferror;
        }
      }
      else
      {
        neword->s_transtatus = INVALID_ITEM ;

        EXEC SQL ROLLBACK WORK ;

        if ( sqlca.sqlcode != 0 )
        {
          neword->s_transtatus = FATAL_SQLERROR;

          sqlerror(NEWORD_SQL, "ROLLBACK FAILED (INVALID ITEM)", __FILE__,
__LINE__ , &sqlca);
          // no point in ferror
        }
      }
    }
    else
    {
      DLCHK( retry_tran );

      sqlerror( NEWORD_SQL, "NEW_WH", __FILE__, __LINE__ , &sqlca);
      goto ferror;
    }

    /*-----------------------------------------------*/
    /*  Return to client                             */
    /*-----------------------------------------------*/

mexit:

    if ( sqlca.sqlcode >= 0 )
    {
      storedProcRc = SQLZ_HOLD_PROC ;
    }
    else
    {
      storedProcRc = SQLZ_DISCONNECT_PROC ;
    }

#ifdef DEBUGIT
    new_debug( neword, in_neword, "SP prior to return");
#endif

    return ( storedProcRc ) ;

sql_error:

    {
      char tempstr[ 4096 ] ;

      DLCHK( retry_tran ) ;

      sprintf( tempstr, "inputItemCount=%d, :next_o_id=%d, :d_id=%d, :w_id=%d",
inputItemCount, next_o_id, d_id, w_id ) ;
      sqlerror( NEWORD_SQL, tempstr , __FILE__, __LINE__, &sqlca ) ;
    }

ferror:

    neword->s_transtatus = FATAL_SQLERROR;

    EXEC SQL ROLLBACK WORK;

    if ( sqlca.sqlcode != 0 )
    {
      sqlerror( NEWORD_SQL, "ROLLBACK FAILED", __FILE__, __LINE__ , &sqlca ) ;
    }


    goto mexit ;
}

/*
** A little function to search for the string "ORIGINAL" given a string and
** it's length
*/
static unsigned char  skip[256] = {8,8,8,8,8,8,8,8,8,8,  /*0-9*/
                                   8,8,8,8,8,8,8,8,8,8,  /*10-19*/
                                   8,8,8,8,8,8,8,8,8,8,  /*20-29*/
                                   8,8,8,8,8,8,8,8,8,8,  /*30-39*/
                                   8,8,8,8,8,8,8,8,8,8,  /*40-49*/
                                   8,8,8,8,8,8,8,8,8,8,  /*50-59*/
                                   8,8,8,8,8,1,8,8,8,8,  /*60-69*/
                                   8,4,8,3,8,8,0,8,2,7,  /*70-79*/
                                   8,8,6,8,8,8,8,8,8,8,  /*80-89*/
                                   8,8,8,8,8,8,8,8,8,8,  /*90-99*/
                                   8,8,8,8,8,8,8,8,8,8,  /*100-109*/
                                   8,8,8,8,8,8,8,8,8,8,  /*110-119*/
                                   8,8,8,8,8,8,8,8,8,8,  /*120-129*/
```

```
                        8,8,8,8,8,8,8,8,8,8,  /*130-139*/
                        8,8,8,8,8,8,8,8,8,8,  /*140-149*/
                        8,8,8,8,8,8,8,8,8,8,  /*150-159*/
                        8,8,8,8,8,8,8,8,8,8,  /*160-169*/
                        8,8,8,8,8,8,8,8,8,8,  /*170-179*/
                        8,8,8,8,8,8,8,8,8,8,  /*180-189*/
                        8,8,8,8,8,8,8,8,8,8,  /*190-199*/
                        8,8,8,8,8,8,8,8,8,8,  /*200-209*/
                        8,8,8,8,8,8,8,8,8,8,  /*210-219*/
                        8,8,8,8,8,8,8,8,8,8,  /*220-229*/
                        8,8,8,8,8,8,8,8,8,8,  /*230-239*/
                        8,8,8,8,8,8,8,8,8,8,  /*240-249*/
                        8,8,8,8};          /*250-254*/

static int is_ORIGINAL( char *string, short length )
{
  char           *cur_string;
  char           *end_string;
  unsigned char   *skips;
  int            skip_dist;
  int            result = 0;

  cur_string = string+7;
  end_string = string + length;
  skips = skip;

  while (cur_string < end_string)
    {
      skip_dist = skips[*cur_string];
      while ( (skip_dist > 0) && (cur_string < end_string) )
        {
          skip_dist = skips[*(cur_string += skip_dist)];
        }

      if (cur_string >= end_string)
        goto exit;

      if ( cur_string[-4] != 'G' )
        goto noMatch;

      if ( memcmp( cur_string-7, "ORIGINAL", 8 ) == 0 )
        {
          result = 1;
          goto exit;
        }
noMatch:
      cur_string += 8;
    } /* end while */

exit:
  return ( result ) ;
}

// ---------------------------------------------------------------
// Order Status SERVER
// ---------------------------------------------------------------

#undef w_id
#undef d_id
#undef c_id_input
#undef o_id
#undef o_entry_d
#undef o_carrier_d
#undef c_id
#undef c_first
#undef c_middle
#undef c_last
#undef c_balance

SQL_API_RC order_status_internal( char *pin, char *pout )
{
   struct in_ordstat_struct  * in_ordstat = (struct in_ordstat_struct *) pin ;
   struct out_ordstat_struct * ordstat = (struct out_ordstat_struct *) pout ;

   struct sqlca sqlca ;

   EXEC SQL BEGIN DECLARE SECTION;

     // From input values

     //##sqlint32   w_id ;
     //##short      d_id;
     sqlint32   c_id_input ;

     struct s_data_type { short len ; char data[ 16 ] ; } c_last_input ;

     // From queries

     // From initial query

     sqlint32   o_id ;
     //##sqlint32   c_id ;
     short      o_carrier_id ;
     //##sqlint64   o_entry_d ;

     char   c_first[ 16 ] ;
     char   c_middle[ 2 ] ;
     //##char   c_last[ 16 ] ;
     double c_balance ;

     // From cursor

     sqlint32   ol_i_id ;
     sqlint32   ol_supply_w_id ;
     short      ol_quantity ;
     float   ol_amount ;
     char    ol_delivery_d [27] ;
     //##char    o_entry_d[ 27 ] ;

   EXEC SQL END DECLARE SECTION;

   //##struct s_data_type { short len ; char data[ 16 ] ; } c_last_input ;

   int     storedProcRc ;
   int     itemArrayIndex = 0 ;

   #define w_id              in_ordstat->s_W_ID ;
   #define d_id              in_ordstat->s_D_ID ;
   #define c_id_input        in_ordstat->s_C_ID
   #define o_id              ordstat->s_O_ID
   #define o_entry_d         ordstat->s_O_ENTRY_D_time
   #define o_carrier_id      ordstat->s_O_CARRIER_ID
   #define c_id              ordstat->s_C_ID
   #define c_first           ordstat->s_C_FIRST
   #define c_middle          ordstat->s_C_MIDDLE
   #define c_last            ordstat->s_C_LAST
   #define c_balance         ordstat->s_C_BALANCE

   EXEC SQL DECLARE read_orderline_cur CURSOR FOR

     SELECT OL_I_ID, OL_SUPPLY_W_ID, OL_QUANTITY, OL_AMOUNT,
OL_DELIVERY_D

      FROM ORDER_LINE

      WHERE OL_W_ID = :w_id
        AND OL_D_ID = :d_id
        AND OL_O_ID = :o_id

      FOR FETCH ONLY ;

   ordstat->deadlocks = -1 ;

#ifdef DEBUGIT
   ord_debug(ordstat, in_ordstat, "SP upon entry");
#endif

retry_tran:

   ordstat->deadlocks ++ ;

   if ( c_id_input == 0 )
   {
     c_last_input.len = strlen( in_ordstat->s_C_LAST ) ;
     memcpy( c_last_input.data , in_ordstat->s_C_LAST , c_last_input.len ) ;

     EXEC SQL

       SELECT O_ID, O_CARRIER_ID, O_ENTRY_D, C_BALANCE, C_FIRST,
C_MIDDLE, C_ID

         INTO :o_id, :o_carrier_id , :o_entry_d , :c_balance, :c_first, :c_middle, :c_id

         FROM TABLE ( ORD_C_LAST(   :w_id
                          , :d_id
                          , :c_last_input
                         )
                  ) AS ORD_C_LAST
     ;
   }
   else
   {
     EXEC SQL

       SELECT O_ID, O_CARRIER_ID, O_ENTRY_D , C_BALANCE, C_FIRST,
C_MIDDLE ,C_LAST

         INTO :o_id, :o_carrier_id , :o_entry_d , :c_balance, :c_first, :c_middle, :c_last

         FROM TABLE ( ORD_C_ID(   :w_id
                          , :d_id
                          , :c_id_input
                         )
                  ) AS ORD_C_ID
     ;
   }

   if ( sqlca.sqlcode != 0 )
   {
     DLCHK( retry_tran );
     sqlerror( ORDSTAT_SQL, "READ CUST and ORDERS", __FILE__, __LINE__ ,
&sqlca ) ;
     goto ferror;
   }
```

```
  /*-----------------------------------------------*/
  /*  Read ORDER_LINEs                             */
  /*-----------------------------------------------*/

  EXEC SQL OPEN read_orderline_cur ;

  if ( sqlca.sqlcode != 0 )
  {
    DLCHK( retry_tran );
    sqlerror(ORDSTAT_SQL, "OPEN CURSOR read_orderline_cur", __FILE__,
__LINE__ , &sqlca ) ;
    goto ferror;
  }

  itemArrayIndex = 0 ;
  {
    do
    {
      EXEC SQL FETCH read_orderline_cur

              INTO :ol_i_id , :ol_supply_w_id , :ol_quantity , :ol_amount , :ol_delivery_d ;

      if ( sqlca.sqlcode == 0 )
      {
        ordstat->item[ itemArrayIndex ].s_OL_I_ID            = ol_i_id ;
        ordstat->item[ itemArrayIndex ].s_OL_SUPPLY_W_ID     = ol_supply_w_id ;
        ordstat->item[ itemArrayIndex ].s_OL_QUANTITY        = ol_quantity ;
        ordstat->item[ itemArrayIndex ].s_OL_AMOUNT          = ol_amount ;
        strcpy(ordstat->item[ itemArrayIndex ].s_OL_DELIVERY_D_time, ol_delivery_d) ;

        itemArrayIndex++;
      }
      else
      if (sqlca.sqlcode < 0 )
      {
        DLCHK( retry_tran ) ;
        sqlerror( ORDSTAT_SQL, "FETCH CURSOR read_orderline_cur" , __FILE__,
__LINE__ , &sqlca ) ;
        goto ferror ;
      }
    }
    while ( sqlca.sqlcode == 0 ) ;
  }

  ordstat->s_ol_cnt = itemArrayIndex ;

  EXEC SQL COMMIT ;

  if ( sqlca.sqlcode == 0 )
  {
    ordstat->s_transtatus = TRAN_OK ;
  }
  else
  {
    DLCHK( retry_tran );
    sqlerror(ORDSTAT_SQL, "COMMIT", __FILE__, __LINE__ , &sqlca);
    goto ferror ;
  }

mexit:

  if ( sqlca.sqlcode >= 0 )
  {
    storedProcRc = SQLZ_HOLD_PROC ;
  }
  else
  {
    storedProcRc = SQLZ_DISCONNECT_PROC ;
  }

#ifdef DEBUGIT
  ord_debug(ordstat, in_ordstat, "SP prior to return");
#endif

  return ( storedProcRc ) ;

ferror:

  ordstat->s_transtatus = FATAL_SQLERROR ;

  EXEC SQL ROLLBACK WORK ;

  if ( sqlca.sqlcode != 0 )
  {
    sqlerror(ORDSTAT_SQL, "ROLLBACK FAILED", __FILE__, __LINE__, &sqlca);
  }

  goto mexit;
}

// ---------------------------------------------------------------
// Delivery SERVER
// ---------------------------------------------------------------

#undef d_id
#undef c_id
#undef w_id
#undef o_carrier_id
#undef ol_delivery_d

SQL_API_RC delivery_internal ( char * pin, char * pout )
{
  struct in_delivery_struct * in_delivery = (struct in_delivery_struct *)  pin ;
  struct out_delivery_struct * delivery   = (struct out_delivery_struct *) pout ;

  struct sqlca sqlca ;

  int storedProcRc ;

  short district_id ;
  sqlint32  customer_id ;

  EXEC SQL BEGIN DECLARE SECTION;

    // input

    //##sqlint32   w_id ;
    //##short      d_id ;
    //##sqlint32   c_id ;
    //##short      o_carrier_id ;
    //##sqlint64   ol_delivery_d ;

    // output

    short      no_o_id_indicator = 0 ;
    sqlint32   no_o_id ;

  EXEC SQL END DECLARE SECTION;

  #define d_id          district_id
  #define c_id          customer_id

  #define w_id          in_delivery->s_W_ID
  #define o_carrier_id  in_delivery->s_O_CARRIER_ID
  #define ol_delivery_d in_delivery->s_O_DELIVERY_D_time

  delivery->deadlocks = -1 ;

#ifdef DEBUGIT
  del_debug( delivery, in_delivery, "SP upon entry");
#endif

  // Deadlock Handling
  // -----------------
  // Since we COMMIT inside the for() loop, we must take special
  // care while handling deadlocks.  This is best explained by
  // an example.
  //
  // Assume we deadlock on d_id=6.  This means that an order from the
  // first 5 districts have already been delivered.  We will then
  // restart the loop (retry_tran).  However, the loop will restart
  // at d_id = 1!  This means that the second (and all subsequent)
  // time through the loop, we will deliver orders for districts that
  // have already been delivered, with the net result being more than
  // 10 orders being delivered.
  //
  // The solution to this problem is to initialize the starting point
  // of the loop *before* the retry_tran label.  This will ensure that
  // if we deadlock, we will restart the loop with the same district
  // that we deadlocked on, and we won't deliver any extra orders.
  //
  // NOTE: If we ever change this back to one COMMIT per transaction
  // (instead of one COMMIT per iteration), then the initialization
  // of d_id must be moved back into the for loop.  (A rollback due
  // to deadlock in this case would rollback all delivered orders so
  // far, so we'd need to re-deliver them all on the next iteration.)

  d_id = 1;

retry_tran:

  delivery->deadlocks++;

  for ( ;  d_id <= DISTRICTS_PER_WAREHOUSE ; d_id++ )
  {
    no_o_id = 0 ;
    no_o_id_indicator = 0 ;

      EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

        SELECT O_ID

                INTO :no_o_id :no_o_id_indicator

         FROM TABLE (  DEL( :w_id , :d_id , :o_carrier_id ) ) AS T ;

        COMMIT ;

      END COMPOUND ;

    if ( sqlca.sqlcode == 0 )
    {
      delivery->s_O_ID[ d_id - 1 ] = no_o_id ;
```

```
    }
    else
    {
      DLCHK( retry_tran );

      sqlerror( DELIVERY_SQL , "DELIVERY", __FILE__, __LINE__ , &sqlca);
      goto ferror ;
    }
  }

  delivery->s_transtatus = TRAN_OK ;

mexit:

  if ( sqlca.sqlcode >= 0 )
  {
    storedProcRc = SQLZ_HOLD_PROC ;
  }
  else
  {
    storedProcRc = SQLZ_DISCONNECT_PROC ;
  }

#ifdef DEBUGIT
  del_debug( delivery, in_delivery, "SP prior to return");
#endif

  return ( storedProcRc ) ;

ferror:

  delivery->s_transtatus = FATAL_SQLERROR ;

  EXEC SQL ROLLBACK WORK ;

  if ( sqlca.sqlcode != 0 )
  {
   sqlerror( DELIVERY_SQL, "ROLLBACK FAILED", __FILE__, __LINE__ , &sqlca ) ;
  }


  goto mexit ;
}

// ---------------------------------------------------------------
// Stored Procedure Stubs
// ---------------------------------------------------------------

SQL_API_RC SQL_API_FN news( char *pin, char *pout )
{
  return new_order_internal( pin, pout ) ;
}

SQL_API_RC SQL_API_FN ords( char *pin, char *pout )
{
  return order_status_internal( pin, pout ) ;
}

SQL_API_RC SQL_API_FN dels ( char * pin, char * pout )
{
  return delivery_internal( pin, pout ) ;
}
```

Src.Srv/uncat-func.ddl

```
-----------------------------------------------------------------------
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----------------------------------------------------------------------
--
-- uncat-func.ddl - Drop table function DDL
--
--

-- STOCK LEVEL
DROP SPECIFIC FUNCTION STOCK_LEVEL %
-- DELIVERY
DROP SPECIFIC FUNCTION DELIVERY %
-- ORDER STATUS
DROP SPECIFIC FUNCTION ORD_C_LAST %
DROP SPECIFIC FUNCTION ORD_C_ID %
-- PAYMENT
DROP SPECIFIC FUNCTION PAY_C_LAST %
DROP SPECIFIC FUNCTION PAY_C_ID %
-- NEW ORDER
DROP SPECIFIC FUNCTION NEW_OL_ALL %
DROP SPECIFIC FUNCTION NEW_OL_LOCAL %
DROP SPECIFIC FUNCTION NEW_WH %
```

Src.Srv/uncat-proc.ddl

```
DROP PROCEDURE news
                (varchar(262),varchar(682));
DROP PROCEDURE news
                (varchar(270),varchar(662));
DROP PROCEDURE news;



DROP PROCEDURE pays;

DROP PROCEDURE ords
                (varchar(42),varchar(822));
DROP PROCEDURE ords
                (varchar(42),varchar(446));
DROP PROCEDURE ords;

DROP PROCEDURE dels
                (varchar(14),varchar(50));
DROP PROCEDURE dels
                (varchar(22),varchar(50));
DROP PROCEDURE dels;



DROP PROCEDURE stks;
```

include/db2tpcc.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * db2tpcc.h - Macros and Miscellany
 *
 */

#ifndef __DB2TPCC_H
#define __DB2TPCC_H

#include <sys/types.h>

#include "lval.h"

/* ******************************************************************** */
/* Transaction Return Codes (s_transtatus)                              */
/* ******************************************************************** */

#define INVALID_ITEM          100
#define TRAN_OK               0
#define FATAL_SQLERROR        -1

/* ******************************************************************** */
/* Definition of Unused and Bad Items                                   */
/* ******************************************************************** */
/* Define unused item ID to be 0. This allows the SUT to determine the     */
/* number of items in the order as required by 2.4.1.3 and 2.4.2.2 since   */
/* the assumption that any item with OL_I_ID = 0 is unused will be true.   */
/* This in turn requires that the value used for an invalid item is        */
/* equal to ITEMS + 1.                                                      */
/* ******************************************************************** */

#define INVALID_ITEM_ID (2 * ITEMS) + 1
#define UNUSED_ITEM_ID 0

#define MIN_WAREHOUSE 1
#define MAX_WAREHOUSE WAREHOUSES

/*****************************************************************************/
/* NURand Constants                                                         */
/* C_C_LAST_RUN and C_C_LAST_LOAD must adhere to clause 2.1.6.              */
/*****************************************************************************/
#define C_C_LAST_RUN  88
#define C_C_LAST_LOAD 173
#define C_C_ID                319
#define C_OL_I_ID             3849
#define A_C_LAST              255
#define A_C_ID                1023
#define A_OL_I_ID             8191

/*****************************************************************************/
```

```
/* Transaction Type Identifiers                                    */
/*******************************************************************/

#define CLIENT_SQL   0
#define NEWORD_SQL   1
#define PAYMENT_SQL  2
#define ORDSTAT_SQL  3
#define DELIVERY_SQL 4
#define STOCKLEV_SQL 5

#define SPGENERAL_PAD 3
#define SPGENERAL_ADJUST sizeof(int16_t)

struct in_neword_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  struct   in_items_struct {
    int32_t  s_OL_I_ID;
    int32_t  s_OL_SUPPLY_W_ID;
    int16_t  s_OL_QUANTITY;
    int16_t  pad1[3];
  } in_item[15];
  int32_t  s_C_ID;
  int32_t  s_W_ID;
  int16_t  s_D_ID;
  int16_t  s_O_OL_CNT;                    /* init by SUT */
  int16_t  s_all_local;
  int16_t  duplicate_items;
};

struct out_neword_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  struct   items_struct {
    float  s_I_PRICE;
    float  s_OL_AMOUNT;
    int16_t  s_S_QUANTITY;
    int16_t  pad2;
    char     s_I_NAME[25];
    char     s_brand_generic;
  } item[15];
  float  s_W_TAX;
  float  s_D_TAX;
  float  s_C_DISCOUNT;
  float  s_total_amount;
  int32_t  s_O_ID;
  int16_t  s_O_OL_CNT;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_LAST[17];
  char     s_C_CREDIT[3];
  char     s_O_ENTRY_D_time[27];

};

struct in_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  float  s_H_AMOUNT;
  int32_t  s_W_ID;
  int32_t  s_C_W_ID;
  int32_t  s_C_ID;
  int16_t  s_C_D_ID;
  int16_t  s_D_ID;
  char     s_C_LAST[17];
};

struct out_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_CREDIT_LIM;
  double  s_C_BALANCE;
  float  s_C_DISCOUNT;
  int32_t  s_C_ID;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_W_STREET_1[21];
  char     s_W_STREET_2[21];
  char     s_W_CITY[21];
  char     s_W_STATE[3];
  char     s_W_ZIP[10];
  char     s_D_STREET_1[21];
  char     s_D_STREET_2[21];
  char     s_D_CITY[21];
  char     s_D_STATE[3];
  char     s_D_ZIP[10];
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_C_STREET_1[21];
  char     s_C_STREET_2[21];
  char     s_C_CITY[21];
  char     s_C_STATE[3];
  char     s_C_ZIP[10];
  char     s_C_PHONE[17];
  char     s_C_CREDIT[3];
  char     s_C_DATA[201];
  char     s_H_DATE_time[27];
  char     s_C_SINCE_time[27];
};

struct in_ordstat_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_C_ID;
  int32_t  s_W_ID;
  int16_t  s_D_ID;
  int16_t  pad1[3];
  char     s_C_LAST[17];
};

struct out_ordstat_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_BALANCE;
  int32_t  s_C_ID;
  int32_t  s_O_ID;
  int16_t  s_O_CARRIER_ID;
  int16_t  s_ol_cnt;
  int16_t  pad1[2];
  struct   oitems_struct {
    double  s_OL_AMOUNT;
    int32_t  s_OL_I_ID;
    int32_t  s_OL_SUPPLY_W_ID;
    int16_t  s_OL_QUANTITY;
    int16_t  pad2;
    char     s_OL_DELIVERY_D_time[27];
  } item[15];
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_O_ENTRY_D_time[27];
  int16_t  pad3[2];
};

struct in_delivery_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_W_ID;
  int16_t  s_O_CARRIER_ID;
};

struct out_delivery_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_O_ID[10];
  int16_t  s_transtatus;
  int16_t  deadlocks;
};

struct in_stocklev_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_threshold;
  int32_t  s_W_ID;
  int16_t  s_D_ID;
};

struct out_stocklev_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_low_stock;
  int16_t  s_transtatus;
  int16_t  deadlocks;
};

/* ******************************************************************* */
/* Transaction Prototypes                                              */
/* ******************************************************************* */

#ifdef __cplusplus
extern "C" {
#endif

extern int neword_sql(struct in_neword_struct*, struct out_neword_struct*);
extern int payment_sql(struct in_payment_struct*, struct out_payment_struct*);
extern int ordstat_sql(struct in_ordstat_struct*, struct out_ordstat_struct*);
extern int delivery_sql(struct in_delivery_struct*, struct out_delivery_struct*);
extern int stocklev_sql(struct in_stocklev_struct*, struct out_stocklev_struct*);

#ifdef __cplusplus
}
#endif

/* ******************************************************************* */
/* DB2 Connect/Disconnect & Thread Context Wrappers                    */
/* ******************************************************************* */

#ifdef __cplusplus
extern "C" {
#endif
```

```
extern int connect_to_TM(char*);
extern int connect_to_TM_auth(char*, char*, char*);
extern int disconnect_from_TM(void);


#ifdef __cplusplus
}
#endif

#endif // __DB2TPCC_H
```

include/lval.h

```
/* lval.h - generated automatically at 20101007.1242 */



#ifndef __LVAL_H

#define __LVAL_H

#define WAREHOUSES 184032

#define DISTRICTS_PER_WAREHOUSE 10

#define CUSTOMERS_PER_DISTRICT 3000

#define ITEMS 100000

#define STOCK_PER_WAREHOUSE 100000

#define MIN_OL_PER_ORDER 5

#define MAX_OL_PER_ORDER 15

#define NU_ORDERS_PER_DISTRICT 900

#endif // __LVAL_H
```

include/tpccapp.h

```
/*******************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*******************************************************************************/

/*
 * tpccapp.h - Application Macros
 *
 */

#ifndef __TPCCAPP_H
#define __TPCCAPP_H

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <time.h>

#define daricall

#include "sqlca.h"
#include "sqlcodes.h"

#ifdef SWAP_ENDIAN
#define SWAP_BYTE(Var)  SwapEndian((void*)&Var, sizeof(Var))

/*******************************************************************************
  FUNCTION: SwapEndian
  PURPOSE: Swap the byte order of a structure
  EXAMPLE: int I=0x12345678; SWAP_BYTE(I); I => 0x78563412;
  IMPLEMENTATION: Fold Addr in half, swap header & tail by XOR op
          e.g.: *a = 0x12 [ Addr + 0];
                *b = 0x78 [ Add + 4 - 0 - 1 = Addr+3];
                *a ^= *b;        // sets *a to 0x6A
                *b ^= *a;        // sets *b to 0x12
                *a ^= *b;        // sets *a to 0x78

           Now *a => 0x78 && *b => 0x12
*******************************************************************************/

void SwapEndian(void *Addr, int nb)
{
  int i;
  for (i=0; i<nb/2; i++)
  {
    char *a = (char*)Addr+i;
    char *b = (char*)Addr+(nb-i-1);

    *a ^= *b;
    *b ^= *a;
    *a ^= *b;
  }
}
#endif   //SWAP_ENDIAN


/*******************************************************************************/
/* SQLCODE Macros                                                              */
/*******************************************************************************/

#define DLCHK(a) \
  if (sqlca.sqlcode == SQL_RC_E911) { goto a; }

#define NACOMPCHK(last) \
  if (sqlca.sqlcode != SQL_RC_E1339) { last = -1; } \
  else { int a = ((sqlca.sqlerrmc[4] == 0x20) ? 0 : sqlca.sqlerrmc[4]-0x30); \
        int b = ((sqlca.sqlerrmc[5] == 0x20) ? 0 : sqlca.sqlerrmc[5]-0x30); \
        if (b == 0) { last = a; } else { last = a * 10 + b; } \
      }

#endif // __TPCCAPP_H
```

include/tpccdbg.h

```
/*******************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*******************************************************************************/

/*
 * tpccdbg.h - Debugging Macros
 *
 */

#ifndef __TPCCDBG_H
#define __TPCCDBG_H

#ifdef __cplusplus
extern "C" {
#endif

extern void sqlerror (int tranType, char *msg, char *file, int line,
                      SQL_STRUCTURE sqlca *psqlca);

extern void new_debug (struct out_neword_struct *neword_ptr,
                       struct in_neword_struct *in_neword_ptr,
                       char *msg);
extern void pay_debug (struct out_payment_struct *payment_ptr,
                       struct in_payment_struct *in_payment_ptr,
                       char *msg);
extern void ord_debug (struct out_ordstat_struct *ordstat_ptr,
                       struct in_ordstat_struct *in_ordstat_ptr,
                       char *msg);
extern void del_debug (struct out_delivery_struct *delivery_ptr,
                       struct in_delivery_struct *in_delivery_ptr,
                       char *msg);
extern void stk_debug (struct out_stocklev_struct *stocklev_ptr,
                       struct in_stocklev_struct *in_stocklev_ptr,
                       char *msg);

extern void new_print (struct out_neword_struct *neword_ptr,
                       struct in_neword_struct *in_neword_ptr,
                       char *filename,
                       char *msg);
extern void pay_print (struct out_payment_struct *payment_ptr,
                       struct in_payment_struct *in_payment_ptr,
                       char *filename,
                       char *msg);
extern void ord_print (struct out_ordstat_struct *ordstat_ptr,
                       struct in_ordstat_struct *in_ordstat_ptr,
                       char *filename,
                       char *msg);
extern void del_print (struct out_delivery_struct *delivery_ptr,
                       struct in_delivery_struct *in_delivery_ptr,
                       char *filename,
                       char *msg);
extern void stk_print (struct out_stocklev_struct *stocklev_ptr,
                       struct in_stocklev_struct *in_stocklev_ptr,
                       char *filename,
                       char *msg);

#ifdef __cplusplus
```

```
}
#endif

#endif // __TPCCDBG_H
```

tpccenv.sh

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# tpccenv.sh - UNIX Environment Setup
#

# The Kit Version
export TPCC_VERSION=CK080718

# The DB2 Instance Name (for DB2)
export DB2INSTANCE=${USER}

# The OS being used (i.e. "UNIX", "LINUX", "WINDOWS")
export PLATFORM=LINUX
export SERVER_PLATFORM=UNIX

# The type of make command and slash used by the OS.
# (i.e. UNIX - "/", WINDOWS - "\").
# These are referenced all over the kit.
export SLASH="/";
export MAKE=make

# Specifies whether or not to use dari stored proc's for the TPC-C driver.  Set to either
DARIVERSION or NONDARI;
#export TPCC_SPTYPE=NOSP
#export TPCC_SPTYPE=SPGENERAL2
export TPCC_SPTYPE=SPGENERAL
#export TPCC_SPTYPE=DARI2SQLDA

# The schema name is typically the SQL autorization ID (or username).
# This is required for runstats and EEE.
export TPCC_SCHEMA=${USER}
export SERVER_TPCC_SCHEMA=${USER}

# DB2 EE/EEE Configuration
export DB2EDITION=EE
#export DB2EDITION=EEE
export DB2NODE=0
export DB2NODES=1;                      # set to the number of nodes you have.  Set to 1
for EE.

# TPCC General Configuration
export TPCC_DBNAME=TPCC
export TPCC_ROOT=${HOME}/tpc-c.ibm
export TPCC_SQLLIB=${HOME}/sqllib
export TPCC_RUNDATA=${HOME}/tpccdata

# TPCC Debug Configuration
# This is the path where all error and debug logs are placed.
# To get debugging from within the stored procedures, you must
# set DB2ENVLIST="TPCC_DEBUGDIR" in tpcc.config.
export TPCC_DEBUGDIR=/tmp

# Specifies where stored procedures should be placed and if they should
# be fenced.
export TPCC_SPDIR=${TPCC_SQLLIB}/function
export TPCC_FENCED=NO

# changes by KMR
# MTE says aaaahhh!  Nooooo!
export EDITOR=emacs
```

10 Appendix B: Tunable Paramaters

10.1. Database Parameters

db2set

DB2_LARGE_PAGE_MEM=DB

DB2_RESOURCE_POLICY=/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/cfg/db2-resource-policy-gid-16clients_ff.cfg

DB2_SELUDI_COMM_BUFFER=Y

DB2_USE_ALTERNATE_PAGE_CLEANING=YES

DB2_MAX_NON_TABLE_LOCKS=500

DB2_RCT_FEATURES=GROUPUPDATE=ON

DB2_TRUSTED_BINDIN=ON

DB2_KEEPTABLELOCK=CONNECTION

DB2_NO_FORK_CHECK=ON

DB2_ALLOCATION_SIZE=8388608

DB2_LOGGER_NON_BUFFERED_IO=ON

DB2_FMP_COMM_HEAPSZ=1200

DB2_APM_PERFORMANCE=ON

DB2_ENABLE_BUFPD=OFF

DB2_PINNED_BP=ON

DB2_SELECTIVITY=ON

DB2ASSUMEUPDATE=ON

DB2CHECKCLIENTINTERVAL=0

DB2_HASH_JOIN=OFF

DB2CHKSQLDA=OFF

DB2_UOW_LOGSPACE_CACHE=14336

DB2_COLLECT_TS_REC_INFO=OFF

DB2COMM=tcpip

db2 cfg

```
Database Configuration for Database tpcc

Database configuration release level            = 0x0d00
Database release level                          = 0x0d00

Database territory                              = US
Database code page                              = 819
Database code set                               = ISO8859-1
Database country/region code                    = 1
Database collating sequence                     = IDENTITY
Alternate collating sequence            (ALT_COLLATE) =
Number compatibility                            = OFF
Varchar2 compatibility                          = OFF
Date compatibility                              = OFF
Database page size                              = 4096

Dynamic SQL Query management            (DYN_QUERY_MGMT) = DISABLE

Statement concentrator                  (STMT_CONC) = OFF

Discovery support for this database     (DISCOVER_DB) = ENABLE

Restrict access                                 = NO
Default query optimization class        (DFT_QUERYOPT) = 5
Degree of parallelism                   (DFT_DEGREE) = 1
Continue upon arithmetic exceptions     (DFT_SQLMATHWARN) = NO
Default refresh age                     (DFT_REFRESH_AGE) = 0
Default maintained table types for opt (DFT_MTTB_TYPES) = SYSTEM
Number of frequent values retained      (NUM_FREQVALUES) = 10
Number of quantiles retained            (NUM_QUANTILES) = 20

Decimal floating point rounding mode  (DECFLT_ROUNDING) = ROUND_HALF_EVEN

Backup pending                                  = NO

All committed transactions have been written to disk    = YES
Rollforward pending                             = NO
Restore pending                                 = NO

Multi-page file allocation enabled              = YES

Log retain for recovery status                  = RECOVERY
User exit for logging status                    = NO

Self tuning memory                      (SELF_TUNING_MEM) = OFF
Size of database shared memory (4KB)  (DATABASE_MEMORY) = 386145152
Database memory threshold               (DB_MEM_THRESH) = 10
Max storage for lock list (4KB)         (LOCKLIST) = 64000
Percent. of lock lists per application  (MAXLOCKS) = 100
Package cache size (4KB)                (PCKCACHESZ) = AUTOMATIC(5000)
Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = 250
Sort list heap (4KB)                    (SORTHEAP) = 16

Database heap (4KB)                     (DBHEAP) = 524288
Catalog cache size (4KB)                (CATALOGCACHE_SZ) = (MAXAPPLS*5)
Log buffer size (4KB)                   (LOGBUFSZ) = 6000
Utilities heap size (4KB)               (UTIL_HEAP_SZ) = 5000
Buffer pool size (pages)                (BUFFPAGE) = 1000
SQL statement heap (4KB)                (STMTHEAP) = AUTOMATIC(150000)
Default application heap (4KB)          (APPLHEAPSZ) = 1000
Application Memory Size (4KB)           (APPL_MEMORY) = AUTOMATIC(40000)
```

```
Statistics heap size (4KB)              (STAT_HEAP_SZ) = AUTOMATIC(4384)

Interval for checking deadlock (ms)         (DLCHKTIME) = 3000
Lock timeout (sec)                      (LOCKTIMEOUT) = -1

Changed pages threshold                 (CHNGPGS_THRESH) = 99
Number of asynchronous page cleaners    (NUM_IOCLEANERS) = 1
Number of I/O servers                   (NUM_IOSERVERS) = 1
Index sort flag                         (INDEXSORT) = YES
Sequential detect flag                  (SEQDETECT) = NO
Default prefetch size (pages)           (DFT_PREFETCH_SZ) = AUTOMATIC

Track modified pages                    (TRACKMOD) = OFF

Default number of containers                       = 1
Default tablespace extentsize (pages)   (DFT_EXTENT_SZ) = 32

Max number of active applications       (MAXAPPLS) = 2000
Average number of active applications   (AVG_APPLS) = AUTOMATIC(1)
Max DB files open per application       (MAXFILOP) = 2200

Log file size (4KB)                     (LOGFILSIZ) = 512000
Number of primary log files             (LOGPRIMARY) = 256
Number of secondary log files           (LOGSECOND) = 0
Changed path to log files               (NEWLOGPATH) =
Path to log files                                  = /dev/tpcc/log
Overflow log path                       (OVERFLOWLOGPATH) =
Mirror log path                         (MIRRORLOGPATH) =
First active log file                              = S0000004.LOG
Block log on disk full                  (BLK_LOG_DSK_FUL) = NO
Block non logged operations             (BLOCKNONLOGGED) = NO
Percent max primary log space by transaction  (MAX_LOG) = 0
Num. of active log files for 1 active UOW(NUM_LOG_SPAN) = 0

Group commit count                      (MINCOMMIT) = 1
Percent log file reclaimed before soft chckpt (SOFTMAX) = 8080
Log retain for recovery enabled         (LOGRETAIN) = RECOVERY
User exit for logging enabled           (USEREXIT) = OFF

HADR database role                                 = STANDARD
HADR local host name                    (HADR_LOCAL_HOST) =
HADR local service name                 (HADR_LOCAL_SVC) =
HADR remote host name                   (HADR_REMOTE_HOST) =
HADR remote service name                (HADR_REMOTE_SVC) =
HADR instance name of remote server     (HADR_REMOTE_INST) =
HADR timeout value                      (HADR_TIMEOUT) = 120
HADR log write synchronization mode     (HADR_SYNCMODE) = NEARSYNC
HADR peer window duration (seconds)     (HADR_PEER_WINDOW) = 0

First log archive method                (LOGARCHMETH1) = LOGRETAIN
Options for logarchmeth1                (LOGARCHOPT1) =
Second log archive method               (LOGARCHMETH2) = OFF
Options for logarchmeth2                (LOGARCHOPT2) =
Failover log archive path               (FAILARCHPATH) =
Number of log archive retries on error  (NUMARCHRETRY) = 5
Log archive retry Delay (secs)          (ARCHRETRYDELAY) = 20
Vendor options                          (VENDOROPT) =

Auto restart enabled                    (AUTORESTART) = ON
Index re-creation time and redo index build  (INDEXREC) = SYSTEM (RESTART)
Log pages during index build            (LOGINDEXBUILD) = OFF
Default number of loadrec sessions      (DFT_LOADREC_SES) = 1
Number of database backups to retain    (NUM_DB_BACKUPS) = 12
Recovery history retention (days)       (REC_HIS_RETENTN) = 366
Auto deletion of recovery objects       (AUTO_DEL_REC_OBJ) = OFF

TSM management class                    (TSM_MGMTCLASS) =
TSM node name                           (TSM_NODENAME) =
TSM owner                               (TSM_OWNER) =
TSM password                            (TSM_PASSWORD) =

Automatic maintenance                   (AUTO_MAINT) = OFF
  Automatic database backup             (AUTO_DB_BACKUP) = OFF
  Automatic table maintenance           (AUTO_TBL_MAINT) = OFF
    Automatic runstats                  (AUTO_RUNSTATS) = OFF
      Automatic statement statistics    (AUTO_STMT_STATS) = OFF
    Automatic statistics profiling      (AUTO_STATS_PROF) = OFF
      Automatic profile updates         (AUTO_PROF_UPD) = OFF
    Automatic reorganization            (AUTO_REORG) = OFF

Auto-Revalidation                       (AUTO_REVAL) = DEFERRED
Currently Committed                     (CUR_COMMIT) = DISABLED
CHAR output with DECIMAL input          (DEC_TO_CHAR_FMT) = NEW
Enable XML Character operations         (ENABLE_XMLCHAR) = YES
WLM Collection Interval (minutes)       (WLM_COLLECT_INT) = 0
Monitor Collect Settings
Request metrics                         (MON_REQ_METRICS) = NONE
Activity metrics                        (MON_ACT_METRICS) = NONE
Object metrics                          (MON_OBJ_METRICS) = NONE
Unit of work events                     (MON_UOW_DATA) = NONE
Lock timeout events                     (MON_LOCKTIMEOUT) = NONE
Deadlock events                         (MON_DEADLOCK) = WITHOUT_HIST
Lock wait events                        (MON_LOCKWAIT) = NONE
Lock wait event threshold               (MON_LW_THRESH) = 5000000
Number of package list entries          (MON_PKGLIST_SZ) = 32
Lock event notification level           (MON_LCK_MSG_LVL) = 1
```

SMTP Server (SMTP_SERVER) =

SQL conditional compilation flags (SQL_CCFLAGS) =

Section actuals setting (SECTION_ACTUALS) = NONE

db2.dbm.out

```
Database Manager Configuration

   Node type = Enterprise Server Edition with local and remote clients

Database manager configuration release level            = 0x0d00

CPU speed (millisec/instruction)             (CPUSPEED) = 3.581944e-07
Communications bandwidth (MB/sec)      (COMM_BANDWIDTH) = 1.000000e+02

Max number of concurrently active databases     (NUMDB) = 1
Federated Database System Support          (FEDERATED) = NO
Transaction processor monitor name        (TP_MON_NAME) =

Default charge-back account           (DFT_ACCOUNT_STR) =

Java Development Kit installation path       (JDK_PATH) = /autobench/sources/db2_tpcc-
1/users/tpcc/sqllib/java/jdk64

Diagnostic error capture level             (DIAGLEVEL) = 3
Notify Level                          (NOTIFYLEVEL) = 3
Diagnostic data directory path             (DIAGPATH) =
Size of rotating db2diag & notify logs (MB)  (DIAGSIZE) = 0

Default database monitor switches
  Buffer pool                    (DFT_MON_BUFPOOL) = OFF
  Lock                           (DFT_MON_LOCK) = OFF
  Sort                           (DFT_MON_SORT) = OFF
  Statement                      (DFT_MON_STMT) = OFF
  Table                          (DFT_MON_TABLE) = OFF
  Timestamp                      (DFT_MON_TIMESTAMP) = OFF
  Unit of work                   (DFT_MON_UOW) = OFF
Monitor health of instance and databases   (HEALTH_MON) = OFF

SYSADM group name                    (SYSADM_GROUP) =
SYSCTRL group name                  (SYSCTRL_GROUP) =
SYSMAINT group name                 (SYSMAINT_GROUP) =
SYSMON group name                    (SYSMON_GROUP) =

Client Userid-Password Plugin         (CLNT_PW_PLUGIN) =
Client Kerberos Plugin              (CLNT_KRB_PLUGIN) =
Group Plugin                          (GROUP_PLUGIN) =
GSS Plugin for Local Authorization    (LOCAL_GSSPLUGIN) =
Server Plugin Mode                  (SRV_PLUGIN_MODE) = UNFENCED
Server List of GSS Plugins      (SRVCON_GSSPLUGIN_LIST) =
Server Userid-Password Plugin        (SRVCON_PW_PLUGIN) =
Server Connection Authentication         (SRVCON_AUTH) = NOT_SPECIFIED
Cluster manager                       (CLUSTER_MGR) =

Database manager authentication       (AUTHENTICATION) = CLIENT
Alternate authentication          (ALTERNATE_AUTH_ENC) = NOT_SPECIFIED
Cataloging allowed without authority   (CATALOG_NOAUTH) = NO
Trust all clients                   (TRUST_ALLCLNTS) = YES
Trusted client authentication         (TRUST_CLNTAUTH) = CLIENT
Bypass federated authentication          (FED_NOAUTH) = NO

Default database path                 (DFTDBPATH) = /autobench/sources/db2_tpcc-1/users/tpcc

Database monitor heap size (4KB)          (MON_HEAP_SZ) = 4096
Java Virtual Machine heap size (4KB)     (JAVA_HEAP_SZ) = 2048
Audit buffer size (4KB)                (AUDIT_BUF_SZ) = 0
Size of instance shared memory (4KB)  (INSTANCE_MEMORY) = 388145152
Backup buffer default size (4KB)           (BACKBUFSZ) = 1024
Restore buffer default size (4KB)          (RESTBUFSZ) = 1024

Agent stack size                     (AGENT_STACK_SZ) = 1024
Sort heap threshold (4KB)                (SHEAPTHRES) = 0

Directory cache support                   (DIR_CACHE) = YES

Application support layer heap size (4KB)   (ASLHEAPSZ) = 15
Max requester I/O block size (bytes)         (RQRIOBLK) = 4096
Query heap size (4KB)                (QUERY_HEAP_SZ) = 1000

Workload impact by throttled utilities(UTIL_IMPACT_LIM) = 10

Priority of agents                       (AGENTPRI) = 59
Agent pool size                     (NUM_POOLAGENTS) = 0
Initial number of agents in pool       (NUM_INITAGENTS) = 0
Max number of coordinating agents     (MAX_COORDAGENTS) = AUTOMATIC(200)
Max number of client connections      (MAX_CONNECTIONS) =
AUTOMATIC(MAX_COORDAGENTS)

Keep fenced process                     (KEEPFENCED) = YES
Number of pooled fenced processes        (FENCED_POOL) = MAX_COORDAGENTS
Initial number of fenced processes     (NUM_INITFENCED) = 0

Index re-creation time and redo index build  (INDEXREC) = RESTART

Transaction manager database name         (TM_DATABASE) = 1ST_CONN
```

```
Transaction resync interval (sec)     (RESYNC_INTERVAL) = 180

SPM name                              (SPM_NAME) =
SPM log size                          (SPM_LOG_FILE_SZ) = 256
SPM resync agent limit                (SPM_MAX_RESYNC) = 20
SPM log path                          (SPM_LOG_PATH) =

TCP/IP Service name                   (SVCENAME) =
Discovery mode                        (DISCOVER) = SEARCH
Discover server instance              (DISCOVER_INST) = ENABLE

SSL server keydb file                 (SSL_SVR_KEYDB) =
SSL server stash file                 (SSL_SVR_STASH) =
SSL server certificate label          (SSL_SVR_LABEL) =
SSL service name                      (SSL_SVCENAME) =
SSL cipher specs                      (SSL_CIPHERSPECS) =
SSL versions                          (SSL_VERSIONS) =
SSL client keydb file                 (SSL_CLNT_KEYDB) =
SSL client stash file                 (SSL_CLNT_STASH) =

Maximum query degree of parallelism   (MAX_QUERYDEGREE) = ANY
Enable intra-partition parallelism    (INTRA_PARALLEL) = NO

Maximum Asynchronous TQs per query    (FEDERATED_ASYNC) = 0

No. of int. communication buffers(4KB)(FCM_NUM_BUFFERS) = AUTOMATIC(4096)
No. of int. communication channels   (FCM_NUM_CHANNELS) = AUTOMATIC(2048)
Node connection elapse time (sec)     (CONN_ELAPSE) = 10
Max number of node connection retries (MAX_CONNRETRIES) = 5
Max time difference between nodes (min) (MAX_TIME_DIFF) = 60

db2start/db2stop timeout (min)        (START_STOP_TIME) = 10
```

db2-resource-policy.cfg

```
<!-- This policy is valid -->
<RESOURCE_POLICY>
  <DATABASE_RESOURCE_POLICY>
    <DBNAME>TPCC</DBNAME>
    <METHOD>CPUMASK</METHOD>
    <RESOURCE_BINDING>
      <RESOURCE>0x000000FF000000FF</RESOURCE>
      <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
      <SERVICE_NAME>DB2_tpcc</SERVICE_NAME>
      <BUFFERPOOL_BINDING>
        <NUM_CLEANERS>4</NUM_CLEANERS>
        <BUFFERPOOL_ID>4</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>20</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>36</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>37</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>53</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>69</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>85</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>101</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>117</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>133</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>149</BUFFERPOOL_ID>
      </BUFFERPOOL_BINDING>
    </RESOURCE_BINDING>
    <RESOURCE_BINDING>
      <RESOURCE>0x000000FF000000FF</RESOURCE>
      <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
      <SERVICE_NAME>DB2_tpcc_5</SERVICE_NAME>
      <BUFFERPOOL_BINDING>
        <NUM_CLEANERS>4</NUM_CLEANERS>
        <BUFFERPOOL_ID>5</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>21</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>38</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>54</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>70</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>86</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>102</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>118</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>134</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>150</BUFFERPOOL_ID>
      </BUFFERPOOL_BINDING>
    </RESOURCE_BINDING>
    <RESOURCE_BINDING>
      <RESOURCE>0x000000FF000000FF</RESOURCE>
      <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
      <SERVICE_NAME>DB2_tpcc_9</SERVICE_NAME>
      <BUFFERPOOL_BINDING>
        <NUM_CLEANERS>4</NUM_CLEANERS>
        <BUFFERPOOL_ID>6</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>22</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>39</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>55</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>71</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>87</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>103</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>119</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>135</BUFFERPOOL_ID>
        <BUFFERPOOL_ID>151</BUFFERPOOL_ID>
      </BUFFERPOOL_BINDING>
    </RESOURCE_BINDING>
```

```
<RESOURCE_BINDING>
  <RESOURCE>0x000000FF000000FF</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_10</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>7</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>23</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>40</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>56</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>72</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>88</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>104</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>120</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>136</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>152</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0x0000FF000000FF00</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_3</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>8</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>24</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>41</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>57</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>73</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>89</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>105</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>121</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>137</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>153</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0x0000FF000000FF00</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_7</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>9</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>25</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>42</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>58</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>74</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>90</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>106</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>122</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>138</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>154</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0x0000FF000000FF00</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_11</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>10</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>26</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>43</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>59</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>75</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>91</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>107</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>123</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>139</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>155</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0x0000FF000000FF00</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_12</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>11</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>27</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>44</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>60</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>76</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>92</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>108</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>124</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>140</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>156</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0x00FF000000FF0000</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_2</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
```

```
    <BUFFERPOOL_ID>12</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>28</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>45</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>61</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>77</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>93</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>109</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>125</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>141</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>157</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0x00FF000000FF0000</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_6</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>13</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>29</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>46</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>62</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>78</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>94</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>110</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>126</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>142</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>158</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0x00FF000000FF0000</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_13</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>14</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>30</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>47</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>63</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>79</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>95</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>111</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>127</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>143</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>159</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0x00FF000000FF0000</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_14</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>15</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>31</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>48</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>64</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>80</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>96</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>112</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>128</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>144</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>160</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0xFF000000FF000000</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_4</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>16</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>32</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>49</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>65</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>81</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>97</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>113</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>129</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>145</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>161</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
  <RESOURCE>0xFF000000FF000000</RESOURCE>
  <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
  <SERVICE_NAME>DB2_tpcc_8</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
    <NUM_CLEANERS>4</NUM_CLEANERS>
    <BUFFERPOOL_ID>17</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>33</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>50</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>66</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>82</BUFFERPOOL_ID>
    <BUFFERPOOL_ID>98</BUFFERPOOL_ID>
```

```
      <BUFFERPOOL_ID>114</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>130</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>146</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>162</BUFFERPOOL_ID>
    </BUFFERPOOL_BINDING>
  </RESOURCE_BINDING>
  <RESOURCE_BINDING>
    <RESOURCE>0xFF000000FF000000</RESOURCE>
    <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
    <SERVICE_NAME>DB2_tpcc_15</SERVICE_NAME>
    <BUFFERPOOL_BINDING>
      <NUM_CLEANERS>4</NUM_CLEANERS>
      <BUFFERPOOL_ID>18</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>34</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>51</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>67</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>83</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>99</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>115</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>131</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>147</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>163</BUFFERPOOL_ID>
    </BUFFERPOOL_BINDING>
  </RESOURCE_BINDING>
  <RESOURCE_BINDING>
    <RESOURCE>0xFF000000FF000000</RESOURCE>
    <DBMEM_PERCENTAGE>6.125</DBMEM_PERCENTAGE>
    <SERVICE_NAME>DB2_tpcc_16</SERVICE_NAME>
    <BUFFERPOOL_BINDING>
      <NUM_CLEANERS>4</NUM_CLEANERS>
      <BUFFERPOOL_ID>19</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>35</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>52</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>68</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>84</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>100</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>116</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>132</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>148</BUFFERPOOL_ID>
      <BUFFERPOOL_ID>164</BUFFERPOOL_ID>
    </BUFFERPOOL_BINDING>
  </RESOURCE_BINDING>
 </DATABASE_RESOURCE_POLICY>
</RESOURCE_POLICY>
```


10.2. Transaction Monitor Parameters

inetInfo_registry.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\InetInfo\Parameters]
"MaxConcurrency"=dword:00000640
"MaxPoolThreads"=dword:00000190
"PoolThreadLimit"=dword:00000640
```

tcpip_parameters_registery.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters]
"ICSDomain"="mshome.net"
"SyncDomainWithMembership"=dword:00000001
"NV Hostname"="tpcc-client1"
"DataBasePath"=hex(2):25,00,53,00,79,00,73,00,74,00,65,00,6d,00,52,00,6f,00,6f,\
  00,74,00,25,00,5c,00,53,00,79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,\
  64,00,72,00,69,00,76,00,65,00,72,00,73,00,5c,00,65,00,74,00,63,00,00,00
"NameServer"=""
"ForwardBroadcasts"=dword:00000000
"IPEnableRouter"=dword:00000000
"Domain"=""
"Hostname"="tpcc-client1"
"SearchList"="austin.ibm.com,linuxperf9025.net"
"UseDomainNameDevolution"=dword:00000001
"EnableICMPRedirect"=dword:00000001
"DeadGWDetectDefault"=dword:00000001
"DontAddDefaultGatewayDefault"=dword:00000000
"MaxUserPort"=dword:0000fffe

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{02E71305-9C31-498F-9EC3-5AEAE7EFEEB9}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,30,00,32,00,45,00,37,00,31,00,33,00,\
  30,00,35,00,2d,00,39,00,43,00,33,00,31,00,2d,00,34,00,39,00,38,00,46,00,2d,\
  00,39,00,45,00,43,00,33,00,2d,00,35,00,41,00,45,00,41,00,45,00,37,00,45,00,\
  46,00,45,00,45,00,42,00,39,00,7d,00,00,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{170923B8-287F-4AA5-9B62-ACC3DF721737}]
"LLInterface"=""
```

```
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,31,00,37,00,30,00,39,00,32,00,33,00,\
  42,00,38,00,2d,00,32,00,38,00,37,00,46,00,2d,00,34,00,41,00,41,00,35,00,2d,\
  00,39,00,42,00,36,00,32,00,2d,00,41,00,43,00,43,00,33,00,44,00,46,00,37,00,\
  32,00,31,00,37,00,33,00,37,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{1F5668B7-A26C-483F-B719-EF9338C9C4DC}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,31,00,46,00,35,00,36,00,36,00,38,00,\
  42,00,37,00,2d,00,41,00,32,00,36,00,43,00,2d,00,34,00,38,00,33,00,46,00,2d,\
  00,42,00,37,00,31,00,39,00,2d,00,45,00,46,00,39,00,33,00,33,00,38,00,43,00,\
  39,00,43,00,34,00,44,00,43,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{511B57FE-B4D6-4121-9D12-C4E2BA398E58}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,35,00,31,00,31,00,42,00,35,00,37,00,\
  46,00,45,00,2d,00,42,00,34,00,44,00,36,00,2d,00,34,00,31,00,32,00,31,00,2d,\
  00,39,00,44,00,31,00,32,00,2d,00,43,00,34,00,45,00,32,00,42,00,41,00,33,00,\
  39,00,38,00,45,00,35,00,38,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{62C84429-9356-407B-9572-B798C6C8AC9A}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,36,00,32,00,43,00,38,00,34,00,34,00,\
  32,00,39,00,2d,00,39,00,33,00,35,00,36,00,2d,00,34,00,30,00,37,00,42,00,2d,\
  00,39,00,35,00,37,00,32,00,2d,00,42,00,37,00,39,00,38,00,43,00,36,00,43,00,\
  38,00,41,00,43,00,39,00,41,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{724248B6-966D-41C0-8E01-F7C08D0E8648}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,37,00,32,00,34,00,32,00,34,00,38,00,\
  42,00,36,00,2d,00,39,00,36,00,36,00,44,00,2d,00,34,00,31,00,43,00,30,00,2d,\
  00,38,00,45,00,30,00,31,00,2d,00,46,00,37,00,43,00,30,00,38,00,44,00,30,00,\
  45,00,38,00,36,00,34,00,38,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{72E75E72-4FF7-4DE0-A81E-B4721EA039EE}]
"LLInterface"=""
```

```
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,37,00,32,00,45,00,37,00,35,00,45,00,\
  37,00,32,00,2d,00,34,00,46,00,46,00,37,00,2d,00,34,00,44,00,45,00,30,00,2d,\
  00,41,00,38,00,31,00,45,00,2d,00,42,00,34,00,37,00,32,00,31,00,45,00,41,00,\
  30,00,33,00,39,00,45,00,45,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{7C1C96DA-E1EC-4771-A1BC-B107F1AD3B8E}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,37,00,43,00,31,00,43,00,39,00,36,00,\
  44,00,41,00,2d,00,45,00,31,00,45,00,43,00,2d,00,34,00,37,00,37,00,31,00,2d,\
  00,41,00,31,00,42,00,43,00,2d,00,42,00,31,00,30,00,37,00,46,00,31,00,41,00,\
  44,00,33,00,42,00,38,00,45,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{C7B5C48B-7281-4F2B-B106-6DAE0493EE4D}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,43,00,37,00,42,00,35,00,43,00,34,00,\
  38,00,42,00,2d,00,37,00,32,00,38,00,31,00,2d,00,34,00,46,00,32,00,42,00,2d,\
  00,42,00,31,00,30,00,36,00,2d,00,36,00,44,00,41,00,45,00,30,00,34,00,39,00,\
  33,00,45,00,45,00,34,00,44,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{D0264C87-0470-4A06-A0D7-63EF81968E35}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,44,00,30,00,32,00,36,00,34,00,43,00,\
  38,00,37,00,2d,00,30,00,34,00,37,00,30,00,2d,00,34,00,41,00,30,00,36,00,2d,\
  00,41,00,30,00,44,00,37,00,2d,00,36,00,33,00,45,00,46,00,38,00,31,00,39,00,\
  36,00,38,00,45,00,33,00,35,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{D7C8DBF2-6FC6-4873-9467-8A0637B1D07A}]
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,44,00,37,00,43,00,38,00,44,00,42,00,\
  46,00,32,00,2d,00,36,00,46,00,43,00,36,00,2d,00,34,00,38,00,37,00,33,00,2d,\
  00,39,00,34,00,36,00,37,00,2d,00,38,00,41,00,30,00,36,00,33,00,37,00,42,00,\
  31,00,44,00,30,00,37,00,41,00,7d,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapt
ers\{DF3B01C1-9D76-427F-BA7A-17AEAF182E30}]
"LLInterface"=""
```

```
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,44,00,46,00,33,00,42,00,30,00,31,00,\
  43,00,31,00,2d,00,39,00,44,00,37,00,36,00,2d,00,34,00,32,00,37,00,46,00,2d,\
  00,42,00,41,00,37,00,41,00,2d,00,31,00,37,00,41,00,45,00,41,00,46,00,31,00,\
  38,00,32,00,45,00,33,00,30,00,7d,00,00,00,00,00
```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Adapters\{F7E95255-579D-4EF7-9A8B-20B4C40B7C48}]

```
"LLInterface"=""
"IpConfig"=hex(7):54,00,63,00,70,00,69,00,70,00,5c,00,50,00,61,00,72,00,61,00,\
  6d,00,65,00,74,00,65,00,72,00,73,00,5c,00,49,00,6e,00,74,00,65,00,72,00,66,\
  00,61,00,63,00,65,00,73,00,5c,00,7b,00,46,00,37,00,45,00,39,00,35,00,32,00,\
  35,00,35,00,2d,00,35,00,37,00,39,00,44,00,2d,00,34,00,45,00,46,00,37,00,2d,\
  00,39,00,41,00,38,00,42,00,2d,00,32,00,30,00,42,00,34,00,43,00,34,00,30,00,\
  42,00,37,00,43,00,34,00,38,00,7d,00,00,00,00,00
```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\DNSRegisteredAdapters]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{02E71305-9C31-498F-9EC3-5AEAE7EFEEB9}]

```
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000000
"NameServer"="10.0.1.254,9.0.7.1"
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000000
"LeaseObtainedTime"=dword:00000000
"T1"=dword:00000000
"T2"=dword:00000000
"LeaseTerminatesTime"=dword:00000000
"AddressType"=dword:00000000
"IsServerNapAware"=dword:00000000
"DhcpConnForceBroadcastFlag"=dword:00000000
"IPAddress"=hex(7):31,00,30,00,2e,00,32,00,2e,00,30,00,2e,00,32,00,31,00,00,00,\
  00,00
"SubnetMask"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,32,00,35,\
  00,35,00,2e,00,30,00,00,00,00,00
"DefaultGateway"=hex(7):00,00
"DefaultGatewayMetric"=hex(7):00,00
```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{065f0c42-703a-11de-9954-806e6f6e6963}]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{170923B8-287F-4AA5-9B62-ACC3DF721737}]

```
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000001
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"DhcpIPAddress"="0.0.0.0"
"DhcpSubnetMask"="255.0.0.0"
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000000
"LeaseObtainedTime"=dword:00000000
"T1"=dword:00000000
"T2"=dword:00000000
"LeaseTerminatesTime"=dword:00000000
"AddressType"=dword:00000000
"IsServerNapAware"=dword:00000000
"DhcpConnForceBroadcastFlag"=dword:00000000
```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{1F5668B7-A26C-483F-B719-EF9338C9C4DC}]

```
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000001
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"DhcpServer"="10.1.1.250"
"Lease"=dword:00015180
"LeaseObtainedTime"=dword:4c487c64
"T1"=dword:4c492524
"T2"=dword:4c49a3b4
"LeaseTerminatesTime"=dword:4c49cde4
"AddressType"=dword:00000000
"IsServerNapAware"=dword:00000000
"DhcpConnForceBroadcastFlag"=dword:00000001
"IPAddress"=hex(7):00,00
"SubnetMask"=hex(7):00,00
"DhcpIPAddress"="10.1.1.97"
"DhcpSubnetMask"="255.255.0.0"
"DhcpInterfaceOptions"=hex:06,00,00,00,00,00,00,00,08,00,00,00,00,00,00,00,e4,\
  cd,49,4c,09,00,07,01,09,00,06,0b,03,00,00,00,00,00,00,00,04,00,00,00,00,00,\
  00,00,e4,cd,49,4c,0a,01,01,fa,01,00,00,00,00,00,00,00,04,00,00,00,00,00,00,\
  00,e4,cd,49,4c,ff,ff,00,00,36,00,00,00,00,00,00,00,04,00,00,00,00,00,00,00,\
  e4,cd,49,4c,0a,01,01,fa,35,00,00,00,00,00,00,00,01,00,00,00,00,00,00,00,e4,\
```

```
  cd,49,4c,05,00,00,00,fc,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,67,7c,\
  48,4c,33,00,00,00,00,00,00,00,04,00,00,00,00,00,00,00,00,e4,cd,49,4c,00,01,51,\
  80
"DhcpGatewayHardware"=hex:0a,01,01,fa,06,00,00,00,00,15,17,bd,e4,fd
"DhcpGatewayHardwareCount"=dword:00000001
"DhcpNameServer"="9.0.7.1 9.0.6.11"
"DhcpDefaultGateway"=hex(7):31,00,30,00,2e,00,31,00,2e,00,31,00,2e,00,32,00,35,\
  00,30,00,00,00,00,00
"DhcpSubnetMaskOpt"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,30,\
  00,2e,00,30,00,00,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interf
aces\{511B57FE-B4D6-4121-9D12-C4E2BA398E58}]
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000001
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"DhcpIPAddress"="0.0.0.0"
"DhcpSubnetMask"="255.0.0.0"
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000000
"LeaseObtainedTime"=dword:00000000
"T1"=dword:00000000
"T2"=dword:00000000
"LeaseTerminatesTime"=dword:00000000
"AddressType"=dword:00000000
"IsServerNapAware"=dword:00000000
"DhcpConnForceBroadcastFlag"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interf
aces\{62C84429-9356-407B-9572-B798C6C8AC9A}]
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000000
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000000
"LeaseObtainedTime"=dword:00000000
"T1"=dword:00000000
"T2"=dword:00000000
"LeaseTerminatesTime"=dword:00000000
"AddressType"=dword:00000000
"IsServerNapAware"=dword:00000000
"DhcpConnForceBroadcastFlag"=dword:00000000
"IPAddress"=hex(7):31,00,30,00,2e,00,32,00,2e,00,31,00,2e,00,32,00,00,00,00,00
"SubnetMask"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,32,00,35,\
  00,35,00,2e,00,30,00,00,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interf
aces\{724248B6-966D-41C0-8E01-F7C08D0E8648}]
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000001
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"DhcpIPAddress"="0.0.0.0"
"DhcpSubnetMask"="255.0.0.0"
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000000
"LeaseObtainedTime"=dword:00000000
"T1"=dword:00000000
"T2"=dword:00000000
"LeaseTerminatesTime"=dword:00000000
"AddressType"=dword:00000000
"IsServerNapAware"=dword:00000000
"DhcpConnForceBroadcastFlag"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interf
aces\{72E75E72-4FF7-4DE0-A81E-B4721EA039EE}]
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000000
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000000
"LeaseObtainedTime"=dword:00000000
"T1"=dword:00000000
"T2"=dword:00000000
"LeaseTerminatesTime"=dword:00000000
"AddressType"=dword:00000000
"IsServerNapAware"=dword:00000000
"DhcpConnForceBroadcastFlag"=dword:00000000
"IPAddress"=hex(7):31,00,30,00,2e,00,33,00,2e,00,31,00,2e,00,32,00,00,00,00,00
"SubnetMask"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,32,00,35,\
  00,35,00,2e,00,30,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{7C1C96DA-E1EC-4771-A1BC-B107F1AD3B8E}]

"UseZeroBroadcast"=dword:00000000

"EnableDeadGWDetect"=dword:00000001

"EnableDHCP"=dword:00000001

"NameServer"=""

"Domain"=""

"RegistrationEnabled"=dword:00000001

"RegisterAdapterName"=dword:00000000

"DhcpIPAddress"="10.1.1.11"

"DhcpSubnetMask"="255.255.0.0"

"DhcpServer"="10.1.1.250"

"Lease"=dword:00015180

"LeaseObtainedTime"=dword:4bfab4e4

"T1"=dword:4bfb5da4

"T2"=dword:4bfbdc34

"LeaseTerminatesTime"=dword:4bfc0664

"AddressType"=dword:00000000

"IsServerNapAware"=dword:00000000

"DhcpConnForceBroadcastFlag"=dword:00000001

"DhcpInterfaceOptions"=hex:03,00,00,00,00,00,00,00,04,00,00,00,00,00,00,00,64,\

  06,fc,4b,0a,01,01,fa,01,00,00,00,00,00,00,00,04,00,00,00,00,00,00,00,64,06,\

  fc,4b,ff,ff,00,00,36,00,00,00,00,00,00,00,04,00,00,00,00,00,00,00,64,06,fc,\

  4b,0a,01,01,fa,35,00,00,00,00,00,00,00,01,00,00,00,00,00,00,00,64,06,fc,4b,\

  05,00,00,00,fc,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,e7,b4,fa,4b,33,\

  00,00,00,00,00,00,00,04,00,00,00,00,00,00,00,64,06,fc,4b,00,01,51,80

"DhcpGatewayHardware"=hex:0a,01,01,fa,06,00,00,00,00,15,17,bd,e4,fd

"DhcpGatewayHardwareCount"=dword:00000001

"DhcpDefaultGateway"=hex(7):31,00,30,00,2e,00,31,00,2e,00,31,00,2e,00,32,00,35,\

  00,30,00,00,00,00,00

"DhcpSubnetMaskOpt"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,30,\

  00,2e,00,30,00,00,00,00,00


[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{C7B5C48B-7281-4F2B-B106-6DAE0493EE4D}]

"UseZeroBroadcast"=dword:00000000

"EnableDeadGWDetect"=dword:00000001

"EnableDHCP"=dword:00000001

"NameServer"=""

"Domain"=""

"RegistrationEnabled"=dword:00000001

"RegisterAdapterName"=dword:00000000

"DhcpIPAddress"="0.0.0.0"

"DhcpSubnetMask"="255.0.0.0"

"DhcpServer"="255.255.255.255"

"Lease"=dword:00000000

"LeaseObtainedTime"=dword:00000000

"T1"=dword:00000000

"T2"=dword:00000000

"LeaseTerminatesTime"=dword:00000000

"AddressType"=dword:00000000

"IsServerNapAware"=dword:00000000

"DhcpConnForceBroadcastFlag"=dword:00000000


[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{D0264C87-0470-4A06-A0D7-63EF81968E35}]

"UseZeroBroadcast"=dword:00000000

"EnableDeadGWDetect"=dword:00000001

"EnableDHCP"=dword:00000001

"NameServer"=""

"Domain"=""

"RegistrationEnabled"=dword:00000001

"RegisterAdapterName"=dword:00000000

"DhcpServer"="255.255.255.255"

"Lease"=dword:00000000

"LeaseObtainedTime"=dword:00000000

"T1"=dword:00000000

"T2"=dword:00000000

"LeaseTerminatesTime"=dword:00000000

"AddressType"=dword:00000000

"IsServerNapAware"=dword:00000000

"DhcpConnForceBroadcastFlag"=dword:00000000

"IPAddress"=hex(7):00,00

"SubnetMask"=hex(7):00,00

"DhcpIPAddress"="0.0.0.0"

"DhcpSubnetMask"="255.0.0.0"


[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{D7C8DBF2-6FC6-4873-9467-8A0637B1D07A}]

"UseZeroBroadcast"=dword:00000000

"EnableDeadGWDetect"=dword:00000001

"EnableDHCP"=dword:00000001

"NameServer"=""

"Domain"=""

"RegistrationEnabled"=dword:00000001

"RegisterAdapterName"=dword:00000000

"DhcpIPAddress"="0.0.0.0"

"DhcpSubnetMask"="255.0.0.0"

"DhcpServer"="255.255.255.255"

"Lease"=dword:00000000

"LeaseObtainedTime"=dword:00000000

"T1"=dword:00000000

"T2"=dword:00000000

"LeaseTerminatesTime"=dword:00000000

"AddressType"=dword:00000000

"IsServerNapAware"=dword:00000000

"DhcpConnForceBroadcastFlag"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{DF3B01C1-9D76-427F-BA7A-17AEAF182E30}]

"UseZeroBroadcast"=dword:00000000

"EnableDeadGWDetect"=dword:00000001

"EnableDHCP"=dword:00000000

"NameServer"="9.0.7.1,9.0.6.11"

"Domain"=""

"RegistrationEnabled"=dword:00000001

"RegisterAdapterName"=dword:00000000

"DhcpServer"="255.255.255.255"

"Lease"=dword:00000000

"LeaseObtainedTime"=dword:00000000

"T1"=dword:00000000

"T2"=dword:00000000

"LeaseTerminatesTime"=dword:00000000

"AddressType"=dword:00000000

"IsServerNapAware"=dword:00000000

"DhcpConnForceBroadcastFlag"=dword:00000000

"IPAddress"=hex(7):39,00,2e,00,33,00,2e,00,31,00,34,00,34,00,2e,00,35,00,37,00,\

  00,00,00,00

"SubnetMask"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,32,00,35,\

  00,35,00,2e,00,30,00,00,00,00,00

"DefaultGateway"=hex(7):39,00,2e,00,33,00,2e,00,31,00,34,00,34,00,2e,00,31,00,\

  00,00,00,00

"DefaultGatewayMetric"=hex(7):30,00,00,00,00,00


[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\{F7E95255-579D-4EF7-9A8B-20B4C40B7C48}]

"UseZeroBroadcast"=dword:00000000

"EnableDeadGWDetect"=dword:00000001

"EnableDHCP"=dword:00000000

"NameServer"=""

"Domain"=""

"RegistrationEnabled"=dword:00000001

"RegisterAdapterName"=dword:00000000

"DhcpServer"="255.255.255.255"

"Lease"=dword:00000000

"LeaseObtainedTime"=dword:00000000

"T1"=dword:00000000

"T2"=dword:00000000

"LeaseTerminatesTime"=dword:00000000

"AddressType"=dword:00000000

"IsServerNapAware"=dword:00000000

"DhcpConnForceBroadcastFlag"=dword:00000000

"IPAddress"=hex(7):31,00,39,00,32,00,2e,00,32,00,2e,00,31,00,2e,00,32,00,00,00,\

  00,00

"SubnetMask"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,32,00,35,\

  00,35,00,2e,00,30,00,00,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\PersistentRoutes]

"0.0.0.0,0.0.0.0,9.3.144.1,-1"=""


[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Winsock]

"HelperDllName"=hex(2):25,00,53,00,79,00,73,00,74,00,65,00,6d,00,52,00,6f,00,\

  6f,00,74,00,25,00,5c,00,53,00,79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,\

  00,77,00,73,00,68,00,74,00,63,00,70,00,69,00,70,00,2e,00,64,00,6c,00,6c,00,\

  00,00

"MaxSockAddrLength"=dword:00000010

"MinSockAddrLength"=dword:00000010

"Mapping"=hex:08,00,00,00,03,00,00,00,02,00,00,00,01,00,00,00,06,00,00,00,02,\

  00,00,00,01,00,00,00,00,00,00,00,02,00,00,00,00,00,00,00,06,00,00,00,02,00,\

  00,00,02,00,00,00,11,00,00,00,02,00,00,00,02,00,00,00,00,00,00,00,02,00,00,\

  00,00,00,00,00,11,00,00,00,02,00,00,00,03,00,00,00,ff,00,00,00,02,00,00,00,\

  03,00,00,00,00,00,00,00

"UseDelayedAcceptance"=dword:00000000
```

tpccCom.tpcc_com_settings.txt

```
Windows Registry Editor Version 5.00


[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\TPCC]

"dbName"="tpcc"

"dbPassword"="tpcc"

"dbType"="DB2"

"dbUserName"="tpcc"

"dlvyLogPath"="c:\\InetPub\\wwwroot\\tpcc\\"

"dlvyQueueLen"=dword:00004e20

"dlvyThreads"=dword:0000000c

"errorLogFile"="c:\\inetpub\\wwwroot\\tpcc\\isapi_err.log"

"htmlTrace"=dword:00000000

"htmlTraceLogFile"="c:\\inetpub\\wwwroot\\tpcc\\isapi.log"

"isapi_trace"=dword:00000000

"nullDB"=dword:00000000

"numServers"=dword:00000001

"numUsers"=dword:0001d4c1

"numWarehouse"=dword:00002cee

"numConnections"=dword:00000041
```

tpcc_software_registry.reg

Transactions not supported
Enable Object pooling
Minimum pool size 68
Maximum pool size 68
Creation timeout 1,900,000,000
Enable Just in time activation
Concurrency Required

10.3. Linux Parameters

/proc/cmdline

```
initrd=mksully/initrd-2.6.32.12-0.7 root=/dev/disk/by-id/scsi-3600a0b800036967f00001a2b4bab6191-part11 elevator=noop unsupported=1 BOOT_IMAGE=mksully/bzImage-2.6.32.12-0.7
```

setup.sh

```
#!/bin/bash

echo  757432 > /proc/sys/vm/nr_hugepages

modprobe qla2xxx; modprobe megaraid_sas; modprobe mpt2sas;

vgscan; vgchange -ay


cd /autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/tools

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/tools/gcreate_ln.sh


/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/tools/setup_aff.sh


echo 0 > /sys/bus/pci/drivers/megaraid_sas/poll_mode_io

for cpu in $(seq 0 63) ; do echo $cpu; echo performance > /sys/devices/system/cpu/cpu${cpu}/cpufreq/scaling_governor; done
```

gcreate_ln.sh

```
#!/bin/bash


{

ssd_disks_file="/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/tools/devices.ssd"

qla_disks_file="/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/tools/devices.qla"

if [ ! -f $ssd_disks_file ]; then

   echo "Could not find input file that contains list of ssds by-id!"

   exit 1

fi

if [ ! -f $qla_disks_file ]; then

   echo "Could not find input file that contains list of qla disks by-id!"

   exit 1

fi

#vgscan

#vgchange -ay

#mount -a

mkdir /dev/tpcc

chmod -R 777 /dev/tpcc

loop=144


i=1

j=1

while read device; do

   let j=($i*2)-1

   s=`printf "%03d" $j`;

   ln -s ${device}-part1 /dev/tpcc/S_$s

   ln -s ${device}-part3 /dev/tpcc/C_$s

   ln -s ${device}-part6 /dev/tpcc/C2_$s

   let j=$j+1

   s=`printf "%03d" $j`;

   ln -s ${device}-part2 /dev/tpcc/S_$s

   ln -s ${device}-part5 /dev/tpcc/C_$s

   ln -s ${device}-part7 /dev/tpcc/C2_$s

   let i=$i+1

done < $ssd_disks_file


i=1

while read device; do

   s=`printf "%03d" $i`;

   ln -s ${device}-part1 /dev/tpcc/W_$s

   ln -s ${device}-part2 /dev/tpcc/D_$s

   ln -s ${device}-part3 /dev/tpcc/N_$s

   ln -s ${device}-part5 /dev/tpcc/O2_$s

   ln -s ${device}-part6 /dev/tpcc/O_$s

   ln -s ${device}-part7 /dev/tpcc/H_$s

   ln -s ${device}-part8 /dev/tpcc/OL_$s

   let i=$i+1

done < $qla_disks_file


i=37

while read device; do

   s=`printf "%03d" $i`;

   ln -s ${device}-part9 /dev/tpcc/OL_$s

   let i=$i+1

done < $qla_disks_file


# I

ln -s /dev/disk/by-id/scsi-3600a0b800034d6ce00001c084bab2ea6-part10 /dev/tpcc/I


# LOG

logdevice=`readlink -f /dev/disk/by-id/scsi-3600605b000a8e5a000000678664f780e`

chmod 777 $logdevice

ln -s /dev/disk/by-id/scsi-3600605b000a8e5a000000678664f780e /dev/tpcc/log

mkdir /dev/buildVG

chmod 777 /dev/buildVG

ln -s /dev/disk/by-id/scsi-3600605b000a8e5a000000678664f780e /dev/buildVG/log

chmod -R 777 /dev/mapper/*

chmod -R 777 /dev/tpcc/

chmod -R 777 /dev/sd*

} 2>&1 | tee $0.out
```

Devices.ssd

```
/dev/disk/by-id/scsi-3600605b000a8d7a80000a9da236e871e
```

/dev/disk/by-id/scsi-3600605b000a8d7a80000a9e123dacc4e

/dev/disk/by-id/scsi-3600605b000a8d7a80000a9e7242dfc0c

/dev/disk/by-id/scsi-3600605b000a8d7a80000a9ed2490ac9a

/dev/disk/by-id/scsi-3600605b000a8d7a80000a9f324e6ca8f

/dev/disk/by-id/scsi-3600605b000a8d7a80000a9fa254e3d54

/dev/disk/by-id/scsi-3600605b000a8d7a80000aa07261189e2

/dev/disk/by-id/scsi-3600605b000a8d7a80000aa0e267c5859

/dev/disk/by-id/scsi-3600605b000a8d7a80000aa1426da84ad

/dev/disk/by-id/scsi-3600605b000a8d7a80000aa1a273cb7f9

/dev/disk/by-id/scsi-3600605b000a8d7a80000aa2127a7c532

/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000XA

/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000QY

/dev/disk/by-id/scsi-3600605b000a8e85000054eb4ffce095f

/dev/disk/by-id/scsi-3600605b000a8e85000010c4901524429

/dev/disk/by-id/scsi-3600605b000a8e8500008082691f16d07

/dev/disk/by-id/scsi-3600605b000a8e5a00007e5758096745c

/dev/disk/by-id/scsi-3600605b000a8e5a00000905f9e8fc9d0

/dev/disk/by-id/scsi-3600605b000a8e3f00000a7ea05c9554b

/dev/disk/by-id/scsi-3600605b000a8e3f00000a7f0062d2d49

/dev/disk/by-id/scsi-3600605b000a8e3f00000a7f50676e785

/dev/disk/by-id/scsi-3600605b000a8e3f00000a7fa06bf97f5

/dev/disk/by-id/scsi-3600605b000a8e3f00000a7ff070f7c29

/dev/disk/by-id/scsi-3600605b000a8e3f00000a80e07f498f9

/dev/disk/by-id/scsi-3600605b000a8e3f00000a81508609feb

/dev/disk/by-id/scsi-3600605b000a8e3f00000a81b08b91020

/dev/disk/by-id/scsi-3600605b000a8e3f00000a8200909de24

/dev/disk/by-id/scsi-3600605b000a8e3f00000a826095fbd69

/dev/disk/by-id/scsi-3600605b000a8e3f00000a82b09b0ca8d

/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000RJ

/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000ZS

/dev/disk/by-id/scsi-3600605b000a8e8500004342b289eb1c0

/dev/disk/by-id/scsi-3600605b000a8e850000a5f164504ffeb

/dev/disk/by-id/scsi-3600605b000a8e8500004342a2892cedd

/dev/disk/by-id/scsi-3600605b000a8e5a000000678665b7472

/dev/disk/by-id/scsi-3600605b000a8e5a00000067866559718

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac20460dce23

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac28468762df

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac2d46ce7d73

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac32471a0c9e

/dev/disk/by-id/scsi-3600605b000a8e3e00000099c963c023d

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac3c47b07f1a

/dev/disk/by-id/scsi-3600605b000a8e3e0000011c812edfa05

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac47486422c0

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac4e48c86480

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac53491aa9da

/dev/disk/by-id/scsi-3600605b000a8e3e00000ac584965108e

/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_5030013B

/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000ZT

/dev/disk/by-id/scsi-3600605b000a8e8500004342a2894ba3c

/dev/disk/by-id/scsi-3600605b000a8e8500004342a2896bf89

/dev/disk/by-id/scsi-3600605b000a8e85000002f18d2362b32

/dev/disk/by-id/scsi-3600605b000a8e5a000000679665d57ce

/dev/disk/by-id/scsi-3600605b000a8e5a00000067866578360

/dev/disk/by-id/scsi-3600000000000000000000003d03f109970

/dev/disk/by-id/scsi-3600605b000a8e610000007f97d4c3c3b

/dev/disk/by-id/scsi-3600000000000000000000003d03f1488ea

/dev/disk/by-id/scsi-3600000000000000000000003d03f167512

/dev/disk/by-id/scsi-3600000000000000000000003d13f1876c8

/dev/disk/by-id/scsi-3600000000000000000000003d13f1a7c7d

/dev/disk/by-id/scsi-3600000000000000000000003d13f1c71ed

/dev/disk/by-id/scsi-3600000000000000000000003d13f1e720d

/dev/disk/by-id/scsi-3600000000000000000000003d13f2067af

/dev/disk/by-id/scsi-3600000000000000000000003d13f226973

/dev/disk/by-id/scsi-3600000000000000000000003d13f246713

/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000Z8

/dev/disk/by-id/scsi-SATA_SG9XCS1B200GEIB_503000XE

/dev/disk/by-id/scsi-3600605b000a8e8500004342a289aa9ba

/dev/disk/by-id/scsi-3600605b000a8e85000039ed341d30faa

/dev/disk/by-id/scsi-3600605b000a8e5a00000067866598cca

/dev/disk/by-id/scsi-3600605b000a8e5a00000067966612155

/dev/disk/by-id/scsi-3600605b000a8e5a000000679665f3c6a

Devices.qla

/dev/disk/by-id/scsi-3600a0b800034d6ce00001c084bab2ea6

/dev/disk/by-id/scsi-3600a0b800034d6ce00001c0a4bab2eda

/dev/disk/by-id/scsi-3600a0b800024589800001a254bab5798

/dev/disk/by-id/scsi-3600a0b800024589800001a274bab57c8

/dev/disk/by-id/scsi-3600a0b800036fefd00001d714bab6032

/dev/disk/by-id/scsi-3600a0b800036fefd00001d734bab6066

/dev/disk/by-id/scsi-3600a0b80003460b700001d584bab33cc

/dev/disk/by-id/scsi-3600a0b80003460b700001d5a4bab33fe

/dev/disk/by-id/scsi-3600a0b800036945800001ae54bab60a2

/dev/disk/by-id/scsi-3600a0b800036945800001ae74bab60da

/dev/disk/by-id/scsi-3600a0b800034d6cd00001d2b4bab31ee

/dev/disk/by-id/scsi-3600a0b800034d6cd00001d2d4bab3224

/dev/disk/by-id/scsi-3600a0b800036985400001aec4bab5930

/dev/disk/by-id/scsi-3600a0b800036985400001aee4bab5960

/dev/disk/by-id/scsi-3600a0b800034d5a8000081244bab344a

/dev/disk/by-id/scsi-3600a0b800034d5a8000081264bab347a

/dev/disk/by-id/scsi-3600a0b800036991b00001b634bab5e24

/dev/disk/by-id/scsi-3600a0b800036991b00001b654bab5e54

/dev/disk/by-id/scsi-3600a0b800034d6bb00001ace4bab32b7

/dev/disk/by-id/scsi-3600a0b800034d6bb00001ad04bab330f

/dev/disk/by-id/scsi-3600a0b800034d6c500001c4f4bab3421

/dev/disk/by-id/scsi-3600a0b800034d6c500001c514bab3479

/dev/disk/by-id/scsi-3600a0b800034608d000026374bab35eb

```
/dev/disk/by-id/scsi-3600a0b800034608d000026394bab3643
/dev/disk/by-id/scsi-3600a0b800036969000001ad74bab5f9b
/dev/disk/by-id/scsi-3600a0b800036969000001ad94bab5ff3
/dev/disk/by-id/scsi-3600a0b8000245851000019b34bab587b
/dev/disk/by-id/scsi-3600a0b8000245851000019b54bab58d3
/dev/disk/by-id/scsi-3600a0b800034d70b00001bd44bab362d
/dev/disk/by-id/scsi-3600a0b800034d70b00001bd64bab3687
/dev/disk/by-id/scsi-3600a0b800034d6bd00001bcf4bab37d7
/dev/disk/by-id/scsi-3600a0b800034d6bd00001bd14bab3831
/dev/disk/by-id/scsi-3600a0b800036967f00001a2b4bab6191
/dev/disk/by-id/scsi-3600a0b800036967f00001a2d4bab61eb
/dev/disk/by-id/scsi-3600a0b800036986400001b084bab5db7
/dev/disk/by-id/scsi-3600a0b800036986400001b0a4bab5e11
```

setup_aff.sh

```
#!/bin/bash

echo "Affinitizing ironville SAS Interrupts"
sas_irqs=`grep mpt2sas /proc/interrupts | awk '{ print $1 }' | sed -e 's/://`
counter=1
for irq in $sas_irqs; do
   echo "Before: /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"
   case "$counter" in
      "1")
         echo "40 > /proc/irq/${irq}/smp_affinity"
         echo  40 > /proc/irq/${irq}/smp_affinity
         ;;
      "2")
         echo "2000000 > /proc/irq/${irq}/smp_affinity"
         echo  2000000 > /proc/irq/${irq}/smp_affinity
         ;;
   esac
   echo "After:  /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"

   (( counter += 1 ))
done


# enable interrupt affinity for the LSI MegaRAID SAS adapter (marmet)
echo "Affinitizing LSI MegaRAID (marmet) SAS Interrupts"
sas_irqs=`grep megasas /proc/interrupts | awk '{ print $1 }' | sed -e 's/://`
counter=1
for irq in $sas_irqs; do
   echo "Before: /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"
   case "$counter" in
      "1")
         echo "200 > /proc/irq/${irq}/smp_affinity"
         echo  200 > /proc/irq/${irq}/smp_affinity
         ;;
      "2")
         echo "4000 > /proc/irq/${irq}/smp_affinity"
         echo  4000 > /proc/irq/${irq}/smp_affinity
         ;;
      "3")
         echo "2 > /proc/irq/${irq}/smp_affinity"
         echo  2 > /proc/irq/${irq}/smp_affinity
         ;;
      "4")
         echo "20000 > /proc/irq/${irq}/smp_affinity"
         echo  20000 > /proc/irq/${irq}/smp_affinity
         ;;
      "5")
         echo "2000000 > /proc/irq/${irq}/smp_affinity"
         echo  2000000 > /proc/irq/${irq}/smp_affinity
         ;;
      "6")
         echo "40000000 > /proc/irq/${irq}/smp_affinity"
         echo  40000000 > /proc/irq/${irq}/smp_affinity
         ;;
   esac
   echo "After:  /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"

   (( counter += 1 ))
done


# enable interrupt affinity for QLogic Quad Port FC adapters
echo "Affinitizing QLogic Interrupts"
qlogic_irqs=`grep "qla2xxx" /proc/interrupts | awk '{ print $1 }' | sed -e 's/://`
counter=1
for irq in $qlogic_irqs; do
   echo "Before: /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"
   case "$counter" in
      "1")
         echo "10000 > /proc/irq/${irq}/smp_affinity"
         echo  10000 > /proc/irq/${irq}/smp_affinity
         ;;
      "2")
         echo "40000 > /proc/irq/${irq}/smp_affinity"
         echo  40000 > /proc/irq/${irq}/smp_affinity
         ;;
      "3")
         echo "4000000 > /proc/irq/${irq}/smp_affinity"
         echo  4000000 > /proc/irq/${irq}/smp_affinity
         ;;
      "4")
```

```
        echo "8000000 > /proc/irq/${irq}/smp_affinity"
        echo  8000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "5")
        echo "1000000 > /proc/irq/${irq}/smp_affinity"
        echo  1000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "6")
        echo "2000000 > /proc/irq/${irq}/smp_affinity"
        echo  2000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "7")
        echo "40000000 > /proc/irq/${irq}/smp_affinity"
        echo  40000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "8")
        echo "20000000 > /proc/irq/${irq}/smp_affinity"
        echo  20000000 > /proc/irq/${irq}/smp_affinity
        ;;
    esac
    echo "After:  /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"

    (( counter += 1 ))
done

echo "Affinitizing Eth0 Interrupts"
eth0_irqs=`grep "eth0" /proc/interrupts | awk '{ print $1 }' | sed -e 's/://`
counter=1
for irq in $eth0_irqs; do
    echo "Before: /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"
    case "$counter" in
      "1")
        echo "800000 > /proc/irq/${irq}/smp_affinity"
        echo  800000 > /proc/irq/${irq}/smp_affinity
        ;;
      "2")
        echo "800000 > /proc/irq/${irq}/smp_affinity"
        echo  800000 > /proc/irq/${irq}/smp_affinity
        ;;
      "3")
        echo "800000 > /proc/irq/${irq}/smp_affinity"
        echo  800000 > /proc/irq/${irq}/smp_affinity
        ;;
      "4")
        echo "800000 > /proc/irq/${irq}/smp_affinity"
        echo  800000 > /proc/irq/${irq}/smp_affinity
        ;;
      "5")
        echo "800000 > /proc/irq/${irq}/smp_affinity"
        echo  800000 > /proc/irq/${irq}/smp_affinity
        ;;
      "6")
        echo "800000 > /proc/irq/${irq}/smp_affinity"
        echo  800000 > /proc/irq/${irq}/smp_affinity
        ;;
      "7")
        echo "800000 > /proc/irq/${irq}/smp_affinity"
        echo  800000 > /proc/irq/${irq}/smp_affinity
        ;;
      "8")
        echo "800000 > /proc/irq/${irq}/smp_affinity"
        echo  800000 > /proc/irq/${irq}/smp_affinity
        ;;
    esac
    echo "After:  /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"

    (( counter += 1 ))
done

echo "Affinitizing Eth1 Interrupts"
eth0_irqs=`grep "eth1" /proc/interrupts | awk '{ print $1 }' | sed -e 's/://`
counter=1
for irq in $eth0_irqs; do
    echo "Before: /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"
    case "$counter" in
      "1")
        echo "80000000 > /proc/irq/${irq}/smp_affinity"
        echo  80000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "2")
        echo "80000000 > /proc/irq/${irq}/smp_affinity"
        echo  80000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "3")
        echo "80000000 > /proc/irq/${irq}/smp_affinity"
        echo  80000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "4")
        echo "80000000 > /proc/irq/${irq}/smp_affinity"
        echo  80000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "5")
        echo "80000000 > /proc/irq/${irq}/smp_affinity"
        echo  80000000 > /proc/irq/${irq}/smp_affinity
        ;;
      "6")
```

```
        echo "80000000 > /proc/irq/${irq}/smp_affinity"
        echo  80000000 > /proc/irq/${irq}/smp_affinity
        ;;
    "7")
        echo "80000000 > /proc/irq/${irq}/smp_affinity"
        echo  80000000 > /proc/irq/${irq}/smp_affinity
        ;;
    "8")
        echo "80000000 > /proc/irq/${irq}/smp_affinity"
        echo  80000000 > /proc/irq/${irq}/smp_affinity
        ;;
  esac
  echo "After:  /proc/irq/${irq}/smp_affinity=[`cat /proc/irq/${irq}/smp_affinity`]"

  (( counter += 1 ))
done


# enable interrupt coalescing for QLogic FC adapters
pushd /sys/devices
echo "Before enabling interrupt coalescing for QLogic FC adapters"
find . -name zio | xargs -I '{}' bash -c 'echo -e "\t{}=[`cat {}`]"'
find . -name zio_timer | xargs -I '{}' bash -c 'echo -e "\t{}=[`cat {}`]"'

echo "Enabling interrupt coalescing for QLogic FC adapters"
find . -name zio | xargs -I '{}' bash -c 'echo 1 > {}'

echo "After enabling interrupt coalescing for QLogic FC adapters"
find . -name zio | xargs -I '{}' bash -c 'echo -e "\t{}=[`cat {}`]"'
find . -name zio_timer | xargs -I '{}' bash -c 'echo -e "\t{}=[`cat {}`]"'
popd

# enable interrupt coalescing for Broadcom Ethernet adapters
echo "Before enabling interupt coalescing for Broadcom Ethernet adapters"
ethtool -c eth0
ethtool -c eth1

echo "Enabling interrupt coalescing for Broadcom Ethernet adapters"
echo "eth0"
ethtool -C eth0 tx-usecs 160 tx-frames 0 rx-usecs 300 rx-frames 0
echo "eth1"
ethtool -C eth1 tx-usecs 160 tx-frames 0 rx-usecs 300 rx-frames 0

echo "After enabling interupt coalescing for Broadcom Ethernet adapters"
ethtool -c eth0
ethtool -c eth1
```

priority-tuning.sh

```
#!/bin/bash

su - tpcc -c "db2pd -edus > /autobench/sources/db2_tpcc-1/users/tpcc/db2pd.output"


echo "Logwriter:"

logwriter_pid=`grep db2loggw /autobench/sources/db2_tpcc-1/users/tpcc/db2pd.output |
sed -e 's/^.\{,25\}\([0-9]\+\)\s.*/\1/`

chrt -f -p 97 $logwriter_pid


echo "pagecleaners"

for pid in `grep db2pclnr /autobench/sources/db2_tpcc-1/users/tpcc/db2pd.output | sed -e
's/^.\{,25\}\([0-9]\+\)\s.*/\1/`; do

        if chrt -o -p 0 $pid; then

            renice -n -1 $pid

        fi

done
```

/etc/sysctl.conf

```
# Disable response to broadcasts.
# You don't want yourself becoming a Smurf amplifier.
net.ipv4.icmp_echo_ignore_broadcasts = 1
# enable route verification on all interfaces
net.ipv4.conf.all.rp_filter = 1
# increase the number of possible inotify(7) watches
fs.inotify.max_user_watches = 65536
# avoid deleting secondary IPs on deleting the primary IP
net.ipv4.conf.default.promote_secondaries = 1
net.ipv4.conf.all.promote_secondaries = 1
kernel.sem = 500        512000  64      2048
kernel.msgmni = 4096
fs.file-max = 524288
net.core.rmem_max = 131071
net.core.wmem_max = 131071
vm.hugetlb_shm_group = 12090
kernel.shmmax = 18446744073709551615
kernel.shmall = 2252921504606846720
kernel.shmmni = 4096
```

restore.sh

```
#!/bin/bash


option=-1
case "$1" in
  "full")
    option=0
    ;;
  "bp")
```

```
        option=1
        ;;
    *)
        echo "Please supply a valid option"
        exit
        ;;
esac

if [ $option -eq 0 ]; then
    # make sure db2 is stopped
    su - tpcc -c "db2stop force;"
    # copy from the xfs backup
    pushd /autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/tools > /dev/null
    ./set-io-readahead.sh 2048
    ./set-io-scheduler.sh noop
      perl ./static.pl &
    staticpid=$!
     ssh -o IdentityFile=/autobench/ssh/autobench-id -o StrictHostKeyChecking=no
tpccboost "perl /root/dynamic.pl"
    # note: tpccboost does not have enough space to keep a full backup of dynamic files so
we will do the last handful from tpccgids backup
    dd if=/backup/OL_067 of=/dev/tpcc/OL_067 iflag=direct oflag=direct bs=256M &
    dd if=/backup/OL_068 of=/dev/tpcc/OL_068 iflag=direct oflag=direct bs=256M &
    dd if=/backup/OL_069 of=/dev/tpcc/OL_069 iflag=direct oflag=direct bs=256M &
    dd if=/backup/OL_070 of=/dev/tpcc/OL_070 iflag=direct oflag=direct bs=256M &
    dd if=/backup/OL_071 of=/dev/tpcc/OL_071 iflag=direct oflag=direct bs=256M &
    dd if=/backup/OL_072 of=/dev/tpcc/OL_072 iflag=direct oflag=direct bs=256M
    ./set-io-scheduler.sh noop
    popd > /dev/null

    # cleanup and unpack the database directory
    echo "Restoring db_home"
    pushd /autobench/sources/db2_tpcc-1 > /dev/null
    rm -Rf db_home
    tar xzf db_home_backup_ddat.tar.gz
    popd > /dev/null
fi
ulimit -v unlimited

# start db2,
su - tpcc -c "db2stop; db2set DB2_LARGE_PAGE_MEM=DB; db2start;"
```

Dynamic.pl

```
#!/usr/bin/perl

use strict;
use Getopt::Long;
use Data::Dumper;
use threads qw(yield);
use threads::shared;
use Thread::Queue;
use Thread::Semaphore;

# disable output buffering
$| = 1;

my %options;
Getopt::Long::Configure("bundling");
Getopt::Long::Configure ("no_auto_abbrev");
GetOptions(\%options,
        'test',             # only run in test mode
        );

if ($options{'test'}) {
    my_print("Running in test mode.\n", 0);
}

my $backupdir = "/backup";
my $num_threads = 36;

my @source_volume_names = ("N_", "O_", "OL_", "H_", "O2_", "W_", "D_", "I");
my %num_arms = ("N_"   => 36,
            "O_"   => 36,
            "OL_"  => 72,
            "H_"   => 36,
            "O2_"  => 36,
            "W_"   => 36,
            "D_"   => 36,
            "I"    =>  0);

my @jobs;
my $manager_thread;

my $job_counter=1;
my $uas=144;

for (my $i=0; $i<@source_volume_names; $i++) {
    $uas=$num_arms{$source_volume_names[$i]};
    if ($uas != 0) {

        for (my $x=0; $x<$uas; $x++) {
            push @jobs, { 'id'   => $job_counter,
                    'type' => "ls",
                    'dest' => "/dev/tpcc/$source_volume_names[$i]" . prepend($x+1),
                    'src'  => "$backupdir/$source_volume_names[$i]" . prepend($x+1) };
            $job_counter++;
```

```
        }
    } else {

        push @jobs, { 'id'   => $job_counter,
                      'type' => "ls",
                      'dest'  => "/dev/tpcc/$source_volume_names[$i]",
                      'src' => "$backupdir/$source_volume_names[$i]" };

        $job_counter++;
    }
}

#print Dumper \@jobs;

my $queue = new Thread::Queue;

for (my $i=0; $i<@jobs; $i++) {
    $queue->enqueue($i);
}

$manager_thread = threads->create('manager_thread', $queue, $num_threads, \@jobs);
$manager_thread->join;

##########################################################################
##############################

sub worker_thread {
    my ($thread_num, $work_queue, $jobs, $finished, $threads_finished, $lock) = @_;

    # sync point
    sleep 5;

    my_print("WORKER : thread=[$thread_num] starting.\n", 2);

    while (!$$finished) {
        my $job_id = $work_queue->dequeue_nb;

        if (defined($job_id)) {
            my_print("WORKER : thread=[$thread_num] retreived job=[" . $jobs[$job_id]-
>{'id'} . "] type=[" . $jobs[$job_id]->{'type'} . "]\n", 2);

            my @args = ();

            if ($options{'test'}) {
                my $str = "";
                for (my $i=0; $i<3; $i++) {
                    $str .= "   ";
                }
                push @args, ("echo", $str, "WORKER : thread=[$thread_num]");
            }

            push @args, ("dd", "if=" . $jobs[$job_id]->{'src'}, "of=" . $jobs[$job_id]->{'dest'},
"iflag=direct", "oflag=direct", "bs=256M");
#           push @args, ("dd", "if=" . $jobs[$job_id]->{'src'}, "of=/dev/null", "iflag=direct",
"bs=256M");

            system(@args);
        } else {
            sleep 1;
        }
    }

    $lock->down(1);
    $$threads_finished++;
    $lock->up(1);
    my_print("WORKER : thread=[$thread_num] finished\n", 2);
}

sub manager_thread {
    my ($job_queue, $thread_count, $jobs) = @_;

    my @threads;
    my $finished : shared = 0;
    my $threads_finished : shared = 0;
    my $lock = new Thread::Semaphore(1);

    my_print("MANAGER : Jobs:\t" . $job_queue->pending . "\n", 1);
    my_print("MANAGER : Parallel Tasks:\t" . $thread_count . "\n", 1);

    my_print("MANAGER : Starting threads:", 1);
    for (my $i=0; $i<$thread_count; $i++) {
        push @threads, threads->create('worker_thread', $i, $job_queue, $jobs, \$finished,
\$threads_finished, $lock);
        print " $i";
    }
    print "\n";

    while ($job_queue->pending) {
        my_print("MANAGER : There are " . $job_queue->pending . " jobs waiting in the
queue\n", 1);
        sleep 5;
    }

    my_print("MANAGER : Cleaning up worker threads\n", 1);
    sleep 10;
    $finished = 1;

    while ($threads_finished != $thread_count) {
```

```
        my_print("MANAGER : There are " . ($thread_count - $threads_finished) . " threads still processing jobs\n", 1);
        sleep 5;
    }

    for (my $i=0; $i<@threads; $i++) {
        $threads[$i]->join;
    }
}

sub my_print {
    my ($msg, $level) = @_;

    my $str = "";
    for (my $i=0; $i<$level; $i++) {
        $str .= "   ";
    }

    print $str . $msg;
}

sub prepend {
    my $num = shift;
    my $length = length($num);
    my $i;
    $length = 3 - $length;
    for ($i=0; $i<$length; $i++) {
        $num = "0$num";
    }
    return $num;
}
```

Static.pl

```
#!/usr/bin/perl

use strict;
use Getopt::Long;
use Data::Dumper;
use threads qw(yield);
use threads::shared;
use Thread::Queue;
use Thread::Semaphore;

# disable output buffering
$| = 1;

my %options;
Getopt::Long::Configure("bundling");
Getopt::Long::Configure ("no_auto_abbrev");
GetOptions(\%options,
        'test',             # only run in test mode
        );

if ($options{'test'}) {
    my_print("Running in test mode.\n", 0);
}

my $backupdir = "/backup";
my $num_threads = 144;

my @source_volume_names = ("S_", "C_", "C2_");
my %num_arms = ("S_"   => 144,
            "C_"   => 144,
            "C2_"  => 144);

my @jobs;
my $manager_thread;

my $job_counter=1;
my $uas=144;

for (my $i=0; $i<@source_volume_names; $i++) {
    $uas=$num_arms{$source_volume_names[$i]};
    if ($uas != 0) {
        for (my $x=0; $x<$uas; $x++) {

            if ($x > 24) {
                $backupdir = "/backup";
            } else {
                $backupdir = "/backupsas";
            }

            push @jobs, { 'id'   => $job_counter,
                        'type' => "ls",
                        'dest' => "/dev/tpcc/$source_volume_names[$i]" . prepend($x+1),
                        'src'  => "$backupdir/$source_volume_names[$i]" . prepend($x+1) };
            $job_counter++;
        }
    } else {

        push @jobs, { 'id'   => $job_counter,
                    'type' => "ls",
                    'dest'  => "/dev/tpcc/$source_volume_names[$i]",
                    'src' => "$backupdir/$source_volume_names[$i]" };

        $job_counter++;
```

```
    }
}

#print Dumper \@jobs;

my $queue = new Thread::Queue;

for (my $i=0; $i<@jobs; $i++) {
    $queue->enqueue($i);
}

$manager_thread = threads->create('manager_thread', $queue, $num_threads, \@jobs);
$manager_thread->join;

##########################################################################
##############################

sub worker_thread {
    my ($thread_num, $work_queue, $jobs, $finished, $threads_finished, $lock) = @_;

    # sync point
    sleep 5;

    my_print("WORKER : thread=[$thread_num] starting.\n", 2);

    while (!$$finished) {
        my $job_id = $work_queue->dequeue_nb;

        if (defined($job_id)) {
            my_print("WORKER : thread=[$thread_num] retreived job=[" . $jobs[$job_id]-
>{'id'} . "] type=[" . $jobs[$job_id]->{'type'} . "]\n", 2);

            my @args = ();

            if ($options{'test'}) {
                my $str = "";
                for (my $i=0; $i<3; $i++) {
                    $str .= "   ";
                }
                push @args, ("echo", $str, "WORKER : thread=[$thread_num]");
            }

            push @args, ("dd", "if=" . $jobs[$job_id]->{'src'}, "of=" . $jobs[$job_id]->{'dest'},
"iflag=direct", "oflag=direct", "bs=8M");
#           push @args, ("dd", "if=" . $jobs[$job_id]->{'src'}, "of=/dev/null", "iflag=direct",
"bs=8M");

            system(@args);
        } else {
            sleep 1;
        }
    }

    $lock->down(1);
    $$threads_finished++;
    $lock->up(1);
    my_print("WORKER : thread=[$thread_num] finished\n", 2);
}

sub manager_thread {
    my ($job_queue, $thread_count, $jobs) = @_;

    my @threads;
    my $finished : shared = 0;
    my $threads_finished : shared = 0;
    my $lock = new Thread::Semaphore(1);

    my_print("MANAGER : Jobs:\t" . $job_queue->pending . "\n", 1);
    my_print("MANAGER : Parallel Tasks:\t" . $thread_count . "\n", 1);

    my_print("MANAGER : Starting threads:", 1);
    for (my $i=0; $i<$thread_count; $i++) {
        push @threads, threads->create('worker_thread', $i, $job_queue, $jobs, \$finished,
\$threads_finished, $lock);
        print " $i";
    }
    print "\n";

    while ($job_queue->pending) {
        my_print("MANAGER : There are " . $job_queue->pending . " jobs waiting in the
queue\n", 1);
        sleep 5;
    }

    my_print("MANAGER : Cleaning up worker threads\n", 1);
    sleep 10;
    $finished = 1;

    while ($threads_finished != $thread_count) {
        my_print("MANAGER : There are " . ($thread_count - $threads_finished) . " threads
still processing jobs\n", 1);
        sleep 5;
    }

    for (my $i=0; $i<@threads; $i++) {
        $threads[$i]->join;
    }
}
```

```
sub my_print {
    my ($msg, $level) = @_;

    my $str = "";
    for (my $i=0; $i<$level; $i++) {
        $str .= "   ";
    }

    print $str . $msg;
}

sub prepend {
    my $num = shift;
    my $length = length($num);
    my $i;
    $length = 3 - $length;
    for ($i=0; $i<$length; $i++) {
        $num = "0$num";
    }
    return $num;
}
```

/proc/version

```
Linux version 2.6.32.12-0.7 (geeko@buildhost) (gcc version 4.3.4 [gcc-4_3-branch
revision 152973] (SUSE Linux) ) #4 SMP Wed Jul 28 12:39:13 CDT 2010
```

11 Appendix C: Database Setup Code

11.1. Database Creation Scripts

create_bufferpool.ddl

```
-----------------------------------------------------------------------------
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----------------------------------------------------------------------------
--
-- create_bufferpool.ddl.sample - Sample Create Bufferpool DDL
--

connect to tpcc;
create bufferpool IBMDEFAULT8K size 1000 pagesize 8192;
create bufferpool IBMDEFAULT16K size 1000 pagesize 16384;
connect reset;
```

create_database.ddl

```
-----------------------------------------------------------------------------
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----------------------------------------------------------------------------
--
-- create_database.ddl.sample - Sample Create Database DDL
--

drop database tpcc;
create database tpcc on /autobench/sources/db2_tpcc-1/db_home using codeset
ISO8859-1 territory US collate using identity catalog tablespace managed by system
using ('/autobench/sources/db2_tpcc-1/db_home/db1catalog');
```

create_tablespace.ddl

```
connect to tpcc;
drop tablespace ts_W_001;
create regular tablespace ts_W_001 pagesize 4K managed by database using ( device
'/dev/tpcc/W_001' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_002;
create regular tablespace ts_W_002 pagesize 4K managed by database using ( device
'/dev/tpcc/W_002' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_003;
create regular tablespace ts_W_003 pagesize 4K managed by database using ( device
'/dev/tpcc/W_003' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_004;
create regular tablespace ts_W_004 pagesize 4K managed by database using ( device
'/dev/tpcc/W_004' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_005;
create regular tablespace ts_W_005 pagesize 4K managed by database using ( device
'/dev/tpcc/W_005' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_006;
create regular tablespace ts_W_006 pagesize 4K managed by database using ( device
'/dev/tpcc/W_006' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_007;
create regular tablespace ts_W_007 pagesize 4K managed by database using ( device
'/dev/tpcc/W_007' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_008;
create regular tablespace ts_W_008 pagesize 4K managed by database using ( device
'/dev/tpcc/W_008' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_009;
create regular tablespace ts_W_009 pagesize 4K managed by database using ( device
'/dev/tpcc/W_009' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_010;
create regular tablespace ts_W_010 pagesize 4K managed by database using ( device
'/dev/tpcc/W_010' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_011;
create regular tablespace ts_W_011 pagesize 4K managed by database using ( device
'/dev/tpcc/W_011' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_012;
create regular tablespace ts_W_012 pagesize 4K managed by database using ( device
'/dev/tpcc/W_012' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_013;
create regular tablespace ts_W_013 pagesize 4K managed by database using ( device
'/dev/tpcc/W_013' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_014;
create regular tablespace ts_W_014 pagesize 4K managed by database using ( device
'/dev/tpcc/W_014' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_015;
create regular tablespace ts_W_015 pagesize 4K managed by database using ( device
'/dev/tpcc/W_015' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_016;
create regular tablespace ts_W_016 pagesize 4K managed by database using ( device
'/dev/tpcc/W_016' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_017;
create regular tablespace ts_W_017 pagesize 4K managed by database using ( device
'/dev/tpcc/W_017' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_018;
create regular tablespace ts_W_018 pagesize 4K managed by database using ( device
'/dev/tpcc/W_018' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_019;
create regular tablespace ts_W_019 pagesize 4K managed by database using ( device
'/dev/tpcc/W_019' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_020;
create regular tablespace ts_W_020 pagesize 4K managed by database using ( device
'/dev/tpcc/W_020' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_021;
create regular tablespace ts_W_021 pagesize 4K managed by database using ( device
'/dev/tpcc/W_021' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_022;
create regular tablespace ts_W_022 pagesize 4K managed by database using ( device
'/dev/tpcc/W_022' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_023;
create regular tablespace ts_W_023 pagesize 4K managed by database using ( device
'/dev/tpcc/W_023' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_024;
create regular tablespace ts_W_024 pagesize 4K managed by database using ( device
'/dev/tpcc/W_024' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_025;
create regular tablespace ts_W_025 pagesize 4K managed by database using ( device
'/dev/tpcc/W_025' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_026;
create regular tablespace ts_W_026 pagesize 4K managed by database using ( device
'/dev/tpcc/W_026' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_027;
create regular tablespace ts_W_027 pagesize 4K managed by database using ( device
'/dev/tpcc/W_027' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_028;
create regular tablespace ts_W_028 pagesize 4K managed by database using ( device
'/dev/tpcc/W_028' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_029;
create regular tablespace ts_W_029 pagesize 4K managed by database using ( device
'/dev/tpcc/W_029' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_030;
create regular tablespace ts_W_030 pagesize 4K managed by database using ( device
'/dev/tpcc/W_030' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_031;
create regular tablespace ts_W_031 pagesize 4K managed by database using ( device
'/dev/tpcc/W_031' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_032;
create regular tablespace ts_W_032 pagesize 4K managed by database using ( device
'/dev/tpcc/W_032' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_033;
```

```
create regular tablespace ts_W_033 pagesize 4K managed by database using ( device
'/dev/tpcc/W_033' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_034;
create regular tablespace ts_W_034 pagesize 4K managed by database using ( device
'/dev/tpcc/W_034' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_035;
create regular tablespace ts_W_035 pagesize 4K managed by database using ( device
'/dev/tpcc/W_035' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_W_036;
create regular tablespace ts_W_036 pagesize 4K managed by database using ( device
'/dev/tpcc/W_036' 1024 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_001;
create regular tablespace ts_D_001 pagesize 4K managed by database using ( device
'/dev/tpcc/D_001' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_002;
create regular tablespace ts_D_002 pagesize 4K managed by database using ( device
'/dev/tpcc/D_002' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_003;
create regular tablespace ts_D_003 pagesize 4K managed by database using ( device
'/dev/tpcc/D_003' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_004;
create regular tablespace ts_D_004 pagesize 4K managed by database using ( device
'/dev/tpcc/D_004' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_005;
create regular tablespace ts_D_005 pagesize 4K managed by database using ( device
'/dev/tpcc/D_005' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_006;
create regular tablespace ts_D_006 pagesize 4K managed by database using ( device
'/dev/tpcc/D_006' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_007;
create regular tablespace ts_D_007 pagesize 4K managed by database using ( device
'/dev/tpcc/D_007' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_008;
create regular tablespace ts_D_008 pagesize 4K managed by database using ( device
'/dev/tpcc/D_008' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_009;
create regular tablespace ts_D_009 pagesize 4K managed by database using ( device
'/dev/tpcc/D_009' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_010;
create regular tablespace ts_D_010 pagesize 4K managed by database using ( device
'/dev/tpcc/D_010' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_011;
create regular tablespace ts_D_011 pagesize 4K managed by database using ( device
'/dev/tpcc/D_011' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_012;
create regular tablespace ts_D_012 pagesize 4K managed by database using ( device
'/dev/tpcc/D_012' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_013;
create regular tablespace ts_D_013 pagesize 4K managed by database using ( device
'/dev/tpcc/D_013' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_014;
create regular tablespace ts_D_014 pagesize 4K managed by database using ( device
'/dev/tpcc/D_014' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_015;
create regular tablespace ts_D_015 pagesize 4K managed by database using ( device
'/dev/tpcc/D_015' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_016;
create regular tablespace ts_D_016 pagesize 4K managed by database using ( device
'/dev/tpcc/D_016' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_017;
create regular tablespace ts_D_017 pagesize 4K managed by database using ( device
'/dev/tpcc/D_017' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_018;
create regular tablespace ts_D_018 pagesize 4K managed by database using ( device
'/dev/tpcc/D_018' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_019;
create regular tablespace ts_D_019 pagesize 4K managed by database using ( device
'/dev/tpcc/D_019' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_020;
create regular tablespace ts_D_020 pagesize 4K managed by database using ( device
'/dev/tpcc/D_020' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_021;
create regular tablespace ts_D_021 pagesize 4K managed by database using ( device
'/dev/tpcc/D_021' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_022;
create regular tablespace ts_D_022 pagesize 4K managed by database using ( device
'/dev/tpcc/D_022' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_023;
create regular tablespace ts_D_023 pagesize 4K managed by database using ( device
'/dev/tpcc/D_023' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_024;
create regular tablespace ts_D_024 pagesize 4K managed by database using ( device
'/dev/tpcc/D_024' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_025;
create regular tablespace ts_D_025 pagesize 4K managed by database using ( device
'/dev/tpcc/D_025' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_026;
create regular tablespace ts_D_026 pagesize 4K managed by database using ( device
'/dev/tpcc/D_026' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_027;
create regular tablespace ts_D_027 pagesize 4K managed by database using ( device
'/dev/tpcc/D_027' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_028;
create regular tablespace ts_D_028 pagesize 4K managed by database using ( device
'/dev/tpcc/D_028' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_029;
create regular tablespace ts_D_029 pagesize 4K managed by database using ( device
'/dev/tpcc/D_029' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_030;
create regular tablespace ts_D_030 pagesize 4K managed by database using ( device
'/dev/tpcc/D_030' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_031;
create regular tablespace ts_D_031 pagesize 4K managed by database using ( device
'/dev/tpcc/D_031' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_032;
create regular tablespace ts_D_032 pagesize 4K managed by database using ( device
'/dev/tpcc/D_032' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_033;
create regular tablespace ts_D_033 pagesize 4K managed by database using ( device
'/dev/tpcc/D_033' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_034;
create regular tablespace ts_D_034 pagesize 4K managed by database using ( device
'/dev/tpcc/D_034' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_035;
create regular tablespace ts_D_035 pagesize 4K managed by database using ( device
'/dev/tpcc/D_035' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_D_036;
create regular tablespace ts_D_036 pagesize 4K managed by database using ( device
'/dev/tpcc/D_036' 6144 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_I;
create regular tablespace ts_I pagesize 8K managed by database using ( device
'/dev/tpcc/I' 6144 ) extentsize 16 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_S_001;
create regular tablespace ts_S_001 pagesize 4K managed by database using ( device
'/dev/tpcc/S_001' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_002;
create regular tablespace ts_S_002 pagesize 4K managed by database using ( device
'/dev/tpcc/S_002' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_003;
create regular tablespace ts_S_003 pagesize 4K managed by database using ( device
'/dev/tpcc/S_003' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_004;
create regular tablespace ts_S_004 pagesize 4K managed by database using ( device
'/dev/tpcc/S_004' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_005;
create regular tablespace ts_S_005 pagesize 4K managed by database using ( device
'/dev/tpcc/S_005' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_006;
create regular tablespace ts_S_006 pagesize 4K managed by database using ( device
'/dev/tpcc/S_006' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
```

```
drop tablespace ts_S_007;
create regular tablespace ts_S_007 pagesize 4K managed by database using ( device
'/dev/tpcc/S_007' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_008;
create regular tablespace ts_S_008 pagesize 4K managed by database using ( device
'/dev/tpcc/S_008' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_009;
create regular tablespace ts_S_009 pagesize 4K managed by database using ( device
'/dev/tpcc/S_009' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_010;
create regular tablespace ts_S_010 pagesize 4K managed by database using ( device
'/dev/tpcc/S_010' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_011;
create regular tablespace ts_S_011 pagesize 4K managed by database using ( device
'/dev/tpcc/S_011' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_012;
create regular tablespace ts_S_012 pagesize 4K managed by database using ( device
'/dev/tpcc/S_012' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_013;
create regular tablespace ts_S_013 pagesize 4K managed by database using ( device
'/dev/tpcc/S_013' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_014;
create regular tablespace ts_S_014 pagesize 4K managed by database using ( device
'/dev/tpcc/S_014' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_015;
create regular tablespace ts_S_015 pagesize 4K managed by database using ( device
'/dev/tpcc/S_015' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_016;
create regular tablespace ts_S_016 pagesize 4K managed by database using ( device
'/dev/tpcc/S_016' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_017;
create regular tablespace ts_S_017 pagesize 4K managed by database using ( device
'/dev/tpcc/S_017' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_018;
create regular tablespace ts_S_018 pagesize 4K managed by database using ( device
'/dev/tpcc/S_018' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_019;
create regular tablespace ts_S_019 pagesize 4K managed by database using ( device
'/dev/tpcc/S_019' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_020;
create regular tablespace ts_S_020 pagesize 4K managed by database using ( device
'/dev/tpcc/S_020' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_021;
create regular tablespace ts_S_021 pagesize 4K managed by database using ( device
'/dev/tpcc/S_021' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_022;
create regular tablespace ts_S_022 pagesize 4K managed by database using ( device
'/dev/tpcc/S_022' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_023;
create regular tablespace ts_S_023 pagesize 4K managed by database using ( device
'/dev/tpcc/S_023' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_024;
create regular tablespace ts_S_024 pagesize 4K managed by database using ( device
'/dev/tpcc/S_024' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_025;
create regular tablespace ts_S_025 pagesize 4K managed by database using ( device
'/dev/tpcc/S_025' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_026;
create regular tablespace ts_S_026 pagesize 4K managed by database using ( device
'/dev/tpcc/S_026' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_027;
create regular tablespace ts_S_027 pagesize 4K managed by database using ( device
'/dev/tpcc/S_027' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_028;
create regular tablespace ts_S_028 pagesize 4K managed by database using ( device
'/dev/tpcc/S_028' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_029;
create regular tablespace ts_S_029 pagesize 4K managed by database using ( device
'/dev/tpcc/S_029' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_030;
create regular tablespace ts_S_030 pagesize 4K managed by database using ( device
'/dev/tpcc/S_030' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_031;
create regular tablespace ts_S_031 pagesize 4K managed by database using ( device
'/dev/tpcc/S_031' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_032;
create regular tablespace ts_S_032 pagesize 4K managed by database using ( device
'/dev/tpcc/S_032' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_033;
create regular tablespace ts_S_033 pagesize 4K managed by database using ( device
'/dev/tpcc/S_033' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_034;
create regular tablespace ts_S_034 pagesize 4K managed by database using ( device
'/dev/tpcc/S_034' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_035;
create regular tablespace ts_S_035 pagesize 4K managed by database using ( device
'/dev/tpcc/S_035' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_036;
create regular tablespace ts_S_036 pagesize 4K managed by database using ( device
'/dev/tpcc/S_036' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_037;
create regular tablespace ts_S_037 pagesize 4K managed by database using ( device
'/dev/tpcc/S_037' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_038;
create regular tablespace ts_S_038 pagesize 4K managed by database using ( device
'/dev/tpcc/S_038' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_039;
create regular tablespace ts_S_039 pagesize 4K managed by database using ( device
'/dev/tpcc/S_039' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_040;
create regular tablespace ts_S_040 pagesize 4K managed by database using ( device
'/dev/tpcc/S_040' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_041;
create regular tablespace ts_S_041 pagesize 4K managed by database using ( device
'/dev/tpcc/S_041' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_042;
create regular tablespace ts_S_042 pagesize 4K managed by database using ( device
'/dev/tpcc/S_042' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_043;
create regular tablespace ts_S_043 pagesize 4K managed by database using ( device
'/dev/tpcc/S_043' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_044;
create regular tablespace ts_S_044 pagesize 4K managed by database using ( device
'/dev/tpcc/S_044' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_045;
```

```
create regular tablespace ts_S_045 pagesize 4K managed by database using ( device
'/dev/tpcc/S_045' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_046;
create regular tablespace ts_S_046 pagesize 4K managed by database using ( device
'/dev/tpcc/S_046' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_047;
create regular tablespace ts_S_047 pagesize 4K managed by database using ( device
'/dev/tpcc/S_047' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_048;
create regular tablespace ts_S_048 pagesize 4K managed by database using ( device
'/dev/tpcc/S_048' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_049;
create regular tablespace ts_S_049 pagesize 4K managed by database using ( device
'/dev/tpcc/S_049' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_050;
create regular tablespace ts_S_050 pagesize 4K managed by database using ( device
'/dev/tpcc/S_050' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_051;
create regular tablespace ts_S_051 pagesize 4K managed by database using ( device
'/dev/tpcc/S_051' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_052;
create regular tablespace ts_S_052 pagesize 4K managed by database using ( device
'/dev/tpcc/S_052' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_053;
create regular tablespace ts_S_053 pagesize 4K managed by database using ( device
'/dev/tpcc/S_053' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_054;
create regular tablespace ts_S_054 pagesize 4K managed by database using ( device
'/dev/tpcc/S_054' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_055;
create regular tablespace ts_S_055 pagesize 4K managed by database using ( device
'/dev/tpcc/S_055' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_056;
create regular tablespace ts_S_056 pagesize 4K managed by database using ( device
'/dev/tpcc/S_056' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_057;
create regular tablespace ts_S_057 pagesize 4K managed by database using ( device
'/dev/tpcc/S_057' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_058;
create regular tablespace ts_S_058 pagesize 4K managed by database using ( device
'/dev/tpcc/S_058' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_059;
create regular tablespace ts_S_059 pagesize 4K managed by database using ( device
'/dev/tpcc/S_059' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_060;
create regular tablespace ts_S_060 pagesize 4K managed by database using ( device
'/dev/tpcc/S_060' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_061;
create regular tablespace ts_S_061 pagesize 4K managed by database using ( device
'/dev/tpcc/S_061' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_062;
create regular tablespace ts_S_062 pagesize 4K managed by database using ( device
'/dev/tpcc/S_062' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_063;
create regular tablespace ts_S_063 pagesize 4K managed by database using ( device
'/dev/tpcc/S_063' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_064;
create regular tablespace ts_S_064 pagesize 4K managed by database using ( device
'/dev/tpcc/S_064' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_065;
create regular tablespace ts_S_065 pagesize 4K managed by database using ( device
'/dev/tpcc/S_065' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_066;
create regular tablespace ts_S_066 pagesize 4K managed by database using ( device
'/dev/tpcc/S_066' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_067;
create regular tablespace ts_S_067 pagesize 4K managed by database using ( device
'/dev/tpcc/S_067' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_068;
create regular tablespace ts_S_068 pagesize 4K managed by database using ( device
'/dev/tpcc/S_068' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_069;
create regular tablespace ts_S_069 pagesize 4K managed by database using ( device
'/dev/tpcc/S_069' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_070;
create regular tablespace ts_S_070 pagesize 4K managed by database using ( device
'/dev/tpcc/S_070' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_071;
create regular tablespace ts_S_071 pagesize 4K managed by database using ( device
'/dev/tpcc/S_071' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_072;
create regular tablespace ts_S_072 pagesize 4K managed by database using ( device
'/dev/tpcc/S_072' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_073;
create regular tablespace ts_S_073 pagesize 4K managed by database using ( device
'/dev/tpcc/S_073' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_074;
create regular tablespace ts_S_074 pagesize 4K managed by database using ( device
'/dev/tpcc/S_074' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_075;
create regular tablespace ts_S_075 pagesize 4K managed by database using ( device
'/dev/tpcc/S_075' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_076;
create regular tablespace ts_S_076 pagesize 4K managed by database using ( device
'/dev/tpcc/S_076' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_077;
create regular tablespace ts_S_077 pagesize 4K managed by database using ( device
'/dev/tpcc/S_077' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_078;
create regular tablespace ts_S_078 pagesize 4K managed by database using ( device
'/dev/tpcc/S_078' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_079;
create regular tablespace ts_S_079 pagesize 4K managed by database using ( device
'/dev/tpcc/S_079' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_080;
create regular tablespace ts_S_080 pagesize 4K managed by database using ( device
'/dev/tpcc/S_080' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_081;
create regular tablespace ts_S_081 pagesize 4K managed by database using ( device
'/dev/tpcc/S_081' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_082;
create regular tablespace ts_S_082 pagesize 4K managed by database using ( device
'/dev/tpcc/S_082' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_083;
```

```
create regular tablespace ts_S_083 pagesize 4K managed by database using ( device
'/dev/tpcc/S_083' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_084;
create regular tablespace ts_S_084 pagesize 4K managed by database using ( device
'/dev/tpcc/S_084' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_085;
create regular tablespace ts_S_085 pagesize 4K managed by database using ( device
'/dev/tpcc/S_085' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_086;
create regular tablespace ts_S_086 pagesize 4K managed by database using ( device
'/dev/tpcc/S_086' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_087;
create regular tablespace ts_S_087 pagesize 4K managed by database using ( device
'/dev/tpcc/S_087' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_088;
create regular tablespace ts_S_088 pagesize 4K managed by database using ( device
'/dev/tpcc/S_088' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_089;
create regular tablespace ts_S_089 pagesize 4K managed by database using ( device
'/dev/tpcc/S_089' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_090;
create regular tablespace ts_S_090 pagesize 4K managed by database using ( device
'/dev/tpcc/S_090' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_091;
create regular tablespace ts_S_091 pagesize 4K managed by database using ( device
'/dev/tpcc/S_091' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_092;
create regular tablespace ts_S_092 pagesize 4K managed by database using ( device
'/dev/tpcc/S_092' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_093;
create regular tablespace ts_S_093 pagesize 4K managed by database using ( device
'/dev/tpcc/S_093' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_094;
create regular tablespace ts_S_094 pagesize 4K managed by database using ( device
'/dev/tpcc/S_094' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_095;
create regular tablespace ts_S_095 pagesize 4K managed by database using ( device
'/dev/tpcc/S_095' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_096;
create regular tablespace ts_S_096 pagesize 4K managed by database using ( device
'/dev/tpcc/S_096' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_097;
create regular tablespace ts_S_097 pagesize 4K managed by database using ( device
'/dev/tpcc/S_097' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_098;
create regular tablespace ts_S_098 pagesize 4K managed by database using ( device
'/dev/tpcc/S_098' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_099;
create regular tablespace ts_S_099 pagesize 4K managed by database using ( device
'/dev/tpcc/S_099' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_100;
create regular tablespace ts_S_100 pagesize 4K managed by database using ( device
'/dev/tpcc/S_100' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_101;
create regular tablespace ts_S_101 pagesize 4K managed by database using ( device
'/dev/tpcc/S_101' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_102;
create regular tablespace ts_S_102 pagesize 4K managed by database using ( device
'/dev/tpcc/S_102' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_103;
create regular tablespace ts_S_103 pagesize 4K managed by database using ( device
'/dev/tpcc/S_103' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_104;
create regular tablespace ts_S_104 pagesize 4K managed by database using ( device
'/dev/tpcc/S_104' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_105;
create regular tablespace ts_S_105 pagesize 4K managed by database using ( device
'/dev/tpcc/S_105' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_106;
create regular tablespace ts_S_106 pagesize 4K managed by database using ( device
'/dev/tpcc/S_106' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_107;
create regular tablespace ts_S_107 pagesize 4K managed by database using ( device
'/dev/tpcc/S_107' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_108;
create regular tablespace ts_S_108 pagesize 4K managed by database using ( device
'/dev/tpcc/S_108' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_109;
create regular tablespace ts_S_109 pagesize 4K managed by database using ( device
'/dev/tpcc/S_109' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_110;
create regular tablespace ts_S_110 pagesize 4K managed by database using ( device
'/dev/tpcc/S_110' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_111;
create regular tablespace ts_S_111 pagesize 4K managed by database using ( device
'/dev/tpcc/S_111' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_112;
create regular tablespace ts_S_112 pagesize 4K managed by database using ( device
'/dev/tpcc/S_112' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_113;
create regular tablespace ts_S_113 pagesize 4K managed by database using ( device
'/dev/tpcc/S_113' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_114;
create regular tablespace ts_S_114 pagesize 4K managed by database using ( device
'/dev/tpcc/S_114' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_115;
create regular tablespace ts_S_115 pagesize 4K managed by database using ( device
'/dev/tpcc/S_115' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_116;
create regular tablespace ts_S_116 pagesize 4K managed by database using ( device
'/dev/tpcc/S_116' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_117;
create regular tablespace ts_S_117 pagesize 4K managed by database using ( device
'/dev/tpcc/S_117' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_118;
create regular tablespace ts_S_118 pagesize 4K managed by database using ( device
'/dev/tpcc/S_118' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_119;
create regular tablespace ts_S_119 pagesize 4K managed by database using ( device
'/dev/tpcc/S_119' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_120;
create regular tablespace ts_S_120 pagesize 4K managed by database using ( device
'/dev/tpcc/S_120' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_121;
```

```
create regular tablespace ts_S_121 pagesize 4K managed by database using ( device
'/dev/tpcc/S_121' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_122;
create regular tablespace ts_S_122 pagesize 4K managed by database using ( device
'/dev/tpcc/S_122' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_123;
create regular tablespace ts_S_123 pagesize 4K managed by database using ( device
'/dev/tpcc/S_123' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_124;
create regular tablespace ts_S_124 pagesize 4K managed by database using ( device
'/dev/tpcc/S_124' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_125;
create regular tablespace ts_S_125 pagesize 4K managed by database using ( device
'/dev/tpcc/S_125' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_126;
create regular tablespace ts_S_126 pagesize 4K managed by database using ( device
'/dev/tpcc/S_126' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_127;
create regular tablespace ts_S_127 pagesize 4K managed by database using ( device
'/dev/tpcc/S_127' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_128;
create regular tablespace ts_S_128 pagesize 4K managed by database using ( device
'/dev/tpcc/S_128' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_129;
create regular tablespace ts_S_129 pagesize 4K managed by database using ( device
'/dev/tpcc/S_129' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_130;
create regular tablespace ts_S_130 pagesize 4K managed by database using ( device
'/dev/tpcc/S_130' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_131;
create regular tablespace ts_S_131 pagesize 4K managed by database using ( device
'/dev/tpcc/S_131' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_132;
create regular tablespace ts_S_132 pagesize 4K managed by database using ( device
'/dev/tpcc/S_132' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_133;
create regular tablespace ts_S_133 pagesize 4K managed by database using ( device
'/dev/tpcc/S_133' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_134;
create regular tablespace ts_S_134 pagesize 4K managed by database using ( device
'/dev/tpcc/S_134' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_135;
create regular tablespace ts_S_135 pagesize 4K managed by database using ( device
'/dev/tpcc/S_135' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_136;
create regular tablespace ts_S_136 pagesize 4K managed by database using ( device
'/dev/tpcc/S_136' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_137;
create regular tablespace ts_S_137 pagesize 4K managed by database using ( device
'/dev/tpcc/S_137' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_138;
create regular tablespace ts_S_138 pagesize 4K managed by database using ( device
'/dev/tpcc/S_138' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_139;
create regular tablespace ts_S_139 pagesize 4K managed by database using ( device
'/dev/tpcc/S_139' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_140;
create regular tablespace ts_S_140 pagesize 4K managed by database using ( device
'/dev/tpcc/S_140' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_141;
create regular tablespace ts_S_141 pagesize 4K managed by database using ( device
'/dev/tpcc/S_141' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_142;
create regular tablespace ts_S_142 pagesize 4K managed by database using ( device
'/dev/tpcc/S_142' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_143;
create regular tablespace ts_S_143 pagesize 4K managed by database using ( device
'/dev/tpcc/S_143' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_S_144;
create regular tablespace ts_S_144 pagesize 4K managed by database using ( device
'/dev/tpcc/S_144' 12404736 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize
4096;
commit;
drop tablespace ts_C_001,is_C2_001;
create regular tablespace ts_C_001 pagesize 4K managed by database using ( device
'/dev/tpcc/C_001' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_001 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_001' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_002,is_C2_002;
create regular tablespace ts_C_002 pagesize 4K managed by database using ( device
'/dev/tpcc/C_002' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_002 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_002' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_003,is_C2_003;
create regular tablespace ts_C_003 pagesize 4K managed by database using ( device
'/dev/tpcc/C_003' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_003 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_003' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_004,is_C2_004;
create regular tablespace ts_C_004 pagesize 4K managed by database using ( device
'/dev/tpcc/C_004' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_004 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_004' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_005,is_C2_005;
create regular tablespace ts_C_005 pagesize 4K managed by database using ( device
'/dev/tpcc/C_005' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_005 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_005' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_006,is_C2_006;
create regular tablespace ts_C_006 pagesize 4K managed by database using ( device
'/dev/tpcc/C_006' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_006 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_006' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_007,is_C2_007;
create regular tablespace ts_C_007 pagesize 4K managed by database using ( device
'/dev/tpcc/C_007' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_007 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_007' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_008,is_C2_008;
create regular tablespace ts_C_008 pagesize 4K managed by database using ( device
'/dev/tpcc/C_008' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_008 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_008' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_009,is_C2_009;
create regular tablespace ts_C_009 pagesize 4K managed by database using ( device
'/dev/tpcc/C_009' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_009 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_009' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_010,is_C2_010;
create regular tablespace ts_C_010 pagesize 4K managed by database using ( device
'/dev/tpcc/C_010' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_010 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_010' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_011,is_C2_011;
create regular tablespace ts_C_011 pagesize 4K managed by database using ( device
'/dev/tpcc/C_011' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
```

```
commit;
create regular tablespace is_C2_011 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_011' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_012,is_C2_012;
create regular tablespace ts_C_012 pagesize 4K managed by database using ( device
'/dev/tpcc/C_012' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_012 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_012' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_013,is_C2_013;
create regular tablespace ts_C_013 pagesize 4K managed by database using ( device
'/dev/tpcc/C_013' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_013 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_013' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_014,is_C2_014;
create regular tablespace ts_C_014 pagesize 4K managed by database using ( device
'/dev/tpcc/C_014' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_014 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_014' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_015,is_C2_015;
create regular tablespace ts_C_015 pagesize 4K managed by database using ( device
'/dev/tpcc/C_015' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_015 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_015' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_016,is_C2_016;
create regular tablespace ts_C_016 pagesize 4K managed by database using ( device
'/dev/tpcc/C_016' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_016 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_016' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_017,is_C2_017;
create regular tablespace ts_C_017 pagesize 4K managed by database using ( device
'/dev/tpcc/C_017' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_017 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_017' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_018,is_C2_018;
create regular tablespace ts_C_018 pagesize 4K managed by database using ( device
'/dev/tpcc/C_018' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_018 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_018' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_019,is_C2_019;
create regular tablespace ts_C_019 pagesize 4K managed by database using ( device
'/dev/tpcc/C_019' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_019 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_019' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_020,is_C2_020;
create regular tablespace ts_C_020 pagesize 4K managed by database using ( device
'/dev/tpcc/C_020' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_020 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_020' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_021,is_C2_021;
create regular tablespace ts_C_021 pagesize 4K managed by database using ( device
'/dev/tpcc/C_021' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_021 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_021' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_022,is_C2_022;
create regular tablespace ts_C_022 pagesize 4K managed by database using ( device
'/dev/tpcc/C_022' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_022 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_022' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_023,is_C2_023;
create regular tablespace ts_C_023 pagesize 4K managed by database using ( device
'/dev/tpcc/C_023' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_023 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_023' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_024,is_C2_024;
create regular tablespace ts_C_024 pagesize 4K managed by database using ( device
'/dev/tpcc/C_024' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_024 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_024' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_025,is_C2_025;
create regular tablespace ts_C_025 pagesize 4K managed by database using ( device
'/dev/tpcc/C_025' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_025 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_025' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_026,is_C2_026;
create regular tablespace ts_C_026 pagesize 4K managed by database using ( device
'/dev/tpcc/C_026' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_026 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_026' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_027,is_C2_027;
create regular tablespace ts_C_027 pagesize 4K managed by database using ( device
'/dev/tpcc/C_027' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_027 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_027' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_028,is_C2_028;
create regular tablespace ts_C_028 pagesize 4K managed by database using ( device
'/dev/tpcc/C_028' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_028 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_028' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_029,is_C2_029;
create regular tablespace ts_C_029 pagesize 4K managed by database using ( device
'/dev/tpcc/C_029' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_029 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_029' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_030,is_C2_030;
create regular tablespace ts_C_030 pagesize 4K managed by database using ( device
'/dev/tpcc/C_030' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_030 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_030' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_031,is_C2_031;
create regular tablespace ts_C_031 pagesize 4K managed by database using ( device
'/dev/tpcc/C_031' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_031 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_031' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_032,is_C2_032;
create regular tablespace ts_C_032 pagesize 4K managed by database using ( device
'/dev/tpcc/C_032' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_032 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_032' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_033,is_C2_033;
create regular tablespace ts_C_033 pagesize 4K managed by database using ( device
'/dev/tpcc/C_033' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_033 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_033' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_034,is_C2_034;
create regular tablespace ts_C_034 pagesize 4K managed by database using ( device
'/dev/tpcc/C_034' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_034 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_034' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_035,is_C2_035;
create regular tablespace ts_C_035 pagesize 4K managed by database using ( device
'/dev/tpcc/C_035' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_035 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_035' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_036,is_C2_036;
create regular tablespace ts_C_036 pagesize 4K managed by database using ( device
'/dev/tpcc/C_036' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_036 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_036' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_037,is_C2_037;
create regular tablespace ts_C_037 pagesize 4K managed by database using ( device
'/dev/tpcc/C_037' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_037 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_037' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_038,is_C2_038;
create regular tablespace ts_C_038 pagesize 4K managed by database using ( device
'/dev/tpcc/C_038' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_038 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_038' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
```

```
drop tablespace ts_C_039,is_C2_039;
create regular tablespace ts_C_039 pagesize 4K managed by database using ( device
'/dev/tpcc/C_039' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_039 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_039' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_040,is_C2_040;
create regular tablespace ts_C_040 pagesize 4K managed by database using ( device
'/dev/tpcc/C_040' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_040 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_040' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_041,is_C2_041;
create regular tablespace ts_C_041 pagesize 4K managed by database using ( device
'/dev/tpcc/C_041' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_041 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_041' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_042,is_C2_042;
create regular tablespace ts_C_042 pagesize 4K managed by database using ( device
'/dev/tpcc/C_042' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_042 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_042' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_043,is_C2_043;
create regular tablespace ts_C_043 pagesize 4K managed by database using ( device
'/dev/tpcc/C_043' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_043 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_043' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_044,is_C2_044;
create regular tablespace ts_C_044 pagesize 4K managed by database using ( device
'/dev/tpcc/C_044' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_044 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_044' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_045,is_C2_045;
create regular tablespace ts_C_045 pagesize 4K managed by database using ( device
'/dev/tpcc/C_045' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_045 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_045' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_046,is_C2_046;
create regular tablespace ts_C_046 pagesize 4K managed by database using ( device
'/dev/tpcc/C_046' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_046 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_046' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_047,is_C2_047;
create regular tablespace ts_C_047 pagesize 4K managed by database using ( device
'/dev/tpcc/C_047' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_047 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_047' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_048,is_C2_048;
create regular tablespace ts_C_048 pagesize 4K managed by database using ( device
'/dev/tpcc/C_048' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_048 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_048' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_049,is_C2_049;
create regular tablespace ts_C_049 pagesize 4K managed by database using ( device
'/dev/tpcc/C_049' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_049 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_049' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_050,is_C2_050;
create regular tablespace ts_C_050 pagesize 4K managed by database using ( device
'/dev/tpcc/C_050' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_050 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_050' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_051,is_C2_051;
create regular tablespace ts_C_051 pagesize 4K managed by database using ( device
'/dev/tpcc/C_051' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_051 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_051' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_052,is_C2_052;
create regular tablespace ts_C_052 pagesize 4K managed by database using ( device
'/dev/tpcc/C_052' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_052 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_052' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_053,is_C2_053;
create regular tablespace ts_C_053 pagesize 4K managed by database using ( device
'/dev/tpcc/C_053' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_053 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_053' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_054,is_C2_054;
create regular tablespace ts_C_054 pagesize 4K managed by database using ( device
'/dev/tpcc/C_054' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_054 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_054' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_055,is_C2_055;
create regular tablespace ts_C_055 pagesize 4K managed by database using ( device
'/dev/tpcc/C_055' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_055 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_055' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_056,is_C2_056;
create regular tablespace ts_C_056 pagesize 4K managed by database using ( device
'/dev/tpcc/C_056' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_056 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_056' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_057,is_C2_057;
create regular tablespace ts_C_057 pagesize 4K managed by database using ( device
'/dev/tpcc/C_057' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_057 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_057' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_058,is_C2_058;
create regular tablespace ts_C_058 pagesize 4K managed by database using ( device
'/dev/tpcc/C_058' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_058 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_058' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_059,is_C2_059;
create regular tablespace ts_C_059 pagesize 4K managed by database using ( device
'/dev/tpcc/C_059' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_059 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_059' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_060,is_C2_060;
create regular tablespace ts_C_060 pagesize 4K managed by database using ( device
'/dev/tpcc/C_060' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_060 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_060' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_061,is_C2_061;
create regular tablespace ts_C_061 pagesize 4K managed by database using ( device
'/dev/tpcc/C_061' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_061 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_061' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_062,is_C2_062;
create regular tablespace ts_C_062 pagesize 4K managed by database using ( device
'/dev/tpcc/C_062' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_062 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_062' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_063,is_C2_063;
create regular tablespace ts_C_063 pagesize 4K managed by database using ( device
'/dev/tpcc/C_063' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_063 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_063' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_064,is_C2_064;
create regular tablespace ts_C_064 pagesize 4K managed by database using ( device
'/dev/tpcc/C_064' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_064 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_064' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_065,is_C2_065;
create regular tablespace ts_C_065 pagesize 4K managed by database using ( device
'/dev/tpcc/C_065' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_065 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_065' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_066,is_C2_066;
create regular tablespace ts_C_066 pagesize 4K managed by database using ( device
'/dev/tpcc/C_066' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
```

```
create regular tablespace is_C2_066 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_066' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_067,is_C2_067;
create regular tablespace ts_C_067 pagesize 4K managed by database using ( device
'/dev/tpcc/C_067' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_067 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_067' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_068,is_C2_068;
create regular tablespace ts_C_068 pagesize 4K managed by database using ( device
'/dev/tpcc/C_068' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_068 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_068' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_069,is_C2_069;
create regular tablespace ts_C_069 pagesize 4K managed by database using ( device
'/dev/tpcc/C_069' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_069 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_069' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_070,is_C2_070;
create regular tablespace ts_C_070 pagesize 4K managed by database using ( device
'/dev/tpcc/C_070' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_070 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_070' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_071,is_C2_071;
create regular tablespace ts_C_071 pagesize 4K managed by database using ( device
'/dev/tpcc/C_071' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_071 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_071' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_072,is_C2_072;
create regular tablespace ts_C_072 pagesize 4K managed by database using ( device
'/dev/tpcc/C_072' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_072 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_072' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_073,is_C2_073;
create regular tablespace ts_C_073 pagesize 4K managed by database using ( device
'/dev/tpcc/C_073' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_073 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_073' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_074,is_C2_074;
create regular tablespace ts_C_074 pagesize 4K managed by database using ( device
'/dev/tpcc/C_074' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_074 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_074' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_075,is_C2_075;
create regular tablespace ts_C_075 pagesize 4K managed by database using ( device
'/dev/tpcc/C_075' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_075 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_075' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_076,is_C2_076;
create regular tablespace ts_C_076 pagesize 4K managed by database using ( device
'/dev/tpcc/C_076' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_076 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_076' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_077,is_C2_077;
create regular tablespace ts_C_077 pagesize 4K managed by database using ( device
'/dev/tpcc/C_077' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_077 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_077' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_078,is_C2_078;
create regular tablespace ts_C_078 pagesize 4K managed by database using ( device
'/dev/tpcc/C_078' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_078 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_078' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_079,is_C2_079;
create regular tablespace ts_C_079 pagesize 4K managed by database using ( device
'/dev/tpcc/C_079' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_079 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_079' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_080,is_C2_080;
create regular tablespace ts_C_080 pagesize 4K managed by database using ( device
'/dev/tpcc/C_080' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_080 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_080' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_081,is_C2_081;
create regular tablespace ts_C_081 pagesize 4K managed by database using ( device
'/dev/tpcc/C_081' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_081 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_081' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_082,is_C2_082;
create regular tablespace ts_C_082 pagesize 4K managed by database using ( device
'/dev/tpcc/C_082' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_082 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_082' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_083,is_C2_083;
create regular tablespace ts_C_083 pagesize 4K managed by database using ( device
'/dev/tpcc/C_083' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_083 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_083' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_084,is_C2_084;
create regular tablespace ts_C_084 pagesize 4K managed by database using ( device
'/dev/tpcc/C_084' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_084 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_084' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_085,is_C2_085;
create regular tablespace ts_C_085 pagesize 4K managed by database using ( device
'/dev/tpcc/C_085' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_085 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_085' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_086,is_C2_086;
create regular tablespace ts_C_086 pagesize 4K managed by database using ( device
'/dev/tpcc/C_086' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_086 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_086' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_087,is_C2_087;
create regular tablespace ts_C_087 pagesize 4K managed by database using ( device
'/dev/tpcc/C_087' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_087 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_087' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_088,is_C2_088;
create regular tablespace ts_C_088 pagesize 4K managed by database using ( device
'/dev/tpcc/C_088' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_088 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_088' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_089,is_C2_089;
create regular tablespace ts_C_089 pagesize 4K managed by database using ( device
'/dev/tpcc/C_089' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_089 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_089' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_090,is_C2_090;
create regular tablespace ts_C_090 pagesize 4K managed by database using ( device
'/dev/tpcc/C_090' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_090 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_090' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_091,is_C2_091;
create regular tablespace ts_C_091 pagesize 4K managed by database using ( device
'/dev/tpcc/C_091' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_091 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_091' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_092,is_C2_092;
create regular tablespace ts_C_092 pagesize 4K managed by database using ( device
'/dev/tpcc/C_092' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_092 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_092' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_093,is_C2_093;
create regular tablespace ts_C_093 pagesize 4K managed by database using ( device
'/dev/tpcc/C_093' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_093 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_093' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_094,is_C2_094;
```

```
create regular tablespace ts_C_094 pagesize 4K managed by database using ( device
'/dev/tpcc/C_094' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_094 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_094' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_095,is_C2_095;
create regular tablespace ts_C_095 pagesize 4K managed by database using ( device
'/dev/tpcc/C_095' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_095 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_095' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_096,is_C2_096;
create regular tablespace ts_C_096 pagesize 4K managed by database using ( device
'/dev/tpcc/C_096' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_096 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_096' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_097,is_C2_097;
create regular tablespace ts_C_097 pagesize 4K managed by database using ( device
'/dev/tpcc/C_097' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_097 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_097' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_098,is_C2_098;
create regular tablespace ts_C_098 pagesize 4K managed by database using ( device
'/dev/tpcc/C_098' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_098 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_098' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_099,is_C2_099;
create regular tablespace ts_C_099 pagesize 4K managed by database using ( device
'/dev/tpcc/C_099' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_099 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_099' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_100,is_C2_100;
create regular tablespace ts_C_100 pagesize 4K managed by database using ( device
'/dev/tpcc/C_100' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_100 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_100' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_101,is_C2_101;
create regular tablespace ts_C_101 pagesize 4K managed by database using ( device
'/dev/tpcc/C_101' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_101 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_101' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_102,is_C2_102;
create regular tablespace ts_C_102 pagesize 4K managed by database using ( device
'/dev/tpcc/C_102' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_102 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_102' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_103,is_C2_103;
create regular tablespace ts_C_103 pagesize 4K managed by database using ( device
'/dev/tpcc/C_103' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_103 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_103' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_104,is_C2_104;
create regular tablespace ts_C_104 pagesize 4K managed by database using ( device
'/dev/tpcc/C_104' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_104 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_104' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_105,is_C2_105;
create regular tablespace ts_C_105 pagesize 4K managed by database using ( device
'/dev/tpcc/C_105' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_105 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_105' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_106,is_C2_106;
create regular tablespace ts_C_106 pagesize 4K managed by database using ( device
'/dev/tpcc/C_106' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_106 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_106' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_107,is_C2_107;
create regular tablespace ts_C_107 pagesize 4K managed by database using ( device
'/dev/tpcc/C_107' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_107 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_107' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_108,is_C2_108;
create regular tablespace ts_C_108 pagesize 4K managed by database using ( device
'/dev/tpcc/C_108' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_108 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_108' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_109,is_C2_109;
create regular tablespace ts_C_109 pagesize 4K managed by database using ( device
'/dev/tpcc/C_109' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_109 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_109' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_110,is_C2_110;
create regular tablespace ts_C_110 pagesize 4K managed by database using ( device
'/dev/tpcc/C_110' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_110 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_110' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_111,is_C2_111;
create regular tablespace ts_C_111 pagesize 4K managed by database using ( device
'/dev/tpcc/C_111' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_111 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_111' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_112,is_C2_112;
create regular tablespace ts_C_112 pagesize 4K managed by database using ( device
'/dev/tpcc/C_112' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_112 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_112' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_113,is_C2_113;
create regular tablespace ts_C_113 pagesize 4K managed by database using ( device
'/dev/tpcc/C_113' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_113 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_113' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_114,is_C2_114;
create regular tablespace ts_C_114 pagesize 4K managed by database using ( device
'/dev/tpcc/C_114' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_114 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_114' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_115,is_C2_115;
create regular tablespace ts_C_115 pagesize 4K managed by database using ( device
'/dev/tpcc/C_115' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_115 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_115' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_116,is_C2_116;
create regular tablespace ts_C_116 pagesize 4K managed by database using ( device
'/dev/tpcc/C_116' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_116 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_116' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_117,is_C2_117;
create regular tablespace ts_C_117 pagesize 4K managed by database using ( device
'/dev/tpcc/C_117' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_117 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_117' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_118,is_C2_118;
create regular tablespace ts_C_118 pagesize 4K managed by database using ( device
'/dev/tpcc/C_118' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_118 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_118' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_119,is_C2_119;
create regular tablespace ts_C_119 pagesize 4K managed by database using ( device
'/dev/tpcc/C_119' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_119 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_119' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_120,is_C2_120;
create regular tablespace ts_C_120 pagesize 4K managed by database using ( device
'/dev/tpcc/C_120' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_120 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_120' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_121,is_C2_121;
create regular tablespace ts_C_121 pagesize 4K managed by database using ( device
'/dev/tpcc/C_121' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_121 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_121' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
```

```
commit;
drop tablespace ts_C_122,is_C2_122;
create regular tablespace ts_C_122 pagesize 4K managed by database using ( device
'/dev/tpcc/C_122' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_122 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_122' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_123,is_C2_123;
create regular tablespace ts_C_123 pagesize 4K managed by database using ( device
'/dev/tpcc/C_123' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_123 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_123' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_124,is_C2_124;
create regular tablespace ts_C_124 pagesize 4K managed by database using ( device
'/dev/tpcc/C_124' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_124 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_124' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_125,is_C2_125;
create regular tablespace ts_C_125 pagesize 4K managed by database using ( device
'/dev/tpcc/C_125' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_125 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_125' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_126,is_C2_126;
create regular tablespace ts_C_126 pagesize 4K managed by database using ( device
'/dev/tpcc/C_126' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_126 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_126' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_127,is_C2_127;
create regular tablespace ts_C_127 pagesize 4K managed by database using ( device
'/dev/tpcc/C_127' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_127 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_127' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_128,is_C2_128;
create regular tablespace ts_C_128 pagesize 4K managed by database using ( device
'/dev/tpcc/C_128' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_128 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_128' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_129,is_C2_129;
create regular tablespace ts_C_129 pagesize 4K managed by database using ( device
'/dev/tpcc/C_129' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_129 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_129' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_130,is_C2_130;
create regular tablespace ts_C_130 pagesize 4K managed by database using ( device
'/dev/tpcc/C_130' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_130 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_130' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_131,is_C2_131;
create regular tablespace ts_C_131 pagesize 4K managed by database using ( device
'/dev/tpcc/C_131' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_131 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_131' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_132,is_C2_132;
create regular tablespace ts_C_132 pagesize 4K managed by database using ( device
'/dev/tpcc/C_132' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_132 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_132' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_133,is_C2_133;
create regular tablespace ts_C_133 pagesize 4K managed by database using ( device
'/dev/tpcc/C_133' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_133 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_133' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_134,is_C2_134;
create regular tablespace ts_C_134 pagesize 4K managed by database using ( device
'/dev/tpcc/C_134' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_134 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_134' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_135,is_C2_135;
create regular tablespace ts_C_135 pagesize 4K managed by database using ( device
'/dev/tpcc/C_135' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_135 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_135' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_136,is_C2_136;
create regular tablespace ts_C_136 pagesize 4K managed by database using ( device
'/dev/tpcc/C_136' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_136 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_136' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_137,is_C2_137;
create regular tablespace ts_C_137 pagesize 4K managed by database using ( device
'/dev/tpcc/C_137' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_137 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_137' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_138,is_C2_138;
create regular tablespace ts_C_138 pagesize 4K managed by database using ( device
'/dev/tpcc/C_138' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_138 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_138' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_139,is_C2_139;
create regular tablespace ts_C_139 pagesize 4K managed by database using ( device
'/dev/tpcc/C_139' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_139 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_139' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_140,is_C2_140;
create regular tablespace ts_C_140 pagesize 4K managed by database using ( device
'/dev/tpcc/C_140' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_140 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_140' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_141,is_C2_141;
create regular tablespace ts_C_141 pagesize 4K managed by database using ( device
'/dev/tpcc/C_141' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_141 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_141' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_142,is_C2_142;
create regular tablespace ts_C_142 pagesize 4K managed by database using ( device
'/dev/tpcc/C_142' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_142 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_142' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_143,is_C2_143;
create regular tablespace ts_C_143 pagesize 4K managed by database using ( device
'/dev/tpcc/C_143' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_143 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_143' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_C_144,is_C2_144;
create regular tablespace ts_C_144 pagesize 4K managed by database using ( device
'/dev/tpcc/C_144' 8939520 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
create regular tablespace is_C2_144 pagesize 8K managed by database using ( device
'/dev/tpcc/C2_144' 313344 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_N_001;
create regular tablespace ts_N_001 pagesize 4K managed by database using ( device
'/dev/tpcc/N_001' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_002;
create regular tablespace ts_N_002 pagesize 4K managed by database using ( device
'/dev/tpcc/N_002' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_003;
create regular tablespace ts_N_003 pagesize 4K managed by database using ( device
'/dev/tpcc/N_003' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_004;
create regular tablespace ts_N_004 pagesize 4K managed by database using ( device
'/dev/tpcc/N_004' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_005;
create regular tablespace ts_N_005 pagesize 4K managed by database using ( device
'/dev/tpcc/N_005' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_006;
create regular tablespace ts_N_006 pagesize 4K managed by database using ( device
'/dev/tpcc/N_006' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_007;
create regular tablespace ts_N_007 pagesize 4K managed by database using ( device
'/dev/tpcc/N_007' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_008;
create regular tablespace ts_N_008 pagesize 4K managed by database using ( device
'/dev/tpcc/N_008' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
```

```
commit;
drop tablespace ts_N_009;
create regular tablespace ts_N_009 pagesize 4K managed by database using ( device
'/dev/tpcc/N_009' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_010;
create regular tablespace ts_N_010 pagesize 4K managed by database using ( device
'/dev/tpcc/N_010' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_011;
create regular tablespace ts_N_011 pagesize 4K managed by database using ( device
'/dev/tpcc/N_011' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_012;
create regular tablespace ts_N_012 pagesize 4K managed by database using ( device
'/dev/tpcc/N_012' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_013;
create regular tablespace ts_N_013 pagesize 4K managed by database using ( device
'/dev/tpcc/N_013' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_014;
create regular tablespace ts_N_014 pagesize 4K managed by database using ( device
'/dev/tpcc/N_014' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_015;
create regular tablespace ts_N_015 pagesize 4K managed by database using ( device
'/dev/tpcc/N_015' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_016;
create regular tablespace ts_N_016 pagesize 4K managed by database using ( device
'/dev/tpcc/N_016' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_017;
create regular tablespace ts_N_017 pagesize 4K managed by database using ( device
'/dev/tpcc/N_017' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_018;
create regular tablespace ts_N_018 pagesize 4K managed by database using ( device
'/dev/tpcc/N_018' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_019;
create regular tablespace ts_N_019 pagesize 4K managed by database using ( device
'/dev/tpcc/N_019' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_020;
create regular tablespace ts_N_020 pagesize 4K managed by database using ( device
'/dev/tpcc/N_020' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_021;
create regular tablespace ts_N_021 pagesize 4K managed by database using ( device
'/dev/tpcc/N_021' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_022;
create regular tablespace ts_N_022 pagesize 4K managed by database using ( device
'/dev/tpcc/N_022' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_023;
create regular tablespace ts_N_023 pagesize 4K managed by database using ( device
'/dev/tpcc/N_023' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_024;
create regular tablespace ts_N_024 pagesize 4K managed by database using ( device
'/dev/tpcc/N_024' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_025;
create regular tablespace ts_N_025 pagesize 4K managed by database using ( device
'/dev/tpcc/N_025' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_026;
create regular tablespace ts_N_026 pagesize 4K managed by database using ( device
'/dev/tpcc/N_026' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_027;
create regular tablespace ts_N_027 pagesize 4K managed by database using ( device
'/dev/tpcc/N_027' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_028;
create regular tablespace ts_N_028 pagesize 4K managed by database using ( device
'/dev/tpcc/N_028' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_029;
create regular tablespace ts_N_029 pagesize 4K managed by database using ( device
'/dev/tpcc/N_029' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_030;
create regular tablespace ts_N_030 pagesize 4K managed by database using ( device
'/dev/tpcc/N_030' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_031;
create regular tablespace ts_N_031 pagesize 4K managed by database using ( device
'/dev/tpcc/N_031' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_032;
create regular tablespace ts_N_032 pagesize 4K managed by database using ( device
'/dev/tpcc/N_032' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_033;
create regular tablespace ts_N_033 pagesize 4K managed by database using ( device
'/dev/tpcc/N_033' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_034;
create regular tablespace ts_N_034 pagesize 4K managed by database using ( device
'/dev/tpcc/N_034' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_035;
create regular tablespace ts_N_035 pagesize 4K managed by database using ( device
'/dev/tpcc/N_035' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_N_036;
create regular tablespace ts_N_036 pagesize 4K managed by database using ( device
'/dev/tpcc/N_036' 534528 ) extentsize 32 bufferpool IBMDEFAULTBP prefetchsize 4096;
commit;
drop tablespace ts_O_001,is_O2_001;
create regular tablespace ts_O_001 pagesize 8K managed by database using ( device
'/dev/tpcc/O_001' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_001 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_001' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_002,is_O2_002;
create regular tablespace ts_O_002 pagesize 8K managed by database using ( device
'/dev/tpcc/O_002' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_002 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_002' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_003,is_O2_003;
create regular tablespace ts_O_003 pagesize 8K managed by database using ( device
'/dev/tpcc/O_003' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_003 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_003' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_004,is_O2_004;
create regular tablespace ts_O_004 pagesize 8K managed by database using ( device
'/dev/tpcc/O_004' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_004 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_004' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_005,is_O2_005;
create regular tablespace ts_O_005 pagesize 8K managed by database using ( device
'/dev/tpcc/O_005' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_005 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_005' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_006,is_O2_006;
create regular tablespace ts_O_006 pagesize 8K managed by database using ( device
'/dev/tpcc/O_006' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_006 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_006' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_007,is_O2_007;
create regular tablespace ts_O_007 pagesize 8K managed by database using ( device
'/dev/tpcc/O_007' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_007 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_007' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_008,is_O2_008;
create regular tablespace ts_O_008 pagesize 8K managed by database using ( device
'/dev/tpcc/O_008' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_008 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_008' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_009,is_O2_009;
create regular tablespace ts_O_009 pagesize 8K managed by database using ( device
'/dev/tpcc/O_009' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_009 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_009' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_010,is_O2_010;
create regular tablespace ts_O_010 pagesize 8K managed by database using ( device
'/dev/tpcc/O_010' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_010 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_010' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_011,is_O2_011;
create regular tablespace ts_O_011 pagesize 8K managed by database using ( device
'/dev/tpcc/O_011' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_011 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_011' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_012,is_O2_012;
create regular tablespace ts_O_012 pagesize 8K managed by database using ( device
'/dev/tpcc/O_012' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
```

```
create regular tablespace is_O2_012 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_012' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_013,is_O2_013;
create regular tablespace ts_O_013 pagesize 8K managed by database using ( device
'/dev/tpcc/O_013' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_013 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_013' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_014,is_O2_014;
create regular tablespace ts_O_014 pagesize 8K managed by database using ( device
'/dev/tpcc/O_014' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_014 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_014' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_015,is_O2_015;
create regular tablespace ts_O_015 pagesize 8K managed by database using ( device
'/dev/tpcc/O_015' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_015 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_015' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_016,is_O2_016;
create regular tablespace ts_O_016 pagesize 8K managed by database using ( device
'/dev/tpcc/O_016' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_016 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_016' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_017,is_O2_017;
create regular tablespace ts_O_017 pagesize 8K managed by database using ( device
'/dev/tpcc/O_017' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_017 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_017' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_018,is_O2_018;
create regular tablespace ts_O_018 pagesize 8K managed by database using ( device
'/dev/tpcc/O_018' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_018 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_018' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_019,is_O2_019;
create regular tablespace ts_O_019 pagesize 8K managed by database using ( device
'/dev/tpcc/O_019' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_019 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_019' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_020,is_O2_020;
create regular tablespace ts_O_020 pagesize 8K managed by database using ( device
'/dev/tpcc/O_020' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_020 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_020' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_021,is_O2_021;
create regular tablespace ts_O_021 pagesize 8K managed by database using ( device
'/dev/tpcc/O_021' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_021 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_021' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_022,is_O2_022;
create regular tablespace ts_O_022 pagesize 8K managed by database using ( device
'/dev/tpcc/O_022' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_022 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_022' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_023,is_O2_023;
create regular tablespace ts_O_023 pagesize 8K managed by database using ( device
'/dev/tpcc/O_023' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_023 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_023' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_024,is_O2_024;
create regular tablespace ts_O_024 pagesize 8K managed by database using ( device
'/dev/tpcc/O_024' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_024 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_024' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_025,is_O2_025;
create regular tablespace ts_O_025 pagesize 8K managed by database using ( device
'/dev/tpcc/O_025' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_025 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_025' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_026,is_O2_026;
create regular tablespace ts_O_026 pagesize 8K managed by database using ( device
'/dev/tpcc/O_026' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_026 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_026' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_027,is_O2_027;
create regular tablespace ts_O_027 pagesize 8K managed by database using ( device
'/dev/tpcc/O_027' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_027 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_027' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_028,is_O2_028;
create regular tablespace ts_O_028 pagesize 8K managed by database using ( device
'/dev/tpcc/O_028' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_028 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_028' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_029,is_O2_029;
create regular tablespace ts_O_029 pagesize 8K managed by database using ( device
'/dev/tpcc/O_029' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_029 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_029' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_030,is_O2_030;
create regular tablespace ts_O_030 pagesize 8K managed by database using ( device
'/dev/tpcc/O_030' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_030 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_030' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_031,is_O2_031;
create regular tablespace ts_O_031 pagesize 8K managed by database using ( device
'/dev/tpcc/O_031' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_031 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_031' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_032,is_O2_032;
create regular tablespace ts_O_032 pagesize 8K managed by database using ( device
'/dev/tpcc/O_032' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_032 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_032' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_033,is_O2_033;
create regular tablespace ts_O_033 pagesize 8K managed by database using ( device
'/dev/tpcc/O_033' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_033 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_033' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_034,is_O2_034;
create regular tablespace ts_O_034 pagesize 8K managed by database using ( device
'/dev/tpcc/O_034' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_034 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_034' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_035,is_O2_035;
create regular tablespace ts_O_035 pagesize 8K managed by database using ( device
'/dev/tpcc/O_035' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_035 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_035' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_O_036,is_O2_036;
create regular tablespace ts_O_036 pagesize 8K managed by database using ( device
'/dev/tpcc/O_036' 1087488 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
create regular tablespace is_O2_036 pagesize 8K managed by database using ( device
'/dev/tpcc/O2_036' 792576 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize 4096;
commit;
drop tablespace ts_OL_001;
create regular tablespace ts_OL_001 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_001' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_002;
create regular tablespace ts_OL_002 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_002' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_003;
create regular tablespace ts_OL_003 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_003' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_004;
create regular tablespace ts_OL_004 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_004' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_005;
```

```
create regular tablespace ts_OL_005 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_005' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_006;
create regular tablespace ts_OL_006 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_006' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_007;
create regular tablespace ts_OL_007 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_007' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_008;
create regular tablespace ts_OL_008 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_008' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_009;
create regular tablespace ts_OL_009 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_009' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_010;
create regular tablespace ts_OL_010 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_010' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_011;
create regular tablespace ts_OL_011 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_011' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_012;
create regular tablespace ts_OL_012 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_012' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_013;
create regular tablespace ts_OL_013 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_013' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_014;
create regular tablespace ts_OL_014 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_014' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_015;
create regular tablespace ts_OL_015 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_015' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_016;
create regular tablespace ts_OL_016 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_016' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_017;
create regular tablespace ts_OL_017 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_017' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_018;
create regular tablespace ts_OL_018 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_018' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_019;
create regular tablespace ts_OL_019 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_019' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_020;
create regular tablespace ts_OL_020 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_020' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_021;
create regular tablespace ts_OL_021 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_021' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_022;
create regular tablespace ts_OL_022 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_022' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_023;
create regular tablespace ts_OL_023 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_023' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_024;
create regular tablespace ts_OL_024 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_024' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_025;
create regular tablespace ts_OL_025 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_025' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_026;
create regular tablespace ts_OL_026 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_026' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_027;
create regular tablespace ts_OL_027 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_027' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_028;
create regular tablespace ts_OL_028 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_028' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_029;
create regular tablespace ts_OL_029 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_029' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_030;
create regular tablespace ts_OL_030 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_030' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_031;
create regular tablespace ts_OL_031 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_031' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_032;
create regular tablespace ts_OL_032 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_032' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_033;
create regular tablespace ts_OL_033 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_033' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_034;
create regular tablespace ts_OL_034 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_034' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_035;
create regular tablespace ts_OL_035 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_035' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_036;
create regular tablespace ts_OL_036 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_036' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_037;
create regular tablespace ts_OL_037 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_037' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_038;
create regular tablespace ts_OL_038 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_038' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_039;
create regular tablespace ts_OL_039 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_039' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_040;
create regular tablespace ts_OL_040 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_040' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_041;
create regular tablespace ts_OL_041 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_041' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_042;
create regular tablespace ts_OL_042 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_042' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_043;
```

```
create regular tablespace ts_OL_043 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_043' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_044;
create regular tablespace ts_OL_044 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_044' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_045;
create regular tablespace ts_OL_045 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_045' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_046;
create regular tablespace ts_OL_046 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_046' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_047;
create regular tablespace ts_OL_047 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_047' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_048;
create regular tablespace ts_OL_048 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_048' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_049;
create regular tablespace ts_OL_049 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_049' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_050;
create regular tablespace ts_OL_050 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_050' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_051;
create regular tablespace ts_OL_051 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_051' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_052;
create regular tablespace ts_OL_052 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_052' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_053;
create regular tablespace ts_OL_053 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_053' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_054;
create regular tablespace ts_OL_054 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_054' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_055;
create regular tablespace ts_OL_055 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_055' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_056;
create regular tablespace ts_OL_056 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_056' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_057;
create regular tablespace ts_OL_057 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_057' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_058;
create regular tablespace ts_OL_058 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_058' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_059;
create regular tablespace ts_OL_059 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_059' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_060;
create regular tablespace ts_OL_060 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_060' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_061;
create regular tablespace ts_OL_061 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_061' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_062;
create regular tablespace ts_OL_062 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_062' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_063;
create regular tablespace ts_OL_063 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_063' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_064;
create regular tablespace ts_OL_064 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_064' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_065;
create regular tablespace ts_OL_065 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_065' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_066;
create regular tablespace ts_OL_066 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_066' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_067;
create regular tablespace ts_OL_067 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_067' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_068;
create regular tablespace ts_OL_068 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_068' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_069;
create regular tablespace ts_OL_069 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_069' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_070;
create regular tablespace ts_OL_070 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_070' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_071;
create regular tablespace ts_OL_071 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_071' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_OL_072;
create regular tablespace ts_OL_072 pagesize 8K managed by database using ( device
'/dev/tpcc/OL_072' 14220288 ) extentsize 32 bufferpool IBMDEFAULT8K prefetchsize
4096;
commit;
drop tablespace ts_H_001;
create regular tablespace ts_H_001 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_001' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_002;
create regular tablespace ts_H_002 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_002' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_003;
create regular tablespace ts_H_003 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_003' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_004;
create regular tablespace ts_H_004 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_004' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_005;
create regular tablespace ts_H_005 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_005' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_006;
create regular tablespace ts_H_006 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_006' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_007;
create regular tablespace ts_H_007 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_007' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_008;
create regular tablespace ts_H_008 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_008' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_009;
create regular tablespace ts_H_009 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_009' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_010;
create regular tablespace ts_H_010 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_010' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_011;
create regular tablespace ts_H_011 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_011' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
```

```
commit;
drop tablespace ts_H_012;
create regular tablespace ts_H_012 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_012' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_013;
create regular tablespace ts_H_013 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_013' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_014;
create regular tablespace ts_H_014 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_014' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_015;
create regular tablespace ts_H_015 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_015' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_016;
create regular tablespace ts_H_016 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_016' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_017;
create regular tablespace ts_H_017 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_017' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_018;
create regular tablespace ts_H_018 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_018' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_019;
create regular tablespace ts_H_019 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_019' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_020;
create regular tablespace ts_H_020 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_020' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_021;
create regular tablespace ts_H_021 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_021' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_022;
create regular tablespace ts_H_022 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_022' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_023;
create regular tablespace ts_H_023 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_023' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_024;
create regular tablespace ts_H_024 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_024' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_025;
create regular tablespace ts_H_025 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_025' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_026;
create regular tablespace ts_H_026 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_026' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_027;
create regular tablespace ts_H_027 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_027' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_028;
create regular tablespace ts_H_028 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_028' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_029;
create regular tablespace ts_H_029 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_029' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_030;
create regular tablespace ts_H_030 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_030' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_031;
create regular tablespace ts_H_031 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_031' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_032;
create regular tablespace ts_H_032 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_032' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_033;
create regular tablespace ts_H_033 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_033' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_034;
create regular tablespace ts_H_034 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_034' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_035;
create regular tablespace ts_H_035 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_035' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
drop tablespace ts_H_036;
create regular tablespace ts_H_036 pagesize 16K  managed by database using ( device
'/dev/tpcc/H_036' 852480 ) extentsize 32 bufferpool IBMDEFAULT16K prefetchsize 4096;
commit;
connect reset;
```

CRCONST_CUSTOMER100.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER100 OFF;
ALTER TABLE CUSTOMER100 DROP CONSTRAINT CUSTOMER100CKC;
ALTER TABLE CUSTOMER100 ADD CONSTRAINT CUSTOMER100CKC CHECK
(C_W_ID BETWEEN 126523 AND 127800);
SET INTEGRITY FOR CUSTOMER100 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER101.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER101 OFF;
ALTER TABLE CUSTOMER101 DROP CONSTRAINT CUSTOMER101CKC;
ALTER TABLE CUSTOMER101 ADD CONSTRAINT CUSTOMER101CKC CHECK
(C_W_ID BETWEEN 127801 AND 129078);
SET INTEGRITY FOR CUSTOMER101 ALL IMMEDIATE UNCHECKED;
connect reset;
```

ALTTBSP_PF_4096.ddl

```
connect to TPCC;
alter tablespace is_C2_001 prefetchsize 4096;
alter tablespace is_C2_002 prefetchsize 4096;
alter tablespace is_C2_003 prefetchsize 4096;
alter tablespace is_C2_004 prefetchsize 4096;
alter tablespace is_C2_005 prefetchsize 4096;
alter tablespace is_C2_006 prefetchsize 4096;
alter tablespace is_C2_007 prefetchsize 4096;
alter tablespace is_C2_008 prefetchsize 4096;
alter tablespace is_C2_009 prefetchsize 4096;
alter tablespace is_C2_010 prefetchsize 4096;
alter tablespace is_C2_011 prefetchsize 4096;
alter tablespace is_C2_012 prefetchsize 4096;
alter tablespace is_C2_013 prefetchsize 4096;
alter tablespace is_C2_014 prefetchsize 4096;
alter tablespace is_C2_015 prefetchsize 4096;
alter tablespace is_C2_016 prefetchsize 4096;
alter tablespace is_C2_017 prefetchsize 4096;
alter tablespace is_C2_018 prefetchsize 4096;
alter tablespace is_C2_019 prefetchsize 4096;
alter tablespace is_C2_020 prefetchsize 4096;
alter tablespace is_C2_021 prefetchsize 4096;
alter tablespace is_C2_022 prefetchsize 4096;
alter tablespace is_C2_023 prefetchsize 4096;
alter tablespace is_C2_024 prefetchsize 4096;
alter tablespace is_C2_025 prefetchsize 4096;
alter tablespace is_C2_026 prefetchsize 4096;
alter tablespace is_C2_027 prefetchsize 4096;
alter tablespace is_C2_028 prefetchsize 4096;
alter tablespace is_C2_029 prefetchsize 4096;
alter tablespace is_C2_030 prefetchsize 4096;
alter tablespace is_C2_031 prefetchsize 4096;
alter tablespace is_C2_032 prefetchsize 4096;
alter tablespace is_C2_033 prefetchsize 4096;
alter tablespace is_C2_034 prefetchsize 4096;
alter tablespace is_C2_035 prefetchsize 4096;
alter tablespace is_C2_036 prefetchsize 4096;
alter tablespace is_C2_037 prefetchsize 4096;
alter tablespace is_C2_038 prefetchsize 4096;
alter tablespace is_C2_039 prefetchsize 4096;
alter tablespace is_C2_040 prefetchsize 4096;
alter tablespace is_C2_041 prefetchsize 4096;
alter tablespace is_C2_042 prefetchsize 4096;
alter tablespace is_C2_043 prefetchsize 4096;
alter tablespace is_C2_044 prefetchsize 4096;
alter tablespace is_C2_045 prefetchsize 4096;
alter tablespace is_C2_046 prefetchsize 4096;
alter tablespace is_C2_047 prefetchsize 4096;
alter tablespace is_C2_048 prefetchsize 4096;
alter tablespace is_C2_049 prefetchsize 4096;
alter tablespace is_C2_050 prefetchsize 4096;
alter tablespace is_C2_051 prefetchsize 4096;
alter tablespace is_C2_052 prefetchsize 4096;
alter tablespace is_C2_053 prefetchsize 4096;
alter tablespace is_C2_054 prefetchsize 4096;
alter tablespace is_C2_055 prefetchsize 4096;
alter tablespace is_C2_056 prefetchsize 4096;
alter tablespace is_C2_057 prefetchsize 4096;
alter tablespace is_C2_058 prefetchsize 4096;
alter tablespace is_C2_059 prefetchsize 4096;
alter tablespace is_C2_060 prefetchsize 4096;
alter tablespace is_C2_061 prefetchsize 4096;
alter tablespace is_C2_062 prefetchsize 4096;
```

```
alter tablespace is_C2_063 prefetchsize 4096;
alter tablespace is_C2_064 prefetchsize 4096;
alter tablespace is_C2_065 prefetchsize 4096;
alter tablespace is_C2_066 prefetchsize 4096;
alter tablespace is_C2_067 prefetchsize 4096;
alter tablespace is_C2_068 prefetchsize 4096;
alter tablespace is_C2_069 prefetchsize 4096;
alter tablespace is_C2_070 prefetchsize 4096;
alter tablespace is_C2_071 prefetchsize 4096;
alter tablespace is_C2_072 prefetchsize 4096;
alter tablespace is_C2_073 prefetchsize 4096;
alter tablespace is_C2_074 prefetchsize 4096;
alter tablespace is_C2_075 prefetchsize 4096;
alter tablespace is_C2_076 prefetchsize 4096;
alter tablespace is_C2_077 prefetchsize 4096;
alter tablespace is_C2_078 prefetchsize 4096;
alter tablespace is_C2_079 prefetchsize 4096;
alter tablespace is_C2_080 prefetchsize 4096;
alter tablespace is_C2_081 prefetchsize 4096;
alter tablespace is_C2_082 prefetchsize 4096;
alter tablespace is_C2_083 prefetchsize 4096;
alter tablespace is_C2_084 prefetchsize 4096;
alter tablespace is_C2_085 prefetchsize 4096;
alter tablespace is_C2_086 prefetchsize 4096;
alter tablespace is_C2_087 prefetchsize 4096;
alter tablespace is_C2_088 prefetchsize 4096;
alter tablespace is_C2_089 prefetchsize 4096;
alter tablespace is_C2_090 prefetchsize 4096;
alter tablespace is_C2_091 prefetchsize 4096;
alter tablespace is_C2_092 prefetchsize 4096;
alter tablespace is_C2_093 prefetchsize 4096;
alter tablespace is_C2_094 prefetchsize 4096;
alter tablespace is_C2_095 prefetchsize 4096;
alter tablespace is_C2_096 prefetchsize 4096;
alter tablespace is_C2_097 prefetchsize 4096;
alter tablespace is_C2_098 prefetchsize 4096;
alter tablespace is_C2_099 prefetchsize 4096;
alter tablespace is_C2_100 prefetchsize 4096;
alter tablespace is_C2_101 prefetchsize 4096;
alter tablespace is_C2_102 prefetchsize 4096;
alter tablespace is_C2_103 prefetchsize 4096;
alter tablespace is_C2_104 prefetchsize 4096;
alter tablespace is_C2_105 prefetchsize 4096;
alter tablespace is_C2_106 prefetchsize 4096;
alter tablespace is_C2_107 prefetchsize 4096;
alter tablespace is_C2_108 prefetchsize 4096;
alter tablespace is_C2_109 prefetchsize 4096;
alter tablespace is_C2_110 prefetchsize 4096;
alter tablespace is_C2_111 prefetchsize 4096;
alter tablespace is_C2_112 prefetchsize 4096;
alter tablespace is_C2_113 prefetchsize 4096;
alter tablespace is_C2_114 prefetchsize 4096;
alter tablespace is_C2_115 prefetchsize 4096;
alter tablespace is_C2_116 prefetchsize 4096;
alter tablespace is_C2_117 prefetchsize 4096;
alter tablespace is_C2_118 prefetchsize 4096;
alter tablespace is_C2_119 prefetchsize 4096;
alter tablespace is_C2_120 prefetchsize 4096;
alter tablespace is_C2_121 prefetchsize 4096;
alter tablespace is_C2_122 prefetchsize 4096;
alter tablespace is_C2_123 prefetchsize 4096;
alter tablespace is_C2_124 prefetchsize 4096;
alter tablespace is_C2_125 prefetchsize 4096;
alter tablespace is_C2_126 prefetchsize 4096;
alter tablespace is_C2_127 prefetchsize 4096;
alter tablespace is_C2_128 prefetchsize 4096;
alter tablespace is_C2_129 prefetchsize 4096;
alter tablespace is_C2_130 prefetchsize 4096;
alter tablespace is_C2_131 prefetchsize 4096;
alter tablespace is_C2_132 prefetchsize 4096;
alter tablespace is_C2_133 prefetchsize 4096;
alter tablespace is_C2_134 prefetchsize 4096;
alter tablespace is_C2_135 prefetchsize 4096;
alter tablespace is_C2_136 prefetchsize 4096;
alter tablespace is_C2_137 prefetchsize 4096;
alter tablespace is_C2_138 prefetchsize 4096;
alter tablespace is_C2_139 prefetchsize 4096;
alter tablespace is_C2_140 prefetchsize 4096;
alter tablespace is_C2_141 prefetchsize 4096;
alter tablespace is_C2_142 prefetchsize 4096;
alter tablespace is_C2_143 prefetchsize 4096;
alter tablespace is_C2_144 prefetchsize 4096;
alter tablespace is_O2_001 prefetchsize 4096;
alter tablespace is_O2_002 prefetchsize 4096;
alter tablespace is_O2_003 prefetchsize 4096;
alter tablespace is_O2_004 prefetchsize 4096;
alter tablespace is_O2_005 prefetchsize 4096;
alter tablespace is_O2_006 prefetchsize 4096;
alter tablespace is_O2_007 prefetchsize 4096;
alter tablespace is_O2_008 prefetchsize 4096;
alter tablespace is_O2_009 prefetchsize 4096;
alter tablespace is_O2_010 prefetchsize 4096;
alter tablespace is_O2_011 prefetchsize 4096;
alter tablespace is_O2_012 prefetchsize 4096;
alter tablespace is_O2_013 prefetchsize 4096;
alter tablespace is_O2_014 prefetchsize 4096;
alter tablespace is_O2_015 prefetchsize 4096;
alter tablespace is_O2_016 prefetchsize 4096;
alter tablespace is_O2_017 prefetchsize 4096;
alter tablespace is_O2_018 prefetchsize 4096;
alter tablespace is_O2_019 prefetchsize 4096;
alter tablespace is_O2_020 prefetchsize 4096;
alter tablespace is_O2_021 prefetchsize 4096;
alter tablespace is_O2_022 prefetchsize 4096;
alter tablespace is_O2_023 prefetchsize 4096;
alter tablespace is_O2_024 prefetchsize 4096;
alter tablespace is_O2_025 prefetchsize 4096;
alter tablespace is_O2_026 prefetchsize 4096;
alter tablespace is_O2_027 prefetchsize 4096;
alter tablespace is_O2_028 prefetchsize 4096;
alter tablespace is_O2_029 prefetchsize 4096;
alter tablespace is_O2_030 prefetchsize 4096;
alter tablespace is_O2_031 prefetchsize 4096;
alter tablespace is_O2_032 prefetchsize 4096;
alter tablespace is_O2_033 prefetchsize 4096;
alter tablespace is_O2_034 prefetchsize 4096;
alter tablespace is_O2_035 prefetchsize 4096;
alter tablespace is_O2_036 prefetchsize 4096;
alter tablespace ts_C_001 prefetchsize 4096;
alter tablespace ts_C_002 prefetchsize 4096;
alter tablespace ts_C_003 prefetchsize 4096;
alter tablespace ts_C_004 prefetchsize 4096;
alter tablespace ts_C_005 prefetchsize 4096;
alter tablespace ts_C_006 prefetchsize 4096;
alter tablespace ts_C_007 prefetchsize 4096;
alter tablespace ts_C_008 prefetchsize 4096;
alter tablespace ts_C_009 prefetchsize 4096;
alter tablespace ts_C_010 prefetchsize 4096;
alter tablespace ts_C_011 prefetchsize 4096;
alter tablespace ts_C_012 prefetchsize 4096;
alter tablespace ts_C_013 prefetchsize 4096;
alter tablespace ts_C_014 prefetchsize 4096;
alter tablespace ts_C_015 prefetchsize 4096;
alter tablespace ts_C_016 prefetchsize 4096;
alter tablespace ts_C_017 prefetchsize 4096;
alter tablespace ts_C_018 prefetchsize 4096;
alter tablespace ts_C_019 prefetchsize 4096;
alter tablespace ts_C_020 prefetchsize 4096;
alter tablespace ts_C_021 prefetchsize 4096;
alter tablespace ts_C_022 prefetchsize 4096;
alter tablespace ts_C_023 prefetchsize 4096;
alter tablespace ts_C_024 prefetchsize 4096;
alter tablespace ts_C_025 prefetchsize 4096;
alter tablespace ts_C_026 prefetchsize 4096;
alter tablespace ts_C_027 prefetchsize 4096;
alter tablespace ts_C_028 prefetchsize 4096;
alter tablespace ts_C_029 prefetchsize 4096;
alter tablespace ts_C_030 prefetchsize 4096;
alter tablespace ts_C_031 prefetchsize 4096;
alter tablespace ts_C_032 prefetchsize 4096;
alter tablespace ts_C_033 prefetchsize 4096;
alter tablespace ts_C_034 prefetchsize 4096;
alter tablespace ts_C_035 prefetchsize 4096;
alter tablespace ts_C_036 prefetchsize 4096;
alter tablespace ts_C_037 prefetchsize 4096;
alter tablespace ts_C_038 prefetchsize 4096;
alter tablespace ts_C_039 prefetchsize 4096;
alter tablespace ts_C_040 prefetchsize 4096;
alter tablespace ts_C_041 prefetchsize 4096;
alter tablespace ts_C_042 prefetchsize 4096;
alter tablespace ts_C_043 prefetchsize 4096;
alter tablespace ts_C_044 prefetchsize 4096;
alter tablespace ts_C_045 prefetchsize 4096;
alter tablespace ts_C_046 prefetchsize 4096;
alter tablespace ts_C_047 prefetchsize 4096;
alter tablespace ts_C_048 prefetchsize 4096;
alter tablespace ts_C_049 prefetchsize 4096;
alter tablespace ts_C_050 prefetchsize 4096;
alter tablespace ts_C_051 prefetchsize 4096;
alter tablespace ts_C_052 prefetchsize 4096;
alter tablespace ts_C_053 prefetchsize 4096;
alter tablespace ts_C_054 prefetchsize 4096;
alter tablespace ts_C_055 prefetchsize 4096;
alter tablespace ts_C_056 prefetchsize 4096;
alter tablespace ts_C_057 prefetchsize 4096;
alter tablespace ts_C_058 prefetchsize 4096;
alter tablespace ts_C_059 prefetchsize 4096;
alter tablespace ts_C_060 prefetchsize 4096;
alter tablespace ts_C_061 prefetchsize 4096;
alter tablespace ts_C_062 prefetchsize 4096;
alter tablespace ts_C_063 prefetchsize 4096;
alter tablespace ts_C_064 prefetchsize 4096;
alter tablespace ts_C_065 prefetchsize 4096;
alter tablespace ts_C_066 prefetchsize 4096;
alter tablespace ts_C_067 prefetchsize 4096;
alter tablespace ts_C_068 prefetchsize 4096;
alter tablespace ts_C_069 prefetchsize 4096;
alter tablespace ts_C_070 prefetchsize 4096;
alter tablespace ts_C_071 prefetchsize 4096;
alter tablespace ts_C_072 prefetchsize 4096;
alter tablespace ts_C_073 prefetchsize 4096;
alter tablespace ts_C_074 prefetchsize 4096;
alter tablespace ts_C_075 prefetchsize 4096;
alter tablespace ts_C_076 prefetchsize 4096;
```

```
alter tablespace ts_C_077 prefetchsize 4096;
alter tablespace ts_C_078 prefetchsize 4096;
alter tablespace ts_C_079 prefetchsize 4096;
alter tablespace ts_C_080 prefetchsize 4096;
alter tablespace ts_C_081 prefetchsize 4096;
alter tablespace ts_C_082 prefetchsize 4096;
alter tablespace ts_C_083 prefetchsize 4096;
alter tablespace ts_C_084 prefetchsize 4096;
alter tablespace ts_C_085 prefetchsize 4096;
alter tablespace ts_C_086 prefetchsize 4096;
alter tablespace ts_C_087 prefetchsize 4096;
alter tablespace ts_C_088 prefetchsize 4096;
alter tablespace ts_C_089 prefetchsize 4096;
alter tablespace ts_C_090 prefetchsize 4096;
alter tablespace ts_C_091 prefetchsize 4096;
alter tablespace ts_C_092 prefetchsize 4096;
alter tablespace ts_C_093 prefetchsize 4096;
alter tablespace ts_C_094 prefetchsize 4096;
alter tablespace ts_C_095 prefetchsize 4096;
alter tablespace ts_C_096 prefetchsize 4096;
alter tablespace ts_C_097 prefetchsize 4096;
alter tablespace ts_C_098 prefetchsize 4096;
alter tablespace ts_C_099 prefetchsize 4096;
alter tablespace ts_C_100 prefetchsize 4096;
alter tablespace ts_C_101 prefetchsize 4096;
alter tablespace ts_C_102 prefetchsize 4096;
alter tablespace ts_C_103 prefetchsize 4096;
alter tablespace ts_C_104 prefetchsize 4096;
alter tablespace ts_C_105 prefetchsize 4096;
alter tablespace ts_C_106 prefetchsize 4096;
alter tablespace ts_C_107 prefetchsize 4096;
alter tablespace ts_C_108 prefetchsize 4096;
alter tablespace ts_C_109 prefetchsize 4096;
alter tablespace ts_C_110 prefetchsize 4096;
alter tablespace ts_C_111 prefetchsize 4096;
alter tablespace ts_C_112 prefetchsize 4096;
alter tablespace ts_C_113 prefetchsize 4096;
alter tablespace ts_C_114 prefetchsize 4096;
alter tablespace ts_C_115 prefetchsize 4096;
alter tablespace ts_C_116 prefetchsize 4096;
alter tablespace ts_C_117 prefetchsize 4096;
alter tablespace ts_C_118 prefetchsize 4096;
alter tablespace ts_C_119 prefetchsize 4096;
alter tablespace ts_C_120 prefetchsize 4096;
alter tablespace ts_C_121 prefetchsize 4096;
alter tablespace ts_C_122 prefetchsize 4096;
alter tablespace ts_C_123 prefetchsize 4096;
alter tablespace ts_C_124 prefetchsize 4096;
alter tablespace ts_C_125 prefetchsize 4096;
alter tablespace ts_C_126 prefetchsize 4096;
alter tablespace ts_C_127 prefetchsize 4096;
alter tablespace ts_C_128 prefetchsize 4096;
alter tablespace ts_C_129 prefetchsize 4096;
alter tablespace ts_C_130 prefetchsize 4096;
alter tablespace ts_C_131 prefetchsize 4096;
alter tablespace ts_C_132 prefetchsize 4096;
alter tablespace ts_C_133 prefetchsize 4096;
alter tablespace ts_C_134 prefetchsize 4096;
alter tablespace ts_C_135 prefetchsize 4096;
alter tablespace ts_C_136 prefetchsize 4096;
alter tablespace ts_C_137 prefetchsize 4096;
alter tablespace ts_C_138 prefetchsize 4096;
alter tablespace ts_C_139 prefetchsize 4096;
alter tablespace ts_C_140 prefetchsize 4096;
alter tablespace ts_C_141 prefetchsize 4096;
alter tablespace ts_C_142 prefetchsize 4096;
alter tablespace ts_C_143 prefetchsize 4096;
alter tablespace ts_C_144 prefetchsize 4096;
alter tablespace ts_D_001 prefetchsize 4096;
alter tablespace ts_D_002 prefetchsize 4096;
alter tablespace ts_D_003 prefetchsize 4096;
alter tablespace ts_D_004 prefetchsize 4096;
alter tablespace ts_D_005 prefetchsize 4096;
alter tablespace ts_D_006 prefetchsize 4096;
alter tablespace ts_D_007 prefetchsize 4096;
alter tablespace ts_D_008 prefetchsize 4096;
alter tablespace ts_D_009 prefetchsize 4096;
alter tablespace ts_D_010 prefetchsize 4096;
alter tablespace ts_D_011 prefetchsize 4096;
alter tablespace ts_D_012 prefetchsize 4096;
alter tablespace ts_D_013 prefetchsize 4096;
alter tablespace ts_D_014 prefetchsize 4096;
alter tablespace ts_D_015 prefetchsize 4096;
alter tablespace ts_D_016 prefetchsize 4096;
alter tablespace ts_D_017 prefetchsize 4096;
alter tablespace ts_D_018 prefetchsize 4096;
alter tablespace ts_D_019 prefetchsize 4096;
alter tablespace ts_D_020 prefetchsize 4096;
alter tablespace ts_D_021 prefetchsize 4096;
alter tablespace ts_D_022 prefetchsize 4096;
alter tablespace ts_D_023 prefetchsize 4096;
alter tablespace ts_D_024 prefetchsize 4096;
alter tablespace ts_D_025 prefetchsize 4096;
alter tablespace ts_D_026 prefetchsize 4096;
alter tablespace ts_D_027 prefetchsize 4096;
alter tablespace ts_D_028 prefetchsize 4096;
alter tablespace ts_D_029 prefetchsize 4096;
alter tablespace ts_D_030 prefetchsize 4096;
alter tablespace ts_D_031 prefetchsize 4096;
alter tablespace ts_D_032 prefetchsize 4096;
alter tablespace ts_D_033 prefetchsize 4096;
alter tablespace ts_D_034 prefetchsize 4096;
alter tablespace ts_D_035 prefetchsize 4096;
alter tablespace ts_D_036 prefetchsize 4096;
alter tablespace ts_H_001 prefetchsize 4096;
alter tablespace ts_H_002 prefetchsize 4096;
alter tablespace ts_H_003 prefetchsize 4096;
alter tablespace ts_H_004 prefetchsize 4096;
alter tablespace ts_H_005 prefetchsize 4096;
alter tablespace ts_H_006 prefetchsize 4096;
alter tablespace ts_H_007 prefetchsize 4096;
alter tablespace ts_H_008 prefetchsize 4096;
alter tablespace ts_H_009 prefetchsize 4096;
alter tablespace ts_H_010 prefetchsize 4096;
alter tablespace ts_H_011 prefetchsize 4096;
alter tablespace ts_H_012 prefetchsize 4096;
alter tablespace ts_H_013 prefetchsize 4096;
alter tablespace ts_H_014 prefetchsize 4096;
alter tablespace ts_H_015 prefetchsize 4096;
alter tablespace ts_H_016 prefetchsize 4096;
alter tablespace ts_H_017 prefetchsize 4096;
alter tablespace ts_H_018 prefetchsize 4096;
alter tablespace ts_H_019 prefetchsize 4096;
alter tablespace ts_H_020 prefetchsize 4096;
alter tablespace ts_H_021 prefetchsize 4096;
alter tablespace ts_H_022 prefetchsize 4096;
alter tablespace ts_H_023 prefetchsize 4096;
alter tablespace ts_H_024 prefetchsize 4096;
alter tablespace ts_H_025 prefetchsize 4096;
alter tablespace ts_H_026 prefetchsize 4096;
alter tablespace ts_H_027 prefetchsize 4096;
alter tablespace ts_H_028 prefetchsize 4096;
alter tablespace ts_H_029 prefetchsize 4096;
alter tablespace ts_H_030 prefetchsize 4096;
alter tablespace ts_H_031 prefetchsize 4096;
alter tablespace ts_H_032 prefetchsize 4096;
alter tablespace ts_H_033 prefetchsize 4096;
alter tablespace ts_H_034 prefetchsize 4096;
alter tablespace ts_H_035 prefetchsize 4096;
alter tablespace ts_H_036 prefetchsize 4096;
alter tablespace ts_I prefetchsize 4096;
alter tablespace ts_N_001 prefetchsize 4096;
alter tablespace ts_N_002 prefetchsize 4096;
alter tablespace ts_N_003 prefetchsize 4096;
alter tablespace ts_N_004 prefetchsize 4096;
alter tablespace ts_N_005 prefetchsize 4096;
alter tablespace ts_N_006 prefetchsize 4096;
alter tablespace ts_N_007 prefetchsize 4096;
alter tablespace ts_N_008 prefetchsize 4096;
alter tablespace ts_N_009 prefetchsize 4096;
alter tablespace ts_N_010 prefetchsize 4096;
alter tablespace ts_N_011 prefetchsize 4096;
alter tablespace ts_N_012 prefetchsize 4096;
alter tablespace ts_N_013 prefetchsize 4096;
alter tablespace ts_N_014 prefetchsize 4096;
alter tablespace ts_N_015 prefetchsize 4096;
alter tablespace ts_N_016 prefetchsize 4096;
alter tablespace ts_N_017 prefetchsize 4096;
alter tablespace ts_N_018 prefetchsize 4096;
alter tablespace ts_N_019 prefetchsize 4096;
alter tablespace ts_N_020 prefetchsize 4096;
alter tablespace ts_N_021 prefetchsize 4096;
alter tablespace ts_N_022 prefetchsize 4096;
alter tablespace ts_N_023 prefetchsize 4096;
alter tablespace ts_N_024 prefetchsize 4096;
alter tablespace ts_N_025 prefetchsize 4096;
alter tablespace ts_N_026 prefetchsize 4096;
alter tablespace ts_N_027 prefetchsize 4096;
alter tablespace ts_N_028 prefetchsize 4096;
alter tablespace ts_N_029 prefetchsize 4096;
alter tablespace ts_N_030 prefetchsize 4096;
alter tablespace ts_N_031 prefetchsize 4096;
alter tablespace ts_N_032 prefetchsize 4096;
alter tablespace ts_N_033 prefetchsize 4096;
alter tablespace ts_N_034 prefetchsize 4096;
alter tablespace ts_N_035 prefetchsize 4096;
alter tablespace ts_N_036 prefetchsize 4096;
alter tablespace ts_OL_001 prefetchsize 4096;
alter tablespace ts_OL_002 prefetchsize 4096;
alter tablespace ts_OL_003 prefetchsize 4096;
alter tablespace ts_OL_004 prefetchsize 4096;
alter tablespace ts_OL_005 prefetchsize 4096;
alter tablespace ts_OL_006 prefetchsize 4096;
alter tablespace ts_OL_007 prefetchsize 4096;
alter tablespace ts_OL_008 prefetchsize 4096;
alter tablespace ts_OL_009 prefetchsize 4096;
alter tablespace ts_OL_010 prefetchsize 4096;
alter tablespace ts_OL_011 prefetchsize 4096;
alter tablespace ts_OL_012 prefetchsize 4096;
alter tablespace ts_OL_013 prefetchsize 4096;
alter tablespace ts_OL_014 prefetchsize 4096;
alter tablespace ts_OL_015 prefetchsize 4096;
alter tablespace ts_OL_016 prefetchsize 4096;
alter tablespace ts_OL_017 prefetchsize 4096;
```

```
alter tablespace ts_OL_018 prefetchsize 4096;
alter tablespace ts_OL_019 prefetchsize 4096;
alter tablespace ts_OL_020 prefetchsize 4096;
alter tablespace ts_OL_021 prefetchsize 4096;
alter tablespace ts_OL_022 prefetchsize 4096;
alter tablespace ts_OL_023 prefetchsize 4096;
alter tablespace ts_OL_024 prefetchsize 4096;
alter tablespace ts_OL_025 prefetchsize 4096;
alter tablespace ts_OL_026 prefetchsize 4096;
alter tablespace ts_OL_027 prefetchsize 4096;
alter tablespace ts_OL_028 prefetchsize 4096;
alter tablespace ts_OL_029 prefetchsize 4096;
alter tablespace ts_OL_030 prefetchsize 4096;
alter tablespace ts_OL_031 prefetchsize 4096;
alter tablespace ts_OL_032 prefetchsize 4096;
alter tablespace ts_OL_033 prefetchsize 4096;
alter tablespace ts_OL_034 prefetchsize 4096;
alter tablespace ts_OL_035 prefetchsize 4096;
alter tablespace ts_OL_036 prefetchsize 4096;
alter tablespace ts_OL_037 prefetchsize 4096;
alter tablespace ts_OL_038 prefetchsize 4096;
alter tablespace ts_OL_039 prefetchsize 4096;
alter tablespace ts_OL_040 prefetchsize 4096;
alter tablespace ts_OL_041 prefetchsize 4096;
alter tablespace ts_OL_042 prefetchsize 4096;
alter tablespace ts_OL_043 prefetchsize 4096;
alter tablespace ts_OL_044 prefetchsize 4096;
alter tablespace ts_OL_045 prefetchsize 4096;
alter tablespace ts_OL_046 prefetchsize 4096;
alter tablespace ts_OL_047 prefetchsize 4096;
alter tablespace ts_OL_048 prefetchsize 4096;
alter tablespace ts_OL_049 prefetchsize 4096;
alter tablespace ts_OL_050 prefetchsize 4096;
alter tablespace ts_OL_051 prefetchsize 4096;
alter tablespace ts_OL_052 prefetchsize 4096;
alter tablespace ts_OL_053 prefetchsize 4096;
alter tablespace ts_OL_054 prefetchsize 4096;
alter tablespace ts_OL_055 prefetchsize 4096;
alter tablespace ts_OL_056 prefetchsize 4096;
alter tablespace ts_OL_057 prefetchsize 4096;
alter tablespace ts_OL_058 prefetchsize 4096;
alter tablespace ts_OL_059 prefetchsize 4096;
alter tablespace ts_OL_060 prefetchsize 4096;
alter tablespace ts_OL_061 prefetchsize 4096;
alter tablespace ts_OL_062 prefetchsize 4096;
alter tablespace ts_OL_063 prefetchsize 4096;
alter tablespace ts_OL_064 prefetchsize 4096;
alter tablespace ts_OL_065 prefetchsize 4096;
alter tablespace ts_OL_066 prefetchsize 4096;
alter tablespace ts_OL_067 prefetchsize 4096;
alter tablespace ts_OL_068 prefetchsize 4096;
alter tablespace ts_OL_069 prefetchsize 4096;
alter tablespace ts_OL_070 prefetchsize 4096;
alter tablespace ts_OL_071 prefetchsize 4096;
alter tablespace ts_OL_072 prefetchsize 4096;
alter tablespace ts_O_001 prefetchsize 4096;
alter tablespace ts_O_002 prefetchsize 4096;
alter tablespace ts_O_003 prefetchsize 4096;
alter tablespace ts_O_004 prefetchsize 4096;
alter tablespace ts_O_005 prefetchsize 4096;
alter tablespace ts_O_006 prefetchsize 4096;
alter tablespace ts_O_007 prefetchsize 4096;
alter tablespace ts_O_008 prefetchsize 4096;
alter tablespace ts_O_009 prefetchsize 4096;
alter tablespace ts_O_010 prefetchsize 4096;
alter tablespace ts_O_011 prefetchsize 4096;
alter tablespace ts_O_012 prefetchsize 4096;
alter tablespace ts_O_013 prefetchsize 4096;
alter tablespace ts_O_014 prefetchsize 4096;
alter tablespace ts_O_015 prefetchsize 4096;
alter tablespace ts_O_016 prefetchsize 4096;
alter tablespace ts_O_017 prefetchsize 4096;
alter tablespace ts_O_018 prefetchsize 4096;
alter tablespace ts_O_019 prefetchsize 4096;
alter tablespace ts_O_020 prefetchsize 4096;
alter tablespace ts_O_021 prefetchsize 4096;
alter tablespace ts_O_022 prefetchsize 4096;
alter tablespace ts_O_023 prefetchsize 4096;
alter tablespace ts_O_024 prefetchsize 4096;
alter tablespace ts_O_025 prefetchsize 4096;
alter tablespace ts_O_026 prefetchsize 4096;
alter tablespace ts_O_027 prefetchsize 4096;
alter tablespace ts_O_028 prefetchsize 4096;
alter tablespace ts_O_029 prefetchsize 4096;
alter tablespace ts_O_030 prefetchsize 4096;
alter tablespace ts_O_031 prefetchsize 4096;
alter tablespace ts_O_032 prefetchsize 4096;
alter tablespace ts_O_033 prefetchsize 4096;
alter tablespace ts_O_034 prefetchsize 4096;
alter tablespace ts_O_035 prefetchsize 4096;
alter tablespace ts_O_036 prefetchsize 4096;
alter tablespace ts_S_001 prefetchsize 4096;
alter tablespace ts_S_002 prefetchsize 4096;
alter tablespace ts_S_003 prefetchsize 4096;
alter tablespace ts_S_004 prefetchsize 4096;
alter tablespace ts_S_005 prefetchsize 4096;
alter tablespace ts_S_006 prefetchsize 4096;
alter tablespace ts_S_007 prefetchsize 4096;
alter tablespace ts_S_008 prefetchsize 4096;
alter tablespace ts_S_009 prefetchsize 4096;
alter tablespace ts_S_010 prefetchsize 4096;
alter tablespace ts_S_011 prefetchsize 4096;
alter tablespace ts_S_012 prefetchsize 4096;
alter tablespace ts_S_013 prefetchsize 4096;
alter tablespace ts_S_014 prefetchsize 4096;
alter tablespace ts_S_015 prefetchsize 4096;
alter tablespace ts_S_016 prefetchsize 4096;
alter tablespace ts_S_017 prefetchsize 4096;
alter tablespace ts_S_018 prefetchsize 4096;
alter tablespace ts_S_019 prefetchsize 4096;
alter tablespace ts_S_020 prefetchsize 4096;
alter tablespace ts_S_021 prefetchsize 4096;
alter tablespace ts_S_022 prefetchsize 4096;
alter tablespace ts_S_023 prefetchsize 4096;
alter tablespace ts_S_024 prefetchsize 4096;
alter tablespace ts_S_025 prefetchsize 4096;
alter tablespace ts_S_026 prefetchsize 4096;
alter tablespace ts_S_027 prefetchsize 4096;
alter tablespace ts_S_028 prefetchsize 4096;
alter tablespace ts_S_029 prefetchsize 4096;
alter tablespace ts_S_030 prefetchsize 4096;
alter tablespace ts_S_031 prefetchsize 4096;
alter tablespace ts_S_032 prefetchsize 4096;
alter tablespace ts_S_033 prefetchsize 4096;
alter tablespace ts_S_034 prefetchsize 4096;
alter tablespace ts_S_035 prefetchsize 4096;
alter tablespace ts_S_036 prefetchsize 4096;
alter tablespace ts_S_037 prefetchsize 4096;
alter tablespace ts_S_038 prefetchsize 4096;
alter tablespace ts_S_039 prefetchsize 4096;
alter tablespace ts_S_040 prefetchsize 4096;
alter tablespace ts_S_041 prefetchsize 4096;
alter tablespace ts_S_042 prefetchsize 4096;
alter tablespace ts_S_043 prefetchsize 4096;
alter tablespace ts_S_044 prefetchsize 4096;
alter tablespace ts_S_045 prefetchsize 4096;
alter tablespace ts_S_046 prefetchsize 4096;
alter tablespace ts_S_047 prefetchsize 4096;
alter tablespace ts_S_048 prefetchsize 4096;
alter tablespace ts_S_049 prefetchsize 4096;
alter tablespace ts_S_050 prefetchsize 4096;
alter tablespace ts_S_051 prefetchsize 4096;
alter tablespace ts_S_052 prefetchsize 4096;
alter tablespace ts_S_053 prefetchsize 4096;
alter tablespace ts_S_054 prefetchsize 4096;
alter tablespace ts_S_055 prefetchsize 4096;
alter tablespace ts_S_056 prefetchsize 4096;
alter tablespace ts_S_057 prefetchsize 4096;
alter tablespace ts_S_058 prefetchsize 4096;
alter tablespace ts_S_059 prefetchsize 4096;
alter tablespace ts_S_060 prefetchsize 4096;
alter tablespace ts_S_061 prefetchsize 4096;
alter tablespace ts_S_062 prefetchsize 4096;
alter tablespace ts_S_063 prefetchsize 4096;
alter tablespace ts_S_064 prefetchsize 4096;
alter tablespace ts_S_065 prefetchsize 4096;
alter tablespace ts_S_066 prefetchsize 4096;
alter tablespace ts_S_067 prefetchsize 4096;
alter tablespace ts_S_068 prefetchsize 4096;
alter tablespace ts_S_069 prefetchsize 4096;
alter tablespace ts_S_070 prefetchsize 4096;
alter tablespace ts_S_071 prefetchsize 4096;
alter tablespace ts_S_072 prefetchsize 4096;
alter tablespace ts_S_073 prefetchsize 4096;
alter tablespace ts_S_074 prefetchsize 4096;
alter tablespace ts_S_075 prefetchsize 4096;
alter tablespace ts_S_076 prefetchsize 4096;
alter tablespace ts_S_077 prefetchsize 4096;
alter tablespace ts_S_078 prefetchsize 4096;
alter tablespace ts_S_079 prefetchsize 4096;
alter tablespace ts_S_080 prefetchsize 4096;
alter tablespace ts_S_081 prefetchsize 4096;
alter tablespace ts_S_082 prefetchsize 4096;
alter tablespace ts_S_083 prefetchsize 4096;
alter tablespace ts_S_084 prefetchsize 4096;
alter tablespace ts_S_085 prefetchsize 4096;
alter tablespace ts_S_086 prefetchsize 4096;
alter tablespace ts_S_087 prefetchsize 4096;
alter tablespace ts_S_088 prefetchsize 4096;
alter tablespace ts_S_089 prefetchsize 4096;
alter tablespace ts_S_090 prefetchsize 4096;
alter tablespace ts_S_091 prefetchsize 4096;
alter tablespace ts_S_092 prefetchsize 4096;
alter tablespace ts_S_093 prefetchsize 4096;
alter tablespace ts_S_094 prefetchsize 4096;
alter tablespace ts_S_095 prefetchsize 4096;
alter tablespace ts_S_096 prefetchsize 4096;
alter tablespace ts_S_097 prefetchsize 4096;
alter tablespace ts_S_098 prefetchsize 4096;
alter tablespace ts_S_099 prefetchsize 4096;
alter tablespace ts_S_100 prefetchsize 4096;
alter tablespace ts_S_101 prefetchsize 4096;
alter tablespace ts_S_102 prefetchsize 4096;
alter tablespace ts_S_103 prefetchsize 4096;
```

```
alter tablespace ts_S_104 prefetchsize 4096;
alter tablespace ts_S_105 prefetchsize 4096;
alter tablespace ts_S_106 prefetchsize 4096;
alter tablespace ts_S_107 prefetchsize 4096;
alter tablespace ts_S_108 prefetchsize 4096;
alter tablespace ts_S_109 prefetchsize 4096;
alter tablespace ts_S_110 prefetchsize 4096;
alter tablespace ts_S_111 prefetchsize 4096;
alter tablespace ts_S_112 prefetchsize 4096;
alter tablespace ts_S_113 prefetchsize 4096;
alter tablespace ts_S_114 prefetchsize 4096;
alter tablespace ts_S_115 prefetchsize 4096;
alter tablespace ts_S_116 prefetchsize 4096;
alter tablespace ts_S_117 prefetchsize 4096;
alter tablespace ts_S_118 prefetchsize 4096;
alter tablespace ts_S_119 prefetchsize 4096;
alter tablespace ts_S_120 prefetchsize 4096;
alter tablespace ts_S_121 prefetchsize 4096;
alter tablespace ts_S_122 prefetchsize 4096;
alter tablespace ts_S_123 prefetchsize 4096;
alter tablespace ts_S_124 prefetchsize 4096;
alter tablespace ts_S_125 prefetchsize 4096;
alter tablespace ts_S_126 prefetchsize 4096;
alter tablespace ts_S_127 prefetchsize 4096;
alter tablespace ts_S_128 prefetchsize 4096;
alter tablespace ts_S_129 prefetchsize 4096;
alter tablespace ts_S_130 prefetchsize 4096;
alter tablespace ts_S_131 prefetchsize 4096;
alter tablespace ts_S_132 prefetchsize 4096;
alter tablespace ts_S_133 prefetchsize 4096;
alter tablespace ts_S_134 prefetchsize 4096;
alter tablespace ts_S_135 prefetchsize 4096;
alter tablespace ts_S_136 prefetchsize 4096;
alter tablespace ts_S_137 prefetchsize 4096;
alter tablespace ts_S_138 prefetchsize 4096;
alter tablespace ts_S_139 prefetchsize 4096;
alter tablespace ts_S_140 prefetchsize 4096;
alter tablespace ts_S_141 prefetchsize 4096;
alter tablespace ts_S_142 prefetchsize 4096;
alter tablespace ts_S_143 prefetchsize 4096;
alter tablespace ts_S_144 prefetchsize 4096;
alter tablespace ts_W_001 prefetchsize 4096;
alter tablespace ts_W_002 prefetchsize 4096;
alter tablespace ts_W_003 prefetchsize 4096;
alter tablespace ts_W_004 prefetchsize 4096;
alter tablespace ts_W_005 prefetchsize 4096;
alter tablespace ts_W_006 prefetchsize 4096;
alter tablespace ts_W_007 prefetchsize 4096;
alter tablespace ts_W_008 prefetchsize 4096;
alter tablespace ts_W_009 prefetchsize 4096;
alter tablespace ts_W_010 prefetchsize 4096;
alter tablespace ts_W_011 prefetchsize 4096;
alter tablespace ts_W_012 prefetchsize 4096;
alter tablespace ts_W_013 prefetchsize 4096;
alter tablespace ts_W_014 prefetchsize 4096;
alter tablespace ts_W_015 prefetchsize 4096;
alter tablespace ts_W_016 prefetchsize 4096;
alter tablespace ts_W_017 prefetchsize 4096;
alter tablespace ts_W_018 prefetchsize 4096;
alter tablespace ts_W_019 prefetchsize 4096;
alter tablespace ts_W_020 prefetchsize 4096;
alter tablespace ts_W_021 prefetchsize 4096;
alter tablespace ts_W_022 prefetchsize 4096;
alter tablespace ts_W_023 prefetchsize 4096;
alter tablespace ts_W_024 prefetchsize 4096;
alter tablespace ts_W_025 prefetchsize 4096;
alter tablespace ts_W_026 prefetchsize 4096;
alter tablespace ts_W_027 prefetchsize 4096;
alter tablespace ts_W_028 prefetchsize 4096;
alter tablespace ts_W_029 prefetchsize 4096;
alter tablespace ts_W_030 prefetchsize 4096;
alter tablespace ts_W_031 prefetchsize 4096;
alter tablespace ts_W_032 prefetchsize 4096;
alter tablespace ts_W_033 prefetchsize 4096;
alter tablespace ts_W_034 prefetchsize 4096;
alter tablespace ts_W_035 prefetchsize 4096;
alter tablespace ts_W_036 prefetchsize 4096;
connect reset;
```

CRCONST_CUSTOMER102.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER102 OFF;
ALTER TABLE CUSTOMER102 DROP CONSTRAINT CUSTOMER102CKC;
ALTER TABLE CUSTOMER102 ADD CONSTRAINT CUSTOMER102CKC CHECK
(C_W_ID BETWEEN 129079 AND 130356);
SET INTEGRITY FOR CUSTOMER102 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER103.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER103 OFF;
ALTER TABLE CUSTOMER103 DROP CONSTRAINT CUSTOMER103CKC;
ALTER TABLE CUSTOMER103 ADD CONSTRAINT CUSTOMER103CKC CHECK
(C_W_ID BETWEEN 130357 AND 131634);
SET INTEGRITY FOR CUSTOMER103 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER104.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER104 OFF;
ALTER TABLE CUSTOMER104 DROP CONSTRAINT CUSTOMER104CKC;
ALTER TABLE CUSTOMER104 ADD CONSTRAINT CUSTOMER104CKC CHECK
(C_W_ID BETWEEN 131635 AND 132912);
SET INTEGRITY FOR CUSTOMER104 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER105.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER105 OFF;
ALTER TABLE CUSTOMER105 DROP CONSTRAINT CUSTOMER105CKC;
ALTER TABLE CUSTOMER105 ADD CONSTRAINT CUSTOMER105CKC CHECK
(C_W_ID BETWEEN 132913 AND 134190);
SET INTEGRITY FOR CUSTOMER105 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER106.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER106 OFF;
ALTER TABLE CUSTOMER106 DROP CONSTRAINT CUSTOMER106CKC;
ALTER TABLE CUSTOMER106 ADD CONSTRAINT CUSTOMER106CKC CHECK
(C_W_ID BETWEEN 134191 AND 135468);
SET INTEGRITY FOR CUSTOMER106 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER107.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER107 OFF;
ALTER TABLE CUSTOMER107 DROP CONSTRAINT CUSTOMER107CKC;
ALTER TABLE CUSTOMER107 ADD CONSTRAINT CUSTOMER107CKC CHECK
(C_W_ID BETWEEN 135469 AND 136746);
SET INTEGRITY FOR CUSTOMER107 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER108.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER108 OFF;
ALTER TABLE CUSTOMER108 DROP CONSTRAINT CUSTOMER108CKC;
ALTER TABLE CUSTOMER108 ADD CONSTRAINT CUSTOMER108CKC CHECK
(C_W_ID BETWEEN 136747 AND 138024);
SET INTEGRITY FOR CUSTOMER108 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER109.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER109 OFF;
ALTER TABLE CUSTOMER109 DROP CONSTRAINT CUSTOMER109CKC;
ALTER TABLE CUSTOMER109 ADD CONSTRAINT CUSTOMER109CKC CHECK
(C_W_ID BETWEEN 138025 AND 139302);
SET INTEGRITY FOR CUSTOMER109 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER10.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER10 OFF;
ALTER TABLE CUSTOMER10 DROP CONSTRAINT CUSTOMER10CKC;
ALTER TABLE CUSTOMER10 ADD CONSTRAINT CUSTOMER10CKC CHECK
(C_W_ID BETWEEN 11503 AND 12780);
SET INTEGRITY FOR CUSTOMER10 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER110.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER110 OFF;
ALTER TABLE CUSTOMER110 DROP CONSTRAINT CUSTOMER110CKC;
ALTER TABLE CUSTOMER110 ADD CONSTRAINT CUSTOMER110CKC CHECK
(C_W_ID BETWEEN 139303 AND 140580);
SET INTEGRITY FOR CUSTOMER110 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER111.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER111 OFF;
ALTER TABLE CUSTOMER111 DROP CONSTRAINT CUSTOMER111CKC;
ALTER TABLE CUSTOMER111 ADD CONSTRAINT CUSTOMER111CKC CHECK
(C_W_ID BETWEEN 140581 AND 141858);
SET INTEGRITY FOR CUSTOMER111 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER112.ddl

```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR CUSTOMER112 OFF;
ALTER TABLE CUSTOMER112 DROP CONSTRAINT CUSTOMER112CKC;
ALTER TABLE CUSTOMER112 ADD CONSTRAINT CUSTOMER112CKC CHECK
(C_W_ID BETWEEN 141859 AND 143136);
SET INTEGRITY FOR CUSTOMER112 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER113.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER113 OFF;
ALTER TABLE CUSTOMER113 DROP CONSTRAINT CUSTOMER113CKC;
ALTER TABLE CUSTOMER113 ADD CONSTRAINT CUSTOMER113CKC CHECK
(C_W_ID BETWEEN 143137 AND 144414);
SET INTEGRITY FOR CUSTOMER113 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER114.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER114 OFF;
ALTER TABLE CUSTOMER114 DROP CONSTRAINT CUSTOMER114CKC;
ALTER TABLE CUSTOMER114 ADD CONSTRAINT CUSTOMER114CKC CHECK
(C_W_ID BETWEEN 144415 AND 145692);
SET INTEGRITY FOR CUSTOMER114 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER115.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER115 OFF;
ALTER TABLE CUSTOMER115 DROP CONSTRAINT CUSTOMER115CKC;
ALTER TABLE CUSTOMER115 ADD CONSTRAINT CUSTOMER115CKC CHECK
(C_W_ID BETWEEN 145693 AND 146970);
SET INTEGRITY FOR CUSTOMER115 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER116.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER116 OFF;
ALTER TABLE CUSTOMER116 DROP CONSTRAINT CUSTOMER116CKC;
ALTER TABLE CUSTOMER116 ADD CONSTRAINT CUSTOMER116CKC CHECK
(C_W_ID BETWEEN 146971 AND 148248);
SET INTEGRITY FOR CUSTOMER116 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER117.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER117 OFF;
ALTER TABLE CUSTOMER117 DROP CONSTRAINT CUSTOMER117CKC;
ALTER TABLE CUSTOMER117 ADD CONSTRAINT CUSTOMER117CKC CHECK
(C_W_ID BETWEEN 148249 AND 149526);
SET INTEGRITY FOR CUSTOMER117 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER118.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER118 OFF;
ALTER TABLE CUSTOMER118 DROP CONSTRAINT CUSTOMER118CKC;
ALTER TABLE CUSTOMER118 ADD CONSTRAINT CUSTOMER118CKC CHECK
(C_W_ID BETWEEN 149527 AND 150804);
SET INTEGRITY FOR CUSTOMER118 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER119.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER119 OFF;
ALTER TABLE CUSTOMER119 DROP CONSTRAINT CUSTOMER119CKC;
ALTER TABLE CUSTOMER119 ADD CONSTRAINT CUSTOMER119CKC CHECK
(C_W_ID BETWEEN 150805 AND 152082);
SET INTEGRITY FOR CUSTOMER119 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER11.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER11 OFF;
ALTER TABLE CUSTOMER11 DROP CONSTRAINT CUSTOMER11CKC;
ALTER TABLE CUSTOMER11 ADD CONSTRAINT CUSTOMER11CKC CHECK
(C_W_ID BETWEEN 12781 AND 14058);
SET INTEGRITY FOR CUSTOMER11 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER120.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER120 OFF;
ALTER TABLE CUSTOMER120 DROP CONSTRAINT CUSTOMER120CKC;
ALTER TABLE CUSTOMER120 ADD CONSTRAINT CUSTOMER120CKC CHECK
(C_W_ID BETWEEN 152083 AND 153360);
SET INTEGRITY FOR CUSTOMER120 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER121.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER121 OFF;
ALTER TABLE CUSTOMER121 DROP CONSTRAINT CUSTOMER121CKC;
ALTER TABLE CUSTOMER121 ADD CONSTRAINT CUSTOMER121CKC CHECK
(C_W_ID BETWEEN 153361 AND 154638);
SET INTEGRITY FOR CUSTOMER121 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER122.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER122 OFF;
ALTER TABLE CUSTOMER122 DROP CONSTRAINT CUSTOMER122CKC;
ALTER TABLE CUSTOMER122 ADD CONSTRAINT CUSTOMER122CKC CHECK
(C_W_ID BETWEEN 154639 AND 155916);
SET INTEGRITY FOR CUSTOMER122 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER123.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER123 OFF;
ALTER TABLE CUSTOMER123 DROP CONSTRAINT CUSTOMER123CKC;
ALTER TABLE CUSTOMER123 ADD CONSTRAINT CUSTOMER123CKC CHECK
(C_W_ID BETWEEN 155917 AND 157194);
SET INTEGRITY FOR CUSTOMER123 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER124.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER124 OFF;
ALTER TABLE CUSTOMER124 DROP CONSTRAINT CUSTOMER124CKC;
ALTER TABLE CUSTOMER124 ADD CONSTRAINT CUSTOMER124CKC CHECK
(C_W_ID BETWEEN 157195 AND 158472);
SET INTEGRITY FOR CUSTOMER124 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER125.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER125 OFF;
ALTER TABLE CUSTOMER125 DROP CONSTRAINT CUSTOMER125CKC;
ALTER TABLE CUSTOMER125 ADD CONSTRAINT CUSTOMER125CKC CHECK
(C_W_ID BETWEEN 158473 AND 159750);
SET INTEGRITY FOR CUSTOMER125 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER126.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER126 OFF;
ALTER TABLE CUSTOMER126 DROP CONSTRAINT CUSTOMER126CKC;
ALTER TABLE CUSTOMER126 ADD CONSTRAINT CUSTOMER126CKC CHECK
(C_W_ID BETWEEN 159751 AND 161028);
SET INTEGRITY FOR CUSTOMER126 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER127.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER127 OFF;
ALTER TABLE CUSTOMER127 DROP CONSTRAINT CUSTOMER127CKC;
ALTER TABLE CUSTOMER127 ADD CONSTRAINT CUSTOMER127CKC CHECK
(C_W_ID BETWEEN 161029 AND 162306);
SET INTEGRITY FOR CUSTOMER127 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER128.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER128 OFF;
ALTER TABLE CUSTOMER128 DROP CONSTRAINT CUSTOMER128CKC;
ALTER TABLE CUSTOMER128 ADD CONSTRAINT CUSTOMER128CKC CHECK
(C_W_ID BETWEEN 162307 AND 163584);
SET INTEGRITY FOR CUSTOMER128 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER129.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER129 OFF;
ALTER TABLE CUSTOMER129 DROP CONSTRAINT CUSTOMER129CKC;
ALTER TABLE CUSTOMER129 ADD CONSTRAINT CUSTOMER129CKC CHECK
(C_W_ID BETWEEN 163585 AND 164862);
SET INTEGRITY FOR CUSTOMER129 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER12.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER12 OFF;
ALTER TABLE CUSTOMER12 DROP CONSTRAINT CUSTOMER12CKC;
ALTER TABLE CUSTOMER12 ADD CONSTRAINT CUSTOMER12CKC CHECK
(C_W_ID BETWEEN 14059 AND 15336);
```

```
SET INTEGRITY FOR CUSTOMER12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER130.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER130 OFF;
ALTER TABLE CUSTOMER130 DROP CONSTRAINT CUSTOMER130CKC;
ALTER TABLE CUSTOMER130 ADD CONSTRAINT CUSTOMER130CKC CHECK
(C_W_ID BETWEEN 164863 AND 166140);
SET INTEGRITY FOR CUSTOMER130 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER131.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER131 OFF;
ALTER TABLE CUSTOMER131 DROP CONSTRAINT CUSTOMER131CKC;
ALTER TABLE CUSTOMER131 ADD CONSTRAINT CUSTOMER131CKC CHECK
(C_W_ID BETWEEN 166141 AND 167418);
SET INTEGRITY FOR CUSTOMER131 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER132.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER132 OFF;
ALTER TABLE CUSTOMER132 DROP CONSTRAINT CUSTOMER132CKC;
ALTER TABLE CUSTOMER132 ADD CONSTRAINT CUSTOMER132CKC CHECK
(C_W_ID BETWEEN 167419 AND 168696);
SET INTEGRITY FOR CUSTOMER132 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER133.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER133 OFF;
ALTER TABLE CUSTOMER133 DROP CONSTRAINT CUSTOMER133CKC;
ALTER TABLE CUSTOMER133 ADD CONSTRAINT CUSTOMER133CKC CHECK
(C_W_ID BETWEEN 168697 AND 169974);
SET INTEGRITY FOR CUSTOMER133 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER134.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER134 OFF;
ALTER TABLE CUSTOMER134 DROP CONSTRAINT CUSTOMER134CKC;
ALTER TABLE CUSTOMER134 ADD CONSTRAINT CUSTOMER134CKC CHECK
(C_W_ID BETWEEN 169975 AND 171252);
SET INTEGRITY FOR CUSTOMER134 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER135.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER135 OFF;
ALTER TABLE CUSTOMER135 DROP CONSTRAINT CUSTOMER135CKC;
ALTER TABLE CUSTOMER135 ADD CONSTRAINT CUSTOMER135CKC CHECK
(C_W_ID BETWEEN 171253 AND 172530);
SET INTEGRITY FOR CUSTOMER135 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER136.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER136 OFF;
ALTER TABLE CUSTOMER136 DROP CONSTRAINT CUSTOMER136CKC;
ALTER TABLE CUSTOMER136 ADD CONSTRAINT CUSTOMER136CKC CHECK
(C_W_ID BETWEEN 172531 AND 173808);
SET INTEGRITY FOR CUSTOMER136 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER137.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER137 OFF;
ALTER TABLE CUSTOMER137 DROP CONSTRAINT CUSTOMER137CKC;
ALTER TABLE CUSTOMER137 ADD CONSTRAINT CUSTOMER137CKC CHECK
(C_W_ID BETWEEN 173809 AND 175086);
SET INTEGRITY FOR CUSTOMER137 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER138.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER138 OFF;
ALTER TABLE CUSTOMER138 DROP CONSTRAINT CUSTOMER138CKC;
ALTER TABLE CUSTOMER138 ADD CONSTRAINT CUSTOMER138CKC CHECK
(C_W_ID BETWEEN 175087 AND 176364);
SET INTEGRITY FOR CUSTOMER138 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER139.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER139 OFF;
ALTER TABLE CUSTOMER139 DROP CONSTRAINT CUSTOMER139CKC;
ALTER TABLE CUSTOMER139 ADD CONSTRAINT CUSTOMER139CKC CHECK
(C_W_ID BETWEEN 176365 AND 177642);
SET INTEGRITY FOR CUSTOMER139 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER13.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER13 OFF;
ALTER TABLE CUSTOMER13 DROP CONSTRAINT CUSTOMER13CKC;
ALTER TABLE CUSTOMER13 ADD CONSTRAINT CUSTOMER13CKC CHECK
(C_W_ID BETWEEN 15337 AND 16614);
SET INTEGRITY FOR CUSTOMER13 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER140.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER140 OFF;
ALTER TABLE CUSTOMER140 DROP CONSTRAINT CUSTOMER140CKC;
ALTER TABLE CUSTOMER140 ADD CONSTRAINT CUSTOMER140CKC CHECK
(C_W_ID BETWEEN 177643 AND 178920);
SET INTEGRITY FOR CUSTOMER140 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER141.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER141 OFF;
ALTER TABLE CUSTOMER141 DROP CONSTRAINT CUSTOMER141CKC;
ALTER TABLE CUSTOMER141 ADD CONSTRAINT CUSTOMER141CKC CHECK
(C_W_ID BETWEEN 178921 AND 180198);
SET INTEGRITY FOR CUSTOMER141 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER142.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER142 OFF;
ALTER TABLE CUSTOMER142 DROP CONSTRAINT CUSTOMER142CKC;
ALTER TABLE CUSTOMER142 ADD CONSTRAINT CUSTOMER142CKC CHECK
(C_W_ID BETWEEN 180199 AND 181476);
SET INTEGRITY FOR CUSTOMER142 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER143.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER143 OFF;
ALTER TABLE CUSTOMER143 DROP CONSTRAINT CUSTOMER143CKC;
ALTER TABLE CUSTOMER143 ADD CONSTRAINT CUSTOMER143CKC CHECK
(C_W_ID BETWEEN 181477 AND 182754);
SET INTEGRITY FOR CUSTOMER143 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER144.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER144 OFF;
ALTER TABLE CUSTOMER144 DROP CONSTRAINT CUSTOMER144CKC;
ALTER TABLE CUSTOMER144 ADD CONSTRAINT CUSTOMER144CKC CHECK
(C_W_ID >= 182755);
SET INTEGRITY FOR CUSTOMER144 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER14.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER14 OFF;
ALTER TABLE CUSTOMER14 DROP CONSTRAINT CUSTOMER14CKC;
ALTER TABLE CUSTOMER14 ADD CONSTRAINT CUSTOMER14CKC CHECK
(C_W_ID BETWEEN 16615 AND 17892);
SET INTEGRITY FOR CUSTOMER14 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER15.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER15 OFF;
ALTER TABLE CUSTOMER15 DROP CONSTRAINT CUSTOMER15CKC;
ALTER TABLE CUSTOMER15 ADD CONSTRAINT CUSTOMER15CKC CHECK
(C_W_ID BETWEEN 17893 AND 19170);
SET INTEGRITY FOR CUSTOMER15 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER16.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER16 OFF;
ALTER TABLE CUSTOMER16 DROP CONSTRAINT CUSTOMER16CKC;
ALTER TABLE CUSTOMER16 ADD CONSTRAINT CUSTOMER16CKC CHECK
(C_W_ID BETWEEN 19171 AND 20448);
SET INTEGRITY FOR CUSTOMER16 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER17.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER17 OFF;
ALTER TABLE CUSTOMER17 DROP CONSTRAINT CUSTOMER17CKC;
ALTER TABLE CUSTOMER17 ADD CONSTRAINT CUSTOMER17CKC CHECK
(C_W_ID BETWEEN 20449 AND 21726);
SET INTEGRITY FOR CUSTOMER17 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER18.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER18 OFF;
ALTER TABLE CUSTOMER18 DROP CONSTRAINT CUSTOMER18CKC;
ALTER TABLE CUSTOMER18 ADD CONSTRAINT CUSTOMER18CKC CHECK
(C_W_ID BETWEEN 21727 AND 23004);
SET INTEGRITY FOR CUSTOMER18 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER19.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER19 OFF;
ALTER TABLE CUSTOMER19 DROP CONSTRAINT CUSTOMER19CKC;
ALTER TABLE CUSTOMER19 ADD CONSTRAINT CUSTOMER19CKC CHECK
(C_W_ID BETWEEN 23005 AND 24282);
SET INTEGRITY FOR CUSTOMER19 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER1.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER1 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID
BETWEEN 1 AND 1278);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER20.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER20 OFF;
ALTER TABLE CUSTOMER20 DROP CONSTRAINT CUSTOMER20CKC;
ALTER TABLE CUSTOMER20 ADD CONSTRAINT CUSTOMER20CKC CHECK
(C_W_ID BETWEEN 24283 AND 25560);
SET INTEGRITY FOR CUSTOMER20 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER21.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER21 OFF;
ALTER TABLE CUSTOMER21 DROP CONSTRAINT CUSTOMER21CKC;
ALTER TABLE CUSTOMER21 ADD CONSTRAINT CUSTOMER21CKC CHECK
(C_W_ID BETWEEN 25561 AND 26838);
SET INTEGRITY FOR CUSTOMER21 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER22.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER22 OFF;
ALTER TABLE CUSTOMER22 DROP CONSTRAINT CUSTOMER22CKC;
ALTER TABLE CUSTOMER22 ADD CONSTRAINT CUSTOMER22CKC CHECK
(C_W_ID BETWEEN 26839 AND 28116);
SET INTEGRITY FOR CUSTOMER22 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER23.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER23 OFF;
ALTER TABLE CUSTOMER23 DROP CONSTRAINT CUSTOMER23CKC;
ALTER TABLE CUSTOMER23 ADD CONSTRAINT CUSTOMER23CKC CHECK
(C_W_ID BETWEEN 28117 AND 29394);
SET INTEGRITY FOR CUSTOMER23 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER24.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER24 OFF;
ALTER TABLE CUSTOMER24 DROP CONSTRAINT CUSTOMER24CKC;
ALTER TABLE CUSTOMER24 ADD CONSTRAINT CUSTOMER24CKC CHECK
(C_W_ID BETWEEN 29395 AND 30672);
SET INTEGRITY FOR CUSTOMER24 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER25.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER25 OFF;
ALTER TABLE CUSTOMER25 DROP CONSTRAINT CUSTOMER25CKC;
ALTER TABLE CUSTOMER25 ADD CONSTRAINT CUSTOMER25CKC CHECK
(C_W_ID BETWEEN 30673 AND 31950);
SET INTEGRITY FOR CUSTOMER25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER26.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER26 OFF;
ALTER TABLE CUSTOMER26 DROP CONSTRAINT CUSTOMER26CKC;
ALTER TABLE CUSTOMER26 ADD CONSTRAINT CUSTOMER26CKC CHECK
(C_W_ID BETWEEN 31951 AND 33228);
SET INTEGRITY FOR CUSTOMER26 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER27.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER27 OFF;
ALTER TABLE CUSTOMER27 DROP CONSTRAINT CUSTOMER27CKC;
ALTER TABLE CUSTOMER27 ADD CONSTRAINT CUSTOMER27CKC CHECK
(C_W_ID BETWEEN 33229 AND 34506);
SET INTEGRITY FOR CUSTOMER27 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER28.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER28 OFF;
ALTER TABLE CUSTOMER28 DROP CONSTRAINT CUSTOMER28CKC;
ALTER TABLE CUSTOMER28 ADD CONSTRAINT CUSTOMER28CKC CHECK
(C_W_ID BETWEEN 34507 AND 35784);
SET INTEGRITY FOR CUSTOMER28 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER29.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER29 OFF;
ALTER TABLE CUSTOMER29 DROP CONSTRAINT CUSTOMER29CKC;
ALTER TABLE CUSTOMER29 ADD CONSTRAINT CUSTOMER29CKC CHECK
(C_W_ID BETWEEN 35785 AND 37062);
SET INTEGRITY FOR CUSTOMER29 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER2.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER2 DROP CONSTRAINT CUSTOMER2CKC;
ALTER TABLE CUSTOMER2 ADD CONSTRAINT CUSTOMER2CKC CHECK (C_W_ID
BETWEEN 1279 AND 2556);
SET INTEGRITY FOR CUSTOMER2 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER30.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER30 OFF;
ALTER TABLE CUSTOMER30 DROP CONSTRAINT CUSTOMER30CKC;
ALTER TABLE CUSTOMER30 ADD CONSTRAINT CUSTOMER30CKC CHECK
(C_W_ID BETWEEN 37063 AND 38340);
SET INTEGRITY FOR CUSTOMER30 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER31.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER31 OFF;
ALTER TABLE CUSTOMER31 DROP CONSTRAINT CUSTOMER31CKC;
ALTER TABLE CUSTOMER31 ADD CONSTRAINT CUSTOMER31CKC CHECK
(C_W_ID BETWEEN 38341 AND 39618);
SET INTEGRITY FOR CUSTOMER31 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER32.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER32 OFF;
ALTER TABLE CUSTOMER32 DROP CONSTRAINT CUSTOMER32CKC;
ALTER TABLE CUSTOMER32 ADD CONSTRAINT CUSTOMER32CKC CHECK
(C_W_ID BETWEEN 39619 AND 40896);
SET INTEGRITY FOR CUSTOMER32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER33.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER33 OFF;
ALTER TABLE CUSTOMER33 DROP CONSTRAINT CUSTOMER33CKC;
ALTER TABLE CUSTOMER33 ADD CONSTRAINT CUSTOMER33CKC CHECK
(C_W_ID BETWEEN 40897 AND 42174);
SET INTEGRITY FOR CUSTOMER33 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER34.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR CUSTOMER34 OFF;
ALTER TABLE CUSTOMER34 DROP CONSTRAINT CUSTOMER34CKC;
ALTER TABLE CUSTOMER34 ADD CONSTRAINT CUSTOMER34CKC CHECK
(C_W_ID BETWEEN 42175 AND 43452);
SET INTEGRITY FOR CUSTOMER34 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER35.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER35 OFF;
ALTER TABLE CUSTOMER35 DROP CONSTRAINT CUSTOMER35CKC;
ALTER TABLE CUSTOMER35 ADD CONSTRAINT CUSTOMER35CKC CHECK
(C_W_ID BETWEEN 43453 AND 44730);
SET INTEGRITY FOR CUSTOMER35 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER36.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER36 OFF;
ALTER TABLE CUSTOMER36 DROP CONSTRAINT CUSTOMER36CKC;
ALTER TABLE CUSTOMER36 ADD CONSTRAINT CUSTOMER36CKC CHECK
(C_W_ID BETWEEN 44731 AND 46008);
SET INTEGRITY FOR CUSTOMER36 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER37.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER37 OFF;
ALTER TABLE CUSTOMER37 DROP CONSTRAINT CUSTOMER37CKC;
ALTER TABLE CUSTOMER37 ADD CONSTRAINT CUSTOMER37CKC CHECK
(C_W_ID BETWEEN 46009 AND 47286);
SET INTEGRITY FOR CUSTOMER37 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER38.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER38 OFF;
ALTER TABLE CUSTOMER38 DROP CONSTRAINT CUSTOMER38CKC;
ALTER TABLE CUSTOMER38 ADD CONSTRAINT CUSTOMER38CKC CHECK
(C_W_ID BETWEEN 47287 AND 48564);
SET INTEGRITY FOR CUSTOMER38 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER39.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER39 OFF;
ALTER TABLE CUSTOMER39 DROP CONSTRAINT CUSTOMER39CKC;
ALTER TABLE CUSTOMER39 ADD CONSTRAINT CUSTOMER39CKC CHECK
(C_W_ID BETWEEN 48565 AND 49842);
SET INTEGRITY FOR CUSTOMER39 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER3.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER3 OFF;
ALTER TABLE CUSTOMER3 DROP CONSTRAINT CUSTOMER3CKC;
ALTER TABLE CUSTOMER3 ADD CONSTRAINT CUSTOMER3CKC CHECK (C_W_ID
BETWEEN 2557 AND 3834);
SET INTEGRITY FOR CUSTOMER3 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER40.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER40 OFF;
ALTER TABLE CUSTOMER40 DROP CONSTRAINT CUSTOMER40CKC;
ALTER TABLE CUSTOMER40 ADD CONSTRAINT CUSTOMER40CKC CHECK
(C_W_ID BETWEEN 49843 AND 51120);
SET INTEGRITY FOR CUSTOMER40 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER41.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER41 OFF;
ALTER TABLE CUSTOMER41 DROP CONSTRAINT CUSTOMER41CKC;
ALTER TABLE CUSTOMER41 ADD CONSTRAINT CUSTOMER41CKC CHECK
(C_W_ID BETWEEN 51121 AND 52398);
SET INTEGRITY FOR CUSTOMER41 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER42.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER42 OFF;
ALTER TABLE CUSTOMER42 DROP CONSTRAINT CUSTOMER42CKC;
ALTER TABLE CUSTOMER42 ADD CONSTRAINT CUSTOMER42CKC CHECK
(C_W_ID BETWEEN 52399 AND 53676);
SET INTEGRITY FOR CUSTOMER42 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER43.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER43 OFF;
ALTER TABLE CUSTOMER43 DROP CONSTRAINT CUSTOMER43CKC;
ALTER TABLE CUSTOMER43 ADD CONSTRAINT CUSTOMER43CKC CHECK
(C_W_ID BETWEEN 53677 AND 54954);
SET INTEGRITY FOR CUSTOMER43 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER44.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER44 OFF;
ALTER TABLE CUSTOMER44 DROP CONSTRAINT CUSTOMER44CKC;
ALTER TABLE CUSTOMER44 ADD CONSTRAINT CUSTOMER44CKC CHECK
(C_W_ID BETWEEN 54955 AND 56232);
SET INTEGRITY FOR CUSTOMER44 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER45.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER45 OFF;
ALTER TABLE CUSTOMER45 DROP CONSTRAINT CUSTOMER45CKC;
ALTER TABLE CUSTOMER45 ADD CONSTRAINT CUSTOMER45CKC CHECK
(C_W_ID BETWEEN 56233 AND 57510);
SET INTEGRITY FOR CUSTOMER45 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER46.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER46 OFF;
ALTER TABLE CUSTOMER46 DROP CONSTRAINT CUSTOMER46CKC;
ALTER TABLE CUSTOMER46 ADD CONSTRAINT CUSTOMER46CKC CHECK
(C_W_ID BETWEEN 57511 AND 58788);
SET INTEGRITY FOR CUSTOMER46 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER47.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER47 OFF;
ALTER TABLE CUSTOMER47 DROP CONSTRAINT CUSTOMER47CKC;
ALTER TABLE CUSTOMER47 ADD CONSTRAINT CUSTOMER47CKC CHECK
(C_W_ID BETWEEN 58789 AND 60066);
SET INTEGRITY FOR CUSTOMER47 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER48.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER48 OFF;
ALTER TABLE CUSTOMER48 DROP CONSTRAINT CUSTOMER48CKC;
ALTER TABLE CUSTOMER48 ADD CONSTRAINT CUSTOMER48CKC CHECK
(C_W_ID BETWEEN 60067 AND 61344);
SET INTEGRITY FOR CUSTOMER48 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER49.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER49 OFF;
ALTER TABLE CUSTOMER49 DROP CONSTRAINT CUSTOMER49CKC;
ALTER TABLE CUSTOMER49 ADD CONSTRAINT CUSTOMER49CKC CHECK
(C_W_ID BETWEEN 61345 AND 62622);
SET INTEGRITY FOR CUSTOMER49 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER4.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER4 OFF;
ALTER TABLE CUSTOMER4 DROP CONSTRAINT CUSTOMER4CKC;
ALTER TABLE CUSTOMER4 ADD CONSTRAINT CUSTOMER4CKC CHECK (C_W_ID
BETWEEN 3835 AND 5112);
SET INTEGRITY FOR CUSTOMER4 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER50.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER50 OFF;
ALTER TABLE CUSTOMER50 DROP CONSTRAINT CUSTOMER50CKC;
ALTER TABLE CUSTOMER50 ADD CONSTRAINT CUSTOMER50CKC CHECK
(C_W_ID BETWEEN 62623 AND 63900);
SET INTEGRITY FOR CUSTOMER50 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER51.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER51 OFF;
ALTER TABLE CUSTOMER51 DROP CONSTRAINT CUSTOMER51CKC;
ALTER TABLE CUSTOMER51 ADD CONSTRAINT CUSTOMER51CKC CHECK
(C_W_ID BETWEEN 63901 AND 65178);
```

```
SET INTEGRITY FOR CUSTOMER51 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER52.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER52 OFF;
ALTER TABLE CUSTOMER52 DROP CONSTRAINT CUSTOMER52CKC;
ALTER TABLE CUSTOMER52 ADD CONSTRAINT CUSTOMER52CKC CHECK
(C_W_ID BETWEEN 65179 AND 66456);
SET INTEGRITY FOR CUSTOMER52 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER53.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER53 OFF;
ALTER TABLE CUSTOMER53 DROP CONSTRAINT CUSTOMER53CKC;
ALTER TABLE CUSTOMER53 ADD CONSTRAINT CUSTOMER53CKC CHECK
(C_W_ID BETWEEN 66457 AND 67734);
SET INTEGRITY FOR CUSTOMER53 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER54.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER54 OFF;
ALTER TABLE CUSTOMER54 DROP CONSTRAINT CUSTOMER54CKC;
ALTER TABLE CUSTOMER54 ADD CONSTRAINT CUSTOMER54CKC CHECK
(C_W_ID BETWEEN 67735 AND 69012);
SET INTEGRITY FOR CUSTOMER54 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER55.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER55 OFF;
ALTER TABLE CUSTOMER55 DROP CONSTRAINT CUSTOMER55CKC;
ALTER TABLE CUSTOMER55 ADD CONSTRAINT CUSTOMER55CKC CHECK
(C_W_ID BETWEEN 69013 AND 70290);
SET INTEGRITY FOR CUSTOMER55 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER56.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER56 OFF;
ALTER TABLE CUSTOMER56 DROP CONSTRAINT CUSTOMER56CKC;
ALTER TABLE CUSTOMER56 ADD CONSTRAINT CUSTOMER56CKC CHECK
(C_W_ID BETWEEN 70291 AND 71568);
SET INTEGRITY FOR CUSTOMER56 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER57.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER57 OFF;
ALTER TABLE CUSTOMER57 DROP CONSTRAINT CUSTOMER57CKC;
ALTER TABLE CUSTOMER57 ADD CONSTRAINT CUSTOMER57CKC CHECK
(C_W_ID BETWEEN 71569 AND 72846);
SET INTEGRITY FOR CUSTOMER57 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER58.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER58 OFF;
ALTER TABLE CUSTOMER58 DROP CONSTRAINT CUSTOMER58CKC;
ALTER TABLE CUSTOMER58 ADD CONSTRAINT CUSTOMER58CKC CHECK
(C_W_ID BETWEEN 72847 AND 74124);
SET INTEGRITY FOR CUSTOMER58 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER59.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER59 OFF;
ALTER TABLE CUSTOMER59 DROP CONSTRAINT CUSTOMER59CKC;
ALTER TABLE CUSTOMER59 ADD CONSTRAINT CUSTOMER59CKC CHECK
(C_W_ID BETWEEN 74125 AND 75402);
SET INTEGRITY FOR CUSTOMER59 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER5.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER5 OFF;
ALTER TABLE CUSTOMER5 DROP CONSTRAINT CUSTOMER5CKC;
ALTER TABLE CUSTOMER5 ADD CONSTRAINT CUSTOMER5CKC CHECK (C_W_ID
BETWEEN 5113 AND 6390);
SET INTEGRITY FOR CUSTOMER5 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER60.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER60 OFF;
ALTER TABLE CUSTOMER60 DROP CONSTRAINT CUSTOMER60CKC;
ALTER TABLE CUSTOMER60 ADD CONSTRAINT CUSTOMER60CKC CHECK
(C_W_ID BETWEEN 75403 AND 76680);
SET INTEGRITY FOR CUSTOMER60 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER61.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER61 OFF;
ALTER TABLE CUSTOMER61 DROP CONSTRAINT CUSTOMER61CKC;
ALTER TABLE CUSTOMER61 ADD CONSTRAINT CUSTOMER61CKC CHECK
(C_W_ID BETWEEN 76681 AND 77958);
SET INTEGRITY FOR CUSTOMER61 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER62.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER62 OFF;
ALTER TABLE CUSTOMER62 DROP CONSTRAINT CUSTOMER62CKC;
ALTER TABLE CUSTOMER62 ADD CONSTRAINT CUSTOMER62CKC CHECK
(C_W_ID BETWEEN 77959 AND 79236);
SET INTEGRITY FOR CUSTOMER62 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER63.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER63 OFF;
ALTER TABLE CUSTOMER63 DROP CONSTRAINT CUSTOMER63CKC;
ALTER TABLE CUSTOMER63 ADD CONSTRAINT CUSTOMER63CKC CHECK
(C_W_ID BETWEEN 79237 AND 80514);
SET INTEGRITY FOR CUSTOMER63 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER64.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER64 OFF;
ALTER TABLE CUSTOMER64 DROP CONSTRAINT CUSTOMER64CKC;
ALTER TABLE CUSTOMER64 ADD CONSTRAINT CUSTOMER64CKC CHECK
(C_W_ID BETWEEN 80515 AND 81792);
SET INTEGRITY FOR CUSTOMER64 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER65.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER65 OFF;
ALTER TABLE CUSTOMER65 DROP CONSTRAINT CUSTOMER65CKC;
ALTER TABLE CUSTOMER65 ADD CONSTRAINT CUSTOMER65CKC CHECK
(C_W_ID BETWEEN 81793 AND 83070);
SET INTEGRITY FOR CUSTOMER65 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER66.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER66 OFF;
ALTER TABLE CUSTOMER66 DROP CONSTRAINT CUSTOMER66CKC;
ALTER TABLE CUSTOMER66 ADD CONSTRAINT CUSTOMER66CKC CHECK
(C_W_ID BETWEEN 83071 AND 84348);
SET INTEGRITY FOR CUSTOMER66 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER67.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER67 OFF;
ALTER TABLE CUSTOMER67 DROP CONSTRAINT CUSTOMER67CKC;
ALTER TABLE CUSTOMER67 ADD CONSTRAINT CUSTOMER67CKC CHECK
(C_W_ID BETWEEN 84349 AND 85626);
SET INTEGRITY FOR CUSTOMER67 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER68.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER68 OFF;
ALTER TABLE CUSTOMER68 DROP CONSTRAINT CUSTOMER68CKC;
ALTER TABLE CUSTOMER68 ADD CONSTRAINT CUSTOMER68CKC CHECK
(C_W_ID BETWEEN 85627 AND 86904);
SET INTEGRITY FOR CUSTOMER68 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER69.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER69 OFF;
ALTER TABLE CUSTOMER69 DROP CONSTRAINT CUSTOMER69CKC;
ALTER TABLE CUSTOMER69 ADD CONSTRAINT CUSTOMER69CKC CHECK
(C_W_ID BETWEEN 86905 AND 88182);
SET INTEGRITY FOR CUSTOMER69 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER6.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER6 OFF;
ALTER TABLE CUSTOMER6 DROP CONSTRAINT CUSTOMER6CKC;
ALTER TABLE CUSTOMER6 ADD CONSTRAINT CUSTOMER6CKC CHECK (C_W_ID
BETWEEN 6391 AND 7668);
SET INTEGRITY FOR CUSTOMER6 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER70.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER70 OFF;
ALTER TABLE CUSTOMER70 DROP CONSTRAINT CUSTOMER70CKC;
ALTER TABLE CUSTOMER70 ADD CONSTRAINT CUSTOMER70CKC CHECK
(C_W_ID BETWEEN 88183 AND 89460);
SET INTEGRITY FOR CUSTOMER70 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER71.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER71 OFF;
ALTER TABLE CUSTOMER71 DROP CONSTRAINT CUSTOMER71CKC;
ALTER TABLE CUSTOMER71 ADD CONSTRAINT CUSTOMER71CKC CHECK
(C_W_ID BETWEEN 89461 AND 90738);
SET INTEGRITY FOR CUSTOMER71 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER72.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER72 OFF;
ALTER TABLE CUSTOMER72 DROP CONSTRAINT CUSTOMER72CKC;
ALTER TABLE CUSTOMER72 ADD CONSTRAINT CUSTOMER72CKC CHECK
(C_W_ID BETWEEN 90739 AND 92016);
SET INTEGRITY FOR CUSTOMER72 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER73.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER73 OFF;
ALTER TABLE CUSTOMER73 DROP CONSTRAINT CUSTOMER73CKC;
ALTER TABLE CUSTOMER73 ADD CONSTRAINT CUSTOMER73CKC CHECK
(C_W_ID BETWEEN 92017 AND 93294);
SET INTEGRITY FOR CUSTOMER73 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER74.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER74 OFF;
ALTER TABLE CUSTOMER74 DROP CONSTRAINT CUSTOMER74CKC;
ALTER TABLE CUSTOMER74 ADD CONSTRAINT CUSTOMER74CKC CHECK
(C_W_ID BETWEEN 93295 AND 94572);
SET INTEGRITY FOR CUSTOMER74 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER75.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER75 OFF;
ALTER TABLE CUSTOMER75 DROP CONSTRAINT CUSTOMER75CKC;
ALTER TABLE CUSTOMER75 ADD CONSTRAINT CUSTOMER75CKC CHECK
(C_W_ID BETWEEN 94573 AND 95850);
SET INTEGRITY FOR CUSTOMER75 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER76.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER76 OFF;
ALTER TABLE CUSTOMER76 DROP CONSTRAINT CUSTOMER76CKC;
ALTER TABLE CUSTOMER76 ADD CONSTRAINT CUSTOMER76CKC CHECK
(C_W_ID BETWEEN 95851 AND 97128);
SET INTEGRITY FOR CUSTOMER76 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER77.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER77 OFF;
ALTER TABLE CUSTOMER77 DROP CONSTRAINT CUSTOMER77CKC;
ALTER TABLE CUSTOMER77 ADD CONSTRAINT CUSTOMER77CKC CHECK
(C_W_ID BETWEEN 97129 AND 98406);
SET INTEGRITY FOR CUSTOMER77 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER78.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER78 OFF;
ALTER TABLE CUSTOMER78 DROP CONSTRAINT CUSTOMER78CKC;
ALTER TABLE CUSTOMER78 ADD CONSTRAINT CUSTOMER78CKC CHECK
(C_W_ID BETWEEN 98407 AND 99684);
SET INTEGRITY FOR CUSTOMER78 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER79.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER79 OFF;
ALTER TABLE CUSTOMER79 DROP CONSTRAINT CUSTOMER79CKC;
ALTER TABLE CUSTOMER79 ADD CONSTRAINT CUSTOMER79CKC CHECK
(C_W_ID BETWEEN 99685 AND 100962);
SET INTEGRITY FOR CUSTOMER79 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER7.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER7 OFF;
ALTER TABLE CUSTOMER7 DROP CONSTRAINT CUSTOMER7CKC;
ALTER TABLE CUSTOMER7 ADD CONSTRAINT CUSTOMER7CKC CHECK (C_W_ID
BETWEEN 7669 AND 8946);
SET INTEGRITY FOR CUSTOMER7 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER80.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER80 OFF;
ALTER TABLE CUSTOMER80 DROP CONSTRAINT CUSTOMER80CKC;
ALTER TABLE CUSTOMER80 ADD CONSTRAINT CUSTOMER80CKC CHECK
(C_W_ID BETWEEN 100963 AND 102240);
SET INTEGRITY FOR CUSTOMER80 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER81.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER81 OFF;
ALTER TABLE CUSTOMER81 DROP CONSTRAINT CUSTOMER81CKC;
ALTER TABLE CUSTOMER81 ADD CONSTRAINT CUSTOMER81CKC CHECK
(C_W_ID BETWEEN 102241 AND 103518);
SET INTEGRITY FOR CUSTOMER81 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER82.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER82 OFF;
ALTER TABLE CUSTOMER82 DROP CONSTRAINT CUSTOMER82CKC;
ALTER TABLE CUSTOMER82 ADD CONSTRAINT CUSTOMER82CKC CHECK
(C_W_ID BETWEEN 103519 AND 104796);
SET INTEGRITY FOR CUSTOMER82 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER83.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER83 OFF;
ALTER TABLE CUSTOMER83 DROP CONSTRAINT CUSTOMER83CKC;
ALTER TABLE CUSTOMER83 ADD CONSTRAINT CUSTOMER83CKC CHECK
(C_W_ID BETWEEN 104797 AND 106074);
SET INTEGRITY FOR CUSTOMER83 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER84.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER84 OFF;
ALTER TABLE CUSTOMER84 DROP CONSTRAINT CUSTOMER84CKC;
ALTER TABLE CUSTOMER84 ADD CONSTRAINT CUSTOMER84CKC CHECK
(C_W_ID BETWEEN 106075 AND 107352);
SET INTEGRITY FOR CUSTOMER84 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER85.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER85 OFF;
ALTER TABLE CUSTOMER85 DROP CONSTRAINT CUSTOMER85CKC;
ALTER TABLE CUSTOMER85 ADD CONSTRAINT CUSTOMER85CKC CHECK
(C_W_ID BETWEEN 107353 AND 108630);
SET INTEGRITY FOR CUSTOMER85 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER86.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER86 OFF;
ALTER TABLE CUSTOMER86 DROP CONSTRAINT CUSTOMER86CKC;
ALTER TABLE CUSTOMER86 ADD CONSTRAINT CUSTOMER86CKC CHECK
(C_W_ID BETWEEN 108631 AND 109908);
SET INTEGRITY FOR CUSTOMER86 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER87.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR CUSTOMER87 OFF;
ALTER TABLE CUSTOMER87 DROP CONSTRAINT CUSTOMER87CKC;
ALTER TABLE CUSTOMER87 ADD CONSTRAINT CUSTOMER87CKC CHECK
(C_W_ID BETWEEN 109909 AND 111186);
SET INTEGRITY FOR CUSTOMER87 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER88.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER88 OFF;
ALTER TABLE CUSTOMER88 DROP CONSTRAINT CUSTOMER88CKC;
ALTER TABLE CUSTOMER88 ADD CONSTRAINT CUSTOMER88CKC CHECK
(C_W_ID BETWEEN 111187 AND 112464);
SET INTEGRITY FOR CUSTOMER88 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER89.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER89 OFF;
ALTER TABLE CUSTOMER89 DROP CONSTRAINT CUSTOMER89CKC;
ALTER TABLE CUSTOMER89 ADD CONSTRAINT CUSTOMER89CKC CHECK
(C_W_ID BETWEEN 112465 AND 113742);
SET INTEGRITY FOR CUSTOMER89 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER8.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER8 OFF;
ALTER TABLE CUSTOMER8 DROP CONSTRAINT CUSTOMER8CKC;
ALTER TABLE CUSTOMER8 ADD CONSTRAINT CUSTOMER8CKC CHECK (C_W_ID
BETWEEN 8947 AND 10224);
SET INTEGRITY FOR CUSTOMER8 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER90.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER90 OFF;
ALTER TABLE CUSTOMER90 DROP CONSTRAINT CUSTOMER90CKC;
ALTER TABLE CUSTOMER90 ADD CONSTRAINT CUSTOMER90CKC CHECK
(C_W_ID BETWEEN 113743 AND 115020);
SET INTEGRITY FOR CUSTOMER90 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER91.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER91 OFF;
ALTER TABLE CUSTOMER91 DROP CONSTRAINT CUSTOMER91CKC;
ALTER TABLE CUSTOMER91 ADD CONSTRAINT CUSTOMER91CKC CHECK
(C_W_ID BETWEEN 115021 AND 116298);
SET INTEGRITY FOR CUSTOMER91 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER92.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER92 OFF;
ALTER TABLE CUSTOMER92 DROP CONSTRAINT CUSTOMER92CKC;
ALTER TABLE CUSTOMER92 ADD CONSTRAINT CUSTOMER92CKC CHECK
(C_W_ID BETWEEN 116299 AND 117576);
SET INTEGRITY FOR CUSTOMER92 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER93.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER93 OFF;
ALTER TABLE CUSTOMER93 DROP CONSTRAINT CUSTOMER93CKC;
ALTER TABLE CUSTOMER93 ADD CONSTRAINT CUSTOMER93CKC CHECK
(C_W_ID BETWEEN 117577 AND 118854);
SET INTEGRITY FOR CUSTOMER93 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER94.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER94 OFF;
ALTER TABLE CUSTOMER94 DROP CONSTRAINT CUSTOMER94CKC;
ALTER TABLE CUSTOMER94 ADD CONSTRAINT CUSTOMER94CKC CHECK
(C_W_ID BETWEEN 118855 AND 120132);
SET INTEGRITY FOR CUSTOMER94 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER95.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER95 OFF;
ALTER TABLE CUSTOMER95 DROP CONSTRAINT CUSTOMER95CKC;
ALTER TABLE CUSTOMER95 ADD CONSTRAINT CUSTOMER95CKC CHECK
(C_W_ID BETWEEN 120133 AND 121410);
SET INTEGRITY FOR CUSTOMER95 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER96.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER96 OFF;
ALTER TABLE CUSTOMER96 DROP CONSTRAINT CUSTOMER96CKC;
ALTER TABLE CUSTOMER96 ADD CONSTRAINT CUSTOMER96CKC CHECK
(C_W_ID BETWEEN 121411 AND 122688);
SET INTEGRITY FOR CUSTOMER96 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER97.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER97 OFF;
ALTER TABLE CUSTOMER97 DROP CONSTRAINT CUSTOMER97CKC;
ALTER TABLE CUSTOMER97 ADD CONSTRAINT CUSTOMER97CKC CHECK
(C_W_ID BETWEEN 122689 AND 123966);
SET INTEGRITY FOR CUSTOMER97 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER98.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER98 OFF;
ALTER TABLE CUSTOMER98 DROP CONSTRAINT CUSTOMER98CKC;
ALTER TABLE CUSTOMER98 ADD CONSTRAINT CUSTOMER98CKC CHECK
(C_W_ID BETWEEN 123967 AND 125244);
SET INTEGRITY FOR CUSTOMER98 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER99.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER99 OFF;
ALTER TABLE CUSTOMER99 DROP CONSTRAINT CUSTOMER99CKC;
ALTER TABLE CUSTOMER99 ADD CONSTRAINT CUSTOMER99CKC CHECK
(C_W_ID BETWEEN 125245 AND 126522);
SET INTEGRITY FOR CUSTOMER99 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_CUSTOMER9.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER9 OFF;
ALTER TABLE CUSTOMER9 DROP CONSTRAINT CUSTOMER9CKC;
ALTER TABLE CUSTOMER9 ADD CONSTRAINT CUSTOMER9CKC CHECK (C_W_ID
BETWEEN 10225 AND 11502);
SET INTEGRITY FOR CUSTOMER9 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT10.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT10 OFF;
ALTER TABLE DISTRICT10 DROP CONSTRAINT DISTRICT10CKC;
ALTER TABLE DISTRICT10 ADD CONSTRAINT DISTRICT10CKC CHECK (D_W_ID
BETWEEN 46009 AND 51120);
SET INTEGRITY FOR DISTRICT10 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT11.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT11 OFF;
ALTER TABLE DISTRICT11 DROP CONSTRAINT DISTRICT11CKC;
ALTER TABLE DISTRICT11 ADD CONSTRAINT DISTRICT11CKC CHECK (D_W_ID
BETWEEN 51121 AND 56232);
SET INTEGRITY FOR DISTRICT11 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT12.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT12 OFF;
ALTER TABLE DISTRICT12 DROP CONSTRAINT DISTRICT12CKC;
ALTER TABLE DISTRICT12 ADD CONSTRAINT DISTRICT12CKC CHECK (D_W_ID
BETWEEN 56233 AND 61344);
SET INTEGRITY FOR DISTRICT12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT13.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT13 OFF;
ALTER TABLE DISTRICT13 DROP CONSTRAINT DISTRICT13CKC;
ALTER TABLE DISTRICT13 ADD CONSTRAINT DISTRICT13CKC CHECK (D_W_ID
BETWEEN 61345 AND 66456);
SET INTEGRITY FOR DISTRICT13 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT14.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT14 OFF;
ALTER TABLE DISTRICT14 DROP CONSTRAINT DISTRICT14CKC;
ALTER TABLE DISTRICT14 ADD CONSTRAINT DISTRICT14CKC CHECK (D_W_ID
BETWEEN 66457 AND 71568);
```

```
SET INTEGRITY FOR DISTRICT14 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT15.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT15 OFF;
ALTER TABLE DISTRICT15 DROP CONSTRAINT DISTRICT15CKC;
ALTER TABLE DISTRICT15 ADD CONSTRAINT DISTRICT15CKC CHECK (D_W_ID
BETWEEN 71569 AND 76680);
SET INTEGRITY FOR DISTRICT15 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT16.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT16 OFF;
ALTER TABLE DISTRICT16 DROP CONSTRAINT DISTRICT16CKC;
ALTER TABLE DISTRICT16 ADD CONSTRAINT DISTRICT16CKC CHECK (D_W_ID
BETWEEN 76681 AND 81792);
SET INTEGRITY FOR DISTRICT16 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT17.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT17 OFF;
ALTER TABLE DISTRICT17 DROP CONSTRAINT DISTRICT17CKC;
ALTER TABLE DISTRICT17 ADD CONSTRAINT DISTRICT17CKC CHECK (D_W_ID
BETWEEN 81793 AND 86904);
SET INTEGRITY FOR DISTRICT17 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT18.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT18 OFF;
ALTER TABLE DISTRICT18 DROP CONSTRAINT DISTRICT18CKC;
ALTER TABLE DISTRICT18 ADD CONSTRAINT DISTRICT18CKC CHECK (D_W_ID
BETWEEN 86905 AND 92016);
SET INTEGRITY FOR DISTRICT18 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT19.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT19 OFF;
ALTER TABLE DISTRICT19 DROP CONSTRAINT DISTRICT19CKC;
ALTER TABLE DISTRICT19 ADD CONSTRAINT DISTRICT19CKC CHECK (D_W_ID
BETWEEN 92017 AND 97128);
SET INTEGRITY FOR DISTRICT19 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT1.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT1 OFF;
ALTER TABLE DISTRICT1 DROP CONSTRAINT DISTRICT1CKC;
ALTER TABLE DISTRICT1 ADD CONSTRAINT DISTRICT1CKC CHECK (D_W_ID
BETWEEN 1 AND 5112);
SET INTEGRITY FOR DISTRICT1 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT20.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT20 OFF;
ALTER TABLE DISTRICT20 DROP CONSTRAINT DISTRICT20CKC;
ALTER TABLE DISTRICT20 ADD CONSTRAINT DISTRICT20CKC CHECK (D_W_ID
BETWEEN 97129 AND 102240);
SET INTEGRITY FOR DISTRICT20 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT21.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT21 OFF;
ALTER TABLE DISTRICT21 DROP CONSTRAINT DISTRICT21CKC;
ALTER TABLE DISTRICT21 ADD CONSTRAINT DISTRICT21CKC CHECK (D_W_ID
BETWEEN 102241 AND 107352);
SET INTEGRITY FOR DISTRICT21 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT22.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT22 OFF;
ALTER TABLE DISTRICT22 DROP CONSTRAINT DISTRICT22CKC;
ALTER TABLE DISTRICT22 ADD CONSTRAINT DISTRICT22CKC CHECK (D_W_ID
BETWEEN 107353 AND 112464);
SET INTEGRITY FOR DISTRICT22 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT23.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT23 OFF;
ALTER TABLE DISTRICT23 DROP CONSTRAINT DISTRICT23CKC;
ALTER TABLE DISTRICT23 ADD CONSTRAINT DISTRICT23CKC CHECK (D_W_ID
BETWEEN 112465 AND 117576);
SET INTEGRITY FOR DISTRICT23 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT24.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT24 OFF;
ALTER TABLE DISTRICT24 DROP CONSTRAINT DISTRICT24CKC;
ALTER TABLE DISTRICT24 ADD CONSTRAINT DISTRICT24CKC CHECK (D_W_ID
BETWEEN 117577 AND 122688);
SET INTEGRITY FOR DISTRICT24 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT25.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT25 OFF;
ALTER TABLE DISTRICT25 DROP CONSTRAINT DISTRICT25CKC;
ALTER TABLE DISTRICT25 ADD CONSTRAINT DISTRICT25CKC CHECK (D_W_ID
BETWEEN 122689 AND 127800);
SET INTEGRITY FOR DISTRICT25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT26.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT26 OFF;
ALTER TABLE DISTRICT26 DROP CONSTRAINT DISTRICT26CKC;
ALTER TABLE DISTRICT26 ADD CONSTRAINT DISTRICT26CKC CHECK (D_W_ID
BETWEEN 127801 AND 132912);
SET INTEGRITY FOR DISTRICT26 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT27.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT27 OFF;
ALTER TABLE DISTRICT27 DROP CONSTRAINT DISTRICT27CKC;
ALTER TABLE DISTRICT27 ADD CONSTRAINT DISTRICT27CKC CHECK (D_W_ID
BETWEEN 132913 AND 138024);
SET INTEGRITY FOR DISTRICT27 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT28.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT28 OFF;
ALTER TABLE DISTRICT28 DROP CONSTRAINT DISTRICT28CKC;
ALTER TABLE DISTRICT28 ADD CONSTRAINT DISTRICT28CKC CHECK (D_W_ID
BETWEEN 138025 AND 143136);
SET INTEGRITY FOR DISTRICT28 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT29.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT29 OFF;
ALTER TABLE DISTRICT29 DROP CONSTRAINT DISTRICT29CKC;
ALTER TABLE DISTRICT29 ADD CONSTRAINT DISTRICT29CKC CHECK (D_W_ID
BETWEEN 143137 AND 148248);
SET INTEGRITY FOR DISTRICT29 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT2.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT2 OFF;
ALTER TABLE DISTRICT2 DROP CONSTRAINT DISTRICT2CKC;
ALTER TABLE DISTRICT2 ADD CONSTRAINT DISTRICT2CKC CHECK (D_W_ID
BETWEEN 5113 AND 10224);
SET INTEGRITY FOR DISTRICT2 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT30.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT30 OFF;
ALTER TABLE DISTRICT30 DROP CONSTRAINT DISTRICT30CKC;
ALTER TABLE DISTRICT30 ADD CONSTRAINT DISTRICT30CKC CHECK (D_W_ID
BETWEEN 148249 AND 153360);
SET INTEGRITY FOR DISTRICT30 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT31.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT31 OFF;
ALTER TABLE DISTRICT31 DROP CONSTRAINT DISTRICT31CKC;
ALTER TABLE DISTRICT31 ADD CONSTRAINT DISTRICT31CKC CHECK (D_W_ID
BETWEEN 153361 AND 158472);
SET INTEGRITY FOR DISTRICT31 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT32.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT32 OFF;
ALTER TABLE DISTRICT32 DROP CONSTRAINT DISTRICT32CKC;
ALTER TABLE DISTRICT32 ADD CONSTRAINT DISTRICT32CKC CHECK (D_W_ID
BETWEEN 158473 AND 163584);
SET INTEGRITY FOR DISTRICT32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT33.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT33 OFF;
ALTER TABLE DISTRICT33 DROP CONSTRAINT DISTRICT33CKC;
ALTER TABLE DISTRICT33 ADD CONSTRAINT DISTRICT33CKC CHECK (D_W_ID
BETWEEN 163585 AND 168696);
SET INTEGRITY FOR DISTRICT33 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT34.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT34 OFF;
ALTER TABLE DISTRICT34 DROP CONSTRAINT DISTRICT34CKC;
ALTER TABLE DISTRICT34 ADD CONSTRAINT DISTRICT34CKC CHECK (D_W_ID
BETWEEN 168697 AND 173808);
SET INTEGRITY FOR DISTRICT34 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT35.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT35 OFF;
ALTER TABLE DISTRICT35 DROP CONSTRAINT DISTRICT35CKC;
ALTER TABLE DISTRICT35 ADD CONSTRAINT DISTRICT35CKC CHECK (D_W_ID
BETWEEN 173809 AND 178920);
SET INTEGRITY FOR DISTRICT35 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT36.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT36 OFF;
ALTER TABLE DISTRICT36 DROP CONSTRAINT DISTRICT36CKC;
ALTER TABLE DISTRICT36 ADD CONSTRAINT DISTRICT36CKC CHECK (D_W_ID >=
178921);
SET INTEGRITY FOR DISTRICT36 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT3.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT3 OFF;
ALTER TABLE DISTRICT3 DROP CONSTRAINT DISTRICT3CKC;
ALTER TABLE DISTRICT3 ADD CONSTRAINT DISTRICT3CKC CHECK (D_W_ID
BETWEEN 10225 AND 15336);
SET INTEGRITY FOR DISTRICT3 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT4.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT4 OFF;
ALTER TABLE DISTRICT4 DROP CONSTRAINT DISTRICT4CKC;
ALTER TABLE DISTRICT4 ADD CONSTRAINT DISTRICT4CKC CHECK (D_W_ID
BETWEEN 15337 AND 20448);
SET INTEGRITY FOR DISTRICT4 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT5.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT5 OFF;
ALTER TABLE DISTRICT5 DROP CONSTRAINT DISTRICT5CKC;
ALTER TABLE DISTRICT5 ADD CONSTRAINT DISTRICT5CKC CHECK (D_W_ID
BETWEEN 20449 AND 25560);
SET INTEGRITY FOR DISTRICT5 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT6.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT6 OFF;
ALTER TABLE DISTRICT6 DROP CONSTRAINT DISTRICT6CKC;
ALTER TABLE DISTRICT6 ADD CONSTRAINT DISTRICT6CKC CHECK (D_W_ID
BETWEEN 25561 AND 30672);
SET INTEGRITY FOR DISTRICT6 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT7.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT7 OFF;
ALTER TABLE DISTRICT7 DROP CONSTRAINT DISTRICT7CKC;
ALTER TABLE DISTRICT7 ADD CONSTRAINT DISTRICT7CKC CHECK (D_W_ID
BETWEEN 30673 AND 35784);
SET INTEGRITY FOR DISTRICT7 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT8.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT8 OFF;
ALTER TABLE DISTRICT8 DROP CONSTRAINT DISTRICT8CKC;
ALTER TABLE DISTRICT8 ADD CONSTRAINT DISTRICT8CKC CHECK (D_W_ID
BETWEEN 35785 AND 40896);
SET INTEGRITY FOR DISTRICT8 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_DISTRICT9.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT9 OFF;
ALTER TABLE DISTRICT9 DROP CONSTRAINT DISTRICT9CKC;
ALTER TABLE DISTRICT9 ADD CONSTRAINT DISTRICT9CKC CHECK (D_W_ID
BETWEEN 40897 AND 46008);
SET INTEGRITY FOR DISTRICT9 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY10.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY10 OFF;
ALTER TABLE HISTORY10 DROP CONSTRAINT HISTORY10CKC;
ALTER TABLE HISTORY10 ADD CONSTRAINT HISTORY10CKC CHECK (H_W_ID
BETWEEN 46009 AND 51120);
SET INTEGRITY FOR HISTORY10 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY11.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY11 OFF;
ALTER TABLE HISTORY11 DROP CONSTRAINT HISTORY11CKC;
ALTER TABLE HISTORY11 ADD CONSTRAINT HISTORY11CKC CHECK (H_W_ID
BETWEEN 51121 AND 56232);
SET INTEGRITY FOR HISTORY11 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY12.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY12 OFF;
ALTER TABLE HISTORY12 DROP CONSTRAINT HISTORY12CKC;
ALTER TABLE HISTORY12 ADD CONSTRAINT HISTORY12CKC CHECK (H_W_ID
BETWEEN 56233 AND 61344);
SET INTEGRITY FOR HISTORY12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY13.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY13 OFF;
ALTER TABLE HISTORY13 DROP CONSTRAINT HISTORY13CKC;
ALTER TABLE HISTORY13 ADD CONSTRAINT HISTORY13CKC CHECK (H_W_ID
BETWEEN 61345 AND 66456);
SET INTEGRITY FOR HISTORY13 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY14.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY14 OFF;
ALTER TABLE HISTORY14 DROP CONSTRAINT HISTORY14CKC;
ALTER TABLE HISTORY14 ADD CONSTRAINT HISTORY14CKC CHECK (H_W_ID
BETWEEN 66457 AND 71568);
SET INTEGRITY FOR HISTORY14 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY15.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY15 OFF;
ALTER TABLE HISTORY15 DROP CONSTRAINT HISTORY15CKC;
ALTER TABLE HISTORY15 ADD CONSTRAINT HISTORY15CKC CHECK (H_W_ID
BETWEEN 71569 AND 76680);
SET INTEGRITY FOR HISTORY15 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY16.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY16 OFF;
ALTER TABLE HISTORY16 DROP CONSTRAINT HISTORY16CKC;
ALTER TABLE HISTORY16 ADD CONSTRAINT HISTORY16CKC CHECK (H_W_ID
BETWEEN 76681 AND 81792);
SET INTEGRITY FOR HISTORY16 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY17.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR HISTORY17 OFF;
ALTER TABLE HISTORY17 DROP CONSTRAINT HISTORY17CKC;
ALTER TABLE HISTORY17 ADD CONSTRAINT HISTORY17CKC CHECK (H_W_ID
BETWEEN 81793 AND 86904);
SET INTEGRITY FOR HISTORY17 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY18.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY18 OFF;
ALTER TABLE HISTORY18 DROP CONSTRAINT HISTORY18CKC;
ALTER TABLE HISTORY18 ADD CONSTRAINT HISTORY18CKC CHECK (H_W_ID
BETWEEN 86905 AND 92016);
SET INTEGRITY FOR HISTORY18 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY19.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY19 OFF;
ALTER TABLE HISTORY19 DROP CONSTRAINT HISTORY19CKC;
ALTER TABLE HISTORY19 ADD CONSTRAINT HISTORY19CKC CHECK (H_W_ID
BETWEEN 92017 AND 97128);
SET INTEGRITY FOR HISTORY19 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY1.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY1 OFF;
ALTER TABLE HISTORY1 DROP CONSTRAINT HISTORY1CKC;
ALTER TABLE HISTORY1 ADD CONSTRAINT HISTORY1CKC CHECK (H_W_ID
BETWEEN 1 AND 5112);
SET INTEGRITY FOR HISTORY1 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY20.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY20 OFF;
ALTER TABLE HISTORY20 DROP CONSTRAINT HISTORY20CKC;
ALTER TABLE HISTORY20 ADD CONSTRAINT HISTORY20CKC CHECK (H_W_ID
BETWEEN 97129 AND 102240);
SET INTEGRITY FOR HISTORY20 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY21.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY21 OFF;
ALTER TABLE HISTORY21 DROP CONSTRAINT HISTORY21CKC;
ALTER TABLE HISTORY21 ADD CONSTRAINT HISTORY21CKC CHECK (H_W_ID
BETWEEN 102241 AND 107352);
SET INTEGRITY FOR HISTORY21 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY22.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY22 OFF;
ALTER TABLE HISTORY22 DROP CONSTRAINT HISTORY22CKC;
ALTER TABLE HISTORY22 ADD CONSTRAINT HISTORY22CKC CHECK (H_W_ID
BETWEEN 107353 AND 112464);
SET INTEGRITY FOR HISTORY22 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY23.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY23 OFF;
ALTER TABLE HISTORY23 DROP CONSTRAINT HISTORY23CKC;
ALTER TABLE HISTORY23 ADD CONSTRAINT HISTORY23CKC CHECK (H_W_ID
BETWEEN 112465 AND 117576);
SET INTEGRITY FOR HISTORY23 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY24.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY24 OFF;
ALTER TABLE HISTORY24 DROP CONSTRAINT HISTORY24CKC;
ALTER TABLE HISTORY24 ADD CONSTRAINT HISTORY24CKC CHECK (H_W_ID
BETWEEN 117577 AND 122688);
SET INTEGRITY FOR HISTORY24 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY25.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY25 OFF;
ALTER TABLE HISTORY25 DROP CONSTRAINT HISTORY25CKC;
ALTER TABLE HISTORY25 ADD CONSTRAINT HISTORY25CKC CHECK (H_W_ID
BETWEEN 122689 AND 127800);
SET INTEGRITY FOR HISTORY25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY26.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY26 OFF;
ALTER TABLE HISTORY26 DROP CONSTRAINT HISTORY26CKC;
ALTER TABLE HISTORY26 ADD CONSTRAINT HISTORY26CKC CHECK (H_W_ID
BETWEEN 127801 AND 132912);
SET INTEGRITY FOR HISTORY26 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY27.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY27 OFF;
ALTER TABLE HISTORY27 DROP CONSTRAINT HISTORY27CKC;
ALTER TABLE HISTORY27 ADD CONSTRAINT HISTORY27CKC CHECK (H_W_ID
BETWEEN 132913 AND 138024);
SET INTEGRITY FOR HISTORY27 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY28.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY28 OFF;
ALTER TABLE HISTORY28 DROP CONSTRAINT HISTORY28CKC;
ALTER TABLE HISTORY28 ADD CONSTRAINT HISTORY28CKC CHECK (H_W_ID
BETWEEN 138025 AND 143136);
SET INTEGRITY FOR HISTORY28 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY29.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY29 OFF;
ALTER TABLE HISTORY29 DROP CONSTRAINT HISTORY29CKC;
ALTER TABLE HISTORY29 ADD CONSTRAINT HISTORY29CKC CHECK (H_W_ID
BETWEEN 143137 AND 148248);
SET INTEGRITY FOR HISTORY29 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY2.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY2 OFF;
ALTER TABLE HISTORY2 DROP CONSTRAINT HISTORY2CKC;
ALTER TABLE HISTORY2 ADD CONSTRAINT HISTORY2CKC CHECK (H_W_ID
BETWEEN 5113 AND 10224);
SET INTEGRITY FOR HISTORY2 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY30.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY30 OFF;
ALTER TABLE HISTORY30 DROP CONSTRAINT HISTORY30CKC;
ALTER TABLE HISTORY30 ADD CONSTRAINT HISTORY30CKC CHECK (H_W_ID
BETWEEN 148249 AND 153360);
SET INTEGRITY FOR HISTORY30 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY31.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY31 OFF;
ALTER TABLE HISTORY31 DROP CONSTRAINT HISTORY31CKC;
ALTER TABLE HISTORY31 ADD CONSTRAINT HISTORY31CKC CHECK (H_W_ID
BETWEEN 153361 AND 158472);
SET INTEGRITY FOR HISTORY31 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY32.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY32 OFF;
ALTER TABLE HISTORY32 DROP CONSTRAINT HISTORY32CKC;
ALTER TABLE HISTORY32 ADD CONSTRAINT HISTORY32CKC CHECK (H_W_ID
BETWEEN 158473 AND 163584);
SET INTEGRITY FOR HISTORY32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY33.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY33 OFF;
ALTER TABLE HISTORY33 DROP CONSTRAINT HISTORY33CKC;
ALTER TABLE HISTORY33 ADD CONSTRAINT HISTORY33CKC CHECK (H_W_ID
BETWEEN 163585 AND 168696);
SET INTEGRITY FOR HISTORY33 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY34.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY34 OFF;
ALTER TABLE HISTORY34 DROP CONSTRAINT HISTORY34CKC;
ALTER TABLE HISTORY34 ADD CONSTRAINT HISTORY34CKC CHECK (H_W_ID
BETWEEN 168697 AND 173808);
```

```
SET INTEGRITY FOR HISTORY34 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY35.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY35 OFF;
ALTER TABLE HISTORY35 DROP CONSTRAINT HISTORY35CKC;
ALTER TABLE HISTORY35 ADD CONSTRAINT HISTORY35CKC CHECK (H_W_ID
BETWEEN 173809 AND 178920);
SET INTEGRITY FOR HISTORY35 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY36.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY36 OFF;
ALTER TABLE HISTORY36 DROP CONSTRAINT HISTORY36CKC;
ALTER TABLE HISTORY36 ADD CONSTRAINT HISTORY36CKC CHECK (H_W_ID >=
178921);
SET INTEGRITY FOR HISTORY36 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY3.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY3 OFF;
ALTER TABLE HISTORY3 DROP CONSTRAINT HISTORY3CKC;
ALTER TABLE HISTORY3 ADD CONSTRAINT HISTORY3CKC CHECK (H_W_ID
BETWEEN 10225 AND 15336);
SET INTEGRITY FOR HISTORY3 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY4.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY4 OFF;
ALTER TABLE HISTORY4 DROP CONSTRAINT HISTORY4CKC;
ALTER TABLE HISTORY4 ADD CONSTRAINT HISTORY4CKC CHECK (H_W_ID
BETWEEN 15337 AND 20448);
SET INTEGRITY FOR HISTORY4 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY5.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY5 OFF;
ALTER TABLE HISTORY5 DROP CONSTRAINT HISTORY5CKC;
ALTER TABLE HISTORY5 ADD CONSTRAINT HISTORY5CKC CHECK (H_W_ID
BETWEEN 20449 AND 25560);
SET INTEGRITY FOR HISTORY5 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY6.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY6 OFF;
ALTER TABLE HISTORY6 DROP CONSTRAINT HISTORY6CKC;
ALTER TABLE HISTORY6 ADD CONSTRAINT HISTORY6CKC CHECK (H_W_ID
BETWEEN 25561 AND 30672);
SET INTEGRITY FOR HISTORY6 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY7.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY7 OFF;
ALTER TABLE HISTORY7 DROP CONSTRAINT HISTORY7CKC;
ALTER TABLE HISTORY7 ADD CONSTRAINT HISTORY7CKC CHECK (H_W_ID
BETWEEN 30673 AND 35784);
SET INTEGRITY FOR HISTORY7 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY8.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY8 OFF;
ALTER TABLE HISTORY8 DROP CONSTRAINT HISTORY8CKC;
ALTER TABLE HISTORY8 ADD CONSTRAINT HISTORY8CKC CHECK (H_W_ID
BETWEEN 35785 AND 40896);
SET INTEGRITY FOR HISTORY8 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_HISTORY9.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY9 OFF;
ALTER TABLE HISTORY9 DROP CONSTRAINT HISTORY9CKC;
ALTER TABLE HISTORY9 ADD CONSTRAINT HISTORY9CKC CHECK (H_W_ID
BETWEEN 40897 AND 46008);
SET INTEGRITY FOR HISTORY9 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER10.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER10 OFF;
ALTER TABLE NEW_ORDER10 DROP CONSTRAINT NEW_ORDER10CKC;
ALTER TABLE NEW_ORDER10 ADD CONSTRAINT NEW_ORDER10CKC CHECK
(NO_W_ID BETWEEN 46009 AND 51120);
SET INTEGRITY FOR NEW_ORDER10 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER11.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER11 OFF;
ALTER TABLE NEW_ORDER11 DROP CONSTRAINT NEW_ORDER11CKC;
ALTER TABLE NEW_ORDER11 ADD CONSTRAINT NEW_ORDER11CKC CHECK
(NO_W_ID BETWEEN 51121 AND 56232);
SET INTEGRITY FOR NEW_ORDER11 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER12.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER12 OFF;
ALTER TABLE NEW_ORDER12 DROP CONSTRAINT NEW_ORDER12CKC;
ALTER TABLE NEW_ORDER12 ADD CONSTRAINT NEW_ORDER12CKC CHECK
(NO_W_ID BETWEEN 56233 AND 61344);
SET INTEGRITY FOR NEW_ORDER12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER13.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER13 OFF;
ALTER TABLE NEW_ORDER13 DROP CONSTRAINT NEW_ORDER13CKC;
ALTER TABLE NEW_ORDER13 ADD CONSTRAINT NEW_ORDER13CKC CHECK
(NO_W_ID BETWEEN 61345 AND 66456);
SET INTEGRITY FOR NEW_ORDER13 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER14.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER14 OFF;
ALTER TABLE NEW_ORDER14 DROP CONSTRAINT NEW_ORDER14CKC;
ALTER TABLE NEW_ORDER14 ADD CONSTRAINT NEW_ORDER14CKC CHECK
(NO_W_ID BETWEEN 66457 AND 71568);
SET INTEGRITY FOR NEW_ORDER14 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER15.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER15 OFF;
ALTER TABLE NEW_ORDER15 DROP CONSTRAINT NEW_ORDER15CKC;
ALTER TABLE NEW_ORDER15 ADD CONSTRAINT NEW_ORDER15CKC CHECK
(NO_W_ID BETWEEN 71569 AND 76680);
SET INTEGRITY FOR NEW_ORDER15 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER16.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER16 OFF;
ALTER TABLE NEW_ORDER16 DROP CONSTRAINT NEW_ORDER16CKC;
ALTER TABLE NEW_ORDER16 ADD CONSTRAINT NEW_ORDER16CKC CHECK
(NO_W_ID BETWEEN 76681 AND 81792);
SET INTEGRITY FOR NEW_ORDER16 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER17.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER17 OFF;
ALTER TABLE NEW_ORDER17 DROP CONSTRAINT NEW_ORDER17CKC;
ALTER TABLE NEW_ORDER17 ADD CONSTRAINT NEW_ORDER17CKC CHECK
(NO_W_ID BETWEEN 81793 AND 86904);
SET INTEGRITY FOR NEW_ORDER17 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER18.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER18 OFF;
ALTER TABLE NEW_ORDER18 DROP CONSTRAINT NEW_ORDER18CKC;
ALTER TABLE NEW_ORDER18 ADD CONSTRAINT NEW_ORDER18CKC CHECK
(NO_W_ID BETWEEN 86905 AND 92016);
SET INTEGRITY FOR NEW_ORDER18 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER19.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER19 OFF;
ALTER TABLE NEW_ORDER19 DROP CONSTRAINT NEW_ORDER19CKC;
ALTER TABLE NEW_ORDER19 ADD CONSTRAINT NEW_ORDER19CKC CHECK
(NO_W_ID BETWEEN 92017 AND 97128);
SET INTEGRITY FOR NEW_ORDER19 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER1.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER1 OFF;
ALTER TABLE NEW_ORDER1 DROP CONSTRAINT NEW_ORDER1CKC;
ALTER TABLE NEW_ORDER1 ADD CONSTRAINT NEW_ORDER1CKC CHECK
(NO_W_ID BETWEEN 1 AND 5112);
SET INTEGRITY FOR NEW_ORDER1 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER20.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER20 OFF;
ALTER TABLE NEW_ORDER20 DROP CONSTRAINT NEW_ORDER20CKC;
ALTER TABLE NEW_ORDER20 ADD CONSTRAINT NEW_ORDER20CKC CHECK
(NO_W_ID BETWEEN 97129 AND 102240);
SET INTEGRITY FOR NEW_ORDER20 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER21.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER21 OFF;
ALTER TABLE NEW_ORDER21 DROP CONSTRAINT NEW_ORDER21CKC;
ALTER TABLE NEW_ORDER21 ADD CONSTRAINT NEW_ORDER21CKC CHECK
(NO_W_ID BETWEEN 102241 AND 107352);
SET INTEGRITY FOR NEW_ORDER21 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER22.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER22 OFF;
ALTER TABLE NEW_ORDER22 DROP CONSTRAINT NEW_ORDER22CKC;
ALTER TABLE NEW_ORDER22 ADD CONSTRAINT NEW_ORDER22CKC CHECK
(NO_W_ID BETWEEN 107353 AND 112464);
SET INTEGRITY FOR NEW_ORDER22 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER23.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER23 OFF;
ALTER TABLE NEW_ORDER23 DROP CONSTRAINT NEW_ORDER23CKC;
ALTER TABLE NEW_ORDER23 ADD CONSTRAINT NEW_ORDER23CKC CHECK
(NO_W_ID BETWEEN 112465 AND 117576);
SET INTEGRITY FOR NEW_ORDER23 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER24.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER24 OFF;
ALTER TABLE NEW_ORDER24 DROP CONSTRAINT NEW_ORDER24CKC;
ALTER TABLE NEW_ORDER24 ADD CONSTRAINT NEW_ORDER24CKC CHECK
(NO_W_ID BETWEEN 117577 AND 122688);
SET INTEGRITY FOR NEW_ORDER24 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER25.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER25 OFF;
ALTER TABLE NEW_ORDER25 DROP CONSTRAINT NEW_ORDER25CKC;
ALTER TABLE NEW_ORDER25 ADD CONSTRAINT NEW_ORDER25CKC CHECK
(NO_W_ID BETWEEN 122689 AND 127800);
SET INTEGRITY FOR NEW_ORDER25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER26.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER26 OFF;
ALTER TABLE NEW_ORDER26 DROP CONSTRAINT NEW_ORDER26CKC;
ALTER TABLE NEW_ORDER26 ADD CONSTRAINT NEW_ORDER26CKC CHECK
(NO_W_ID BETWEEN 127801 AND 132912);
SET INTEGRITY FOR NEW_ORDER26 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER27.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER27 OFF;
ALTER TABLE NEW_ORDER27 DROP CONSTRAINT NEW_ORDER27CKC;
ALTER TABLE NEW_ORDER27 ADD CONSTRAINT NEW_ORDER27CKC CHECK
(NO_W_ID BETWEEN 132913 AND 138024);
SET INTEGRITY FOR NEW_ORDER27 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER28.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER28 OFF;
ALTER TABLE NEW_ORDER28 DROP CONSTRAINT NEW_ORDER28CKC;
ALTER TABLE NEW_ORDER28 ADD CONSTRAINT NEW_ORDER28CKC CHECK
(NO_W_ID BETWEEN 138025 AND 143136);
SET INTEGRITY FOR NEW_ORDER28 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER29.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER29 OFF;
ALTER TABLE NEW_ORDER29 DROP CONSTRAINT NEW_ORDER29CKC;
ALTER TABLE NEW_ORDER29 ADD CONSTRAINT NEW_ORDER29CKC CHECK
(NO_W_ID BETWEEN 143137 AND 148248);
SET INTEGRITY FOR NEW_ORDER29 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER2.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER2 OFF;
ALTER TABLE NEW_ORDER2 DROP CONSTRAINT NEW_ORDER2CKC;
ALTER TABLE NEW_ORDER2 ADD CONSTRAINT NEW_ORDER2CKC CHECK
(NO_W_ID BETWEEN 5113 AND 10224);
SET INTEGRITY FOR NEW_ORDER2 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER30.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER30 OFF;
ALTER TABLE NEW_ORDER30 DROP CONSTRAINT NEW_ORDER30CKC;
ALTER TABLE NEW_ORDER30 ADD CONSTRAINT NEW_ORDER30CKC CHECK
(NO_W_ID BETWEEN 148249 AND 153360);
SET INTEGRITY FOR NEW_ORDER30 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER31.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER31 OFF;
ALTER TABLE NEW_ORDER31 DROP CONSTRAINT NEW_ORDER31CKC;
ALTER TABLE NEW_ORDER31 ADD CONSTRAINT NEW_ORDER31CKC CHECK
(NO_W_ID BETWEEN 153361 AND 158472);
SET INTEGRITY FOR NEW_ORDER31 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER32.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER32 OFF;
ALTER TABLE NEW_ORDER32 DROP CONSTRAINT NEW_ORDER32CKC;
ALTER TABLE NEW_ORDER32 ADD CONSTRAINT NEW_ORDER32CKC CHECK
(NO_W_ID BETWEEN 158473 AND 163584);
SET INTEGRITY FOR NEW_ORDER32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER33.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER33 OFF;
ALTER TABLE NEW_ORDER33 DROP CONSTRAINT NEW_ORDER33CKC;
ALTER TABLE NEW_ORDER33 ADD CONSTRAINT NEW_ORDER33CKC CHECK
(NO_W_ID BETWEEN 163585 AND 168696);
SET INTEGRITY FOR NEW_ORDER33 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER34.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER34 OFF;
ALTER TABLE NEW_ORDER34 DROP CONSTRAINT NEW_ORDER34CKC;
ALTER TABLE NEW_ORDER34 ADD CONSTRAINT NEW_ORDER34CKC CHECK
(NO_W_ID BETWEEN 168697 AND 173808);
SET INTEGRITY FOR NEW_ORDER34 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER35.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER35 OFF;
ALTER TABLE NEW_ORDER35 DROP CONSTRAINT NEW_ORDER35CKC;
ALTER TABLE NEW_ORDER35 ADD CONSTRAINT NEW_ORDER35CKC CHECK
(NO_W_ID BETWEEN 173809 AND 178920);
SET INTEGRITY FOR NEW_ORDER35 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER36.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER36 OFF;
ALTER TABLE NEW_ORDER36 DROP CONSTRAINT NEW_ORDER36CKC;
ALTER TABLE NEW_ORDER36 ADD CONSTRAINT NEW_ORDER36CKC CHECK
(NO_W_ID >= 178921);
SET INTEGRITY FOR NEW_ORDER36 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER3.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR NEW_ORDER3 OFF;
ALTER TABLE NEW_ORDER3 DROP CONSTRAINT NEW_ORDER3CKC;
ALTER TABLE NEW_ORDER3 ADD CONSTRAINT NEW_ORDER3CKC CHECK
(NO_W_ID BETWEEN 10225 AND 15336);
SET INTEGRITY FOR NEW_ORDER3 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER4.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER4 OFF;
ALTER TABLE NEW_ORDER4 DROP CONSTRAINT NEW_ORDER4CKC;
ALTER TABLE NEW_ORDER4 ADD CONSTRAINT NEW_ORDER4CKC CHECK
(NO_W_ID BETWEEN 15337 AND 20448);
SET INTEGRITY FOR NEW_ORDER4 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER5.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER5 OFF;
ALTER TABLE NEW_ORDER5 DROP CONSTRAINT NEW_ORDER5CKC;
ALTER TABLE NEW_ORDER5 ADD CONSTRAINT NEW_ORDER5CKC CHECK
(NO_W_ID BETWEEN 20449 AND 25560);
SET INTEGRITY FOR NEW_ORDER5 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER6.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER6 OFF;
ALTER TABLE NEW_ORDER6 DROP CONSTRAINT NEW_ORDER6CKC;
ALTER TABLE NEW_ORDER6 ADD CONSTRAINT NEW_ORDER6CKC CHECK
(NO_W_ID BETWEEN 25561 AND 30672);
SET INTEGRITY FOR NEW_ORDER6 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER7.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER7 OFF;
ALTER TABLE NEW_ORDER7 DROP CONSTRAINT NEW_ORDER7CKC;
ALTER TABLE NEW_ORDER7 ADD CONSTRAINT NEW_ORDER7CKC CHECK
(NO_W_ID BETWEEN 30673 AND 35784);
SET INTEGRITY FOR NEW_ORDER7 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER8.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER8 OFF;
ALTER TABLE NEW_ORDER8 DROP CONSTRAINT NEW_ORDER8CKC;
ALTER TABLE NEW_ORDER8 ADD CONSTRAINT NEW_ORDER8CKC CHECK
(NO_W_ID BETWEEN 35785 AND 40896);
SET INTEGRITY FOR NEW_ORDER8 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER9.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDER9 OFF;
ALTER TABLE NEW_ORDER9 DROP CONSTRAINT NEW_ORDER9CKC;
ALTER TABLE NEW_ORDER9 ADD CONSTRAINT NEW_ORDER9CKC CHECK
(NO_W_ID BETWEEN 40897 AND 46008);
SET INTEGRITY FOR NEW_ORDER9 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE10.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE10 OFF;
ALTER TABLE ORDER_LINE10 DROP CONSTRAINT ORDER_LINE10CKC;
ALTER TABLE ORDER_LINE10 ADD CONSTRAINT ORDER_LINE10CKC CHECK
(OL_W_ID BETWEEN 23005 AND 25560);
SET INTEGRITY FOR ORDER_LINE10 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE11.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE11 OFF;
ALTER TABLE ORDER_LINE11 DROP CONSTRAINT ORDER_LINE11CKC;
ALTER TABLE ORDER_LINE11 ADD CONSTRAINT ORDER_LINE11CKC CHECK
(OL_W_ID BETWEEN 25561 AND 28116);
SET INTEGRITY FOR ORDER_LINE11 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE12.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE12 OFF;
ALTER TABLE ORDER_LINE12 DROP CONSTRAINT ORDER_LINE12CKC;
ALTER TABLE ORDER_LINE12 ADD CONSTRAINT ORDER_LINE12CKC CHECK
(OL_W_ID BETWEEN 28117 AND 30672);
SET INTEGRITY FOR ORDER_LINE12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE13.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE13 OFF;
ALTER TABLE ORDER_LINE13 DROP CONSTRAINT ORDER_LINE13CKC;
ALTER TABLE ORDER_LINE13 ADD CONSTRAINT ORDER_LINE13CKC CHECK
(OL_W_ID BETWEEN 30673 AND 33228);
SET INTEGRITY FOR ORDER_LINE13 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE14.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE14 OFF;
ALTER TABLE ORDER_LINE14 DROP CONSTRAINT ORDER_LINE14CKC;
ALTER TABLE ORDER_LINE14 ADD CONSTRAINT ORDER_LINE14CKC CHECK
(OL_W_ID BETWEEN 33229 AND 35784);
SET INTEGRITY FOR ORDER_LINE14 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE15.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE15 OFF;
ALTER TABLE ORDER_LINE15 DROP CONSTRAINT ORDER_LINE15CKC;
ALTER TABLE ORDER_LINE15 ADD CONSTRAINT ORDER_LINE15CKC CHECK
(OL_W_ID BETWEEN 35785 AND 38340);
SET INTEGRITY FOR ORDER_LINE15 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE16.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE16 OFF;
ALTER TABLE ORDER_LINE16 DROP CONSTRAINT ORDER_LINE16CKC;
ALTER TABLE ORDER_LINE16 ADD CONSTRAINT ORDER_LINE16CKC CHECK
(OL_W_ID BETWEEN 38341 AND 40896);
SET INTEGRITY FOR ORDER_LINE16 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE17.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE17 OFF;
ALTER TABLE ORDER_LINE17 DROP CONSTRAINT ORDER_LINE17CKC;
ALTER TABLE ORDER_LINE17 ADD CONSTRAINT ORDER_LINE17CKC CHECK
(OL_W_ID BETWEEN 40897 AND 43452);
SET INTEGRITY FOR ORDER_LINE17 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE18.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE18 OFF;
ALTER TABLE ORDER_LINE18 DROP CONSTRAINT ORDER_LINE18CKC;
ALTER TABLE ORDER_LINE18 ADD CONSTRAINT ORDER_LINE18CKC CHECK
(OL_W_ID BETWEEN 43453 AND 46008);
SET INTEGRITY FOR ORDER_LINE18 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE19.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE19 OFF;
ALTER TABLE ORDER_LINE19 DROP CONSTRAINT ORDER_LINE19CKC;
ALTER TABLE ORDER_LINE19 ADD CONSTRAINT ORDER_LINE19CKC CHECK
(OL_W_ID BETWEEN 46009 AND 48564);
SET INTEGRITY FOR ORDER_LINE19 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE1.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE1 OFF;
ALTER TABLE ORDER_LINE1 DROP CONSTRAINT ORDER_LINE1CKC;
ALTER TABLE ORDER_LINE1 ADD CONSTRAINT ORDER_LINE1CKC CHECK
(OL_W_ID BETWEEN 1 AND 2556);
SET INTEGRITY FOR ORDER_LINE1 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE20.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE20 OFF;
ALTER TABLE ORDER_LINE20 DROP CONSTRAINT ORDER_LINE20CKC;
ALTER TABLE ORDER_LINE20 ADD CONSTRAINT ORDER_LINE20CKC CHECK
(OL_W_ID BETWEEN 48565 AND 51120);
SET INTEGRITY FOR ORDER_LINE20 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE21.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE21 OFF;
ALTER TABLE ORDER_LINE21 DROP CONSTRAINT ORDER_LINE21CKC;
ALTER TABLE ORDER_LINE21 ADD CONSTRAINT ORDER_LINE21CKC CHECK
(OL_W_ID BETWEEN 51121 AND 53676);
```

```
SET INTEGRITY FOR ORDER_LINE21 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE22.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE22 OFF;
ALTER TABLE ORDER_LINE22 DROP CONSTRAINT ORDER_LINE22CKC;
ALTER TABLE ORDER_LINE22 ADD CONSTRAINT ORDER_LINE22CKC CHECK
(OL_W_ID BETWEEN 53677 AND 56232);
SET INTEGRITY FOR ORDER_LINE22 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE23.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE23 OFF;
ALTER TABLE ORDER_LINE23 DROP CONSTRAINT ORDER_LINE23CKC;
ALTER TABLE ORDER_LINE23 ADD CONSTRAINT ORDER_LINE23CKC CHECK
(OL_W_ID BETWEEN 56233 AND 58788);
SET INTEGRITY FOR ORDER_LINE23 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE24.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE24 OFF;
ALTER TABLE ORDER_LINE24 DROP CONSTRAINT ORDER_LINE24CKC;
ALTER TABLE ORDER_LINE24 ADD CONSTRAINT ORDER_LINE24CKC CHECK
(OL_W_ID BETWEEN 58789 AND 61344);
SET INTEGRITY FOR ORDER_LINE24 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE25.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE25 OFF;
ALTER TABLE ORDER_LINE25 DROP CONSTRAINT ORDER_LINE25CKC;
ALTER TABLE ORDER_LINE25 ADD CONSTRAINT ORDER_LINE25CKC CHECK
(OL_W_ID BETWEEN 61345 AND 63900);
SET INTEGRITY FOR ORDER_LINE25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE26.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE26 OFF;
ALTER TABLE ORDER_LINE26 DROP CONSTRAINT ORDER_LINE26CKC;
ALTER TABLE ORDER_LINE26 ADD CONSTRAINT ORDER_LINE26CKC CHECK
(OL_W_ID BETWEEN 63901 AND 66456);
SET INTEGRITY FOR ORDER_LINE26 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE27.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE27 OFF;
ALTER TABLE ORDER_LINE27 DROP CONSTRAINT ORDER_LINE27CKC;
ALTER TABLE ORDER_LINE27 ADD CONSTRAINT ORDER_LINE27CKC CHECK
(OL_W_ID BETWEEN 66457 AND 69012);
SET INTEGRITY FOR ORDER_LINE27 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE28.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE28 OFF;
ALTER TABLE ORDER_LINE28 DROP CONSTRAINT ORDER_LINE28CKC;
ALTER TABLE ORDER_LINE28 ADD CONSTRAINT ORDER_LINE28CKC CHECK
(OL_W_ID BETWEEN 69013 AND 71568);
SET INTEGRITY FOR ORDER_LINE28 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE29.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE29 OFF;
ALTER TABLE ORDER_LINE29 DROP CONSTRAINT ORDER_LINE29CKC;
ALTER TABLE ORDER_LINE29 ADD CONSTRAINT ORDER_LINE29CKC CHECK
(OL_W_ID BETWEEN 71569 AND 74124);
SET INTEGRITY FOR ORDER_LINE29 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE2.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE2 OFF;
ALTER TABLE ORDER_LINE2 DROP CONSTRAINT ORDER_LINE2CKC;
ALTER TABLE ORDER_LINE2 ADD CONSTRAINT ORDER_LINE2CKC CHECK
(OL_W_ID BETWEEN 2557 AND 5112);
SET INTEGRITY FOR ORDER_LINE2 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE30.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE30 OFF;
ALTER TABLE ORDER_LINE30 DROP CONSTRAINT ORDER_LINE30CKC;
ALTER TABLE ORDER_LINE30 ADD CONSTRAINT ORDER_LINE30CKC CHECK
(OL_W_ID BETWEEN 74125 AND 76680);
SET INTEGRITY FOR ORDER_LINE30 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE31.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE31 OFF;
ALTER TABLE ORDER_LINE31 DROP CONSTRAINT ORDER_LINE31CKC;
ALTER TABLE ORDER_LINE31 ADD CONSTRAINT ORDER_LINE31CKC CHECK
(OL_W_ID BETWEEN 76681 AND 79236);
SET INTEGRITY FOR ORDER_LINE31 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE32.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE32 OFF;
ALTER TABLE ORDER_LINE32 DROP CONSTRAINT ORDER_LINE32CKC;
ALTER TABLE ORDER_LINE32 ADD CONSTRAINT ORDER_LINE32CKC CHECK
(OL_W_ID BETWEEN 79237 AND 81792);
SET INTEGRITY FOR ORDER_LINE32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE33.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE33 OFF;
ALTER TABLE ORDER_LINE33 DROP CONSTRAINT ORDER_LINE33CKC;
ALTER TABLE ORDER_LINE33 ADD CONSTRAINT ORDER_LINE33CKC CHECK
(OL_W_ID BETWEEN 81793 AND 84348);
SET INTEGRITY FOR ORDER_LINE33 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE34.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE34 OFF;
ALTER TABLE ORDER_LINE34 DROP CONSTRAINT ORDER_LINE34CKC;
ALTER TABLE ORDER_LINE34 ADD CONSTRAINT ORDER_LINE34CKC CHECK
(OL_W_ID BETWEEN 84349 AND 86904);
SET INTEGRITY FOR ORDER_LINE34 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE35.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE35 OFF;
ALTER TABLE ORDER_LINE35 DROP CONSTRAINT ORDER_LINE35CKC;
ALTER TABLE ORDER_LINE35 ADD CONSTRAINT ORDER_LINE35CKC CHECK
(OL_W_ID BETWEEN 86905 AND 89460);
SET INTEGRITY FOR ORDER_LINE35 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE36.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE36 OFF;
ALTER TABLE ORDER_LINE36 DROP CONSTRAINT ORDER_LINE36CKC;
ALTER TABLE ORDER_LINE36 ADD CONSTRAINT ORDER_LINE36CKC CHECK
(OL_W_ID BETWEEN 89461 AND 92016);
SET INTEGRITY FOR ORDER_LINE36 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE37.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE37 OFF;
ALTER TABLE ORDER_LINE37 DROP CONSTRAINT ORDER_LINE37CKC;
ALTER TABLE ORDER_LINE37 ADD CONSTRAINT ORDER_LINE37CKC CHECK
(OL_W_ID BETWEEN 92017 AND 94572);
SET INTEGRITY FOR ORDER_LINE37 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE38.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE38 OFF;
ALTER TABLE ORDER_LINE38 DROP CONSTRAINT ORDER_LINE38CKC;
ALTER TABLE ORDER_LINE38 ADD CONSTRAINT ORDER_LINE38CKC CHECK
(OL_W_ID BETWEEN 94573 AND 97128);
SET INTEGRITY FOR ORDER_LINE38 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE39.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE39 OFF;
ALTER TABLE ORDER_LINE39 DROP CONSTRAINT ORDER_LINE39CKC;
ALTER TABLE ORDER_LINE39 ADD CONSTRAINT ORDER_LINE39CKC CHECK
(OL_W_ID BETWEEN 97129 AND 99684);
SET INTEGRITY FOR ORDER_LINE39 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE3.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE3 OFF;
ALTER TABLE ORDER_LINE3 DROP CONSTRAINT ORDER_LINE3CKC;
ALTER TABLE ORDER_LINE3 ADD CONSTRAINT ORDER_LINE3CKC CHECK
(OL_W_ID BETWEEN 5113 AND 7668);
SET INTEGRITY FOR ORDER_LINE3 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE40.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE40 OFF;
ALTER TABLE ORDER_LINE40 DROP CONSTRAINT ORDER_LINE40CKC;
ALTER TABLE ORDER_LINE40 ADD CONSTRAINT ORDER_LINE40CKC CHECK
(OL_W_ID BETWEEN 99685 AND 102240);
SET INTEGRITY FOR ORDER_LINE40 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE41.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE41 OFF;
ALTER TABLE ORDER_LINE41 DROP CONSTRAINT ORDER_LINE41CKC;
ALTER TABLE ORDER_LINE41 ADD CONSTRAINT ORDER_LINE41CKC CHECK
(OL_W_ID BETWEEN 102241 AND 104796);
SET INTEGRITY FOR ORDER_LINE41 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE42.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE42 OFF;
ALTER TABLE ORDER_LINE42 DROP CONSTRAINT ORDER_LINE42CKC;
ALTER TABLE ORDER_LINE42 ADD CONSTRAINT ORDER_LINE42CKC CHECK
(OL_W_ID BETWEEN 104797 AND 107352);
SET INTEGRITY FOR ORDER_LINE42 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE43.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE43 OFF;
ALTER TABLE ORDER_LINE43 DROP CONSTRAINT ORDER_LINE43CKC;
ALTER TABLE ORDER_LINE43 ADD CONSTRAINT ORDER_LINE43CKC CHECK
(OL_W_ID BETWEEN 107353 AND 109908);
SET INTEGRITY FOR ORDER_LINE43 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE44.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE44 OFF;
ALTER TABLE ORDER_LINE44 DROP CONSTRAINT ORDER_LINE44CKC;
ALTER TABLE ORDER_LINE44 ADD CONSTRAINT ORDER_LINE44CKC CHECK
(OL_W_ID BETWEEN 109909 AND 112464);
SET INTEGRITY FOR ORDER_LINE44 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE45.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE45 OFF;
ALTER TABLE ORDER_LINE45 DROP CONSTRAINT ORDER_LINE45CKC;
ALTER TABLE ORDER_LINE45 ADD CONSTRAINT ORDER_LINE45CKC CHECK
(OL_W_ID BETWEEN 112465 AND 115020);
SET INTEGRITY FOR ORDER_LINE45 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE46.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE46 OFF;
ALTER TABLE ORDER_LINE46 DROP CONSTRAINT ORDER_LINE46CKC;
ALTER TABLE ORDER_LINE46 ADD CONSTRAINT ORDER_LINE46CKC CHECK
(OL_W_ID BETWEEN 115021 AND 117576);
SET INTEGRITY FOR ORDER_LINE46 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE47.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE47 OFF;
ALTER TABLE ORDER_LINE47 DROP CONSTRAINT ORDER_LINE47CKC;
ALTER TABLE ORDER_LINE47 ADD CONSTRAINT ORDER_LINE47CKC CHECK
(OL_W_ID BETWEEN 117577 AND 120132);
SET INTEGRITY FOR ORDER_LINE47 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE48.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE48 OFF;
ALTER TABLE ORDER_LINE48 DROP CONSTRAINT ORDER_LINE48CKC;
ALTER TABLE ORDER_LINE48 ADD CONSTRAINT ORDER_LINE48CKC CHECK
(OL_W_ID BETWEEN 120133 AND 122688);
SET INTEGRITY FOR ORDER_LINE48 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE49.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE49 OFF;
ALTER TABLE ORDER_LINE49 DROP CONSTRAINT ORDER_LINE49CKC;
ALTER TABLE ORDER_LINE49 ADD CONSTRAINT ORDER_LINE49CKC CHECK
(OL_W_ID BETWEEN 122689 AND 125244);
SET INTEGRITY FOR ORDER_LINE49 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE4.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE4 OFF;
ALTER TABLE ORDER_LINE4 DROP CONSTRAINT ORDER_LINE4CKC;
ALTER TABLE ORDER_LINE4 ADD CONSTRAINT ORDER_LINE4CKC CHECK
(OL_W_ID BETWEEN 7669 AND 10224);
SET INTEGRITY FOR ORDER_LINE4 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE50.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE50 OFF;
ALTER TABLE ORDER_LINE50 DROP CONSTRAINT ORDER_LINE50CKC;
ALTER TABLE ORDER_LINE50 ADD CONSTRAINT ORDER_LINE50CKC CHECK
(OL_W_ID BETWEEN 125245 AND 127800);
SET INTEGRITY FOR ORDER_LINE50 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE51.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE51 OFF;
ALTER TABLE ORDER_LINE51 DROP CONSTRAINT ORDER_LINE51CKC;
ALTER TABLE ORDER_LINE51 ADD CONSTRAINT ORDER_LINE51CKC CHECK
(OL_W_ID BETWEEN 127801 AND 130356);
SET INTEGRITY FOR ORDER_LINE51 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE52.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE52 OFF;
ALTER TABLE ORDER_LINE52 DROP CONSTRAINT ORDER_LINE52CKC;
ALTER TABLE ORDER_LINE52 ADD CONSTRAINT ORDER_LINE52CKC CHECK
(OL_W_ID BETWEEN 130357 AND 132912);
SET INTEGRITY FOR ORDER_LINE52 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE53.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE53 OFF;
ALTER TABLE ORDER_LINE53 DROP CONSTRAINT ORDER_LINE53CKC;
ALTER TABLE ORDER_LINE53 ADD CONSTRAINT ORDER_LINE53CKC CHECK
(OL_W_ID BETWEEN 132913 AND 135468);
SET INTEGRITY FOR ORDER_LINE53 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE54.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE54 OFF;
ALTER TABLE ORDER_LINE54 DROP CONSTRAINT ORDER_LINE54CKC;
ALTER TABLE ORDER_LINE54 ADD CONSTRAINT ORDER_LINE54CKC CHECK
(OL_W_ID BETWEEN 135469 AND 138024);
SET INTEGRITY FOR ORDER_LINE54 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE55.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE55 OFF;
ALTER TABLE ORDER_LINE55 DROP CONSTRAINT ORDER_LINE55CKC;
ALTER TABLE ORDER_LINE55 ADD CONSTRAINT ORDER_LINE55CKC CHECK
(OL_W_ID BETWEEN 138025 AND 140580);
SET INTEGRITY FOR ORDER_LINE55 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE56.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE56 OFF;
ALTER TABLE ORDER_LINE56 DROP CONSTRAINT ORDER_LINE56CKC;
ALTER TABLE ORDER_LINE56 ADD CONSTRAINT ORDER_LINE56CKC CHECK
(OL_W_ID BETWEEN 140581 AND 143136);
SET INTEGRITY FOR ORDER_LINE56 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE57.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR ORDER_LINE57 OFF;
ALTER TABLE ORDER_LINE57 DROP CONSTRAINT ORDER_LINE57CKC;
ALTER TABLE ORDER_LINE57 ADD CONSTRAINT ORDER_LINE57CKC CHECK
(OL_W_ID BETWEEN 143137 AND 145692);
SET INTEGRITY FOR ORDER_LINE57 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE58.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE58 OFF;
ALTER TABLE ORDER_LINE58 DROP CONSTRAINT ORDER_LINE58CKC;
ALTER TABLE ORDER_LINE58 ADD CONSTRAINT ORDER_LINE58CKC CHECK
(OL_W_ID BETWEEN 145693 AND 148248);
SET INTEGRITY FOR ORDER_LINE58 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE59.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE59 OFF;
ALTER TABLE ORDER_LINE59 DROP CONSTRAINT ORDER_LINE59CKC;
ALTER TABLE ORDER_LINE59 ADD CONSTRAINT ORDER_LINE59CKC CHECK
(OL_W_ID BETWEEN 148249 AND 150804);
SET INTEGRITY FOR ORDER_LINE59 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE5.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE5 OFF;
ALTER TABLE ORDER_LINE5 DROP CONSTRAINT ORDER_LINE5CKC;
ALTER TABLE ORDER_LINE5 ADD CONSTRAINT ORDER_LINE5CKC CHECK
(OL_W_ID BETWEEN 10225 AND 12780);
SET INTEGRITY FOR ORDER_LINE5 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE60.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE60 OFF;
ALTER TABLE ORDER_LINE60 DROP CONSTRAINT ORDER_LINE60CKC;
ALTER TABLE ORDER_LINE60 ADD CONSTRAINT ORDER_LINE60CKC CHECK
(OL_W_ID BETWEEN 150805 AND 153360);
SET INTEGRITY FOR ORDER_LINE60 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE61.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE61 OFF;
ALTER TABLE ORDER_LINE61 DROP CONSTRAINT ORDER_LINE61CKC;
ALTER TABLE ORDER_LINE61 ADD CONSTRAINT ORDER_LINE61CKC CHECK
(OL_W_ID BETWEEN 153361 AND 155916);
SET INTEGRITY FOR ORDER_LINE61 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE62.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE62 OFF;
ALTER TABLE ORDER_LINE62 DROP CONSTRAINT ORDER_LINE62CKC;
ALTER TABLE ORDER_LINE62 ADD CONSTRAINT ORDER_LINE62CKC CHECK
(OL_W_ID BETWEEN 155917 AND 158472);
SET INTEGRITY FOR ORDER_LINE62 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE63.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE63 OFF;
ALTER TABLE ORDER_LINE63 DROP CONSTRAINT ORDER_LINE63CKC;
ALTER TABLE ORDER_LINE63 ADD CONSTRAINT ORDER_LINE63CKC CHECK
(OL_W_ID BETWEEN 158473 AND 161028);
SET INTEGRITY FOR ORDER_LINE63 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE64.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE64 OFF;
ALTER TABLE ORDER_LINE64 DROP CONSTRAINT ORDER_LINE64CKC;
ALTER TABLE ORDER_LINE64 ADD CONSTRAINT ORDER_LINE64CKC CHECK
(OL_W_ID BETWEEN 161029 AND 163584);
SET INTEGRITY FOR ORDER_LINE64 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE65.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE65 OFF;
ALTER TABLE ORDER_LINE65 DROP CONSTRAINT ORDER_LINE65CKC;
ALTER TABLE ORDER_LINE65 ADD CONSTRAINT ORDER_LINE65CKC CHECK
(OL_W_ID BETWEEN 163585 AND 166140);
SET INTEGRITY FOR ORDER_LINE65 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE66.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE66 OFF;
ALTER TABLE ORDER_LINE66 DROP CONSTRAINT ORDER_LINE66CKC;
ALTER TABLE ORDER_LINE66 ADD CONSTRAINT ORDER_LINE66CKC CHECK
(OL_W_ID BETWEEN 166141 AND 168696);
SET INTEGRITY FOR ORDER_LINE66 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE67.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE67 OFF;
ALTER TABLE ORDER_LINE67 DROP CONSTRAINT ORDER_LINE67CKC;
ALTER TABLE ORDER_LINE67 ADD CONSTRAINT ORDER_LINE67CKC CHECK
(OL_W_ID BETWEEN 168697 AND 171252);
SET INTEGRITY FOR ORDER_LINE67 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE68.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE68 OFF;
ALTER TABLE ORDER_LINE68 DROP CONSTRAINT ORDER_LINE68CKC;
ALTER TABLE ORDER_LINE68 ADD CONSTRAINT ORDER_LINE68CKC CHECK
(OL_W_ID BETWEEN 171253 AND 173808);
SET INTEGRITY FOR ORDER_LINE68 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE69.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE69 OFF;
ALTER TABLE ORDER_LINE69 DROP CONSTRAINT ORDER_LINE69CKC;
ALTER TABLE ORDER_LINE69 ADD CONSTRAINT ORDER_LINE69CKC CHECK
(OL_W_ID BETWEEN 173809 AND 176364);
SET INTEGRITY FOR ORDER_LINE69 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE6.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE6 OFF;
ALTER TABLE ORDER_LINE6 DROP CONSTRAINT ORDER_LINE6CKC;
ALTER TABLE ORDER_LINE6 ADD CONSTRAINT ORDER_LINE6CKC CHECK
(OL_W_ID BETWEEN 12781 AND 15336);
SET INTEGRITY FOR ORDER_LINE6 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE70.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE70 OFF;
ALTER TABLE ORDER_LINE70 DROP CONSTRAINT ORDER_LINE70CKC;
ALTER TABLE ORDER_LINE70 ADD CONSTRAINT ORDER_LINE70CKC CHECK
(OL_W_ID BETWEEN 176365 AND 178920);
SET INTEGRITY FOR ORDER_LINE70 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE71.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE71 OFF;
ALTER TABLE ORDER_LINE71 DROP CONSTRAINT ORDER_LINE71CKC;
ALTER TABLE ORDER_LINE71 ADD CONSTRAINT ORDER_LINE71CKC CHECK
(OL_W_ID BETWEEN 178921 AND 181476);
SET INTEGRITY FOR ORDER_LINE71 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE72.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE72 OFF;
ALTER TABLE ORDER_LINE72 DROP CONSTRAINT ORDER_LINE72CKC;
ALTER TABLE ORDER_LINE72 ADD CONSTRAINT ORDER_LINE72CKC CHECK
(OL_W_ID >= 181477);
SET INTEGRITY FOR ORDER_LINE72 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE7.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE7 OFF;
ALTER TABLE ORDER_LINE7 DROP CONSTRAINT ORDER_LINE7CKC;
ALTER TABLE ORDER_LINE7 ADD CONSTRAINT ORDER_LINE7CKC CHECK
(OL_W_ID BETWEEN 15337 AND 17892);
SET INTEGRITY FOR ORDER_LINE7 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE8.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE8 OFF;
ALTER TABLE ORDER_LINE8 DROP CONSTRAINT ORDER_LINE8CKC;
ALTER TABLE ORDER_LINE8 ADD CONSTRAINT ORDER_LINE8CKC CHECK
(OL_W_ID BETWEEN 17893 AND 20448);
```

```
SET INTEGRITY FOR ORDER_LINE8 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDER_LINE9.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE9 OFF;
ALTER TABLE ORDER_LINE9 DROP CONSTRAINT ORDER_LINE9CKC;
ALTER TABLE ORDER_LINE9 ADD CONSTRAINT ORDER_LINE9CKC CHECK
(OL_W_ID BETWEEN 20449 AND 23004);
SET INTEGRITY FOR ORDER_LINE9 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS10.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS10 OFF;
ALTER TABLE ORDERS10 DROP CONSTRAINT ORDERS10CKC;
ALTER TABLE ORDERS10 ADD CONSTRAINT ORDERS10CKC CHECK (O_W_ID
BETWEEN 46009 AND 51120);
SET INTEGRITY FOR ORDERS10 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS11.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS11 OFF;
ALTER TABLE ORDERS11 DROP CONSTRAINT ORDERS11CKC;
ALTER TABLE ORDERS11 ADD CONSTRAINT ORDERS11CKC CHECK (O_W_ID
BETWEEN 51121 AND 56232);
SET INTEGRITY FOR ORDERS11 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS12.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS12 OFF;
ALTER TABLE ORDERS12 DROP CONSTRAINT ORDERS12CKC;
ALTER TABLE ORDERS12 ADD CONSTRAINT ORDERS12CKC CHECK (O_W_ID
BETWEEN 56233 AND 61344);
SET INTEGRITY FOR ORDERS12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS13.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS13 OFF;
ALTER TABLE ORDERS13 DROP CONSTRAINT ORDERS13CKC;
ALTER TABLE ORDERS13 ADD CONSTRAINT ORDERS13CKC CHECK (O_W_ID
BETWEEN 61345 AND 66456);
SET INTEGRITY FOR ORDERS13 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS14.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS14 OFF;
ALTER TABLE ORDERS14 DROP CONSTRAINT ORDERS14CKC;
ALTER TABLE ORDERS14 ADD CONSTRAINT ORDERS14CKC CHECK (O_W_ID
BETWEEN 66457 AND 71568);
SET INTEGRITY FOR ORDERS14 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS15.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS15 OFF;
ALTER TABLE ORDERS15 DROP CONSTRAINT ORDERS15CKC;
ALTER TABLE ORDERS15 ADD CONSTRAINT ORDERS15CKC CHECK (O_W_ID
BETWEEN 71569 AND 76680);
SET INTEGRITY FOR ORDERS15 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS16.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS16 OFF;
ALTER TABLE ORDERS16 DROP CONSTRAINT ORDERS16CKC;
ALTER TABLE ORDERS16 ADD CONSTRAINT ORDERS16CKC CHECK (O_W_ID
BETWEEN 76681 AND 81792);
SET INTEGRITY FOR ORDERS16 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS17.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS17 OFF;
ALTER TABLE ORDERS17 DROP CONSTRAINT ORDERS17CKC;
ALTER TABLE ORDERS17 ADD CONSTRAINT ORDERS17CKC CHECK (O_W_ID
BETWEEN 81793 AND 86904);
SET INTEGRITY FOR ORDERS17 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS18.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS18 OFF;
ALTER TABLE ORDERS18 DROP CONSTRAINT ORDERS18CKC;
ALTER TABLE ORDERS18 ADD CONSTRAINT ORDERS18CKC CHECK (O_W_ID
BETWEEN 86905 AND 92016);
SET INTEGRITY FOR ORDERS18 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS19.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS19 OFF;
ALTER TABLE ORDERS19 DROP CONSTRAINT ORDERS19CKC;
ALTER TABLE ORDERS19 ADD CONSTRAINT ORDERS19CKC CHECK (O_W_ID
BETWEEN 92017 AND 97128);
SET INTEGRITY FOR ORDERS19 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS1.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS1 OFF;
ALTER TABLE ORDERS1 DROP CONSTRAINT ORDERS1CKC;
ALTER TABLE ORDERS1 ADD CONSTRAINT ORDERS1CKC CHECK (O_W_ID
BETWEEN 1 AND 5112);
SET INTEGRITY FOR ORDERS1 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS20.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS20 OFF;
ALTER TABLE ORDERS20 DROP CONSTRAINT ORDERS20CKC;
ALTER TABLE ORDERS20 ADD CONSTRAINT ORDERS20CKC CHECK (O_W_ID
BETWEEN 97129 AND 102240);
SET INTEGRITY FOR ORDERS20 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS21.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS21 OFF;
ALTER TABLE ORDERS21 DROP CONSTRAINT ORDERS21CKC;
ALTER TABLE ORDERS21 ADD CONSTRAINT ORDERS21CKC CHECK (O_W_ID
BETWEEN 102241 AND 107352);
SET INTEGRITY FOR ORDERS21 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS22.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS22 OFF;
ALTER TABLE ORDERS22 DROP CONSTRAINT ORDERS22CKC;
ALTER TABLE ORDERS22 ADD CONSTRAINT ORDERS22CKC CHECK (O_W_ID
BETWEEN 107353 AND 112464);
SET INTEGRITY FOR ORDERS22 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS23.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS23 OFF;
ALTER TABLE ORDERS23 DROP CONSTRAINT ORDERS23CKC;
ALTER TABLE ORDERS23 ADD CONSTRAINT ORDERS23CKC CHECK (O_W_ID
BETWEEN 112465 AND 117576);
SET INTEGRITY FOR ORDERS23 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS24.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS24 OFF;
ALTER TABLE ORDERS24 DROP CONSTRAINT ORDERS24CKC;
ALTER TABLE ORDERS24 ADD CONSTRAINT ORDERS24CKC CHECK (O_W_ID
BETWEEN 117577 AND 122688);
SET INTEGRITY FOR ORDERS24 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS25.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS25 OFF;
ALTER TABLE ORDERS25 DROP CONSTRAINT ORDERS25CKC;
ALTER TABLE ORDERS25 ADD CONSTRAINT ORDERS25CKC CHECK (O_W_ID
BETWEEN 122689 AND 127800);
SET INTEGRITY FOR ORDERS25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS26.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS26 OFF;
ALTER TABLE ORDERS26 DROP CONSTRAINT ORDERS26CKC;
ALTER TABLE ORDERS26 ADD CONSTRAINT ORDERS26CKC CHECK (O_W_ID
BETWEEN 127801 AND 132912);
SET INTEGRITY FOR ORDERS26 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS27.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS27 OFF;
ALTER TABLE ORDERS27 DROP CONSTRAINT ORDERS27CKC;
ALTER TABLE ORDERS27 ADD CONSTRAINT ORDERS27CKC CHECK (O_W_ID
BETWEEN 132913 AND 138024);
SET INTEGRITY FOR ORDERS27 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS28.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS28 OFF;
ALTER TABLE ORDERS28 DROP CONSTRAINT ORDERS28CKC;
ALTER TABLE ORDERS28 ADD CONSTRAINT ORDERS28CKC CHECK (O_W_ID
BETWEEN 138025 AND 143136);
SET INTEGRITY FOR ORDERS28 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS29.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS29 OFF;
ALTER TABLE ORDERS29 DROP CONSTRAINT ORDERS29CKC;
ALTER TABLE ORDERS29 ADD CONSTRAINT ORDERS29CKC CHECK (O_W_ID
BETWEEN 143137 AND 148248);
SET INTEGRITY FOR ORDERS29 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS2.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS2 OFF;
ALTER TABLE ORDERS2 DROP CONSTRAINT ORDERS2CKC;
ALTER TABLE ORDERS2 ADD CONSTRAINT ORDERS2CKC CHECK (O_W_ID
BETWEEN 5113 AND 10224);
SET INTEGRITY FOR ORDERS2 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS30.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS30 OFF;
ALTER TABLE ORDERS30 DROP CONSTRAINT ORDERS30CKC;
ALTER TABLE ORDERS30 ADD CONSTRAINT ORDERS30CKC CHECK (O_W_ID
BETWEEN 148249 AND 153360);
SET INTEGRITY FOR ORDERS30 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS31.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS31 OFF;
ALTER TABLE ORDERS31 DROP CONSTRAINT ORDERS31CKC;
ALTER TABLE ORDERS31 ADD CONSTRAINT ORDERS31CKC CHECK (O_W_ID
BETWEEN 153361 AND 158472);
SET INTEGRITY FOR ORDERS31 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS32.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS32 OFF;
ALTER TABLE ORDERS32 DROP CONSTRAINT ORDERS32CKC;
ALTER TABLE ORDERS32 ADD CONSTRAINT ORDERS32CKC CHECK (O_W_ID
BETWEEN 158473 AND 163584);
SET INTEGRITY FOR ORDERS32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS33.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS33 OFF;
ALTER TABLE ORDERS33 DROP CONSTRAINT ORDERS33CKC;
ALTER TABLE ORDERS33 ADD CONSTRAINT ORDERS33CKC CHECK (O_W_ID
BETWEEN 163585 AND 168696);
SET INTEGRITY FOR ORDERS33 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS34.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS34 OFF;
ALTER TABLE ORDERS34 DROP CONSTRAINT ORDERS34CKC;
ALTER TABLE ORDERS34 ADD CONSTRAINT ORDERS34CKC CHECK (O_W_ID
BETWEEN 168697 AND 173808);
SET INTEGRITY FOR ORDERS34 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS35.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS35 OFF;
ALTER TABLE ORDERS35 DROP CONSTRAINT ORDERS35CKC;
ALTER TABLE ORDERS35 ADD CONSTRAINT ORDERS35CKC CHECK (O_W_ID
BETWEEN 173809 AND 178920);
SET INTEGRITY FOR ORDERS35 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS36.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS36 OFF;
ALTER TABLE ORDERS36 DROP CONSTRAINT ORDERS36CKC;
ALTER TABLE ORDERS36 ADD CONSTRAINT ORDERS36CKC CHECK (O_W_ID >=
178921);
SET INTEGRITY FOR ORDERS36 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS3.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS3 OFF;
ALTER TABLE ORDERS3 DROP CONSTRAINT ORDERS3CKC;
ALTER TABLE ORDERS3 ADD CONSTRAINT ORDERS3CKC CHECK (O_W_ID
BETWEEN 10225 AND 15336);
SET INTEGRITY FOR ORDERS3 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS4.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS4 OFF;
ALTER TABLE ORDERS4 DROP CONSTRAINT ORDERS4CKC;
ALTER TABLE ORDERS4 ADD CONSTRAINT ORDERS4CKC CHECK (O_W_ID
BETWEEN 15337 AND 20448);
SET INTEGRITY FOR ORDERS4 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS5.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS5 OFF;
ALTER TABLE ORDERS5 DROP CONSTRAINT ORDERS5CKC;
ALTER TABLE ORDERS5 ADD CONSTRAINT ORDERS5CKC CHECK (O_W_ID
BETWEEN 20449 AND 25560);
SET INTEGRITY FOR ORDERS5 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS6.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS6 OFF;
ALTER TABLE ORDERS6 DROP CONSTRAINT ORDERS6CKC;
ALTER TABLE ORDERS6 ADD CONSTRAINT ORDERS6CKC CHECK (O_W_ID
BETWEEN 25561 AND 30672);
SET INTEGRITY FOR ORDERS6 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS7.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS7 OFF;
ALTER TABLE ORDERS7 DROP CONSTRAINT ORDERS7CKC;
ALTER TABLE ORDERS7 ADD CONSTRAINT ORDERS7CKC CHECK (O_W_ID
BETWEEN 30673 AND 35784);
SET INTEGRITY FOR ORDERS7 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS8.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS8 OFF;
ALTER TABLE ORDERS8 DROP CONSTRAINT ORDERS8CKC;
ALTER TABLE ORDERS8 ADD CONSTRAINT ORDERS8CKC CHECK (O_W_ID
BETWEEN 35785 AND 40896);
SET INTEGRITY FOR ORDERS8 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_ORDERS9.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS9 OFF;
ALTER TABLE ORDERS9 DROP CONSTRAINT ORDERS9CKC;
ALTER TABLE ORDERS9 ADD CONSTRAINT ORDERS9CKC CHECK (O_W_ID
BETWEEN 40897 AND 46008);
SET INTEGRITY FOR ORDERS9 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK100.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK100 OFF;
ALTER TABLE STOCK100 DROP CONSTRAINT STOCK100CKC;
ALTER TABLE STOCK100 ADD CONSTRAINT STOCK100CKC CHECK (S_W_ID
BETWEEN 126523 AND 127800);
SET INTEGRITY FOR STOCK100 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK101.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR STOCK101 OFF;
ALTER TABLE STOCK101 DROP CONSTRAINT STOCK101CKC;
ALTER TABLE STOCK101 ADD CONSTRAINT STOCK101CKC CHECK (S_W_ID
BETWEEN 127801 AND 129078);
SET INTEGRITY FOR STOCK101 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK102.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK102 OFF;
ALTER TABLE STOCK102 DROP CONSTRAINT STOCK102CKC;
ALTER TABLE STOCK102 ADD CONSTRAINT STOCK102CKC CHECK (S_W_ID
BETWEEN 129079 AND 130356);
SET INTEGRITY FOR STOCK102 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK103.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK103 OFF;
ALTER TABLE STOCK103 DROP CONSTRAINT STOCK103CKC;
ALTER TABLE STOCK103 ADD CONSTRAINT STOCK103CKC CHECK (S_W_ID
BETWEEN 130357 AND 131634);
SET INTEGRITY FOR STOCK103 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK104.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK104 OFF;
ALTER TABLE STOCK104 DROP CONSTRAINT STOCK104CKC;
ALTER TABLE STOCK104 ADD CONSTRAINT STOCK104CKC CHECK (S_W_ID
BETWEEN 131635 AND 132912);
SET INTEGRITY FOR STOCK104 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK105.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK105 OFF;
ALTER TABLE STOCK105 DROP CONSTRAINT STOCK105CKC;
ALTER TABLE STOCK105 ADD CONSTRAINT STOCK105CKC CHECK (S_W_ID
BETWEEN 132913 AND 134190);
SET INTEGRITY FOR STOCK105 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK106.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK106 OFF;
ALTER TABLE STOCK106 DROP CONSTRAINT STOCK106CKC;
ALTER TABLE STOCK106 ADD CONSTRAINT STOCK106CKC CHECK (S_W_ID
BETWEEN 134191 AND 135468);
SET INTEGRITY FOR STOCK106 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK107.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK107 OFF;
ALTER TABLE STOCK107 DROP CONSTRAINT STOCK107CKC;
ALTER TABLE STOCK107 ADD CONSTRAINT STOCK107CKC CHECK (S_W_ID
BETWEEN 135469 AND 136746);
SET INTEGRITY FOR STOCK107 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK108.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK108 OFF;
ALTER TABLE STOCK108 DROP CONSTRAINT STOCK108CKC;
ALTER TABLE STOCK108 ADD CONSTRAINT STOCK108CKC CHECK (S_W_ID
BETWEEN 136747 AND 138024);
SET INTEGRITY FOR STOCK108 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK109.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK109 OFF;
ALTER TABLE STOCK109 DROP CONSTRAINT STOCK109CKC;
ALTER TABLE STOCK109 ADD CONSTRAINT STOCK109CKC CHECK (S_W_ID
BETWEEN 138025 AND 139302);
SET INTEGRITY FOR STOCK109 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK10.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK10 OFF;
ALTER TABLE STOCK10 DROP CONSTRAINT STOCK10CKC;
ALTER TABLE STOCK10 ADD CONSTRAINT STOCK10CKC CHECK (S_W_ID
BETWEEN 11503 AND 12780);
SET INTEGRITY FOR STOCK10 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK110.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK110 OFF;
ALTER TABLE STOCK110 DROP CONSTRAINT STOCK110CKC;
ALTER TABLE STOCK110 ADD CONSTRAINT STOCK110CKC CHECK (S_W_ID
BETWEEN 139303 AND 140580);
SET INTEGRITY FOR STOCK110 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK111.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK111 OFF;
ALTER TABLE STOCK111 DROP CONSTRAINT STOCK111CKC;
ALTER TABLE STOCK111 ADD CONSTRAINT STOCK111CKC CHECK (S_W_ID
BETWEEN 140581 AND 141858);
SET INTEGRITY FOR STOCK111 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK112.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK112 OFF;
ALTER TABLE STOCK112 DROP CONSTRAINT STOCK112CKC;
ALTER TABLE STOCK112 ADD CONSTRAINT STOCK112CKC CHECK (S_W_ID
BETWEEN 141859 AND 143136);
SET INTEGRITY FOR STOCK112 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK113.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK113 OFF;
ALTER TABLE STOCK113 DROP CONSTRAINT STOCK113CKC;
ALTER TABLE STOCK113 ADD CONSTRAINT STOCK113CKC CHECK (S_W_ID
BETWEEN 143137 AND 144414);
SET INTEGRITY FOR STOCK113 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK114.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK114 OFF;
ALTER TABLE STOCK114 DROP CONSTRAINT STOCK114CKC;
ALTER TABLE STOCK114 ADD CONSTRAINT STOCK114CKC CHECK (S_W_ID
BETWEEN 144415 AND 145692);
SET INTEGRITY FOR STOCK114 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK115.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK115 OFF;
ALTER TABLE STOCK115 DROP CONSTRAINT STOCK115CKC;
ALTER TABLE STOCK115 ADD CONSTRAINT STOCK115CKC CHECK (S_W_ID
BETWEEN 145693 AND 146970);
SET INTEGRITY FOR STOCK115 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK116.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK116 OFF;
ALTER TABLE STOCK116 DROP CONSTRAINT STOCK116CKC;
ALTER TABLE STOCK116 ADD CONSTRAINT STOCK116CKC CHECK (S_W_ID
BETWEEN 146971 AND 148248);
SET INTEGRITY FOR STOCK116 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK117.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK117 OFF;
ALTER TABLE STOCK117 DROP CONSTRAINT STOCK117CKC;
ALTER TABLE STOCK117 ADD CONSTRAINT STOCK117CKC CHECK (S_W_ID
BETWEEN 148249 AND 149526);
SET INTEGRITY FOR STOCK117 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK118.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK118 OFF;
ALTER TABLE STOCK118 DROP CONSTRAINT STOCK118CKC;
ALTER TABLE STOCK118 ADD CONSTRAINT STOCK118CKC CHECK (S_W_ID
BETWEEN 149527 AND 150804);
SET INTEGRITY FOR STOCK118 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK119.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK119 OFF;
ALTER TABLE STOCK119 DROP CONSTRAINT STOCK119CKC;
ALTER TABLE STOCK119 ADD CONSTRAINT STOCK119CKC CHECK (S_W_ID
BETWEEN 150805 AND 152082);
```

```
SET INTEGRITY FOR STOCK119 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK11.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK11 OFF;
ALTER TABLE STOCK11 DROP CONSTRAINT STOCK11CKC;
ALTER TABLE STOCK11 ADD CONSTRAINT STOCK11CKC CHECK (S_W_ID
BETWEEN 12781 AND 14058);
SET INTEGRITY FOR STOCK11 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK120.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK120 OFF;
ALTER TABLE STOCK120 DROP CONSTRAINT STOCK120CKC;
ALTER TABLE STOCK120 ADD CONSTRAINT STOCK120CKC CHECK (S_W_ID
BETWEEN 152083 AND 153360);
SET INTEGRITY FOR STOCK120 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK121.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK121 OFF;
ALTER TABLE STOCK121 DROP CONSTRAINT STOCK121CKC;
ALTER TABLE STOCK121 ADD CONSTRAINT STOCK121CKC CHECK (S_W_ID
BETWEEN 153361 AND 154638);
SET INTEGRITY FOR STOCK121 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK122.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK122 OFF;
ALTER TABLE STOCK122 DROP CONSTRAINT STOCK122CKC;
ALTER TABLE STOCK122 ADD CONSTRAINT STOCK122CKC CHECK (S_W_ID
BETWEEN 154639 AND 155916);
SET INTEGRITY FOR STOCK122 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK123.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK123 OFF;
ALTER TABLE STOCK123 DROP CONSTRAINT STOCK123CKC;
ALTER TABLE STOCK123 ADD CONSTRAINT STOCK123CKC CHECK (S_W_ID
BETWEEN 155917 AND 157194);
SET INTEGRITY FOR STOCK123 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK124.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK124 OFF;
ALTER TABLE STOCK124 DROP CONSTRAINT STOCK124CKC;
ALTER TABLE STOCK124 ADD CONSTRAINT STOCK124CKC CHECK (S_W_ID
BETWEEN 157195 AND 158472);
SET INTEGRITY FOR STOCK124 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK125.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK125 OFF;
ALTER TABLE STOCK125 DROP CONSTRAINT STOCK125CKC;
ALTER TABLE STOCK125 ADD CONSTRAINT STOCK125CKC CHECK (S_W_ID
BETWEEN 158473 AND 159750);
SET INTEGRITY FOR STOCK125 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK126.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK126 OFF;
ALTER TABLE STOCK126 DROP CONSTRAINT STOCK126CKC;
ALTER TABLE STOCK126 ADD CONSTRAINT STOCK126CKC CHECK (S_W_ID
BETWEEN 159751 AND 161028);
SET INTEGRITY FOR STOCK126 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK127.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK127 OFF;
ALTER TABLE STOCK127 DROP CONSTRAINT STOCK127CKC;
ALTER TABLE STOCK127 ADD CONSTRAINT STOCK127CKC CHECK (S_W_ID
BETWEEN 161029 AND 162306);
SET INTEGRITY FOR STOCK127 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK128.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK128 OFF;
ALTER TABLE STOCK128 DROP CONSTRAINT STOCK128CKC;
ALTER TABLE STOCK128 ADD CONSTRAINT STOCK128CKC CHECK (S_W_ID
BETWEEN 162307 AND 163584);
SET INTEGRITY FOR STOCK128 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK129.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK129 OFF;
ALTER TABLE STOCK129 DROP CONSTRAINT STOCK129CKC;
ALTER TABLE STOCK129 ADD CONSTRAINT STOCK129CKC CHECK (S_W_ID
BETWEEN 163585 AND 164862);
SET INTEGRITY FOR STOCK129 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK12.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK12 OFF;
ALTER TABLE STOCK12 DROP CONSTRAINT STOCK12CKC;
ALTER TABLE STOCK12 ADD CONSTRAINT STOCK12CKC CHECK (S_W_ID
BETWEEN 14059 AND 15336);
SET INTEGRITY FOR STOCK12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK130.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK130 OFF;
ALTER TABLE STOCK130 DROP CONSTRAINT STOCK130CKC;
ALTER TABLE STOCK130 ADD CONSTRAINT STOCK130CKC CHECK (S_W_ID
BETWEEN 164863 AND 166140);
SET INTEGRITY FOR STOCK130 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK131.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK131 OFF;
ALTER TABLE STOCK131 DROP CONSTRAINT STOCK131CKC;
ALTER TABLE STOCK131 ADD CONSTRAINT STOCK131CKC CHECK (S_W_ID
BETWEEN 166141 AND 167418);
SET INTEGRITY FOR STOCK131 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK132.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK132 OFF;
ALTER TABLE STOCK132 DROP CONSTRAINT STOCK132CKC;
ALTER TABLE STOCK132 ADD CONSTRAINT STOCK132CKC CHECK (S_W_ID
BETWEEN 167419 AND 168696);
SET INTEGRITY FOR STOCK132 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK133.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK133 OFF;
ALTER TABLE STOCK133 DROP CONSTRAINT STOCK133CKC;
ALTER TABLE STOCK133 ADD CONSTRAINT STOCK133CKC CHECK (S_W_ID
BETWEEN 168697 AND 169974);
SET INTEGRITY FOR STOCK133 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK134.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK134 OFF;
ALTER TABLE STOCK134 DROP CONSTRAINT STOCK134CKC;
ALTER TABLE STOCK134 ADD CONSTRAINT STOCK134CKC CHECK (S_W_ID
BETWEEN 169975 AND 171252);
SET INTEGRITY FOR STOCK134 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK135.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK135 OFF;
ALTER TABLE STOCK135 DROP CONSTRAINT STOCK135CKC;
ALTER TABLE STOCK135 ADD CONSTRAINT STOCK135CKC CHECK (S_W_ID
BETWEEN 171253 AND 172530);
SET INTEGRITY FOR STOCK135 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK136.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK136 OFF;
ALTER TABLE STOCK136 DROP CONSTRAINT STOCK136CKC;
ALTER TABLE STOCK136 ADD CONSTRAINT STOCK136CKC CHECK (S_W_ID
BETWEEN 172531 AND 173808);
SET INTEGRITY FOR STOCK136 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK137.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK137 OFF;
ALTER TABLE STOCK137 DROP CONSTRAINT STOCK137CKC;
ALTER TABLE STOCK137 ADD CONSTRAINT STOCK137CKC CHECK (S_W_ID
BETWEEN 173809 AND 175086);
SET INTEGRITY FOR STOCK137 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK138.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK138 OFF;
ALTER TABLE STOCK138 DROP CONSTRAINT STOCK138CKC;
ALTER TABLE STOCK138 ADD CONSTRAINT STOCK138CKC CHECK (S_W_ID
BETWEEN 175087 AND 176364);
SET INTEGRITY FOR STOCK138 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK139.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK139 OFF;
ALTER TABLE STOCK139 DROP CONSTRAINT STOCK139CKC;
ALTER TABLE STOCK139 ADD CONSTRAINT STOCK139CKC CHECK (S_W_ID
BETWEEN 176365 AND 177642);
SET INTEGRITY FOR STOCK139 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK13.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK13 OFF;
ALTER TABLE STOCK13 DROP CONSTRAINT STOCK13CKC;
ALTER TABLE STOCK13 ADD CONSTRAINT STOCK13CKC CHECK (S_W_ID
BETWEEN 15337 AND 16614);
SET INTEGRITY FOR STOCK13 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK140.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK140 OFF;
ALTER TABLE STOCK140 DROP CONSTRAINT STOCK140CKC;
ALTER TABLE STOCK140 ADD CONSTRAINT STOCK140CKC CHECK (S_W_ID
BETWEEN 177643 AND 178920);
SET INTEGRITY FOR STOCK140 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK141.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK141 OFF;
ALTER TABLE STOCK141 DROP CONSTRAINT STOCK141CKC;
ALTER TABLE STOCK141 ADD CONSTRAINT STOCK141CKC CHECK (S_W_ID
BETWEEN 178921 AND 180198);
SET INTEGRITY FOR STOCK141 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK142.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK142 OFF;
ALTER TABLE STOCK142 DROP CONSTRAINT STOCK142CKC;
ALTER TABLE STOCK142 ADD CONSTRAINT STOCK142CKC CHECK (S_W_ID
BETWEEN 180199 AND 181476);
SET INTEGRITY FOR STOCK142 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK143.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK143 OFF;
ALTER TABLE STOCK143 DROP CONSTRAINT STOCK143CKC;
ALTER TABLE STOCK143 ADD CONSTRAINT STOCK143CKC CHECK (S_W_ID
BETWEEN 181477 AND 182754);
SET INTEGRITY FOR STOCK143 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK144.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK144 OFF;
ALTER TABLE STOCK144 DROP CONSTRAINT STOCK144CKC;
ALTER TABLE STOCK144 ADD CONSTRAINT STOCK144CKC CHECK (S_W_ID >=
182755);
SET INTEGRITY FOR STOCK144 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK14.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK14 OFF;
ALTER TABLE STOCK14 DROP CONSTRAINT STOCK14CKC;
ALTER TABLE STOCK14 ADD CONSTRAINT STOCK14CKC CHECK (S_W_ID
BETWEEN 16615 AND 17892);
SET INTEGRITY FOR STOCK14 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK15.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK15 OFF;
ALTER TABLE STOCK15 DROP CONSTRAINT STOCK15CKC;
ALTER TABLE STOCK15 ADD CONSTRAINT STOCK15CKC CHECK (S_W_ID
BETWEEN 17893 AND 19170);
SET INTEGRITY FOR STOCK15 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK16.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK16 OFF;
ALTER TABLE STOCK16 DROP CONSTRAINT STOCK16CKC;
ALTER TABLE STOCK16 ADD CONSTRAINT STOCK16CKC CHECK (S_W_ID
BETWEEN 19171 AND 20448);
SET INTEGRITY FOR STOCK16 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK17.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK17 OFF;
ALTER TABLE STOCK17 DROP CONSTRAINT STOCK17CKC;
ALTER TABLE STOCK17 ADD CONSTRAINT STOCK17CKC CHECK (S_W_ID
BETWEEN 20449 AND 21726);
SET INTEGRITY FOR STOCK17 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK18.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK18 OFF;
ALTER TABLE STOCK18 DROP CONSTRAINT STOCK18CKC;
ALTER TABLE STOCK18 ADD CONSTRAINT STOCK18CKC CHECK (S_W_ID
BETWEEN 21727 AND 23004);
SET INTEGRITY FOR STOCK18 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK19.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK19 OFF;
ALTER TABLE STOCK19 DROP CONSTRAINT STOCK19CKC;
ALTER TABLE STOCK19 ADD CONSTRAINT STOCK19CKC CHECK (S_W_ID
BETWEEN 23005 AND 24282);
SET INTEGRITY FOR STOCK19 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK1.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK1 OFF;
ALTER TABLE STOCK1 DROP CONSTRAINT STOCK1CKC;
ALTER TABLE STOCK1 ADD CONSTRAINT STOCK1CKC CHECK (S_W_ID BETWEEN
1 AND 1278);
SET INTEGRITY FOR STOCK1 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK20.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK20 OFF;
ALTER TABLE STOCK20 DROP CONSTRAINT STOCK20CKC;
ALTER TABLE STOCK20 ADD CONSTRAINT STOCK20CKC CHECK (S_W_ID
BETWEEN 24283 AND 25560);
SET INTEGRITY FOR STOCK20 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK21.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK21 OFF;
ALTER TABLE STOCK21 DROP CONSTRAINT STOCK21CKC;
ALTER TABLE STOCK21 ADD CONSTRAINT STOCK21CKC CHECK (S_W_ID
BETWEEN 25561 AND 26838);
SET INTEGRITY FOR STOCK21 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK22.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK22 OFF;
ALTER TABLE STOCK22 DROP CONSTRAINT STOCK22CKC;
ALTER TABLE STOCK22 ADD CONSTRAINT STOCK22CKC CHECK (S_W_ID
BETWEEN 26839 AND 28116);
SET INTEGRITY FOR STOCK22 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK23.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR STOCK23 OFF;
ALTER TABLE STOCK23 DROP CONSTRAINT STOCK23CKC;
ALTER TABLE STOCK23 ADD CONSTRAINT STOCK23CKC CHECK (S_W_ID
BETWEEN 28117 AND 29394);
SET INTEGRITY FOR STOCK23 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK24.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK24 OFF;
ALTER TABLE STOCK24 DROP CONSTRAINT STOCK24CKC;
ALTER TABLE STOCK24 ADD CONSTRAINT STOCK24CKC CHECK (S_W_ID
BETWEEN 29395 AND 30672);
SET INTEGRITY FOR STOCK24 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK25.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK25 OFF;
ALTER TABLE STOCK25 DROP CONSTRAINT STOCK25CKC;
ALTER TABLE STOCK25 ADD CONSTRAINT STOCK25CKC CHECK (S_W_ID
BETWEEN 30673 AND 31950);
SET INTEGRITY FOR STOCK25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK26.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK26 OFF;
ALTER TABLE STOCK26 DROP CONSTRAINT STOCK26CKC;
ALTER TABLE STOCK26 ADD CONSTRAINT STOCK26CKC CHECK (S_W_ID
BETWEEN 31951 AND 33228);
SET INTEGRITY FOR STOCK26 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK27.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK27 OFF;
ALTER TABLE STOCK27 DROP CONSTRAINT STOCK27CKC;
ALTER TABLE STOCK27 ADD CONSTRAINT STOCK27CKC CHECK (S_W_ID
BETWEEN 33229 AND 34506);
SET INTEGRITY FOR STOCK27 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK28.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK28 OFF;
ALTER TABLE STOCK28 DROP CONSTRAINT STOCK28CKC;
ALTER TABLE STOCK28 ADD CONSTRAINT STOCK28CKC CHECK (S_W_ID
BETWEEN 34507 AND 35784);
SET INTEGRITY FOR STOCK28 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK29.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK29 OFF;
ALTER TABLE STOCK29 DROP CONSTRAINT STOCK29CKC;
ALTER TABLE STOCK29 ADD CONSTRAINT STOCK29CKC CHECK (S_W_ID
BETWEEN 35785 AND 37062);
SET INTEGRITY FOR STOCK29 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK2.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK2 OFF;
ALTER TABLE STOCK2 DROP CONSTRAINT STOCK2CKC;
ALTER TABLE STOCK2 ADD CONSTRAINT STOCK2CKC CHECK (S_W_ID BETWEEN
1279 AND 2556);
SET INTEGRITY FOR STOCK2 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK30.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK30 OFF;
ALTER TABLE STOCK30 DROP CONSTRAINT STOCK30CKC;
ALTER TABLE STOCK30 ADD CONSTRAINT STOCK30CKC CHECK (S_W_ID
BETWEEN 37063 AND 38340);
SET INTEGRITY FOR STOCK30 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK31.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK31 OFF;
ALTER TABLE STOCK31 DROP CONSTRAINT STOCK31CKC;
ALTER TABLE STOCK31 ADD CONSTRAINT STOCK31CKC CHECK (S_W_ID
BETWEEN 38341 AND 39618);
SET INTEGRITY FOR STOCK31 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK32.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK32 OFF;
ALTER TABLE STOCK32 DROP CONSTRAINT STOCK32CKC;
ALTER TABLE STOCK32 ADD CONSTRAINT STOCK32CKC CHECK (S_W_ID
BETWEEN 39619 AND 40896);
SET INTEGRITY FOR STOCK32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK33.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK33 OFF;
ALTER TABLE STOCK33 DROP CONSTRAINT STOCK33CKC;
ALTER TABLE STOCK33 ADD CONSTRAINT STOCK33CKC CHECK (S_W_ID
BETWEEN 40897 AND 42174);
SET INTEGRITY FOR STOCK33 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK34.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK34 OFF;
ALTER TABLE STOCK34 DROP CONSTRAINT STOCK34CKC;
ALTER TABLE STOCK34 ADD CONSTRAINT STOCK34CKC CHECK (S_W_ID
BETWEEN 42175 AND 43452);
SET INTEGRITY FOR STOCK34 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK35.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK35 OFF;
ALTER TABLE STOCK35 DROP CONSTRAINT STOCK35CKC;
ALTER TABLE STOCK35 ADD CONSTRAINT STOCK35CKC CHECK (S_W_ID
BETWEEN 43453 AND 44730);
SET INTEGRITY FOR STOCK35 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK36.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK36 OFF;
ALTER TABLE STOCK36 DROP CONSTRAINT STOCK36CKC;
ALTER TABLE STOCK36 ADD CONSTRAINT STOCK36CKC CHECK (S_W_ID
BETWEEN 44731 AND 46008);
SET INTEGRITY FOR STOCK36 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK37.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK37 OFF;
ALTER TABLE STOCK37 DROP CONSTRAINT STOCK37CKC;
ALTER TABLE STOCK37 ADD CONSTRAINT STOCK37CKC CHECK (S_W_ID
BETWEEN 46009 AND 47286);
SET INTEGRITY FOR STOCK37 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK38.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK38 OFF;
ALTER TABLE STOCK38 DROP CONSTRAINT STOCK38CKC;
ALTER TABLE STOCK38 ADD CONSTRAINT STOCK38CKC CHECK (S_W_ID
BETWEEN 47287 AND 48564);
SET INTEGRITY FOR STOCK38 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK39.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK39 OFF;
ALTER TABLE STOCK39 DROP CONSTRAINT STOCK39CKC;
ALTER TABLE STOCK39 ADD CONSTRAINT STOCK39CKC CHECK (S_W_ID
BETWEEN 48565 AND 49842);
SET INTEGRITY FOR STOCK39 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK3.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK3 OFF;
ALTER TABLE STOCK3 DROP CONSTRAINT STOCK3CKC;
ALTER TABLE STOCK3 ADD CONSTRAINT STOCK3CKC CHECK (S_W_ID BETWEEN
2557 AND 3834);
SET INTEGRITY FOR STOCK3 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK40.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK40 OFF;
ALTER TABLE STOCK40 DROP CONSTRAINT STOCK40CKC;
ALTER TABLE STOCK40 ADD CONSTRAINT STOCK40CKC CHECK (S_W_ID
BETWEEN 49843 AND 51120);
```

```
SET INTEGRITY FOR STOCK40 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK41.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK41 OFF;
ALTER TABLE STOCK41 DROP CONSTRAINT STOCK41CKC;
ALTER TABLE STOCK41 ADD CONSTRAINT STOCK41CKC CHECK (S_W_ID
BETWEEN 51121 AND 52398);
SET INTEGRITY FOR STOCK41 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK42.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK42 OFF;
ALTER TABLE STOCK42 DROP CONSTRAINT STOCK42CKC;
ALTER TABLE STOCK42 ADD CONSTRAINT STOCK42CKC CHECK (S_W_ID
BETWEEN 52399 AND 53676);
SET INTEGRITY FOR STOCK42 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK43.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK43 OFF;
ALTER TABLE STOCK43 DROP CONSTRAINT STOCK43CKC;
ALTER TABLE STOCK43 ADD CONSTRAINT STOCK43CKC CHECK (S_W_ID
BETWEEN 53677 AND 54954);
SET INTEGRITY FOR STOCK43 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK44.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK44 OFF;
ALTER TABLE STOCK44 DROP CONSTRAINT STOCK44CKC;
ALTER TABLE STOCK44 ADD CONSTRAINT STOCK44CKC CHECK (S_W_ID
BETWEEN 54955 AND 56232);
SET INTEGRITY FOR STOCK44 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK45.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK45 OFF;
ALTER TABLE STOCK45 DROP CONSTRAINT STOCK45CKC;
ALTER TABLE STOCK45 ADD CONSTRAINT STOCK45CKC CHECK (S_W_ID
BETWEEN 56233 AND 57510);
SET INTEGRITY FOR STOCK45 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK46.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK46 OFF;
ALTER TABLE STOCK46 DROP CONSTRAINT STOCK46CKC;
ALTER TABLE STOCK46 ADD CONSTRAINT STOCK46CKC CHECK (S_W_ID
BETWEEN 57511 AND 58788);
SET INTEGRITY FOR STOCK46 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK47.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK47 OFF;
ALTER TABLE STOCK47 DROP CONSTRAINT STOCK47CKC;
ALTER TABLE STOCK47 ADD CONSTRAINT STOCK47CKC CHECK (S_W_ID
BETWEEN 58789 AND 60066);
SET INTEGRITY FOR STOCK47 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK48.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK48 OFF;
ALTER TABLE STOCK48 DROP CONSTRAINT STOCK48CKC;
ALTER TABLE STOCK48 ADD CONSTRAINT STOCK48CKC CHECK (S_W_ID
BETWEEN 60067 AND 61344);
SET INTEGRITY FOR STOCK48 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK49.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK49 OFF;
ALTER TABLE STOCK49 DROP CONSTRAINT STOCK49CKC;
ALTER TABLE STOCK49 ADD CONSTRAINT STOCK49CKC CHECK (S_W_ID
BETWEEN 61345 AND 62622);
SET INTEGRITY FOR STOCK49 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK4.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK4 OFF;
ALTER TABLE STOCK4 DROP CONSTRAINT STOCK4CKC;
ALTER TABLE STOCK4 ADD CONSTRAINT STOCK4CKC CHECK (S_W_ID BETWEEN
3835 AND 5112);
SET INTEGRITY FOR STOCK4 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK50.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK50 OFF;
ALTER TABLE STOCK50 DROP CONSTRAINT STOCK50CKC;
ALTER TABLE STOCK50 ADD CONSTRAINT STOCK50CKC CHECK (S_W_ID
BETWEEN 62623 AND 63900);
SET INTEGRITY FOR STOCK50 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK51.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK51 OFF;
ALTER TABLE STOCK51 DROP CONSTRAINT STOCK51CKC;
ALTER TABLE STOCK51 ADD CONSTRAINT STOCK51CKC CHECK (S_W_ID
BETWEEN 63901 AND 65178);
SET INTEGRITY FOR STOCK51 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK52.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK52 OFF;
ALTER TABLE STOCK52 DROP CONSTRAINT STOCK52CKC;
ALTER TABLE STOCK52 ADD CONSTRAINT STOCK52CKC CHECK (S_W_ID
BETWEEN 65179 AND 66456);
SET INTEGRITY FOR STOCK52 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK53.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK53 OFF;
ALTER TABLE STOCK53 DROP CONSTRAINT STOCK53CKC;
ALTER TABLE STOCK53 ADD CONSTRAINT STOCK53CKC CHECK (S_W_ID
BETWEEN 66457 AND 67734);
SET INTEGRITY FOR STOCK53 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK54.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK54 OFF;
ALTER TABLE STOCK54 DROP CONSTRAINT STOCK54CKC;
ALTER TABLE STOCK54 ADD CONSTRAINT STOCK54CKC CHECK (S_W_ID
BETWEEN 67735 AND 69012);
SET INTEGRITY FOR STOCK54 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK55.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK55 OFF;
ALTER TABLE STOCK55 DROP CONSTRAINT STOCK55CKC;
ALTER TABLE STOCK55 ADD CONSTRAINT STOCK55CKC CHECK (S_W_ID
BETWEEN 69013 AND 70290);
SET INTEGRITY FOR STOCK55 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK56.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK56 OFF;
ALTER TABLE STOCK56 DROP CONSTRAINT STOCK56CKC;
ALTER TABLE STOCK56 ADD CONSTRAINT STOCK56CKC CHECK (S_W_ID
BETWEEN 70291 AND 71568);
SET INTEGRITY FOR STOCK56 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK57.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK57 OFF;
ALTER TABLE STOCK57 DROP CONSTRAINT STOCK57CKC;
ALTER TABLE STOCK57 ADD CONSTRAINT STOCK57CKC CHECK (S_W_ID
BETWEEN 71569 AND 72846);
SET INTEGRITY FOR STOCK57 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK58.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK58 OFF;
ALTER TABLE STOCK58 DROP CONSTRAINT STOCK58CKC;
ALTER TABLE STOCK58 ADD CONSTRAINT STOCK58CKC CHECK (S_W_ID
BETWEEN 72847 AND 74124);
SET INTEGRITY FOR STOCK58 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK59.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK59 OFF;
ALTER TABLE STOCK59 DROP CONSTRAINT STOCK59CKC;
ALTER TABLE STOCK59 ADD CONSTRAINT STOCK59CKC CHECK (S_W_ID
BETWEEN 74125 AND 75402);
SET INTEGRITY FOR STOCK59 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK5.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK5 OFF;
ALTER TABLE STOCK5 DROP CONSTRAINT STOCK5CKC;
ALTER TABLE STOCK5 ADD CONSTRAINT STOCK5CKC CHECK (S_W_ID BETWEEN
5113 AND 6390);
SET INTEGRITY FOR STOCK5 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK60.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK60 OFF;
ALTER TABLE STOCK60 DROP CONSTRAINT STOCK60CKC;
ALTER TABLE STOCK60 ADD CONSTRAINT STOCK60CKC CHECK (S_W_ID
BETWEEN 75403 AND 76680);
SET INTEGRITY FOR STOCK60 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK61.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK61 OFF;
ALTER TABLE STOCK61 DROP CONSTRAINT STOCK61CKC;
ALTER TABLE STOCK61 ADD CONSTRAINT STOCK61CKC CHECK (S_W_ID
BETWEEN 76681 AND 77958);
SET INTEGRITY FOR STOCK61 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK62.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK62 OFF;
ALTER TABLE STOCK62 DROP CONSTRAINT STOCK62CKC;
ALTER TABLE STOCK62 ADD CONSTRAINT STOCK62CKC CHECK (S_W_ID
BETWEEN 77959 AND 79236);
SET INTEGRITY FOR STOCK62 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK63.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK63 OFF;
ALTER TABLE STOCK63 DROP CONSTRAINT STOCK63CKC;
ALTER TABLE STOCK63 ADD CONSTRAINT STOCK63CKC CHECK (S_W_ID
BETWEEN 79237 AND 80514);
SET INTEGRITY FOR STOCK63 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK64.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK64 OFF;
ALTER TABLE STOCK64 DROP CONSTRAINT STOCK64CKC;
ALTER TABLE STOCK64 ADD CONSTRAINT STOCK64CKC CHECK (S_W_ID
BETWEEN 80515 AND 81792);
SET INTEGRITY FOR STOCK64 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK65.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID
BETWEEN 81793 AND 83070);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK66.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK66 OFF;
ALTER TABLE STOCK66 DROP CONSTRAINT STOCK66CKC;
ALTER TABLE STOCK66 ADD CONSTRAINT STOCK66CKC CHECK (S_W_ID
BETWEEN 83071 AND 84348);
SET INTEGRITY FOR STOCK66 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK67.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK67 OFF;
ALTER TABLE STOCK67 DROP CONSTRAINT STOCK67CKC;
ALTER TABLE STOCK67 ADD CONSTRAINT STOCK67CKC CHECK (S_W_ID
BETWEEN 84349 AND 85626);
SET INTEGRITY FOR STOCK67 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK68.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK68 OFF;
ALTER TABLE STOCK68 DROP CONSTRAINT STOCK68CKC;
ALTER TABLE STOCK68 ADD CONSTRAINT STOCK68CKC CHECK (S_W_ID
BETWEEN 85627 AND 86904);
SET INTEGRITY FOR STOCK68 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK69.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK69 OFF;
ALTER TABLE STOCK69 DROP CONSTRAINT STOCK69CKC;
ALTER TABLE STOCK69 ADD CONSTRAINT STOCK69CKC CHECK (S_W_ID
BETWEEN 86905 AND 88182);
SET INTEGRITY FOR STOCK69 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK6.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK6 OFF;
ALTER TABLE STOCK6 DROP CONSTRAINT STOCK6CKC;
ALTER TABLE STOCK6 ADD CONSTRAINT STOCK6CKC CHECK (S_W_ID BETWEEN
6391 AND 7668);
SET INTEGRITY FOR STOCK6 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK70.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK70 OFF;
ALTER TABLE STOCK70 DROP CONSTRAINT STOCK70CKC;
ALTER TABLE STOCK70 ADD CONSTRAINT STOCK70CKC CHECK (S_W_ID
BETWEEN 88183 AND 89460);
SET INTEGRITY FOR STOCK70 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK71.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK71 OFF;
ALTER TABLE STOCK71 DROP CONSTRAINT STOCK71CKC;
ALTER TABLE STOCK71 ADD CONSTRAINT STOCK71CKC CHECK (S_W_ID
BETWEEN 89461 AND 90738);
SET INTEGRITY FOR STOCK71 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK72.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK72 OFF;
ALTER TABLE STOCK72 DROP CONSTRAINT STOCK72CKC;
ALTER TABLE STOCK72 ADD CONSTRAINT STOCK72CKC CHECK (S_W_ID
BETWEEN 90739 AND 92016);
SET INTEGRITY FOR STOCK72 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK73.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK73 OFF;
ALTER TABLE STOCK73 DROP CONSTRAINT STOCK73CKC;
ALTER TABLE STOCK73 ADD CONSTRAINT STOCK73CKC CHECK (S_W_ID
BETWEEN 92017 AND 93294);
SET INTEGRITY FOR STOCK73 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK74.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK74 OFF;
ALTER TABLE STOCK74 DROP CONSTRAINT STOCK74CKC;
ALTER TABLE STOCK74 ADD CONSTRAINT STOCK74CKC CHECK (S_W_ID
BETWEEN 93295 AND 94572);
SET INTEGRITY FOR STOCK74 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK75.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK75 OFF;
ALTER TABLE STOCK75 DROP CONSTRAINT STOCK75CKC;
ALTER TABLE STOCK75 ADD CONSTRAINT STOCK75CKC CHECK (S_W_ID
BETWEEN 94573 AND 95850);
SET INTEGRITY FOR STOCK75 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK76.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR STOCK76 OFF;
ALTER TABLE STOCK76 DROP CONSTRAINT STOCK76CKC;
ALTER TABLE STOCK76 ADD CONSTRAINT STOCK76CKC CHECK (S_W_ID
BETWEEN 95851 AND 97128);
SET INTEGRITY FOR STOCK76 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK77.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK77 OFF;
ALTER TABLE STOCK77 DROP CONSTRAINT STOCK77CKC;
ALTER TABLE STOCK77 ADD CONSTRAINT STOCK77CKC CHECK (S_W_ID
BETWEEN 97129 AND 98406);
SET INTEGRITY FOR STOCK77 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK78.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK78 OFF;
ALTER TABLE STOCK78 DROP CONSTRAINT STOCK78CKC;
ALTER TABLE STOCK78 ADD CONSTRAINT STOCK78CKC CHECK (S_W_ID
BETWEEN 98407 AND 99684);
SET INTEGRITY FOR STOCK78 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK79.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK79 OFF;
ALTER TABLE STOCK79 DROP CONSTRAINT STOCK79CKC;
ALTER TABLE STOCK79 ADD CONSTRAINT STOCK79CKC CHECK (S_W_ID
BETWEEN 99685 AND 100962);
SET INTEGRITY FOR STOCK79 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK7.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK7 OFF;
ALTER TABLE STOCK7 DROP CONSTRAINT STOCK7CKC;
ALTER TABLE STOCK7 ADD CONSTRAINT STOCK7CKC CHECK (S_W_ID BETWEEN
7669 AND 8946);
SET INTEGRITY FOR STOCK7 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK80.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK80 OFF;
ALTER TABLE STOCK80 DROP CONSTRAINT STOCK80CKC;
ALTER TABLE STOCK80 ADD CONSTRAINT STOCK80CKC CHECK (S_W_ID
BETWEEN 100963 AND 102240);
SET INTEGRITY FOR STOCK80 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK81.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK81 OFF;
ALTER TABLE STOCK81 DROP CONSTRAINT STOCK81CKC;
ALTER TABLE STOCK81 ADD CONSTRAINT STOCK81CKC CHECK (S_W_ID
BETWEEN 102241 AND 103518);
SET INTEGRITY FOR STOCK81 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK82.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK82 OFF;
ALTER TABLE STOCK82 DROP CONSTRAINT STOCK82CKC;
ALTER TABLE STOCK82 ADD CONSTRAINT STOCK82CKC CHECK (S_W_ID
BETWEEN 103519 AND 104796);
SET INTEGRITY FOR STOCK82 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK83.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK83 OFF;
ALTER TABLE STOCK83 DROP CONSTRAINT STOCK83CKC;
ALTER TABLE STOCK83 ADD CONSTRAINT STOCK83CKC CHECK (S_W_ID
BETWEEN 104797 AND 106074);
SET INTEGRITY FOR STOCK83 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK84.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK84 OFF;
ALTER TABLE STOCK84 DROP CONSTRAINT STOCK84CKC;
ALTER TABLE STOCK84 ADD CONSTRAINT STOCK84CKC CHECK (S_W_ID
BETWEEN 106075 AND 107352);
SET INTEGRITY FOR STOCK84 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK85.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK85 OFF;
ALTER TABLE STOCK85 DROP CONSTRAINT STOCK85CKC;
ALTER TABLE STOCK85 ADD CONSTRAINT STOCK85CKC CHECK (S_W_ID
BETWEEN 107353 AND 108630);
SET INTEGRITY FOR STOCK85 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK86.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK86 OFF;
ALTER TABLE STOCK86 DROP CONSTRAINT STOCK86CKC;
ALTER TABLE STOCK86 ADD CONSTRAINT STOCK86CKC CHECK (S_W_ID
BETWEEN 108631 AND 109908);
SET INTEGRITY FOR STOCK86 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK87.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK87 OFF;
ALTER TABLE STOCK87 DROP CONSTRAINT STOCK87CKC;
ALTER TABLE STOCK87 ADD CONSTRAINT STOCK87CKC CHECK (S_W_ID
BETWEEN 109909 AND 111186);
SET INTEGRITY FOR STOCK87 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK88.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK88 OFF;
ALTER TABLE STOCK88 DROP CONSTRAINT STOCK88CKC;
ALTER TABLE STOCK88 ADD CONSTRAINT STOCK88CKC CHECK (S_W_ID
BETWEEN 111187 AND 112464);
SET INTEGRITY FOR STOCK88 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK89.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK89 OFF;
ALTER TABLE STOCK89 DROP CONSTRAINT STOCK89CKC;
ALTER TABLE STOCK89 ADD CONSTRAINT STOCK89CKC CHECK (S_W_ID
BETWEEN 112465 AND 113742);
SET INTEGRITY FOR STOCK89 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK8.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK8 OFF;
ALTER TABLE STOCK8 DROP CONSTRAINT STOCK8CKC;
ALTER TABLE STOCK8 ADD CONSTRAINT STOCK8CKC CHECK (S_W_ID BETWEEN
8947 AND 10224);
SET INTEGRITY FOR STOCK8 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK90.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK90 OFF;
ALTER TABLE STOCK90 DROP CONSTRAINT STOCK90CKC;
ALTER TABLE STOCK90 ADD CONSTRAINT STOCK90CKC CHECK (S_W_ID
BETWEEN 113743 AND 115020);
SET INTEGRITY FOR STOCK90 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK91.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK91 OFF;
ALTER TABLE STOCK91 DROP CONSTRAINT STOCK91CKC;
ALTER TABLE STOCK91 ADD CONSTRAINT STOCK91CKC CHECK (S_W_ID
BETWEEN 115021 AND 116298);
SET INTEGRITY FOR STOCK91 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK92.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK92 OFF;
ALTER TABLE STOCK92 DROP CONSTRAINT STOCK92CKC;
ALTER TABLE STOCK92 ADD CONSTRAINT STOCK92CKC CHECK (S_W_ID
BETWEEN 116299 AND 117576);
SET INTEGRITY FOR STOCK92 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK93.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK93 OFF;
ALTER TABLE STOCK93 DROP CONSTRAINT STOCK93CKC;
ALTER TABLE STOCK93 ADD CONSTRAINT STOCK93CKC CHECK (S_W_ID
BETWEEN 117577 AND 118854);
```

```
SET INTEGRITY FOR STOCK93 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK94.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK94 OFF;
ALTER TABLE STOCK94 DROP CONSTRAINT STOCK94CKC;
ALTER TABLE STOCK94 ADD CONSTRAINT STOCK94CKC CHECK (S_W_ID
BETWEEN 118855 AND 120132);
SET INTEGRITY FOR STOCK94 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK95.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK95 OFF;
ALTER TABLE STOCK95 DROP CONSTRAINT STOCK95CKC;
ALTER TABLE STOCK95 ADD CONSTRAINT STOCK95CKC CHECK (S_W_ID
BETWEEN 120133 AND 121410);
SET INTEGRITY FOR STOCK95 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK96.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK96 OFF;
ALTER TABLE STOCK96 DROP CONSTRAINT STOCK96CKC;
ALTER TABLE STOCK96 ADD CONSTRAINT STOCK96CKC CHECK (S_W_ID
BETWEEN 121411 AND 122688);
SET INTEGRITY FOR STOCK96 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK97.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK97 OFF;
ALTER TABLE STOCK97 DROP CONSTRAINT STOCK97CKC;
ALTER TABLE STOCK97 ADD CONSTRAINT STOCK97CKC CHECK (S_W_ID
BETWEEN 122689 AND 123966);
SET INTEGRITY FOR STOCK97 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK98.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK98 OFF;
ALTER TABLE STOCK98 DROP CONSTRAINT STOCK98CKC;
ALTER TABLE STOCK98 ADD CONSTRAINT STOCK98CKC CHECK (S_W_ID
BETWEEN 123967 AND 125244);
SET INTEGRITY FOR STOCK98 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK99.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK99 OFF;
ALTER TABLE STOCK99 DROP CONSTRAINT STOCK99CKC;
ALTER TABLE STOCK99 ADD CONSTRAINT STOCK99CKC CHECK (S_W_ID
BETWEEN 125245 AND 126522);
SET INTEGRITY FOR STOCK99 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_STOCK9.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK9 OFF;
ALTER TABLE STOCK9 DROP CONSTRAINT STOCK9CKC;
ALTER TABLE STOCK9 ADD CONSTRAINT STOCK9CKC CHECK (S_W_ID BETWEEN
10225 AND 11502);
SET INTEGRITY FOR STOCK9 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE10.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE10 OFF;
ALTER TABLE WAREHOUSE10 DROP CONSTRAINT WAREHOUSE10CKC;
ALTER TABLE WAREHOUSE10 ADD CONSTRAINT WAREHOUSE10CKC CHECK
(W_ID BETWEEN 46009 AND 51120);
SET INTEGRITY FOR WAREHOUSE10 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE11.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE11 OFF;
ALTER TABLE WAREHOUSE11 DROP CONSTRAINT WAREHOUSE11CKC;
ALTER TABLE WAREHOUSE11 ADD CONSTRAINT WAREHOUSE11CKC CHECK
(W_ID BETWEEN 51121 AND 56232);
SET INTEGRITY FOR WAREHOUSE11 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE12.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE12 OFF;
ALTER TABLE WAREHOUSE12 DROP CONSTRAINT WAREHOUSE12CKC;
ALTER TABLE WAREHOUSE12 ADD CONSTRAINT WAREHOUSE12CKC CHECK
(W_ID BETWEEN 56233 AND 61344);
SET INTEGRITY FOR WAREHOUSE12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE13.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE13 OFF;
ALTER TABLE WAREHOUSE13 DROP CONSTRAINT WAREHOUSE13CKC;
ALTER TABLE WAREHOUSE13 ADD CONSTRAINT WAREHOUSE13CKC CHECK
(W_ID BETWEEN 61345 AND 66456);
SET INTEGRITY FOR WAREHOUSE13 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE14.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE14 OFF;
ALTER TABLE WAREHOUSE14 DROP CONSTRAINT WAREHOUSE14CKC;
ALTER TABLE WAREHOUSE14 ADD CONSTRAINT WAREHOUSE14CKC CHECK
(W_ID BETWEEN 66457 AND 71568);
SET INTEGRITY FOR WAREHOUSE14 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE15.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE15 OFF;
ALTER TABLE WAREHOUSE15 DROP CONSTRAINT WAREHOUSE15CKC;
ALTER TABLE WAREHOUSE15 ADD CONSTRAINT WAREHOUSE15CKC CHECK
(W_ID BETWEEN 71569 AND 76680);
SET INTEGRITY FOR WAREHOUSE15 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE16.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE16 OFF;
ALTER TABLE WAREHOUSE16 DROP CONSTRAINT WAREHOUSE16CKC;
ALTER TABLE WAREHOUSE16 ADD CONSTRAINT WAREHOUSE16CKC CHECK
(W_ID BETWEEN 76681 AND 81792);
SET INTEGRITY FOR WAREHOUSE16 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE17.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE17 OFF;
ALTER TABLE WAREHOUSE17 DROP CONSTRAINT WAREHOUSE17CKC;
ALTER TABLE WAREHOUSE17 ADD CONSTRAINT WAREHOUSE17CKC CHECK
(W_ID BETWEEN 81793 AND 86904);
SET INTEGRITY FOR WAREHOUSE17 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE18.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE18 OFF;
ALTER TABLE WAREHOUSE18 DROP CONSTRAINT WAREHOUSE18CKC;
ALTER TABLE WAREHOUSE18 ADD CONSTRAINT WAREHOUSE18CKC CHECK
(W_ID BETWEEN 86905 AND 92016);
SET INTEGRITY FOR WAREHOUSE18 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE19.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE19 OFF;
ALTER TABLE WAREHOUSE19 DROP CONSTRAINT WAREHOUSE19CKC;
ALTER TABLE WAREHOUSE19 ADD CONSTRAINT WAREHOUSE19CKC CHECK
(W_ID BETWEEN 92017 AND 97128);
SET INTEGRITY FOR WAREHOUSE19 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE1.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE1 OFF;
ALTER TABLE WAREHOUSE1 DROP CONSTRAINT WAREHOUSE1CKC;
ALTER TABLE WAREHOUSE1 ADD CONSTRAINT WAREHOUSE1CKC CHECK (W_ID
BETWEEN 1 AND 5112);
SET INTEGRITY FOR WAREHOUSE1 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE20.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE20 OFF;
ALTER TABLE WAREHOUSE20 DROP CONSTRAINT WAREHOUSE20CKC;
ALTER TABLE WAREHOUSE20 ADD CONSTRAINT WAREHOUSE20CKC CHECK
(W_ID BETWEEN 97129 AND 102240);
SET INTEGRITY FOR WAREHOUSE20 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE21.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE21 OFF;
ALTER TABLE WAREHOUSE21 DROP CONSTRAINT WAREHOUSE21CKC;
ALTER TABLE WAREHOUSE21 ADD CONSTRAINT WAREHOUSE21CKC CHECK
(W_ID BETWEEN 102241 AND 107352);
SET INTEGRITY FOR WAREHOUSE21 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE22.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE22 OFF;
ALTER TABLE WAREHOUSE22 DROP CONSTRAINT WAREHOUSE22CKC;
ALTER TABLE WAREHOUSE22 ADD CONSTRAINT WAREHOUSE22CKC CHECK
(W_ID BETWEEN 107353 AND 112464);
SET INTEGRITY FOR WAREHOUSE22 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE23.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE23 OFF;
ALTER TABLE WAREHOUSE23 DROP CONSTRAINT WAREHOUSE23CKC;
ALTER TABLE WAREHOUSE23 ADD CONSTRAINT WAREHOUSE23CKC CHECK
(W_ID BETWEEN 112465 AND 117576);
SET INTEGRITY FOR WAREHOUSE23 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE24.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE24 OFF;
ALTER TABLE WAREHOUSE24 DROP CONSTRAINT WAREHOUSE24CKC;
ALTER TABLE WAREHOUSE24 ADD CONSTRAINT WAREHOUSE24CKC CHECK
(W_ID BETWEEN 117577 AND 122688);
SET INTEGRITY FOR WAREHOUSE24 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE25.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE25 OFF;
ALTER TABLE WAREHOUSE25 DROP CONSTRAINT WAREHOUSE25CKC;
ALTER TABLE WAREHOUSE25 ADD CONSTRAINT WAREHOUSE25CKC CHECK
(W_ID BETWEEN 122689 AND 127800);
SET INTEGRITY FOR WAREHOUSE25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE26.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE26 OFF;
ALTER TABLE WAREHOUSE26 DROP CONSTRAINT WAREHOUSE26CKC;
ALTER TABLE WAREHOUSE26 ADD CONSTRAINT WAREHOUSE26CKC CHECK
(W_ID BETWEEN 127801 AND 132912);
SET INTEGRITY FOR WAREHOUSE26 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE27.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE27 OFF;
ALTER TABLE WAREHOUSE27 DROP CONSTRAINT WAREHOUSE27CKC;
ALTER TABLE WAREHOUSE27 ADD CONSTRAINT WAREHOUSE27CKC CHECK
(W_ID BETWEEN 132913 AND 138024);
SET INTEGRITY FOR WAREHOUSE27 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE28.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE28 OFF;
ALTER TABLE WAREHOUSE28 DROP CONSTRAINT WAREHOUSE28CKC;
ALTER TABLE WAREHOUSE28 ADD CONSTRAINT WAREHOUSE28CKC CHECK
(W_ID BETWEEN 138025 AND 143136);
SET INTEGRITY FOR WAREHOUSE28 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE29.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE29 OFF;
ALTER TABLE WAREHOUSE29 DROP CONSTRAINT WAREHOUSE29CKC;
ALTER TABLE WAREHOUSE29 ADD CONSTRAINT WAREHOUSE29CKC CHECK
(W_ID BETWEEN 143137 AND 148248);
SET INTEGRITY FOR WAREHOUSE29 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE2.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE2 OFF;
ALTER TABLE WAREHOUSE2 DROP CONSTRAINT WAREHOUSE2CKC;
ALTER TABLE WAREHOUSE2 ADD CONSTRAINT WAREHOUSE2CKC CHECK (W_ID
BETWEEN 5113 AND 10224);
SET INTEGRITY FOR WAREHOUSE2 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE30.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE30 OFF;
ALTER TABLE WAREHOUSE30 DROP CONSTRAINT WAREHOUSE30CKC;
ALTER TABLE WAREHOUSE30 ADD CONSTRAINT WAREHOUSE30CKC CHECK
(W_ID BETWEEN 148249 AND 153360);
SET INTEGRITY FOR WAREHOUSE30 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE31.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE31 OFF;
ALTER TABLE WAREHOUSE31 DROP CONSTRAINT WAREHOUSE31CKC;
ALTER TABLE WAREHOUSE31 ADD CONSTRAINT WAREHOUSE31CKC CHECK
(W_ID BETWEEN 153361 AND 158472);
SET INTEGRITY FOR WAREHOUSE31 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE32.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE32 OFF;
ALTER TABLE WAREHOUSE32 DROP CONSTRAINT WAREHOUSE32CKC;
ALTER TABLE WAREHOUSE32 ADD CONSTRAINT WAREHOUSE32CKC CHECK
(W_ID BETWEEN 158473 AND 163584);
SET INTEGRITY FOR WAREHOUSE32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE33.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE33 OFF;
ALTER TABLE WAREHOUSE33 DROP CONSTRAINT WAREHOUSE33CKC;
ALTER TABLE WAREHOUSE33 ADD CONSTRAINT WAREHOUSE33CKC CHECK
(W_ID BETWEEN 163585 AND 168696);
SET INTEGRITY FOR WAREHOUSE33 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE34.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE34 OFF;
ALTER TABLE WAREHOUSE34 DROP CONSTRAINT WAREHOUSE34CKC;
ALTER TABLE WAREHOUSE34 ADD CONSTRAINT WAREHOUSE34CKC CHECK
(W_ID BETWEEN 168697 AND 173808);
SET INTEGRITY FOR WAREHOUSE34 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE35.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE35 OFF;
ALTER TABLE WAREHOUSE35 DROP CONSTRAINT WAREHOUSE35CKC;
ALTER TABLE WAREHOUSE35 ADD CONSTRAINT WAREHOUSE35CKC CHECK
(W_ID BETWEEN 173809 AND 178920);
SET INTEGRITY FOR WAREHOUSE35 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE36.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE36 OFF;
ALTER TABLE WAREHOUSE36 DROP CONSTRAINT WAREHOUSE36CKC;
ALTER TABLE WAREHOUSE36 ADD CONSTRAINT WAREHOUSE36CKC CHECK
(W_ID >= 178921);
SET INTEGRITY FOR WAREHOUSE36 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE3.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE3 OFF;
ALTER TABLE WAREHOUSE3 DROP CONSTRAINT WAREHOUSE3CKC;
ALTER TABLE WAREHOUSE3 ADD CONSTRAINT WAREHOUSE3CKC CHECK (W_ID
BETWEEN 10225 AND 15336);
SET INTEGRITY FOR WAREHOUSE3 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE4.ddl
```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE4 OFF;
ALTER TABLE WAREHOUSE4 DROP CONSTRAINT WAREHOUSE4CKC;
ALTER TABLE WAREHOUSE4 ADD CONSTRAINT WAREHOUSE4CKC CHECK (W_ID
BETWEEN 15337 AND 20448);
SET INTEGRITY FOR WAREHOUSE4 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE5.ddl
```
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR WAREHOUSE5 OFF;
ALTER TABLE WAREHOUSE5 DROP CONSTRAINT WAREHOUSE5CKC;
ALTER TABLE WAREHOUSE5 ADD CONSTRAINT WAREHOUSE5CKC CHECK (W_ID
BETWEEN 20449 AND 25560);
SET INTEGRITY FOR WAREHOUSE5 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE6.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE6 OFF;
ALTER TABLE WAREHOUSE6 DROP CONSTRAINT WAREHOUSE6CKC;
ALTER TABLE WAREHOUSE6 ADD CONSTRAINT WAREHOUSE6CKC CHECK (W_ID
BETWEEN 25561 AND 30672);
SET INTEGRITY FOR WAREHOUSE6 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE7.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE7 OFF;
ALTER TABLE WAREHOUSE7 DROP CONSTRAINT WAREHOUSE7CKC;
ALTER TABLE WAREHOUSE7 ADD CONSTRAINT WAREHOUSE7CKC CHECK (W_ID
BETWEEN 30673 AND 35784);
SET INTEGRITY FOR WAREHOUSE7 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE8.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE8 OFF;
ALTER TABLE WAREHOUSE8 DROP CONSTRAINT WAREHOUSE8CKC;
ALTER TABLE WAREHOUSE8 ADD CONSTRAINT WAREHOUSE8CKC CHECK (W_ID
BETWEEN 35785 AND 40896);
SET INTEGRITY FOR WAREHOUSE8 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_WAREHOUSE9.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE9 OFF;
ALTER TABLE WAREHOUSE9 DROP CONSTRAINT WAREHOUSE9CKC;
ALTER TABLE WAREHOUSE9 ADD CONSTRAINT WAREHOUSE9CKC CHECK (W_ID
BETWEEN 40897 AND 46008);
SET INTEGRITY FOR WAREHOUSE9 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRIDX_CUST_IDXB100.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB100;
CREATE INDEX CUST_IDXB100
        ON CUSTOMER100(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB101.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB101;
CREATE INDEX CUST_IDXB101
        ON CUSTOMER101(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB102.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB102;
CREATE INDEX CUST_IDXB102
        ON CUSTOMER102(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB103.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB103;
CREATE INDEX CUST_IDXB103
        ON CUSTOMER103(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB104.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB104;
CREATE INDEX CUST_IDXB104
        ON CUSTOMER104(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB105.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB105;
CREATE INDEX CUST_IDXB105
        ON CUSTOMER105(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB106.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB106;
CREATE INDEX CUST_IDXB106
        ON CUSTOMER106(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB107.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB107;
CREATE INDEX CUST_IDXB107
        ON CUSTOMER107(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB108.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB108;
CREATE INDEX CUST_IDXB108
        ON CUSTOMER108(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB109.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB109;
CREATE INDEX CUST_IDXB109
        ON CUSTOMER109(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB10.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB10;
CREATE INDEX CUST_IDXB10
        ON CUSTOMER10(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB110.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB110;
CREATE INDEX CUST_IDXB110
        ON CUSTOMER110(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB111.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB111;
CREATE INDEX CUST_IDXB111
        ON CUSTOMER111(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB112.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB112;
CREATE INDEX CUST_IDXB112
        ON CUSTOMER112(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB113.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB113;
CREATE INDEX CUST_IDXB113
        ON CUSTOMER113(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB114.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB114;
CREATE INDEX CUST_IDXB114
        ON CUSTOMER114(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB115.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB115;
CREATE INDEX CUST_IDXB115
                ON CUSTOMER115(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB116.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB116;
CREATE INDEX CUST_IDXB116
                ON CUSTOMER116(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB117.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB117;
CREATE INDEX CUST_IDXB117
                ON CUSTOMER117(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB118.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB118;
CREATE INDEX CUST_IDXB118
                ON CUSTOMER118(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB119.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB119;
CREATE INDEX CUST_IDXB119
                ON CUSTOMER119(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB11.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB11;
CREATE INDEX CUST_IDXB11
                ON CUSTOMER11(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB120.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB120;
CREATE INDEX CUST_IDXB120
                ON CUSTOMER120(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB121.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB121;
CREATE INDEX CUST_IDXB121
                ON CUSTOMER121(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB122.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB122;
CREATE INDEX CUST_IDXB122
                ON CUSTOMER122(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB123.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB123;
CREATE INDEX CUST_IDXB123
                ON CUSTOMER123(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB124.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB124;
CREATE INDEX CUST_IDXB124
                ON CUSTOMER124(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB125.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB125;
CREATE INDEX CUST_IDXB125
                ON CUSTOMER125(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB126.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB126;
CREATE INDEX CUST_IDXB126
                ON CUSTOMER126(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB127.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB127;
CREATE INDEX CUST_IDXB127
                ON CUSTOMER127(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB128.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB128;
CREATE INDEX CUST_IDXB128
                ON CUSTOMER128(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB129.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB129;
CREATE INDEX CUST_IDXB129
                ON CUSTOMER129(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB12.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB12;
CREATE INDEX CUST_IDXB12
                ON CUSTOMER12(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB130.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB130;
CREATE INDEX CUST_IDXB130
                ON CUSTOMER130(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB131.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB131;
CREATE INDEX CUST_IDXB131
                ON CUSTOMER131(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB132.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB132;
CREATE INDEX CUST_IDXB132
                ON CUSTOMER132(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB133.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB133;
CREATE INDEX CUST_IDXB133
                ON CUSTOMER133(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB134.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB134;
```

```
CREATE INDEX CUST_IDXB134
                ON CUSTOMER134(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB135.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB135;
CREATE INDEX CUST_IDXB135
                ON CUSTOMER135(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB136.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB136;
CREATE INDEX CUST_IDXB136
                ON CUSTOMER136(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB137.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB137;
CREATE INDEX CUST_IDXB137
                ON CUSTOMER137(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB138.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB138;
CREATE INDEX CUST_IDXB138
                ON CUSTOMER138(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB139.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB139;
CREATE INDEX CUST_IDXB139
                ON CUSTOMER139(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB13.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB13;
CREATE INDEX CUST_IDXB13
                ON CUSTOMER13(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB140.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB140;
CREATE INDEX CUST_IDXB140
                ON CUSTOMER140(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB141.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB141;
CREATE INDEX CUST_IDXB141
                ON CUSTOMER141(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB142.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB142;
CREATE INDEX CUST_IDXB142
                ON CUSTOMER142(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB143.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB143;
CREATE INDEX CUST_IDXB143
                ON CUSTOMER143(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB144.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB144;
CREATE INDEX CUST_IDXB144
                ON CUSTOMER144(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB14.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB14;
CREATE INDEX CUST_IDXB14
                ON CUSTOMER14(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB15.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB15;
CREATE INDEX CUST_IDXB15
                ON CUSTOMER15(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB16.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB16;
CREATE INDEX CUST_IDXB16
                ON CUSTOMER16(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB17.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB17;
CREATE INDEX CUST_IDXB17
                ON CUSTOMER17(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB18.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB18;
CREATE INDEX CUST_IDXB18
                ON CUSTOMER18(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB19.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB19;
CREATE INDEX CUST_IDXB19
                ON CUSTOMER19(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB1.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB1;
CREATE INDEX CUST_IDXB1
                ON CUSTOMER1(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB20.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB20;
CREATE INDEX CUST_IDXB20
                ON CUSTOMER20(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB21.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB21;
CREATE INDEX CUST_IDXB21
                ON CUSTOMER21(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB22.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB22;
CREATE INDEX CUST_IDXB22
                ON CUSTOMER22(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
```

```
connect reset;
```

CRIDX_CUST_IDXB23.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB23;
CREATE INDEX CUST_IDXB23
                ON CUSTOMER23(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB24.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB24;
CREATE INDEX CUST_IDXB24
                ON CUSTOMER24(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB25.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB25;
CREATE INDEX CUST_IDXB25
                ON CUSTOMER25(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB26.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB26;
CREATE INDEX CUST_IDXB26
                ON CUSTOMER26(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB27.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB27;
CREATE INDEX CUST_IDXB27
                ON CUSTOMER27(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB28.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB28;
CREATE INDEX CUST_IDXB28
                ON CUSTOMER28(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB29.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB29;
CREATE INDEX CUST_IDXB29
                ON CUSTOMER29(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB2.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB2;
CREATE INDEX CUST_IDXB2
                ON CUSTOMER2(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB30.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB30;
CREATE INDEX CUST_IDXB30
                ON CUSTOMER30(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB31.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB31;
CREATE INDEX CUST_IDXB31
                ON CUSTOMER31(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB32.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB32;
CREATE INDEX CUST_IDXB32
                ON CUSTOMER32(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB33.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB33;
CREATE INDEX CUST_IDXB33
                ON CUSTOMER33(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB34.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB34;
CREATE INDEX CUST_IDXB34
                ON CUSTOMER34(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB35.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB35;
CREATE INDEX CUST_IDXB35
                ON CUSTOMER35(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB36.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB36;
CREATE INDEX CUST_IDXB36
                ON CUSTOMER36(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB37.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB37;
CREATE INDEX CUST_IDXB37
                ON CUSTOMER37(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB38.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB38;
CREATE INDEX CUST_IDXB38
                ON CUSTOMER38(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB39.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB39;
CREATE INDEX CUST_IDXB39
                ON CUSTOMER39(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB3.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB3;
CREATE INDEX CUST_IDXB3
                ON CUSTOMER3(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB40.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB40;
CREATE INDEX CUST_IDXB40
                ON CUSTOMER40(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB41.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB41;
CREATE INDEX CUST_IDXB41
                ON CUSTOMER41(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB42.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB42;
CREATE INDEX CUST_IDXB42
                ON CUSTOMER42(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB43.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB43;
CREATE INDEX CUST_IDXB43
                ON CUSTOMER43(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB44.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB44;
CREATE INDEX CUST_IDXB44
                ON CUSTOMER44(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB45.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB45;
CREATE INDEX CUST_IDXB45
                ON CUSTOMER45(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB46.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB46;
CREATE INDEX CUST_IDXB46
                ON CUSTOMER46(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB47.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB47;
CREATE INDEX CUST_IDXB47
                ON CUSTOMER47(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB48.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB48;
CREATE INDEX CUST_IDXB48
                ON CUSTOMER48(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB49.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB49;
CREATE INDEX CUST_IDXB49
                ON CUSTOMER49(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB4.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB4;
CREATE INDEX CUST_IDXB4
                ON CUSTOMER4(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB50.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB50;
CREATE INDEX CUST_IDXB50
                ON CUSTOMER50(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB51.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB51;
CREATE INDEX CUST_IDXB51
                ON CUSTOMER51(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB52.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB52;
CREATE INDEX CUST_IDXB52
                ON CUSTOMER52(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB53.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB53;
CREATE INDEX CUST_IDXB53
                ON CUSTOMER53(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB54.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB54;
CREATE INDEX CUST_IDXB54
                ON CUSTOMER54(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB55.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB55;
CREATE INDEX CUST_IDXB55
                ON CUSTOMER55(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB56.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB56;
CREATE INDEX CUST_IDXB56
                ON CUSTOMER56(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB57.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB57;
CREATE INDEX CUST_IDXB57
                ON CUSTOMER57(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB58.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB58;
CREATE INDEX CUST_IDXB58
                ON CUSTOMER58(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB59.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB59;
CREATE INDEX CUST_IDXB59
                ON CUSTOMER59(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB5.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB5;
CREATE INDEX CUST_IDXB5
                ON CUSTOMER5(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB60.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB60;
CREATE INDEX CUST_IDXB60
                ON CUSTOMER60(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB61.ddl
```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB61;
```

```
CREATE INDEX CUST_IDXB61
                ON CUSTOMER61(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB62.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB62;
CREATE INDEX CUST_IDXB62
                ON CUSTOMER62(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB63.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB63;
CREATE INDEX CUST_IDXB63
                ON CUSTOMER63(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB64.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB64;
CREATE INDEX CUST_IDXB64
                ON CUSTOMER64(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB65.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB65;
CREATE INDEX CUST_IDXB65
                ON CUSTOMER65(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB66.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB66;
CREATE INDEX CUST_IDXB66
                ON CUSTOMER66(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB67.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB67;
CREATE INDEX CUST_IDXB67
                ON CUSTOMER67(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB68.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB68;
CREATE INDEX CUST_IDXB68
                ON CUSTOMER68(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB69.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB69;
CREATE INDEX CUST_IDXB69
                ON CUSTOMER69(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB6.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB6;
CREATE INDEX CUST_IDXB6
                ON CUSTOMER6(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB70.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB70;
CREATE INDEX CUST_IDXB70
                ON CUSTOMER70(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB71.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB71;
CREATE INDEX CUST_IDXB71
                ON CUSTOMER71(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB72.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB72;
CREATE INDEX CUST_IDXB72
                ON CUSTOMER72(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB73.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB73;
CREATE INDEX CUST_IDXB73
                ON CUSTOMER73(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB74.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB74;
CREATE INDEX CUST_IDXB74
                ON CUSTOMER74(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB75.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB75;
CREATE INDEX CUST_IDXB75
                ON CUSTOMER75(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB76.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB76;
CREATE INDEX CUST_IDXB76
                ON CUSTOMER76(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB77.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB77;
CREATE INDEX CUST_IDXB77
                ON CUSTOMER77(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB78.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB78;
CREATE INDEX CUST_IDXB78
                ON CUSTOMER78(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB79.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB79;
CREATE INDEX CUST_IDXB79
                ON CUSTOMER79(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB7.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB7;
CREATE INDEX CUST_IDXB7
                ON CUSTOMER7(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB80.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB80;
CREATE INDEX CUST_IDXB80
                ON CUSTOMER80(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
```

```
connect reset;
```

CRIDX_CUST_IDXB81.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB81;
CREATE INDEX CUST_IDXB81
                ON CUSTOMER81(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB82.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB82;
CREATE INDEX CUST_IDXB82
                ON CUSTOMER82(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB83.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB83;
CREATE INDEX CUST_IDXB83
                ON CUSTOMER83(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB84.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB84;
CREATE INDEX CUST_IDXB84
                ON CUSTOMER84(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB85.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB85;
CREATE INDEX CUST_IDXB85
                ON CUSTOMER85(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB86.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB86;
CREATE INDEX CUST_IDXB86
                ON CUSTOMER86(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB87.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB87;
CREATE INDEX CUST_IDXB87
                ON CUSTOMER87(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB88.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB88;
CREATE INDEX CUST_IDXB88
                ON CUSTOMER88(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB89.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB89;
CREATE INDEX CUST_IDXB89
                ON CUSTOMER89(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB8.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB8;
CREATE INDEX CUST_IDXB8
                ON CUSTOMER8(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB90.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB90;
CREATE INDEX CUST_IDXB90
                ON CUSTOMER90(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB91.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB91;
CREATE INDEX CUST_IDXB91
                ON CUSTOMER91(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB92.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB92;
CREATE INDEX CUST_IDXB92
                ON CUSTOMER92(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB93.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB93;
CREATE INDEX CUST_IDXB93
                ON CUSTOMER93(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB94.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB94;
CREATE INDEX CUST_IDXB94
                ON CUSTOMER94(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB95.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB95;
CREATE INDEX CUST_IDXB95
                ON CUSTOMER95(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB96.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB96;
CREATE INDEX CUST_IDXB96
                ON CUSTOMER96(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB97.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB97;
CREATE INDEX CUST_IDXB97
                ON CUSTOMER97(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB98.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB98;
CREATE INDEX CUST_IDXB98
                ON CUSTOMER98(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB99.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB99;
CREATE INDEX CUST_IDXB99
                ON CUSTOMER99(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_CUST_IDXB9.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB9;
CREATE INDEX CUST_IDXB9
                ON CUSTOMER9(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID)
PCTFREE 0;
connect reset;
```

CRIDX_ORDR_IDXB10.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB10;
CREATE INDEX ORDR_IDXB10
                ON ORDERS10(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB11.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB11;
CREATE INDEX ORDR_IDXB11
                ON ORDERS11(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB12.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB12;
CREATE INDEX ORDR_IDXB12
                ON ORDERS12(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB13.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB13;
CREATE INDEX ORDR_IDXB13
                ON ORDERS13(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB14.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB14;
CREATE INDEX ORDR_IDXB14
                ON ORDERS14(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB15.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB15;
CREATE INDEX ORDR_IDXB15
                ON ORDERS15(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB16.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB16;
CREATE INDEX ORDR_IDXB16
                ON ORDERS16(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB17.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB17;
CREATE INDEX ORDR_IDXB17
                ON ORDERS17(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB18.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB18;
CREATE INDEX ORDR_IDXB18
                ON ORDERS18(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB19.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB19;
CREATE INDEX ORDR_IDXB19
                ON ORDERS19(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB1.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB1;
CREATE INDEX ORDR_IDXB1
                ON ORDERS1(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB20.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB20;
CREATE INDEX ORDR_IDXB20
                ON ORDERS20(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB21.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB21;
CREATE INDEX ORDR_IDXB21
                ON ORDERS21(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB22.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB22;
CREATE INDEX ORDR_IDXB22
                ON ORDERS22(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB23.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB23;
CREATE INDEX ORDR_IDXB23
                ON ORDERS23(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB24.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB24;
CREATE INDEX ORDR_IDXB24
                ON ORDERS24(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB25.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB25;
CREATE INDEX ORDR_IDXB25
                ON ORDERS25(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB26.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB26;
CREATE INDEX ORDR_IDXB26
                ON ORDERS26(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB27.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB27;
CREATE INDEX ORDR_IDXB27
                ON ORDERS27(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB28.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB28;
CREATE INDEX ORDR_IDXB28
                ON ORDERS28(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB29.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB29;
CREATE INDEX ORDR_IDXB29
                ON ORDERS29(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB2.ddl
```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB2;
```

```
CREATE INDEX ORDR_IDXB2
                ON ORDERS2(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB30.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB30;
CREATE INDEX ORDR_IDXB30
                ON ORDERS30(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB31.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB31;
CREATE INDEX ORDR_IDXB31
                ON ORDERS31(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB32.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB32;
CREATE INDEX ORDR_IDXB32
                ON ORDERS32(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB33.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB33;
CREATE INDEX ORDR_IDXB33
                ON ORDERS33(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB34.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB34;
CREATE INDEX ORDR_IDXB34
                ON ORDERS34(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB35.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB35;
CREATE INDEX ORDR_IDXB35
                ON ORDERS35(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB36.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB36;
CREATE INDEX ORDR_IDXB36
                ON ORDERS36(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB3.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB3;
CREATE INDEX ORDR_IDXB3
                ON ORDERS3(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB4.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB4;
CREATE INDEX ORDR_IDXB4
                ON ORDERS4(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB5.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB5;
CREATE INDEX ORDR_IDXB5
                ON ORDERS5(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB6.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB6;
CREATE INDEX ORDR_IDXB6
                ON ORDERS6(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB7.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB7;
CREATE INDEX ORDR_IDXB7
                ON ORDERS7(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB8.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB8;
CREATE INDEX ORDR_IDXB8
                ON ORDERS8(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRIDX_ORDR_IDXB9.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB9;
CREATE INDEX ORDR_IDXB9
                ON ORDERS9(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20
LEVEL2 PCTFREE 20;
connect reset;
```

CRTB_CUSTOMER100.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER100;
CREATE TABLE CUSTOMER100
            (
             C_ID          INTEGER       NOT NULL,
             C_STATE       CHAR(2)       NOT NULL,
             C_ZIP         CHAR(9)       NOT NULL,
             C_PHONE       CHAR(16)      NOT NULL,
             C_SINCE       TIMESTAMP     NOT NULL,
             C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
             C_MIDDLE      CHAR(2)       NOT NULL,
             C_CREDIT      CHAR(2)       NOT NULL,
             C_DISCOUNT    REAL          NOT NULL,
             C_DATA        VARCHAR(500)  NOT NULL,
             C_LAST        VARCHAR(16)   NOT NULL,
             C_FIRST       VARCHAR(16)   NOT NULL,
             C_STREET_1    VARCHAR(20)   NOT NULL,
             C_STREET_2    VARCHAR(20)   NOT NULL,
             C_CITY        VARCHAR(20)   NOT NULL,
             C_D_ID        SMALLINT      NOT NULL,
             C_W_ID        INTEGER       NOT NULL,
             C_DELIVERY_CNT INTEGER      NOT NULL,
             C_BALANCE     DECIMAL(12,2) NOT NULL,
             C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
             C_PAYMENT_CNT INTEGER       NOT NULL
            )
            IN ts_C_100
            INDEX IN is_C2_100
            ORGANIZE BY KEY SEQUENCE (
             C_ID STARTING FROM 1 ENDING AT 3000,
             C_W_ID STARTING FROM 126523 ENDING AT 127800,
             C_D_ID STARTING FROM 1 ENDING AT 10
            )
            ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER101.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER101;
CREATE TABLE CUSTOMER101
            (
             C_ID          INTEGER       NOT NULL,
             C_STATE       CHAR(2)       NOT NULL,
             C_ZIP         CHAR(9)       NOT NULL,
             C_PHONE       CHAR(16)      NOT NULL,
             C_SINCE       TIMESTAMP     NOT NULL,
             C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
             C_MIDDLE      CHAR(2)       NOT NULL,
             C_CREDIT      CHAR(2)       NOT NULL,
             C_DISCOUNT    REAL          NOT NULL,
             C_DATA        VARCHAR(500)  NOT NULL,
             C_LAST        VARCHAR(16)   NOT NULL,
             C_FIRST       VARCHAR(16)   NOT NULL,
             C_STREET_1    VARCHAR(20)   NOT NULL,
             C_STREET_2    VARCHAR(20)   NOT NULL,
             C_CITY        VARCHAR(20)   NOT NULL,
             C_D_ID        SMALLINT      NOT NULL,
             C_W_ID        INTEGER       NOT NULL,
```

```
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_101
        INDEX IN is_C2_101
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 127801 ENDING AT 129078,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER102.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER102;
CREATE TABLE CUSTOMER102
        (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER     NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_102
        INDEX IN is_C2_102
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 129079 ENDING AT 130356,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER103.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER103;
CREATE TABLE CUSTOMER103
        (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER     NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_103
        INDEX IN is_C2_103
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 130357 ENDING AT 131634,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER104.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER104;
CREATE TABLE CUSTOMER104
        (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER     NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_104
        INDEX IN is_C2_104
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 131635 ENDING AT 132912,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER105.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER105;
CREATE TABLE CUSTOMER105
        (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER     NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_105
        INDEX IN is_C2_105
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 132913 ENDING AT 134190,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER106.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER106;
CREATE TABLE CUSTOMER106
        (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER     NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
```

```
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_106
       INDEX IN is_C2_106
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 134191 ENDING AT 135468,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER107.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER107;
CREATE TABLE CUSTOMER107
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_107
       INDEX IN is_C2_107
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 135469 ENDING AT 136746,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER108.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER108;
CREATE TABLE CUSTOMER108
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_108
       INDEX IN is_C2_108
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 136747 ENDING AT 138024,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER109.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER109;
CREATE TABLE CUSTOMER109
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_109
       INDEX IN is_C2_109
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 138025 ENDING AT 139302,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER10.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER10;
CREATE TABLE CUSTOMER10
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_010
       INDEX IN is_C2_010
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 11503 ENDING AT 12780,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER110.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER110;
CREATE TABLE CUSTOMER110
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
```

```
)
IN ts_C_110
INDEX IN is_C2_110
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 139303 ENDING AT 140580,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER111.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER111;
CREATE TABLE CUSTOMER111
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_111
INDEX IN is_C2_111
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 140581 ENDING AT 141858,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER112.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER112;
CREATE TABLE CUSTOMER112
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_112
INDEX IN is_C2_112
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 141859 ENDING AT 143136,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER113.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER113;
CREATE TABLE CUSTOMER113
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_113
INDEX IN is_C2_113
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 143137 ENDING AT 144414,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER114.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER114;
CREATE TABLE CUSTOMER114
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_114
INDEX IN is_C2_114
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 144415 ENDING AT 145692,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER115.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER115;
CREATE TABLE CUSTOMER115
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_115
```

```
        INDEX IN is_C2_115
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 145693 ENDING AT 146970,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER116.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER116;
CREATE TABLE CUSTOMER116
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_116
        INDEX IN is_C2_116
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 146971 ENDING AT 148248,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER117.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER117;
CREATE TABLE CUSTOMER117
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_117
        INDEX IN is_C2_117
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 148249 ENDING AT 149526,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER118.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER118;
CREATE TABLE CUSTOMER118
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_118
        INDEX IN is_C2_118
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 149527 ENDING AT 150804,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER119.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER119;
CREATE TABLE CUSTOMER119
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_119
        INDEX IN is_C2_119
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 150805 ENDING AT 152082,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER11.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER11;
CREATE TABLE CUSTOMER11
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_011
        INDEX IN is_C2_011
        ORGANIZE BY KEY SEQUENCE (
```

```
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 12781 ENDING AT 14058,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER120.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER120;
CREATE TABLE CUSTOMER120
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_120
           INDEX IN is_C2_120
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 152083 ENDING AT 153360,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER121.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER121;
CREATE TABLE CUSTOMER121
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_121
           INDEX IN is_C2_121
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 153361 ENDING AT 154638,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER122.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER122;
CREATE TABLE CUSTOMER122
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_122
           INDEX IN is_C2_122
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 154639 ENDING AT 155916,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER123.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER123;
CREATE TABLE CUSTOMER123
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_123
           INDEX IN is_C2_123
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 155917 ENDING AT 157194,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER124.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER124;
CREATE TABLE CUSTOMER124
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_124
           INDEX IN is_C2_124
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 157195 ENDING AT 158472,
```

```
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER125.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER125;
CREATE TABLE CUSTOMER125
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER       NOT NULL
       )
       IN ts_C_125
       INDEX IN is_C2_125
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 158473 ENDING AT 159750,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER126.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER126;
CREATE TABLE CUSTOMER126
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER       NOT NULL
       )
       IN ts_C_126
       INDEX IN is_C2_126
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 159751 ENDING AT 161028,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER127.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER127;
CREATE TABLE CUSTOMER127
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER       NOT NULL
       )
       IN ts_C_127
       INDEX IN is_C2_127
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 161029 ENDING AT 162306,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER128.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER128;
CREATE TABLE CUSTOMER128
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER       NOT NULL
       )
       IN ts_C_128
       INDEX IN is_C2_128
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 162307 ENDING AT 163584,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER129.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER129;
CREATE TABLE CUSTOMER129
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER       NOT NULL
       )
       IN ts_C_129
       INDEX IN is_C2_129
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 163585 ENDING AT 164862,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
```

```
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER12.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER12;
CREATE TABLE CUSTOMER12
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_012
        INDEX IN is_C2_012
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 14059 ENDING AT 15336,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER130.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER130;
CREATE TABLE CUSTOMER130
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_130
        INDEX IN is_C2_130
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 164863 ENDING AT 166140,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER131.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER131;
CREATE TABLE CUSTOMER131
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_131
        INDEX IN is_C2_131
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 166141 ENDING AT 167418,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER132.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER132;
CREATE TABLE CUSTOMER132
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_132
        INDEX IN is_C2_132
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 167419 ENDING AT 168696,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER133.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER133;
CREATE TABLE CUSTOMER133
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_133
        INDEX IN is_C2_133
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 168697 ENDING AT 169974,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER134.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER134;
CREATE TABLE CUSTOMER134
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_134
        INDEX IN is_C2_134
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 169975 ENDING AT 171252,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER135.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER135;
CREATE TABLE CUSTOMER135
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_135
        INDEX IN is_C2_135
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 171253 ENDING AT 172530,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER136.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER136;
CREATE TABLE CUSTOMER136
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_136
        INDEX IN is_C2_136
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 172531 ENDING AT 173808,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER137.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER137;
CREATE TABLE CUSTOMER137
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_137
        INDEX IN is_C2_137
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 173809 ENDING AT 175086,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER138.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER138;
CREATE TABLE CUSTOMER138
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_138
        INDEX IN is_C2_138
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 175087 ENDING AT 176364,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER139.ddl

```
connect to TPCC in share mode;
```

```
DROP TABLE CUSTOMER139;
CREATE TABLE CUSTOMER139
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_139
        INDEX IN is_C2_139
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 176365 ENDING AT 177642,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER13.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER13;
CREATE TABLE CUSTOMER13
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_013
        INDEX IN is_C2_013
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 15337 ENDING AT 16614,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER140.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER140;
CREATE TABLE CUSTOMER140
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_140
        INDEX IN is_C2_140
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 177643 ENDING AT 178920,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER141.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER141;
CREATE TABLE CUSTOMER141
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_141
        INDEX IN is_C2_141
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 178921 ENDING AT 180198,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER142.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER142;
CREATE TABLE CUSTOMER142
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_142
        INDEX IN is_C2_142
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 180199 ENDING AT 181476,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER143.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER143;
CREATE TABLE CUSTOMER143
```

```
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_143
INDEX IN is_C2_143
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 181477 ENDING AT 182754,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER144.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER144;
CREATE TABLE CUSTOMER144
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_144
INDEX IN is_C2_144
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 182755 ENDING AT 184032,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER14.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER14;
CREATE TABLE CUSTOMER14
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_014
INDEX IN is_C2_014
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 16615 ENDING AT 17892,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER15.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER15;
CREATE TABLE CUSTOMER15
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_015
INDEX IN is_C2_015
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 17893 ENDING AT 19170,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER16.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER16;
CREATE TABLE CUSTOMER16
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_016
INDEX IN is_C2_016
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 19171 ENDING AT 20448,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER17.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER17;
CREATE TABLE CUSTOMER17
(
 C_ID          INTEGER       NOT NULL,
```

```
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_017
        INDEX IN is_C2_017
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 20449 ENDING AT 21726,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER18.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER18;
CREATE TABLE CUSTOMER18
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_018
        INDEX IN is_C2_018
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 21727 ENDING AT 23004,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER19.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER19;
CREATE TABLE CUSTOMER19
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_019
        INDEX IN is_C2_019
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 23005 ENDING AT 24282,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER1.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER1;
CREATE TABLE CUSTOMER1
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_001
        INDEX IN is_C2_001
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 1 ENDING AT 1278,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER20.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER20;
CREATE TABLE CUSTOMER20
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_020
        INDEX IN is_C2_020
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 24283 ENDING AT 25560,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER21.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER21;
CREATE TABLE CUSTOMER21
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
```

```
 C_PHONE        CHAR(16)      NOT NULL,
 C_SINCE        TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_021
INDEX IN is_C2_021
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 25561 ENDING AT 26838,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER22.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER22;
CREATE TABLE CUSTOMER22
(
 C_ID            INTEGER       NOT NULL,
 C_STATE         CHAR(2)       NOT NULL,
 C_ZIP           CHAR(9)       NOT NULL,
 C_PHONE         CHAR(16)      NOT NULL,
 C_SINCE         TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_022
INDEX IN is_C2_022
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 26839 ENDING AT 28116,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER23.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER23;
CREATE TABLE CUSTOMER23
(
 C_ID            INTEGER       NOT NULL,
 C_STATE         CHAR(2)       NOT NULL,
 C_ZIP           CHAR(9)       NOT NULL,
 C_PHONE         CHAR(16)      NOT NULL,
 C_SINCE         TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_023
INDEX IN is_C2_023
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 28117 ENDING AT 29394,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER24.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER24;
CREATE TABLE CUSTOMER24
(
 C_ID            INTEGER       NOT NULL,
 C_STATE         CHAR(2)       NOT NULL,
 C_ZIP           CHAR(9)       NOT NULL,
 C_PHONE         CHAR(16)      NOT NULL,
 C_SINCE         TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_024
INDEX IN is_C2_024
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 29395 ENDING AT 30672,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER25.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER25;
CREATE TABLE CUSTOMER25
(
 C_ID            INTEGER       NOT NULL,
 C_STATE         CHAR(2)       NOT NULL,
 C_ZIP           CHAR(9)       NOT NULL,
 C_PHONE         CHAR(16)      NOT NULL,
 C_SINCE         TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_025
INDEX IN is_C2_025
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 30673 ENDING AT 31950,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER26.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER26;
CREATE TABLE CUSTOMER26
(
 C_ID            INTEGER       NOT NULL,
 C_STATE         CHAR(2)       NOT NULL,
 C_ZIP           CHAR(9)       NOT NULL,
 C_PHONE         CHAR(16)      NOT NULL,
 C_SINCE         TIMESTAMP     NOT NULL,
```

```
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_026
       INDEX IN is_C2_026
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 31951 ENDING AT 33228,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER27.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER27;
CREATE TABLE CUSTOMER27
       (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_027
       INDEX IN is_C2_027
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 33229 ENDING AT 34506,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER28.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER28;
CREATE TABLE CUSTOMER28
       (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_028
       INDEX IN is_C2_028
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 34507 ENDING AT 35784,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER29.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER29;
CREATE TABLE CUSTOMER29
       (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_029
       INDEX IN is_C2_029
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 35785 ENDING AT 37062,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER2.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER2;
CREATE TABLE CUSTOMER2
       (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_002
       INDEX IN is_C2_002
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 1279 ENDING AT 2556,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER30.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER30;
CREATE TABLE CUSTOMER30
       (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
```

```
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_030
        INDEX IN is_C2_030
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 37063 ENDING AT 38340,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER31.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER31;
CREATE TABLE CUSTOMER31
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_031
        INDEX IN is_C2_031
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 38341 ENDING AT 39618,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER32.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER32;
CREATE TABLE CUSTOMER32
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_032
        INDEX IN is_C2_032
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 39619 ENDING AT 40896,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER33.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER33;
CREATE TABLE CUSTOMER33
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_033
        INDEX IN is_C2_033
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 40897 ENDING AT 42174,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER34.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER34;
CREATE TABLE CUSTOMER34
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_034
        INDEX IN is_C2_034
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 42175 ENDING AT 43452,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER35.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER35;
CREATE TABLE CUSTOMER35
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
```

```
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
       )
       IN ts_C_035
       INDEX IN is_C2_035
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 43453 ENDING AT 44730,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER36.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER36;
CREATE TABLE CUSTOMER36
       (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
       )
       IN ts_C_036
       INDEX IN is_C2_036
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 44731 ENDING AT 46008,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER37.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER37;
CREATE TABLE CUSTOMER37
       (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
       )
       IN ts_C_037
       INDEX IN is_C2_037
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 46009 ENDING AT 47286,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER38.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER38;
CREATE TABLE CUSTOMER38
       (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
       )
       IN ts_C_038
       INDEX IN is_C2_038
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 47287 ENDING AT 48564,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER39.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER39;
CREATE TABLE CUSTOMER39
       (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
       )
       IN ts_C_039
       INDEX IN is_C2_039
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 48565 ENDING AT 49842,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER3.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER3;
CREATE TABLE CUSTOMER3
       (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
```

```
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_003
        INDEX IN is_C2_003
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 2557 ENDING AT 3834,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER40.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER40;
CREATE TABLE CUSTOMER40
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_040
        INDEX IN is_C2_040
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 49843 ENDING AT 51120,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER41.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER41;
CREATE TABLE CUSTOMER41
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_041
        INDEX IN is_C2_041
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 51121 ENDING AT 52398,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER42.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER42;
CREATE TABLE CUSTOMER42
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_042
        INDEX IN is_C2_042
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 52399 ENDING AT 53676,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER43.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER43;
CREATE TABLE CUSTOMER43
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
        )
        IN ts_C_043
        INDEX IN is_C2_043
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 53677 ENDING AT 54954,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER44.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER44;
CREATE TABLE CUSTOMER44
        (
        C_ID           INTEGER       NOT NULL,
        C_STATE        CHAR(2)       NOT NULL,
        C_ZIP          CHAR(9)       NOT NULL,
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
```

```
 C_D_ID         SMALLINT      NOT NULL,
 C_W_ID         INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER      NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_044
INDEX IN is_C2_044
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 54955 ENDING AT 56232,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER45.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER45;
CREATE TABLE CUSTOMER45
(
 C_ID           INTEGER       NOT NULL,
 C_STATE        CHAR(2)       NOT NULL,
 C_ZIP          CHAR(9)       NOT NULL,
 C_PHONE        CHAR(16)      NOT NULL,
 C_SINCE        TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_045
INDEX IN is_C2_045
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 56233 ENDING AT 57510,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER46.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER46;
CREATE TABLE CUSTOMER46
(
 C_ID           INTEGER       NOT NULL,
 C_STATE        CHAR(2)       NOT NULL,
 C_ZIP          CHAR(9)       NOT NULL,
 C_PHONE        CHAR(16)      NOT NULL,
 C_SINCE        TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_046
INDEX IN is_C2_046
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 57511 ENDING AT 58788,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER47.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER47;
CREATE TABLE CUSTOMER47
(
 C_ID           INTEGER       NOT NULL,
 C_STATE        CHAR(2)       NOT NULL,
 C_ZIP          CHAR(9)       NOT NULL,
 C_PHONE        CHAR(16)      NOT NULL,
 C_SINCE        TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_047
INDEX IN is_C2_047
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 58789 ENDING AT 60066,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER48.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER48;
CREATE TABLE CUSTOMER48
(
 C_ID           INTEGER       NOT NULL,
 C_STATE        CHAR(2)       NOT NULL,
 C_ZIP          CHAR(9)       NOT NULL,
 C_PHONE        CHAR(16)      NOT NULL,
 C_SINCE        TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
 C_DELIVERY_CNT  INTEGER       NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER       NOT NULL
)
IN ts_C_048
INDEX IN is_C2_048
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 60067 ENDING AT 61344,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER49.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER49;
CREATE TABLE CUSTOMER49
(
 C_ID           INTEGER       NOT NULL,
 C_STATE        CHAR(2)       NOT NULL,
 C_ZIP          CHAR(9)       NOT NULL,
 C_PHONE        CHAR(16)      NOT NULL,
 C_SINCE        TIMESTAMP     NOT NULL,
 C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
 C_MIDDLE        CHAR(2)       NOT NULL,
 C_CREDIT        CHAR(2)       NOT NULL,
 C_DISCOUNT      REAL          NOT NULL,
 C_DATA          VARCHAR(500)  NOT NULL,
 C_LAST          VARCHAR(16)   NOT NULL,
 C_FIRST         VARCHAR(16)   NOT NULL,
 C_STREET_1      VARCHAR(20)   NOT NULL,
 C_STREET_2      VARCHAR(20)   NOT NULL,
 C_CITY          VARCHAR(20)   NOT NULL,
 C_D_ID          SMALLINT      NOT NULL,
 C_W_ID          INTEGER       NOT NULL,
```

```
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_049
        INDEX IN is_C2_049
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 61345 ENDING AT 62622,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER4.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER4;
CREATE TABLE CUSTOMER4
        (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_004
        INDEX IN is_C2_004
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 3835 ENDING AT 5112,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER50.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER50;
CREATE TABLE CUSTOMER50
        (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_050
        INDEX IN is_C2_050
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 62623 ENDING AT 63900,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER51.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER51;
CREATE TABLE CUSTOMER51
        (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_051
        INDEX IN is_C2_051
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 63901 ENDING AT 65178,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER52.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER52;
CREATE TABLE CUSTOMER52
        (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
        )
        IN ts_C_052
        INDEX IN is_C2_052
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 65179 ENDING AT 66456,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER53.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER53;
CREATE TABLE CUSTOMER53
        (
        C_ID            INTEGER       NOT NULL,
        C_STATE         CHAR(2)       NOT NULL,
        C_ZIP           CHAR(9)       NOT NULL,
        C_PHONE         CHAR(16)      NOT NULL,
        C_SINCE         TIMESTAMP     NOT NULL,
        C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,
        C_MIDDLE        CHAR(2)       NOT NULL,
        C_CREDIT        CHAR(2)       NOT NULL,
        C_DISCOUNT      REAL          NOT NULL,
        C_DATA          VARCHAR(500)  NOT NULL,
        C_LAST          VARCHAR(16)   NOT NULL,
        C_FIRST         VARCHAR(16)   NOT NULL,
        C_STREET_1      VARCHAR(20)   NOT NULL,
        C_STREET_2      VARCHAR(20)   NOT NULL,
        C_CITY          VARCHAR(20)   NOT NULL,
        C_D_ID          SMALLINT      NOT NULL,
        C_W_ID          INTEGER       NOT NULL,
        C_DELIVERY_CNT  INTEGER       NOT NULL,
        C_BALANCE       DECIMAL(12,2) NOT NULL,
```

```
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_053
       INDEX IN is_C2_053
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 66457 ENDING AT 67734,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER54.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER54;
CREATE TABLE CUSTOMER54
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_054
       INDEX IN is_C2_054
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 67735 ENDING AT 69012,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER55.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER55;
CREATE TABLE CUSTOMER55
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_055
       INDEX IN is_C2_055
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 69013 ENDING AT 70290,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER56.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER56;
CREATE TABLE CUSTOMER56
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_056
       INDEX IN is_C2_056
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 70291 ENDING AT 71568,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER57.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER57;
CREATE TABLE CUSTOMER57
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
       )
       IN ts_C_057
       INDEX IN is_C2_057
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 71569 ENDING AT 72846,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER58.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER58;
CREATE TABLE CUSTOMER58
       (
        C_ID          INTEGER       NOT NULL,
        C_STATE       CHAR(2)       NOT NULL,
        C_ZIP         CHAR(9)       NOT NULL,
        C_PHONE       CHAR(16)      NOT NULL,
        C_SINCE       TIMESTAMP     NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)       NOT NULL,
        C_CREDIT      CHAR(2)       NOT NULL,
        C_DISCOUNT    REAL          NOT NULL,
        C_DATA        VARCHAR(500)  NOT NULL,
        C_LAST        VARCHAR(16)   NOT NULL,
        C_FIRST       VARCHAR(16)   NOT NULL,
        C_STREET_1    VARCHAR(20)   NOT NULL,
        C_STREET_2    VARCHAR(20)   NOT NULL,
        C_CITY        VARCHAR(20)   NOT NULL,
        C_D_ID        SMALLINT      NOT NULL,
        C_W_ID        INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT   INTEGER       NOT NULL
```

```
)
IN ts_C_058
INDEX IN is_C2_058
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 72847 ENDING AT 74124,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER59.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER59;
CREATE TABLE CUSTOMER59
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_059
INDEX IN is_C2_059
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 74125 ENDING AT 75402,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER5.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER5;
CREATE TABLE CUSTOMER5
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_005
INDEX IN is_C2_005
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 5113 ENDING AT 6390,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER60.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER60;
CREATE TABLE CUSTOMER60
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_060
INDEX IN is_C2_060
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 75403 ENDING AT 76680,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER61.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER61;
CREATE TABLE CUSTOMER61
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_061
INDEX IN is_C2_061
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 76681 ENDING AT 77958,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER62.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER62;
CREATE TABLE CUSTOMER62
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_062
```

```
        INDEX IN is_C2_062
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 77959 ENDING AT 79236,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER63.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER63;
CREATE TABLE CUSTOMER63
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_063
        INDEX IN is_C2_063
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 79237 ENDING AT 80514,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER64.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER64;
CREATE TABLE CUSTOMER64
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_064
        INDEX IN is_C2_064
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 80515 ENDING AT 81792,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER65.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER65;
CREATE TABLE CUSTOMER65
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_065
        INDEX IN is_C2_065
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 81793 ENDING AT 83070,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER66.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER66;
CREATE TABLE CUSTOMER66
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_066
        INDEX IN is_C2_066
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 83071 ENDING AT 84348,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER67.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER67;
CREATE TABLE CUSTOMER67
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_067
        INDEX IN is_C2_067
        ORGANIZE BY KEY SEQUENCE (
```

```
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 84349 ENDING AT 85626,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER68.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER68;
CREATE TABLE CUSTOMER68
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_068
           INDEX IN is_C2_068
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 85627 ENDING AT 86904,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER69.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER69;
CREATE TABLE CUSTOMER69
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_069
           INDEX IN is_C2_069
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 86905 ENDING AT 88182,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER6.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER6;
CREATE TABLE CUSTOMER6
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_006
           INDEX IN is_C2_006
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 6391 ENDING AT 7668,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER70.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER70;
CREATE TABLE CUSTOMER70
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_070
           INDEX IN is_C2_070
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 88183 ENDING AT 89460,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER71.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER71;
CREATE TABLE CUSTOMER71
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_071
           INDEX IN is_C2_071
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 89461 ENDING AT 90738,
```

```
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER72.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER72;
CREATE TABLE CUSTOMER72
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_072
           INDEX IN is_C2_072
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 90739 ENDING AT 92016,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER73.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER73;
CREATE TABLE CUSTOMER73
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_073
           INDEX IN is_C2_073
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 92017 ENDING AT 93294,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER74.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER74;
CREATE TABLE CUSTOMER74
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_074
           INDEX IN is_C2_074
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 93295 ENDING AT 94572,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER75.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER75;
CREATE TABLE CUSTOMER75
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_075
           INDEX IN is_C2_075
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 94573 ENDING AT 95850,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER76.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER76;
CREATE TABLE CUSTOMER76
           (
            C_ID          INTEGER       NOT NULL,
            C_STATE       CHAR(2)       NOT NULL,
            C_ZIP         CHAR(9)       NOT NULL,
            C_PHONE       CHAR(16)      NOT NULL,
            C_SINCE       TIMESTAMP     NOT NULL,
            C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
            C_MIDDLE      CHAR(2)       NOT NULL,
            C_CREDIT      CHAR(2)       NOT NULL,
            C_DISCOUNT    REAL          NOT NULL,
            C_DATA        VARCHAR(500)  NOT NULL,
            C_LAST        VARCHAR(16)   NOT NULL,
            C_FIRST       VARCHAR(16)   NOT NULL,
            C_STREET_1    VARCHAR(20)   NOT NULL,
            C_STREET_2    VARCHAR(20)   NOT NULL,
            C_CITY        VARCHAR(20)   NOT NULL,
            C_D_ID        SMALLINT      NOT NULL,
            C_W_ID        INTEGER       NOT NULL,
            C_DELIVERY_CNT INTEGER      NOT NULL,
            C_BALANCE     DECIMAL(12,2) NOT NULL,
            C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
            C_PAYMENT_CNT INTEGER       NOT NULL
           )
           IN ts_C_076
           INDEX IN is_C2_076
           ORGANIZE BY KEY SEQUENCE (
            C_ID STARTING FROM 1 ENDING AT 3000,
            C_W_ID STARTING FROM 95851 ENDING AT 97128,
            C_D_ID STARTING FROM 1 ENDING AT 10
           )
```

```
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER77.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER77;
CREATE TABLE CUSTOMER77
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_077
        INDEX IN is_C2_077
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 97129 ENDING AT 98406,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER78.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER78;
CREATE TABLE CUSTOMER78
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_078
        INDEX IN is_C2_078
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 98407 ENDING AT 99684,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER79.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER79;
CREATE TABLE CUSTOMER79
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_079
        INDEX IN is_C2_079
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 99685 ENDING AT 100962,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER7.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER7;
CREATE TABLE CUSTOMER7
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_007
        INDEX IN is_C2_007
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 7669 ENDING AT 8946,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER80.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER80;
CREATE TABLE CUSTOMER80
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_080
        INDEX IN is_C2_080
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 100963 ENDING AT 102240,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER81.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER81;
CREATE TABLE CUSTOMER81
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_081
        INDEX IN is_C2_081
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 102241 ENDING AT 103518,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER82.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER82;
CREATE TABLE CUSTOMER82
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_082
        INDEX IN is_C2_082
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 103519 ENDING AT 104796,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER83.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER83;
CREATE TABLE CUSTOMER83
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_083
        INDEX IN is_C2_083
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 104797 ENDING AT 106074,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER84.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER84;
CREATE TABLE CUSTOMER84
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_084
        INDEX IN is_C2_084
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 106075 ENDING AT 107352,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER85.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER85;
CREATE TABLE CUSTOMER85
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_085
        INDEX IN is_C2_085
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 107353 ENDING AT 108630,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER86.ddl

```
connect to TPCC in share mode;
```

```
DROP TABLE CUSTOMER86;
CREATE TABLE CUSTOMER86
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_086
        INDEX IN is_C2_086
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 108631 ENDING AT 109908,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER87.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER87;
CREATE TABLE CUSTOMER87
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_087
        INDEX IN is_C2_087
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 109909 ENDING AT 111186,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER88.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER88;
CREATE TABLE CUSTOMER88
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_088
        INDEX IN is_C2_088
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 111187 ENDING AT 112464,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER89.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER89;
CREATE TABLE CUSTOMER89
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_089
        INDEX IN is_C2_089
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 112465 ENDING AT 113742,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER8.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER8;
CREATE TABLE CUSTOMER8
        (
         C_ID          INTEGER       NOT NULL,
         C_STATE       CHAR(2)       NOT NULL,
         C_ZIP         CHAR(9)       NOT NULL,
         C_PHONE       CHAR(16)      NOT NULL,
         C_SINCE       TIMESTAMP     NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)       NOT NULL,
         C_CREDIT      CHAR(2)       NOT NULL,
         C_DISCOUNT    REAL          NOT NULL,
         C_DATA        VARCHAR(500)  NOT NULL,
         C_LAST        VARCHAR(16)   NOT NULL,
         C_FIRST       VARCHAR(16)   NOT NULL,
         C_STREET_1    VARCHAR(20)   NOT NULL,
         C_STREET_2    VARCHAR(20)   NOT NULL,
         C_CITY        VARCHAR(20)   NOT NULL,
         C_D_ID        SMALLINT      NOT NULL,
         C_W_ID        INTEGER       NOT NULL,
         C_DELIVERY_CNT INTEGER      NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER       NOT NULL
        )
        IN ts_C_008
        INDEX IN is_C2_008
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 8947 ENDING AT 10224,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER90.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER90;
CREATE TABLE CUSTOMER90
```

```
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_090
INDEX IN is_C2_090
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 113743 ENDING AT 115020,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER91.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER91;
CREATE TABLE CUSTOMER91
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_091
INDEX IN is_C2_091
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 115021 ENDING AT 116298,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER92.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER92;
CREATE TABLE CUSTOMER92
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_092
INDEX IN is_C2_092
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 116299 ENDING AT 117576,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER93.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER93;
CREATE TABLE CUSTOMER93
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_093
INDEX IN is_C2_093
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 117577 ENDING AT 118854,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER94.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER94;
CREATE TABLE CUSTOMER94
(
 C_ID          INTEGER       NOT NULL,
 C_STATE       CHAR(2)       NOT NULL,
 C_ZIP         CHAR(9)       NOT NULL,
 C_PHONE       CHAR(16)      NOT NULL,
 C_SINCE       TIMESTAMP     NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)       NOT NULL,
 C_CREDIT      CHAR(2)       NOT NULL,
 C_DISCOUNT    REAL          NOT NULL,
 C_DATA        VARCHAR(500)  NOT NULL,
 C_LAST        VARCHAR(16)   NOT NULL,
 C_FIRST       VARCHAR(16)   NOT NULL,
 C_STREET_1    VARCHAR(20)   NOT NULL,
 C_STREET_2    VARCHAR(20)   NOT NULL,
 C_CITY        VARCHAR(20)   NOT NULL,
 C_D_ID        SMALLINT      NOT NULL,
 C_W_ID        INTEGER       NOT NULL,
 C_DELIVERY_CNT INTEGER      NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER       NOT NULL
)
IN ts_C_094
INDEX IN is_C2_094
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 118855 ENDING AT 120132,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER95.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER95;
CREATE TABLE CUSTOMER95
(
 C_ID          INTEGER       NOT NULL,
```

```
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_C_095
INDEX IN is_C2_095
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 120133 ENDING AT 121410,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER96.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER96;
CREATE TABLE CUSTOMER96
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_C_096
INDEX IN is_C2_096
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 121411 ENDING AT 122688,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER97.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER97;
CREATE TABLE CUSTOMER97
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_C_097
INDEX IN is_C2_097
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 122689 ENDING AT 123966,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER98.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER98;
CREATE TABLE CUSTOMER98
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_C_098
INDEX IN is_C2_098
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 123967 ENDING AT 125244,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER99.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER99;
CREATE TABLE CUSTOMER99
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_C_099
INDEX IN is_C2_099
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 125245 ENDING AT 126522,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_CUSTOMER9.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER9;
CREATE TABLE CUSTOMER9
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
```

```
        C_PHONE        CHAR(16)      NOT NULL,
        C_SINCE        TIMESTAMP     NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)       NOT NULL,
        C_CREDIT       CHAR(2)       NOT NULL,
        C_DISCOUNT     REAL          NOT NULL,
        C_DATA         VARCHAR(500)  NOT NULL,
        C_LAST         VARCHAR(16)   NOT NULL,
        C_FIRST        VARCHAR(16)   NOT NULL,
        C_STREET_1     VARCHAR(20)   NOT NULL,
        C_STREET_2     VARCHAR(20)   NOT NULL,
        C_CITY         VARCHAR(20)   NOT NULL,
        C_D_ID         SMALLINT      NOT NULL,
        C_W_ID         INTEGER       NOT NULL,
        C_DELIVERY_CNT INTEGER       NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER       NOT NULL
       )
       IN ts_C_009
       INDEX IN is_C2_009
       ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 10225 ENDING AT 11502,
        C_D_ID STARTING FROM 1 ENDING AT 10
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT10.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT10;
CREATE TABLE DISTRICT10
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_010
       INDEX IN ts_D_010
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 46009 ENDING AT 51120
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT11.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT11;
CREATE TABLE DISTRICT11
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_011
       INDEX IN ts_D_011
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 51121 ENDING AT 56232
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT12.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT12;
CREATE TABLE DISTRICT12
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_012
       INDEX IN ts_D_012
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 56233 ENDING AT 61344
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT13.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT13;
CREATE TABLE DISTRICT13
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_013
       INDEX IN ts_D_013
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 61345 ENDING AT 66456
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT14.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT14;
CREATE TABLE DISTRICT14
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_014
       INDEX IN ts_D_014
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 66457 ENDING AT 71568
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT15.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT15;
CREATE TABLE DISTRICT15
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_015
       INDEX IN ts_D_015
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 71569 ENDING AT 76680
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT16.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT16;
CREATE TABLE DISTRICT16
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_016
        INDEX IN ts_D_016
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 76681 ENDING AT 81792
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT17.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT17;
CREATE TABLE DISTRICT17
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_017
        INDEX IN ts_D_017
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 81793 ENDING AT 86904
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT18.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT18;
CREATE TABLE DISTRICT18
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_018
        INDEX IN ts_D_018
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 86905 ENDING AT 92016
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT19.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT19;
CREATE TABLE DISTRICT19
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_019
        INDEX IN ts_D_019
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 92017 ENDING AT 97128
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT1.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT1;
CREATE TABLE DISTRICT1
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_001
        INDEX IN ts_D_001
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 1 ENDING AT 5112
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT20.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT20;
CREATE TABLE DISTRICT20
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_020
        INDEX IN ts_D_020
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 97129 ENDING AT 102240
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT21.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT21;
CREATE TABLE DISTRICT21
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_021
        INDEX IN ts_D_021
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 102241 ENDING AT 107352
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT22.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT22;
```

```
CREATE TABLE DISTRICT22
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_022
        INDEX IN ts_D_022
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 107353 ENDING AT 112464
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT23.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT23;
CREATE TABLE DISTRICT23
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_023
        INDEX IN ts_D_023
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 112465 ENDING AT 117576
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT24.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT24;
CREATE TABLE DISTRICT24
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_024
        INDEX IN ts_D_024
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 117577 ENDING AT 122688
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT25.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT25;
CREATE TABLE DISTRICT25
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_025
        INDEX IN ts_D_025
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 122689 ENDING AT 127800
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT26.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT26;
CREATE TABLE DISTRICT26
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_026
        INDEX IN ts_D_026
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 127801 ENDING AT 132912
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT27.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT27;
CREATE TABLE DISTRICT27
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_027
        INDEX IN ts_D_027
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 132913 ENDING AT 138024
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT28.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT28;
CREATE TABLE DISTRICT28
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_D_028
        INDEX IN ts_D_028
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 138025 ENDING AT 143136
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT29.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT29;
CREATE TABLE DISTRICT29
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
```

```
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)   NOT NULL,
 D_NAME      CHAR(10)        NOT NULL,
 D_STREET_1  CHAR(20)        NOT NULL,
 D_STREET_2  CHAR(20)        NOT NULL,
 D_CITY      CHAR(20)        NOT NULL,
 D_STATE     CHAR(2)         NOT NULL,
 D_ZIP       CHAR(9)         NOT NULL,
 D_ID        SMALLINT        NOT NULL,
 D_W_ID      INTEGER         NOT NULL
)
IN ts_D_029
INDEX IN ts_D_029
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 143137 ENDING AT 148248
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT2.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT2;
CREATE TABLE DISTRICT2
(
 D_NEXT_O_ID INTEGER         NOT NULL,
 D_TAX       REAL            NOT NULL,
 D_YTD       DECIMAL(12,2)   NOT NULL,
 D_NAME      CHAR(10)        NOT NULL,
 D_STREET_1  CHAR(20)        NOT NULL,
 D_STREET_2  CHAR(20)        NOT NULL,
 D_CITY      CHAR(20)        NOT NULL,
 D_STATE     CHAR(2)         NOT NULL,
 D_ZIP       CHAR(9)         NOT NULL,
 D_ID        SMALLINT        NOT NULL,
 D_W_ID      INTEGER         NOT NULL
)
IN ts_D_002
INDEX IN ts_D_002
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 5113 ENDING AT 10224
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT30.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT30;
CREATE TABLE DISTRICT30
(
 D_NEXT_O_ID INTEGER         NOT NULL,
 D_TAX       REAL            NOT NULL,
 D_YTD       DECIMAL(12,2)   NOT NULL,
 D_NAME      CHAR(10)        NOT NULL,
 D_STREET_1  CHAR(20)        NOT NULL,
 D_STREET_2  CHAR(20)        NOT NULL,
 D_CITY      CHAR(20)        NOT NULL,
 D_STATE     CHAR(2)         NOT NULL,
 D_ZIP       CHAR(9)         NOT NULL,
 D_ID        SMALLINT        NOT NULL,
 D_W_ID      INTEGER         NOT NULL
)
IN ts_D_030
INDEX IN ts_D_030
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 148249 ENDING AT 153360
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT31.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT31;
CREATE TABLE DISTRICT31
(
 D_NEXT_O_ID INTEGER         NOT NULL,
 D_TAX       REAL            NOT NULL,
 D_YTD       DECIMAL(12,2)   NOT NULL,
 D_NAME      CHAR(10)        NOT NULL,
 D_STREET_1  CHAR(20)        NOT NULL,
 D_STREET_2  CHAR(20)        NOT NULL,
 D_CITY      CHAR(20)        NOT NULL,
 D_STATE     CHAR(2)         NOT NULL,
 D_ZIP       CHAR(9)         NOT NULL,
 D_ID        SMALLINT        NOT NULL,
 D_W_ID      INTEGER         NOT NULL
)
IN ts_D_031
INDEX IN ts_D_031
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 153361 ENDING AT 158472
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT32.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT32;
CREATE TABLE DISTRICT32
(
 D_NEXT_O_ID INTEGER         NOT NULL,
 D_TAX       REAL            NOT NULL,
 D_YTD       DECIMAL(12,2)   NOT NULL,
 D_NAME      CHAR(10)        NOT NULL,
 D_STREET_1  CHAR(20)        NOT NULL,
 D_STREET_2  CHAR(20)        NOT NULL,
 D_CITY      CHAR(20)        NOT NULL,
 D_STATE     CHAR(2)         NOT NULL,
 D_ZIP       CHAR(9)         NOT NULL,
 D_ID        SMALLINT        NOT NULL,
 D_W_ID      INTEGER         NOT NULL
)
IN ts_D_032
INDEX IN ts_D_032
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 158473 ENDING AT 163584
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT33.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT33;
CREATE TABLE DISTRICT33
(
 D_NEXT_O_ID INTEGER         NOT NULL,
 D_TAX       REAL            NOT NULL,
 D_YTD       DECIMAL(12,2)   NOT NULL,
 D_NAME      CHAR(10)        NOT NULL,
 D_STREET_1  CHAR(20)        NOT NULL,
 D_STREET_2  CHAR(20)        NOT NULL,
 D_CITY      CHAR(20)        NOT NULL,
 D_STATE     CHAR(2)         NOT NULL,
 D_ZIP       CHAR(9)         NOT NULL,
 D_ID        SMALLINT        NOT NULL,
 D_W_ID      INTEGER         NOT NULL
)
IN ts_D_033
INDEX IN ts_D_033
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 163585 ENDING AT 168696
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT34.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT34;
CREATE TABLE DISTRICT34
(
 D_NEXT_O_ID INTEGER         NOT NULL,
 D_TAX       REAL            NOT NULL,
 D_YTD       DECIMAL(12,2)   NOT NULL,
 D_NAME      CHAR(10)        NOT NULL,
 D_STREET_1  CHAR(20)        NOT NULL,
 D_STREET_2  CHAR(20)        NOT NULL,
 D_CITY      CHAR(20)        NOT NULL,
 D_STATE     CHAR(2)         NOT NULL,
 D_ZIP       CHAR(9)         NOT NULL,
 D_ID        SMALLINT        NOT NULL,
 D_W_ID      INTEGER         NOT NULL
)
IN ts_D_034
INDEX IN ts_D_034
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 168697 ENDING AT 173808
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT35.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT35;
CREATE TABLE DISTRICT35
(
 D_NEXT_O_ID INTEGER         NOT NULL,
 D_TAX       REAL            NOT NULL,
 D_YTD       DECIMAL(12,2)   NOT NULL,
 D_NAME      CHAR(10)        NOT NULL,
```

```
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_035
       INDEX IN ts_D_035
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 173809 ENDING AT 178920
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT36.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT36;
CREATE TABLE DISTRICT36
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_036
       INDEX IN ts_D_036
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 178921 ENDING AT 184032
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT3.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT3;
CREATE TABLE DISTRICT3
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_003
       INDEX IN ts_D_003
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 10225 ENDING AT 15336
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT4.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT4;
CREATE TABLE DISTRICT4
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_004
       INDEX IN ts_D_004
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 15337 ENDING AT 20448
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT5.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT5;
CREATE TABLE DISTRICT5
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_005
       INDEX IN ts_D_005
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 20449 ENDING AT 25560
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT6.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT6;
CREATE TABLE DISTRICT6
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_006
       INDEX IN ts_D_006
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 25561 ENDING AT 30672
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT7.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT7;
CREATE TABLE DISTRICT7
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_007
       INDEX IN ts_D_007
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 30673 ENDING AT 35784
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT8.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT8;
CREATE TABLE DISTRICT8
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
```

```
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_008
       INDEX IN ts_D_008
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 35785 ENDING AT 40896
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_DISTRICT9.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT9;
CREATE TABLE DISTRICT9
       (
        D_NEXT_O_ID INTEGER        NOT NULL,
        D_TAX       REAL           NOT NULL,
        D_YTD       DECIMAL(12,2)  NOT NULL,
        D_NAME      CHAR(10)       NOT NULL,
        D_STREET_1  CHAR(20)       NOT NULL,
        D_STREET_2  CHAR(20)       NOT NULL,
        D_CITY      CHAR(20)       NOT NULL,
        D_STATE     CHAR(2)        NOT NULL,
        D_ZIP       CHAR(9)        NOT NULL,
        D_ID        SMALLINT       NOT NULL,
        D_W_ID      INTEGER        NOT NULL
       )
       IN ts_D_009
       INDEX IN ts_D_009
       ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 40897 ENDING AT 46008
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_HISTORY10.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY10;
CREATE TABLE HISTORY10
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_010
       INDEX IN ts_H_010;
ALTER TABLE HISTORY10 APPEND ON;
connect reset;
```

CRTB_HISTORY11.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY11;
CREATE TABLE HISTORY11
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_011
       INDEX IN ts_H_011;
ALTER TABLE HISTORY11 APPEND ON;
connect reset;
```

CRTB_HISTORY12.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY12;
CREATE TABLE HISTORY12
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_012
       INDEX IN ts_H_012;
ALTER TABLE HISTORY12 APPEND ON;
connect reset;
```

CRTB_HISTORY13.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY13;
CREATE TABLE HISTORY13
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_013
       INDEX IN ts_H_013;
ALTER TABLE HISTORY13 APPEND ON;
connect reset;
```

CRTB_HISTORY14.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY14;
CREATE TABLE HISTORY14
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_014
       INDEX IN ts_H_014;
ALTER TABLE HISTORY14 APPEND ON;
connect reset;
```

CRTB_HISTORY15.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY15;
CREATE TABLE HISTORY15
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_015
       INDEX IN ts_H_015;
ALTER TABLE HISTORY15 APPEND ON;
connect reset;
```

CRTB_HISTORY16.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY16;
CREATE TABLE HISTORY16
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_016
       INDEX IN ts_H_016;
ALTER TABLE HISTORY16 APPEND ON;
connect reset;
```

CRTB_HISTORY17.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY17;
CREATE TABLE HISTORY17
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
```

```
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_017
       INDEX IN ts_H_017;
ALTER TABLE HISTORY17 APPEND ON;
connect reset;
```

CRTB_HISTORY18.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY18;
CREATE TABLE HISTORY18
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_018
       INDEX IN ts_H_018;
ALTER TABLE HISTORY18 APPEND ON;
connect reset;
```

CRTB_HISTORY19.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY19;
CREATE TABLE HISTORY19
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_019
       INDEX IN ts_H_019;
ALTER TABLE HISTORY19 APPEND ON;
connect reset;
```

CRTB_HISTORY1.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY1;
CREATE TABLE HISTORY1
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_001
       INDEX IN ts_H_001;
ALTER TABLE HISTORY1 APPEND ON;
connect reset;
```

CRTB_HISTORY20.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY20;
CREATE TABLE HISTORY20
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_020
       INDEX IN ts_H_020;
ALTER TABLE HISTORY20 APPEND ON;
connect reset;
```

CRTB_HISTORY21.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY21;
CREATE TABLE HISTORY21
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_021
       INDEX IN ts_H_021;
ALTER TABLE HISTORY21 APPEND ON;
connect reset;
```

CRTB_HISTORY22.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY22;
CREATE TABLE HISTORY22
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_022
       INDEX IN ts_H_022;
ALTER TABLE HISTORY22 APPEND ON;
connect reset;
```

CRTB_HISTORY23.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY23;
CREATE TABLE HISTORY23
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_023
       INDEX IN ts_H_023;
ALTER TABLE HISTORY23 APPEND ON;
connect reset;
```

CRTB_HISTORY24.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY24;
CREATE TABLE HISTORY24
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_024
       INDEX IN ts_H_024;
ALTER TABLE HISTORY24 APPEND ON;
connect reset;
```

CRTB_HISTORY25.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY25;
CREATE TABLE HISTORY25
       (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
       )
       IN ts_H_025
       INDEX IN ts_H_025;
ALTER TABLE HISTORY25 APPEND ON;
connect reset;
```

CRTB_HISTORY26.ddl

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY26;
CREATE TABLE HISTORY26
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_026
        INDEX IN ts_H_026;
ALTER TABLE HISTORY26 APPEND ON;
connect reset;
```

CRTB_HISTORY27.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY27;
CREATE TABLE HISTORY27
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_027
        INDEX IN ts_H_027;
ALTER TABLE HISTORY27 APPEND ON;
connect reset;
```

CRTB_HISTORY28.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY28;
CREATE TABLE HISTORY28
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_028
        INDEX IN ts_H_028;
ALTER TABLE HISTORY28 APPEND ON;
connect reset;
```

CRTB_HISTORY29.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY29;
CREATE TABLE HISTORY29
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_029
        INDEX IN ts_H_029;
ALTER TABLE HISTORY29 APPEND ON;
connect reset;
```

CRTB_HISTORY2.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY2;
CREATE TABLE HISTORY2
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_002
        INDEX IN ts_H_002;
ALTER TABLE HISTORY2 APPEND ON;
connect reset;
```

CRTB_HISTORY30.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY30;
CREATE TABLE HISTORY30
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_030
        INDEX IN ts_H_030;
ALTER TABLE HISTORY30 APPEND ON;
connect reset;
```

CRTB_HISTORY31.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY31;
CREATE TABLE HISTORY31
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_031
        INDEX IN ts_H_031;
ALTER TABLE HISTORY31 APPEND ON;
connect reset;
```

CRTB_HISTORY32.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY32;
CREATE TABLE HISTORY32
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_032
        INDEX IN ts_H_032;
ALTER TABLE HISTORY32 APPEND ON;
connect reset;
```

CRTB_HISTORY33.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY33;
CREATE TABLE HISTORY33
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_033
        INDEX IN ts_H_033;
ALTER TABLE HISTORY33 APPEND ON;
connect reset;
```

CRTB_HISTORY34.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY34;
CREATE TABLE HISTORY34
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_034
        INDEX IN ts_H_034;
```

```
ALTER TABLE HISTORY34 APPEND ON;
connect reset;
```

CRTB_HISTORY35.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY35;
CREATE TABLE HISTORY35
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_035
        INDEX IN ts_H_035;
ALTER TABLE HISTORY35 APPEND ON;
connect reset;
```

CRTB_HISTORY36.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY36;
CREATE TABLE HISTORY36
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_036
        INDEX IN ts_H_036;
ALTER TABLE HISTORY36 APPEND ON;
connect reset;
```

CRTB_HISTORY3.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY3;
CREATE TABLE HISTORY3
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_003
        INDEX IN ts_H_003;
ALTER TABLE HISTORY3 APPEND ON;
connect reset;
```

CRTB_HISTORY4.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY4;
CREATE TABLE HISTORY4
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_004
        INDEX IN ts_H_004;
ALTER TABLE HISTORY4 APPEND ON;
connect reset;
```

CRTB_HISTORY5.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY5;
CREATE TABLE HISTORY5
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_005
        INDEX IN ts_H_005;
ALTER TABLE HISTORY5 APPEND ON;
connect reset;
```

CRTB_HISTORY6.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY6;
CREATE TABLE HISTORY6
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_006
        INDEX IN ts_H_006;
ALTER TABLE HISTORY6 APPEND ON;
connect reset;
```

CRTB_HISTORY7.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY7;
CREATE TABLE HISTORY7
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_007
        INDEX IN ts_H_007;
ALTER TABLE HISTORY7 APPEND ON;
connect reset;
```

CRTB_HISTORY8.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY8;
CREATE TABLE HISTORY8
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_008
        INDEX IN ts_H_008;
ALTER TABLE HISTORY8 APPEND ON;
connect reset;
```

CRTB_HISTORY9.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY9;
CREATE TABLE HISTORY9
        (
         H_C_ID        INTEGER      NOT NULL,
         H_C_D_ID      SMALLINT     NOT NULL,
         H_C_W_ID      INTEGER      NOT NULL,
         H_D_ID        SMALLINT     NOT NULL,
         H_W_ID        INTEGER      NOT NULL,
         H_DATE        TIMESTAMP    NOT NULL,
         H_AMOUNT      DECIMAL(6,2) NOT NULL,
         H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_H_009
        INDEX IN ts_H_009;
ALTER TABLE HISTORY9 APPEND ON;
connect reset;
```

CRTB_ITEM.ddl

```
connect to TPCC in share mode;
DROP TABLE ITEM;
CREATE TABLE ITEM
        (
         I_NAME        CHAR(24)     NOT NULL,
         I_PRICE       DECIMAL(5,2) NOT NULL,
         I_DATA        VARCHAR(50)  NOT NULL,
```

```
        I_IM_ID       INTEGER     NOT NULL,
        I_ID          INTEGER     NOT NULL
        )
        IN ts_I
        INDEX IN ts_I
        ORGANIZE BY KEY SEQUENCE (
        I_ID STARTING FROM 1 ENDING AT 100000
        )
        ALLOW OVERFLOW;
ALTER TABLE ITEM LOCKSIZE TABLE;
connect reset;
```

CRTB_NEW_ORDER10.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER10;
CREATE TABLE NEW_ORDER10
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_010
        INDEX IN ts_N_010
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 46009 ENDING AT 51120,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER11.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER11;
CREATE TABLE NEW_ORDER11
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_011
        INDEX IN ts_N_011
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 51121 ENDING AT 56232,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER12.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER12;
CREATE TABLE NEW_ORDER12
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_012
        INDEX IN ts_N_012
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 56233 ENDING AT 61344,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER13.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER13;
CREATE TABLE NEW_ORDER13
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_013
        INDEX IN ts_N_013
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 61345 ENDING AT 66456,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER14.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER14;
CREATE TABLE NEW_ORDER14
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_014
        INDEX IN ts_N_014
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 66457 ENDING AT 71568,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER15.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER15;
CREATE TABLE NEW_ORDER15
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_015
        INDEX IN ts_N_015
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 71569 ENDING AT 76680,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER16.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER16;
CREATE TABLE NEW_ORDER16
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_016
        INDEX IN ts_N_016
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 76681 ENDING AT 81792,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER17.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER17;
CREATE TABLE NEW_ORDER17
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_017
        INDEX IN ts_N_017
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 81793 ENDING AT 86904,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER18.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER18;
CREATE TABLE NEW_ORDER18
        (
        NO_O_ID       INTEGER     NOT NULL,
        NO_D_ID       SMALLINT    NOT NULL,
        NO_W_ID       INTEGER     NOT NULL
        )
        IN ts_N_018
        INDEX IN ts_N_018
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 86905 ENDING AT 92016,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER19.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER19;
CREATE TABLE NEW_ORDER19
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_019
        INDEX IN ts_N_019
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 92017 ENDING AT 97128,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER1.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER1;
CREATE TABLE NEW_ORDER1
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_001
        INDEX IN ts_N_001
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 1 ENDING AT 5112,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER20.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER20;
CREATE TABLE NEW_ORDER20
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_020
        INDEX IN ts_N_020
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 97129 ENDING AT 102240,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER21.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER21;
CREATE TABLE NEW_ORDER21
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_021
        INDEX IN ts_N_021
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 102241 ENDING AT 107352,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER22.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER22;
CREATE TABLE NEW_ORDER22
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_022
        INDEX IN ts_N_022
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 107353 ENDING AT 112464,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER23.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER23;
CREATE TABLE NEW_ORDER23
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_023
        INDEX IN ts_N_023
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 112465 ENDING AT 117576,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER24.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER24;
CREATE TABLE NEW_ORDER24
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_024
        INDEX IN ts_N_024
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 117577 ENDING AT 122688,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER25.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER25;
CREATE TABLE NEW_ORDER25
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_025
        INDEX IN ts_N_025
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 122689 ENDING AT 127800,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER26.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER26;
CREATE TABLE NEW_ORDER26
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_026
        INDEX IN ts_N_026
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 127801 ENDING AT 132912,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3740
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER27.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER27;
CREATE TABLE NEW_ORDER27
        (
         NO_O_ID        INTEGER     NOT NULL,
         NO_D_ID        SMALLINT    NOT NULL,
         NO_W_ID        INTEGER     NOT NULL
        )
        IN ts_N_027
        INDEX IN ts_N_027
        ORGANIZE BY KEY SEQUENCE (
```

```
        NO_W_ID STARTING FROM 132913 ENDING AT 138024,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER28.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER28;
CREATE TABLE NEW_ORDER28
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_028
       INDEX IN ts_N_028
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 138025 ENDING AT 143136,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER29.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER29;
CREATE TABLE NEW_ORDER29
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_029
       INDEX IN ts_N_029
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 143137 ENDING AT 148248,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER2.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER2;
CREATE TABLE NEW_ORDER2
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_002
       INDEX IN ts_N_002
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 5113 ENDING AT 10224,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER30.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER30;
CREATE TABLE NEW_ORDER30
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_030
       INDEX IN ts_N_030
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 148249 ENDING AT 153360,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER31.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER31;
CREATE TABLE NEW_ORDER31
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_031
       INDEX IN ts_N_031
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 153361 ENDING AT 158472,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER32.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER32;
CREATE TABLE NEW_ORDER32
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_032
       INDEX IN ts_N_032
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 158473 ENDING AT 163584,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER33.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER33;
CREATE TABLE NEW_ORDER33
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_033
       INDEX IN ts_N_033
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 163585 ENDING AT 168696,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER34.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER34;
CREATE TABLE NEW_ORDER34
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_034
       INDEX IN ts_N_034
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 168697 ENDING AT 173808,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER35.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER35;
CREATE TABLE NEW_ORDER35
       (
        NO_O_ID      INTEGER    NOT NULL,
        NO_D_ID      SMALLINT   NOT NULL,
        NO_W_ID      INTEGER    NOT NULL
       )
       IN ts_N_035
       INDEX IN ts_N_035
       ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 173809 ENDING AT 178920,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3740
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER36.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER36;
CREATE TABLE NEW_ORDER36
```

```
(
 NO_O_ID        INTEGER     NOT NULL,
 NO_D_ID        SMALLINT    NOT NULL,
 NO_W_ID         INTEGER     NOT NULL
)
IN ts_N_036
INDEX IN ts_N_036
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 178921 ENDING AT 184032,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 1900 ENDING AT 3740
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER3.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER3;
CREATE TABLE NEW_ORDER3
(
 NO_O_ID        INTEGER     NOT NULL,
 NO_D_ID        SMALLINT    NOT NULL,
 NO_W_ID         INTEGER     NOT NULL
)
IN ts_N_003
INDEX IN ts_N_003
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 10225 ENDING AT 15336,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 1900 ENDING AT 3740
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER4.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER4;
CREATE TABLE NEW_ORDER4
(
 NO_O_ID        INTEGER     NOT NULL,
 NO_D_ID        SMALLINT    NOT NULL,
 NO_W_ID         INTEGER     NOT NULL
)
IN ts_N_004
INDEX IN ts_N_004
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 15337 ENDING AT 20448,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 1900 ENDING AT 3740
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER5.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER5;
CREATE TABLE NEW_ORDER5
(
 NO_O_ID        INTEGER     NOT NULL,
 NO_D_ID        SMALLINT    NOT NULL,
 NO_W_ID         INTEGER     NOT NULL
)
IN ts_N_005
INDEX IN ts_N_005
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 20449 ENDING AT 25560,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 1900 ENDING AT 3740
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER6.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER6;
CREATE TABLE NEW_ORDER6
(
 NO_O_ID        INTEGER     NOT NULL,
 NO_D_ID        SMALLINT    NOT NULL,
 NO_W_ID         INTEGER     NOT NULL
)
IN ts_N_006
INDEX IN ts_N_006
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 25561 ENDING AT 30672,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 1900 ENDING AT 3740
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER7.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER7;
CREATE TABLE NEW_ORDER7
(
 NO_O_ID        INTEGER     NOT NULL,
 NO_D_ID        SMALLINT    NOT NULL,
 NO_W_ID         INTEGER     NOT NULL
)
IN ts_N_007
INDEX IN ts_N_007
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 30673 ENDING AT 35784,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 1900 ENDING AT 3740
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER8.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER8;
CREATE TABLE NEW_ORDER8
(
 NO_O_ID        INTEGER     NOT NULL,
 NO_D_ID        SMALLINT    NOT NULL,
 NO_W_ID         INTEGER     NOT NULL
)
IN ts_N_008
INDEX IN ts_N_008
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 35785 ENDING AT 40896,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 1900 ENDING AT 3740
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_NEW_ORDER9.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDER9;
CREATE TABLE NEW_ORDER9
(
 NO_O_ID        INTEGER     NOT NULL,
 NO_D_ID        SMALLINT    NOT NULL,
 NO_W_ID         INTEGER     NOT NULL
)
IN ts_N_009
INDEX IN ts_N_009
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 40897 ENDING AT 46008,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 1900 ENDING AT 3740
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE10.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE10;
CREATE TABLE ORDER_LINE10
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_010
INDEX IN ts_OL_010
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 23005 ENDING AT 25560,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE11.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE11;
CREATE TABLE ORDER_LINE11
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
```

```
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_011
       INDEX IN ts_OL_011
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 25561 ENDING AT 28116,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE12.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE12;
CREATE TABLE ORDER_LINE12
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_012
       INDEX IN ts_OL_012
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 28117 ENDING AT 30672,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE13.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE13;
CREATE TABLE ORDER_LINE13
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_013
       INDEX IN ts_OL_013
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 30673 ENDING AT 33228,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE14.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE14;
CREATE TABLE ORDER_LINE14
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_014
       INDEX IN ts_OL_014
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 33229 ENDING AT 35784,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE15.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE15;
CREATE TABLE ORDER_LINE15
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_015
       INDEX IN ts_OL_015
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 35785 ENDING AT 38340,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE16.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE16;
CREATE TABLE ORDER_LINE16
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_016
       INDEX IN ts_OL_016
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 38341 ENDING AT 40896,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE17.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE17;
CREATE TABLE ORDER_LINE17
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_017
       INDEX IN ts_OL_017
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 40897 ENDING AT 43452,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE18.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE18;
CREATE TABLE ORDER_LINE18
```

```
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_018
INDEX IN ts_OL_018
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 43453 ENDING AT 46008,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE19.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE19;
CREATE TABLE ORDER_LINE19
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_019
INDEX IN ts_OL_019
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 46009 ENDING AT 48564,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE1.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE1;
CREATE TABLE ORDER_LINE1
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_001
INDEX IN ts_OL_001
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 1 ENDING AT 2556,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE20.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE20;
CREATE TABLE ORDER_LINE20
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_020
INDEX IN ts_OL_020
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 48565 ENDING AT 51120,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE21.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE21;
CREATE TABLE ORDER_LINE21
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_021
INDEX IN ts_OL_021
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 51121 ENDING AT 53676,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE22.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE22;
CREATE TABLE ORDER_LINE22
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_022
INDEX IN ts_OL_022
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 53677 ENDING AT 56232,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE23.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE23;
CREATE TABLE ORDER_LINE23
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_023
INDEX IN ts_OL_023
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 56233 ENDING AT 58788,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE24.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE24;
CREATE TABLE ORDER_LINE24
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_024
        INDEX IN ts_OL_024
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 58789 ENDING AT 61344,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE25.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE25;
CREATE TABLE ORDER_LINE25
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_025
        INDEX IN ts_OL_025
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 61345 ENDING AT 63900,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE26.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE26;
CREATE TABLE ORDER_LINE26
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_026
        INDEX IN ts_OL_026
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 63901 ENDING AT 66456,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE27.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE27;
CREATE TABLE ORDER_LINE27
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_027
        INDEX IN ts_OL_027
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 66457 ENDING AT 69012,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE28.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE28;
CREATE TABLE ORDER_LINE28
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_028
        INDEX IN ts_OL_028
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 69013 ENDING AT 71568,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE29.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE29;
CREATE TABLE ORDER_LINE29
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_029
        INDEX IN ts_OL_029
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 71569 ENDING AT 74124,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE2.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE2;
CREATE TABLE ORDER_LINE2
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_002
        INDEX IN ts_OL_002
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 2557 ENDING AT 5112,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
```

```
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE30.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE30;
CREATE TABLE ORDER_LINE30
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_030
        INDEX IN ts_OL_030
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 74125 ENDING AT 76680,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE31.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE31;
CREATE TABLE ORDER_LINE31
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_031
        INDEX IN ts_OL_031
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 76681 ENDING AT 79236,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE32.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE32;
CREATE TABLE ORDER_LINE32
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_032
        INDEX IN ts_OL_032
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 79237 ENDING AT 81792,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE33.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE33;
CREATE TABLE ORDER_LINE33
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_033
        INDEX IN ts_OL_033
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 81793 ENDING AT 84348,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE34.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE34;
CREATE TABLE ORDER_LINE34
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_034
        INDEX IN ts_OL_034
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 84349 ENDING AT 86904,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE35.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE35;
CREATE TABLE ORDER_LINE35
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_035
        INDEX IN ts_OL_035
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 86905 ENDING AT 89460,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE36.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE36;
CREATE TABLE ORDER_LINE36
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_036
        INDEX IN ts_OL_036
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 89461 ENDING AT 92016,
```

```
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE37.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE37;
CREATE TABLE ORDER_LINE37
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_037
       INDEX IN ts_OL_037
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 92017 ENDING AT 94572,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE38.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE38;
CREATE TABLE ORDER_LINE38
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_038
       INDEX IN ts_OL_038
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 94573 ENDING AT 97128,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE39.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE39;
CREATE TABLE ORDER_LINE39
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_039
       INDEX IN ts_OL_039
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 97129 ENDING AT 99684,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE3.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE3;
CREATE TABLE ORDER_LINE3
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_003
       INDEX IN ts_OL_003
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 5113 ENDING AT 7668,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE40.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE40;
CREATE TABLE ORDER_LINE40
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_040
       INDEX IN ts_OL_040
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 99685 ENDING AT 102240,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE41.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE41;
CREATE TABLE ORDER_LINE41
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_041
       INDEX IN ts_OL_041
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 102241 ENDING AT 104796,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE42.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE42;
CREATE TABLE ORDER_LINE42
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
```

```
        IN ts_OL_042
        INDEX IN ts_OL_042
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 104797 ENDING AT 107352,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE43.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE43;
CREATE TABLE ORDER_LINE43
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_043
        INDEX IN ts_OL_043
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 107353 ENDING AT 109908,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE44.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE44;
CREATE TABLE ORDER_LINE44
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_044
        INDEX IN ts_OL_044
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 109909 ENDING AT 112464,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE45.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE45;
CREATE TABLE ORDER_LINE45
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_045
        INDEX IN ts_OL_045
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 112465 ENDING AT 115020,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE46.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE46;
CREATE TABLE ORDER_LINE46
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_046
        INDEX IN ts_OL_046
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 115021 ENDING AT 117576,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE47.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE47;
CREATE TABLE ORDER_LINE47
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_047
        INDEX IN ts_OL_047
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 117577 ENDING AT 120132,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE48.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE48;
CREATE TABLE ORDER_LINE48
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_048
        INDEX IN ts_OL_048
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 120133 ENDING AT 122688,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE49.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE49;
CREATE TABLE ORDER_LINE49
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
```

```
        OL_D_ID         SMALLINT     NOT NULL,
        OL_W_ID         INTEGER      NOT NULL,
        OL_NUMBER       SMALLINT     NOT NULL
        )
        IN ts_OL_049
        INDEX IN ts_OL_049
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 122689 ENDING AT 125244,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE4.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE4;
CREATE TABLE ORDER_LINE4
        (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_004
        INDEX IN ts_OL_004
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 7669 ENDING AT 10224,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE50.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE50;
CREATE TABLE ORDER_LINE50
        (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_050
        INDEX IN ts_OL_050
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 125245 ENDING AT 127800,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE51.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE51;
CREATE TABLE ORDER_LINE51
        (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_051
        INDEX IN ts_OL_051
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 127801 ENDING AT 130356,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE52.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE52;
CREATE TABLE ORDER_LINE52
        (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_052
        INDEX IN ts_OL_052
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 130357 ENDING AT 132912,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE53.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE53;
CREATE TABLE ORDER_LINE53
        (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_053
        INDEX IN ts_OL_053
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 132913 ENDING AT 135468,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE54.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE54;
CREATE TABLE ORDER_LINE54
        (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_054
        INDEX IN ts_OL_054
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 135469 ENDING AT 138024,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE55.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE55;
CREATE TABLE ORDER_LINE55
        (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
```

```
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_055
       INDEX IN ts_OL_055
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 138025 ENDING AT 140580,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE56.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE56;
CREATE TABLE ORDER_LINE56
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_056
       INDEX IN ts_OL_056
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 140581 ENDING AT 143136,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE57.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE57;
CREATE TABLE ORDER_LINE57
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_057
       INDEX IN ts_OL_057
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 143137 ENDING AT 145692,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE58.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE58;
CREATE TABLE ORDER_LINE58
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_058
       INDEX IN ts_OL_058
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 145693 ENDING AT 148248,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE59.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE59;
CREATE TABLE ORDER_LINE59
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_059
       INDEX IN ts_OL_059
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 148249 ENDING AT 150804,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE5.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE5;
CREATE TABLE ORDER_LINE5
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_005
       INDEX IN ts_OL_005
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 10225 ENDING AT 12780,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE60.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE60;
CREATE TABLE ORDER_LINE60
       (
        OL_DELIVERY_D    TIMESTAMP    NOT NULL,
        OL_AMOUNT        DECIMAL(6,2) NOT NULL,
        OL_I_ID          INTEGER      NOT NULL,
        OL_SUPPLY_W_ID   INTEGER      NOT NULL,
        OL_QUANTITY      SMALLINT     NOT NULL,
        OL_DIST_INFO     CHAR(24)     NOT NULL,
        OL_O_ID          INTEGER      NOT NULL,
        OL_D_ID          SMALLINT     NOT NULL,
        OL_W_ID          INTEGER      NOT NULL,
        OL_NUMBER        SMALLINT     NOT NULL
       )
       IN ts_OL_060
       INDEX IN ts_OL_060
       ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 150805 ENDING AT 153360,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3740,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE61.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE61;
CREATE TABLE ORDER_LINE61
```

```
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_061
INDEX IN ts_OL_061
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 153361 ENDING AT 155916,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE62.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE62;
CREATE TABLE ORDER_LINE62
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_062
INDEX IN ts_OL_062
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 155917 ENDING AT 158472,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE63.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE63;
CREATE TABLE ORDER_LINE63
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_063
INDEX IN ts_OL_063
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 158473 ENDING AT 161028,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE64.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE64;
CREATE TABLE ORDER_LINE64
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_064
INDEX IN ts_OL_064
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 161029 ENDING AT 163584,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE65.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE65;
CREATE TABLE ORDER_LINE65
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_065
INDEX IN ts_OL_065
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 163585 ENDING AT 166140,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE66.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE66;
CREATE TABLE ORDER_LINE66
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_066
INDEX IN ts_OL_066
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 166141 ENDING AT 168696,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE67.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE67;
CREATE TABLE ORDER_LINE67
(
 OL_DELIVERY_D    TIMESTAMP    NOT NULL,
 OL_AMOUNT        DECIMAL(6,2) NOT NULL,
 OL_I_ID          INTEGER      NOT NULL,
 OL_SUPPLY_W_ID   INTEGER      NOT NULL,
 OL_QUANTITY      SMALLINT     NOT NULL,
 OL_DIST_INFO     CHAR(24)     NOT NULL,
 OL_O_ID          INTEGER      NOT NULL,
 OL_D_ID          SMALLINT     NOT NULL,
 OL_W_ID          INTEGER      NOT NULL,
 OL_NUMBER        SMALLINT     NOT NULL
)
IN ts_OL_067
INDEX IN ts_OL_067
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 168697 ENDING AT 171252,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3740,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE68.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE68;
CREATE TABLE ORDER_LINE68
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_068
        INDEX IN ts_OL_068
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 171253 ENDING AT 173808,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE69.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE69;
CREATE TABLE ORDER_LINE69
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_069
        INDEX IN ts_OL_069
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 173809 ENDING AT 176364,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE6.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE6;
CREATE TABLE ORDER_LINE6
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_006
        INDEX IN ts_OL_006
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 12781 ENDING AT 15336,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE70.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE70;
CREATE TABLE ORDER_LINE70
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_070
        INDEX IN ts_OL_070
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 176365 ENDING AT 178920,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE71.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE71;
CREATE TABLE ORDER_LINE71
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_071
        INDEX IN ts_OL_071
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 178921 ENDING AT 181476,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE72.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE72;
CREATE TABLE ORDER_LINE72
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_072
        INDEX IN ts_OL_072
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 181477 ENDING AT 184032,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE7.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE7;
CREATE TABLE ORDER_LINE7
        (
         OL_DELIVERY_D    TIMESTAMP    NOT NULL,
         OL_AMOUNT        DECIMAL(6,2) NOT NULL,
         OL_I_ID          INTEGER      NOT NULL,
         OL_SUPPLY_W_ID   INTEGER      NOT NULL,
         OL_QUANTITY      SMALLINT     NOT NULL,
         OL_DIST_INFO     CHAR(24)     NOT NULL,
         OL_O_ID          INTEGER      NOT NULL,
         OL_D_ID          SMALLINT     NOT NULL,
         OL_W_ID          INTEGER      NOT NULL,
         OL_NUMBER        SMALLINT     NOT NULL
        )
        IN ts_OL_007
        INDEX IN ts_OL_007
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 15337 ENDING AT 17892,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3740,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
```

```
            ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE8.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE8;
CREATE TABLE ORDER_LINE8
            (
             OL_DELIVERY_D    TIMESTAMP    NOT NULL,
             OL_AMOUNT        DECIMAL(6,2) NOT NULL,
             OL_I_ID          INTEGER      NOT NULL,
             OL_SUPPLY_W_ID   INTEGER      NOT NULL,
             OL_QUANTITY      SMALLINT     NOT NULL,
             OL_DIST_INFO     CHAR(24)     NOT NULL,
             OL_O_ID          INTEGER      NOT NULL,
             OL_D_ID          SMALLINT     NOT NULL,
             OL_W_ID          INTEGER      NOT NULL,
             OL_NUMBER        SMALLINT     NOT NULL
            )
            IN ts_OL_008
            INDEX IN ts_OL_008
            ORGANIZE BY KEY SEQUENCE (
             OL_W_ID STARTING FROM 17893 ENDING AT 20448,
             OL_D_ID STARTING FROM 1 ENDING AT 10,
             OL_O_ID STARTING FROM 1 ENDING AT 3740,
             OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDER_LINE9.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE9;
CREATE TABLE ORDER_LINE9
            (
             OL_DELIVERY_D    TIMESTAMP    NOT NULL,
             OL_AMOUNT        DECIMAL(6,2) NOT NULL,
             OL_I_ID          INTEGER      NOT NULL,
             OL_SUPPLY_W_ID   INTEGER      NOT NULL,
             OL_QUANTITY      SMALLINT     NOT NULL,
             OL_DIST_INFO     CHAR(24)     NOT NULL,
             OL_O_ID          INTEGER      NOT NULL,
             OL_D_ID          SMALLINT     NOT NULL,
             OL_W_ID          INTEGER      NOT NULL,
             OL_NUMBER        SMALLINT     NOT NULL
            )
            IN ts_OL_009
            INDEX IN ts_OL_009
            ORGANIZE BY KEY SEQUENCE (
             OL_W_ID STARTING FROM 20449 ENDING AT 23004,
             OL_D_ID STARTING FROM 1 ENDING AT 10,
             OL_O_ID STARTING FROM 1 ENDING AT 3740,
             OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS10.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS10;
CREATE TABLE ORDERS10
            (
             O_C_ID          INTEGER     NOT NULL,
             O_ENTRY_D       TIMESTAMP   NOT NULL,
             O_CARRIER_ID    SMALLINT    NOT NULL,
             O_OL_CNT        SMALLINT    NOT NULL,
             O_ALL_LOCAL     SMALLINT    NOT NULL,
             O_ID            INTEGER     NOT NULL,
             O_W_ID          INTEGER     NOT NULL,
             O_D_ID          SMALLINT    NOT NULL
            )
            IN ts_O_010
            INDEX IN is_O2_010
            ORGANIZE BY KEY SEQUENCE (
             O_ID STARTING FROM 1 ENDING AT 3740,
             O_W_ID STARTING FROM 46009 ENDING AT 51120,
             O_D_ID STARTING FROM 1 ENDING AT 10
            )
            ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS11.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS11;
CREATE TABLE ORDERS11
            (
             O_C_ID          INTEGER     NOT NULL,
             O_ENTRY_D       TIMESTAMP   NOT NULL,
             O_CARRIER_ID    SMALLINT    NOT NULL,
             O_OL_CNT        SMALLINT    NOT NULL,
             O_ALL_LOCAL     SMALLINT    NOT NULL,
             O_ID            INTEGER     NOT NULL,
             O_W_ID          INTEGER     NOT NULL,
             O_D_ID          SMALLINT    NOT NULL
            )
            IN ts_O_011
            INDEX IN is_O2_011
            ORGANIZE BY KEY SEQUENCE (
             O_ID STARTING FROM 1 ENDING AT 3740,
             O_W_ID STARTING FROM 51121 ENDING AT 56232,
             O_D_ID STARTING FROM 1 ENDING AT 10
            )
            ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS12.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS12;
CREATE TABLE ORDERS12
            (
             O_C_ID          INTEGER     NOT NULL,
             O_ENTRY_D       TIMESTAMP   NOT NULL,
             O_CARRIER_ID    SMALLINT    NOT NULL,
             O_OL_CNT        SMALLINT    NOT NULL,
             O_ALL_LOCAL     SMALLINT    NOT NULL,
             O_ID            INTEGER     NOT NULL,
             O_W_ID          INTEGER     NOT NULL,
             O_D_ID          SMALLINT    NOT NULL
            )
            IN ts_O_012
            INDEX IN is_O2_012
            ORGANIZE BY KEY SEQUENCE (
             O_ID STARTING FROM 1 ENDING AT 3740,
             O_W_ID STARTING FROM 56233 ENDING AT 61344,
             O_D_ID STARTING FROM 1 ENDING AT 10
            )
            ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS13.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS13;
CREATE TABLE ORDERS13
            (
             O_C_ID          INTEGER     NOT NULL,
             O_ENTRY_D       TIMESTAMP   NOT NULL,
             O_CARRIER_ID    SMALLINT    NOT NULL,
             O_OL_CNT        SMALLINT    NOT NULL,
             O_ALL_LOCAL     SMALLINT    NOT NULL,
             O_ID            INTEGER     NOT NULL,
             O_W_ID          INTEGER     NOT NULL,
             O_D_ID          SMALLINT    NOT NULL
            )
            IN ts_O_013
            INDEX IN is_O2_013
            ORGANIZE BY KEY SEQUENCE (
             O_ID STARTING FROM 1 ENDING AT 3740,
             O_W_ID STARTING FROM 61345 ENDING AT 66456,
             O_D_ID STARTING FROM 1 ENDING AT 10
            )
            ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS14.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS14;
CREATE TABLE ORDERS14
            (
             O_C_ID          INTEGER     NOT NULL,
             O_ENTRY_D       TIMESTAMP   NOT NULL,
             O_CARRIER_ID    SMALLINT    NOT NULL,
             O_OL_CNT        SMALLINT    NOT NULL,
             O_ALL_LOCAL     SMALLINT    NOT NULL,
             O_ID            INTEGER     NOT NULL,
             O_W_ID          INTEGER     NOT NULL,
             O_D_ID          SMALLINT    NOT NULL
            )
            IN ts_O_014
            INDEX IN is_O2_014
            ORGANIZE BY KEY SEQUENCE (
             O_ID STARTING FROM 1 ENDING AT 3740,
             O_W_ID STARTING FROM 66457 ENDING AT 71568,
             O_D_ID STARTING FROM 1 ENDING AT 10
            )
            ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS15.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS15;
CREATE TABLE ORDERS15
            (
             O_C_ID          INTEGER     NOT NULL,
             O_ENTRY_D       TIMESTAMP   NOT NULL,
```

```
 O_CARRIER_ID    SMALLINT    NOT NULL,
 O_OL_CNT        SMALLINT    NOT NULL,
 O_ALL_LOCAL     SMALLINT    NOT NULL,
 O_ID            INTEGER     NOT NULL,
 O_W_ID          INTEGER     NOT NULL,
 O_D_ID          SMALLINT    NOT NULL
)
IN ts_O_015
INDEX IN is_O2_015
ORGANIZE BY KEY SEQUENCE (
 O_ID STARTING FROM 1 ENDING AT 3740,
 O_W_ID STARTING FROM 71569 ENDING AT 76680,
 O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS16.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS16;
CREATE TABLE ORDERS16
(
 O_C_ID          INTEGER     NOT NULL,
 O_ENTRY_D       TIMESTAMP   NOT NULL,
 O_CARRIER_ID    SMALLINT    NOT NULL,
 O_OL_CNT        SMALLINT    NOT NULL,
 O_ALL_LOCAL     SMALLINT    NOT NULL,
 O_ID            INTEGER     NOT NULL,
 O_W_ID          INTEGER     NOT NULL,
 O_D_ID          SMALLINT    NOT NULL
)
IN ts_O_016
INDEX IN is_O2_016
ORGANIZE BY KEY SEQUENCE (
 O_ID STARTING FROM 1 ENDING AT 3740,
 O_W_ID STARTING FROM 76681 ENDING AT 81792,
 O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS17.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS17;
CREATE TABLE ORDERS17
(
 O_C_ID          INTEGER     NOT NULL,
 O_ENTRY_D       TIMESTAMP   NOT NULL,
 O_CARRIER_ID    SMALLINT    NOT NULL,
 O_OL_CNT        SMALLINT    NOT NULL,
 O_ALL_LOCAL     SMALLINT    NOT NULL,
 O_ID            INTEGER     NOT NULL,
 O_W_ID          INTEGER     NOT NULL,
 O_D_ID          SMALLINT    NOT NULL
)
IN ts_O_017
INDEX IN is_O2_017
ORGANIZE BY KEY SEQUENCE (
 O_ID STARTING FROM 1 ENDING AT 3740,
 O_W_ID STARTING FROM 81793 ENDING AT 86904,
 O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS18.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS18;
CREATE TABLE ORDERS18
(
 O_C_ID          INTEGER     NOT NULL,
 O_ENTRY_D       TIMESTAMP   NOT NULL,
 O_CARRIER_ID    SMALLINT    NOT NULL,
 O_OL_CNT        SMALLINT    NOT NULL,
 O_ALL_LOCAL     SMALLINT    NOT NULL,
 O_ID            INTEGER     NOT NULL,
 O_W_ID          INTEGER     NOT NULL,
 O_D_ID          SMALLINT    NOT NULL
)
IN ts_O_018
INDEX IN is_O2_018
ORGANIZE BY KEY SEQUENCE (
 O_ID STARTING FROM 1 ENDING AT 3740,
 O_W_ID STARTING FROM 86905 ENDING AT 92016,
 O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS19.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS19;
CREATE TABLE ORDERS19
(
 O_C_ID          INTEGER     NOT NULL,
 O_ENTRY_D       TIMESTAMP   NOT NULL,
 O_CARRIER_ID    SMALLINT    NOT NULL,
 O_OL_CNT        SMALLINT    NOT NULL,
 O_ALL_LOCAL     SMALLINT    NOT NULL,
 O_ID            INTEGER     NOT NULL,
 O_W_ID          INTEGER     NOT NULL,
 O_D_ID          SMALLINT    NOT NULL
)
IN ts_O_019
INDEX IN is_O2_019
ORGANIZE BY KEY SEQUENCE (
 O_ID STARTING FROM 1 ENDING AT 3740,
 O_W_ID STARTING FROM 92017 ENDING AT 97128,
 O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS1.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS1;
CREATE TABLE ORDERS1
(
 O_C_ID          INTEGER     NOT NULL,
 O_ENTRY_D       TIMESTAMP   NOT NULL,
 O_CARRIER_ID    SMALLINT    NOT NULL,
 O_OL_CNT        SMALLINT    NOT NULL,
 O_ALL_LOCAL     SMALLINT    NOT NULL,
 O_ID            INTEGER     NOT NULL,
 O_W_ID          INTEGER     NOT NULL,
 O_D_ID          SMALLINT    NOT NULL
)
IN ts_O_001
INDEX IN is_O2_001
ORGANIZE BY KEY SEQUENCE (
 O_ID STARTING FROM 1 ENDING AT 3740,
 O_W_ID STARTING FROM 1 ENDING AT 5112,
 O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS20.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS20;
CREATE TABLE ORDERS20
(
 O_C_ID          INTEGER     NOT NULL,
 O_ENTRY_D       TIMESTAMP   NOT NULL,
 O_CARRIER_ID    SMALLINT    NOT NULL,
 O_OL_CNT        SMALLINT    NOT NULL,
 O_ALL_LOCAL     SMALLINT    NOT NULL,
 O_ID            INTEGER     NOT NULL,
 O_W_ID          INTEGER     NOT NULL,
 O_D_ID          SMALLINT    NOT NULL
)
IN ts_O_020
INDEX IN is_O2_020
ORGANIZE BY KEY SEQUENCE (
 O_ID STARTING FROM 1 ENDING AT 3740,
 O_W_ID STARTING FROM 97129 ENDING AT 102240,
 O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS21.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS21;
CREATE TABLE ORDERS21
(
 O_C_ID          INTEGER     NOT NULL,
 O_ENTRY_D       TIMESTAMP   NOT NULL,
 O_CARRIER_ID    SMALLINT    NOT NULL,
 O_OL_CNT        SMALLINT    NOT NULL,
 O_ALL_LOCAL     SMALLINT    NOT NULL,
 O_ID            INTEGER     NOT NULL,
 O_W_ID          INTEGER     NOT NULL,
 O_D_ID          SMALLINT    NOT NULL
)
IN ts_O_021
INDEX IN is_O2_021
ORGANIZE BY KEY SEQUENCE (
 O_ID STARTING FROM 1 ENDING AT 3740,
 O_W_ID STARTING FROM 102241 ENDING AT 107352,
 O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS22.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS22;
CREATE TABLE ORDERS22
        (
         O_C_ID        INTEGER     NOT NULL,
         O_ENTRY_D      TIMESTAMP   NOT NULL,
         O_CARRIER_ID    SMALLINT    NOT NULL,
         O_OL_CNT       SMALLINT    NOT NULL,
         O_ALL_LOCAL     SMALLINT    NOT NULL,
         O_ID          INTEGER     NOT NULL,
         O_W_ID         INTEGER     NOT NULL,
         O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_O_022
        INDEX IN is_O2_022
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3740,
         O_W_ID STARTING FROM 107353 ENDING AT 112464,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS23.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS23;
CREATE TABLE ORDERS23
        (
         O_C_ID        INTEGER     NOT NULL,
         O_ENTRY_D      TIMESTAMP   NOT NULL,
         O_CARRIER_ID    SMALLINT    NOT NULL,
         O_OL_CNT       SMALLINT    NOT NULL,
         O_ALL_LOCAL     SMALLINT    NOT NULL,
         O_ID          INTEGER     NOT NULL,
         O_W_ID         INTEGER     NOT NULL,
         O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_O_023
        INDEX IN is_O2_023
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3740,
         O_W_ID STARTING FROM 112465 ENDING AT 117576,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS24.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS24;
CREATE TABLE ORDERS24
        (
         O_C_ID        INTEGER     NOT NULL,
         O_ENTRY_D      TIMESTAMP   NOT NULL,
         O_CARRIER_ID    SMALLINT    NOT NULL,
         O_OL_CNT       SMALLINT    NOT NULL,
         O_ALL_LOCAL     SMALLINT    NOT NULL,
         O_ID          INTEGER     NOT NULL,
         O_W_ID         INTEGER     NOT NULL,
         O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_O_024
        INDEX IN is_O2_024
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3740,
         O_W_ID STARTING FROM 117577 ENDING AT 122688,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS25.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS25;
CREATE TABLE ORDERS25
        (
         O_C_ID        INTEGER     NOT NULL,
         O_ENTRY_D      TIMESTAMP   NOT NULL,
         O_CARRIER_ID    SMALLINT    NOT NULL,
         O_OL_CNT       SMALLINT    NOT NULL,
         O_ALL_LOCAL     SMALLINT    NOT NULL,
         O_ID          INTEGER     NOT NULL,
         O_W_ID         INTEGER     NOT NULL,
         O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_O_025
        INDEX IN is_O2_025
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3740,
         O_W_ID STARTING FROM 122689 ENDING AT 127800,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS26.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS26;
CREATE TABLE ORDERS26
        (
         O_C_ID        INTEGER     NOT NULL,
         O_ENTRY_D      TIMESTAMP   NOT NULL,
         O_CARRIER_ID    SMALLINT    NOT NULL,
         O_OL_CNT       SMALLINT    NOT NULL,
         O_ALL_LOCAL     SMALLINT    NOT NULL,
         O_ID          INTEGER     NOT NULL,
         O_W_ID         INTEGER     NOT NULL,
         O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_O_026
        INDEX IN is_O2_026
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3740,
         O_W_ID STARTING FROM 127801 ENDING AT 132912,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS27.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS27;
CREATE TABLE ORDERS27
        (
         O_C_ID        INTEGER     NOT NULL,
         O_ENTRY_D      TIMESTAMP   NOT NULL,
         O_CARRIER_ID    SMALLINT    NOT NULL,
         O_OL_CNT       SMALLINT    NOT NULL,
         O_ALL_LOCAL     SMALLINT    NOT NULL,
         O_ID          INTEGER     NOT NULL,
         O_W_ID         INTEGER     NOT NULL,
         O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_O_027
        INDEX IN is_O2_027
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3740,
         O_W_ID STARTING FROM 132913 ENDING AT 138024,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS28.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS28;
CREATE TABLE ORDERS28
        (
         O_C_ID        INTEGER     NOT NULL,
         O_ENTRY_D      TIMESTAMP   NOT NULL,
         O_CARRIER_ID    SMALLINT    NOT NULL,
         O_OL_CNT       SMALLINT    NOT NULL,
         O_ALL_LOCAL     SMALLINT    NOT NULL,
         O_ID          INTEGER     NOT NULL,
         O_W_ID         INTEGER     NOT NULL,
         O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_O_028
        INDEX IN is_O2_028
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3740,
         O_W_ID STARTING FROM 138025 ENDING AT 143136,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS29.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS29;
CREATE TABLE ORDERS29
        (
         O_C_ID        INTEGER     NOT NULL,
         O_ENTRY_D      TIMESTAMP   NOT NULL,
         O_CARRIER_ID    SMALLINT    NOT NULL,
         O_OL_CNT       SMALLINT    NOT NULL,
         O_ALL_LOCAL     SMALLINT    NOT NULL,
         O_ID          INTEGER     NOT NULL,
         O_W_ID         INTEGER     NOT NULL,
         O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_O_029
        INDEX IN is_O2_029
        ORGANIZE BY KEY SEQUENCE (
```

```
            O_ID STARTING FROM 1 ENDING AT 3740,
            O_W_ID STARTING FROM 143137 ENDING AT 148248,
            O_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS2.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS2;
CREATE TABLE ORDERS2
           (
            O_C_ID        INTEGER     NOT NULL,
            O_ENTRY_D     TIMESTAMP   NOT NULL,
            O_CARRIER_ID  SMALLINT    NOT NULL,
            O_OL_CNT      SMALLINT    NOT NULL,
            O_ALL_LOCAL   SMALLINT    NOT NULL,
            O_ID          INTEGER     NOT NULL,
            O_W_ID        INTEGER     NOT NULL,
            O_D_ID        SMALLINT    NOT NULL
           )
           IN ts_O_002
           INDEX IN is_O2_002
           ORGANIZE BY KEY SEQUENCE (
            O_ID STARTING FROM 1 ENDING AT 3740,
            O_W_ID STARTING FROM 5113 ENDING AT 10224,
            O_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS30.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS30;
CREATE TABLE ORDERS30
           (
            O_C_ID        INTEGER     NOT NULL,
            O_ENTRY_D     TIMESTAMP   NOT NULL,
            O_CARRIER_ID  SMALLINT    NOT NULL,
            O_OL_CNT      SMALLINT    NOT NULL,
            O_ALL_LOCAL   SMALLINT    NOT NULL,
            O_ID          INTEGER     NOT NULL,
            O_W_ID        INTEGER     NOT NULL,
            O_D_ID        SMALLINT    NOT NULL
           )
           IN ts_O_030
           INDEX IN is_O2_030
           ORGANIZE BY KEY SEQUENCE (
            O_ID STARTING FROM 1 ENDING AT 3740,
            O_W_ID STARTING FROM 148249 ENDING AT 153360,
            O_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS31.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS31;
CREATE TABLE ORDERS31
           (
            O_C_ID        INTEGER     NOT NULL,
            O_ENTRY_D     TIMESTAMP   NOT NULL,
            O_CARRIER_ID  SMALLINT    NOT NULL,
            O_OL_CNT      SMALLINT    NOT NULL,
            O_ALL_LOCAL   SMALLINT    NOT NULL,
            O_ID          INTEGER     NOT NULL,
            O_W_ID        INTEGER     NOT NULL,
            O_D_ID        SMALLINT    NOT NULL
           )
           IN ts_O_031
           INDEX IN is_O2_031
           ORGANIZE BY KEY SEQUENCE (
            O_ID STARTING FROM 1 ENDING AT 3740,
            O_W_ID STARTING FROM 153361 ENDING AT 158472,
            O_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS32.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS32;
CREATE TABLE ORDERS32
           (
            O_C_ID        INTEGER     NOT NULL,
            O_ENTRY_D     TIMESTAMP   NOT NULL,
            O_CARRIER_ID  SMALLINT    NOT NULL,
            O_OL_CNT      SMALLINT    NOT NULL,
            O_ALL_LOCAL   SMALLINT    NOT NULL,
            O_ID          INTEGER     NOT NULL,
            O_W_ID        INTEGER     NOT NULL,
            O_D_ID        SMALLINT    NOT NULL
           )
           IN ts_O_032
           INDEX IN is_O2_032
           ORGANIZE BY KEY SEQUENCE (
            O_ID STARTING FROM 1 ENDING AT 3740,
            O_W_ID STARTING FROM 158473 ENDING AT 163584,
            O_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS33.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS33;
CREATE TABLE ORDERS33
           (
            O_C_ID        INTEGER     NOT NULL,
            O_ENTRY_D     TIMESTAMP   NOT NULL,
            O_CARRIER_ID  SMALLINT    NOT NULL,
            O_OL_CNT      SMALLINT    NOT NULL,
            O_ALL_LOCAL   SMALLINT    NOT NULL,
            O_ID          INTEGER     NOT NULL,
            O_W_ID        INTEGER     NOT NULL,
            O_D_ID        SMALLINT    NOT NULL
           )
           IN ts_O_033
           INDEX IN is_O2_033
           ORGANIZE BY KEY SEQUENCE (
            O_ID STARTING FROM 1 ENDING AT 3740,
            O_W_ID STARTING FROM 163585 ENDING AT 168696,
            O_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS34.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS34;
CREATE TABLE ORDERS34
           (
            O_C_ID        INTEGER     NOT NULL,
            O_ENTRY_D     TIMESTAMP   NOT NULL,
            O_CARRIER_ID  SMALLINT    NOT NULL,
            O_OL_CNT      SMALLINT    NOT NULL,
            O_ALL_LOCAL   SMALLINT    NOT NULL,
            O_ID          INTEGER     NOT NULL,
            O_W_ID        INTEGER     NOT NULL,
            O_D_ID        SMALLINT    NOT NULL
           )
           IN ts_O_034
           INDEX IN is_O2_034
           ORGANIZE BY KEY SEQUENCE (
            O_ID STARTING FROM 1 ENDING AT 3740,
            O_W_ID STARTING FROM 168697 ENDING AT 173808,
            O_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS35.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS35;
CREATE TABLE ORDERS35
           (
            O_C_ID        INTEGER     NOT NULL,
            O_ENTRY_D     TIMESTAMP   NOT NULL,
            O_CARRIER_ID  SMALLINT    NOT NULL,
            O_OL_CNT      SMALLINT    NOT NULL,
            O_ALL_LOCAL   SMALLINT    NOT NULL,
            O_ID          INTEGER     NOT NULL,
            O_W_ID        INTEGER     NOT NULL,
            O_D_ID        SMALLINT    NOT NULL
           )
           IN ts_O_035
           INDEX IN is_O2_035
           ORGANIZE BY KEY SEQUENCE (
            O_ID STARTING FROM 1 ENDING AT 3740,
            O_W_ID STARTING FROM 173809 ENDING AT 178920,
            O_D_ID STARTING FROM 1 ENDING AT 10
           )
           ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS36.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS36;
CREATE TABLE ORDERS36
           (
            O_C_ID        INTEGER     NOT NULL,
            O_ENTRY_D     TIMESTAMP   NOT NULL,
            O_CARRIER_ID  SMALLINT    NOT NULL,
            O_OL_CNT      SMALLINT    NOT NULL,
```

```
        O_ALL_LOCAL     SMALLINT    NOT NULL,
        O_ID            INTEGER     NOT NULL,
        O_W_ID          INTEGER     NOT NULL,
        O_D_ID          SMALLINT    NOT NULL
        )
        IN ts_O_036
        INDEX IN is_O2_036
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3740,
        O_W_ID STARTING FROM 178921 ENDING AT 184032,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS3.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS3;
CREATE TABLE ORDERS3
        (
        O_C_ID          INTEGER     NOT NULL,
        O_ENTRY_D       TIMESTAMP   NOT NULL,
        O_CARRIER_ID    SMALLINT    NOT NULL,
        O_OL_CNT        SMALLINT    NOT NULL,
        O_ALL_LOCAL     SMALLINT    NOT NULL,
        O_ID            INTEGER     NOT NULL,
        O_W_ID          INTEGER     NOT NULL,
        O_D_ID          SMALLINT    NOT NULL
        )
        IN ts_O_003
        INDEX IN is_O2_003
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3740,
        O_W_ID STARTING FROM 10225 ENDING AT 15336,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS4.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS4;
CREATE TABLE ORDERS4
        (
        O_C_ID          INTEGER     NOT NULL,
        O_ENTRY_D       TIMESTAMP   NOT NULL,
        O_CARRIER_ID    SMALLINT    NOT NULL,
        O_OL_CNT        SMALLINT    NOT NULL,
        O_ALL_LOCAL     SMALLINT    NOT NULL,
        O_ID            INTEGER     NOT NULL,
        O_W_ID          INTEGER     NOT NULL,
        O_D_ID          SMALLINT    NOT NULL
        )
        IN ts_O_004
        INDEX IN is_O2_004
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3740,
        O_W_ID STARTING FROM 15337 ENDING AT 20448,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS5.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS5;
CREATE TABLE ORDERS5
        (
        O_C_ID          INTEGER     NOT NULL,
        O_ENTRY_D       TIMESTAMP   NOT NULL,
        O_CARRIER_ID    SMALLINT    NOT NULL,
        O_OL_CNT        SMALLINT    NOT NULL,
        O_ALL_LOCAL     SMALLINT    NOT NULL,
        O_ID            INTEGER     NOT NULL,
        O_W_ID          INTEGER     NOT NULL,
        O_D_ID          SMALLINT    NOT NULL
        )
        IN ts_O_005
        INDEX IN is_O2_005
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3740,
        O_W_ID STARTING FROM 20449 ENDING AT 25560,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS6.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS6;
CREATE TABLE ORDERS6
        (
        O_C_ID          INTEGER     NOT NULL,
        O_ENTRY_D       TIMESTAMP   NOT NULL,
        O_CARRIER_ID    SMALLINT    NOT NULL,
        O_OL_CNT        SMALLINT    NOT NULL,
        O_ALL_LOCAL     SMALLINT    NOT NULL,
        O_ID            INTEGER     NOT NULL,
        O_W_ID          INTEGER     NOT NULL,
        O_D_ID          SMALLINT    NOT NULL
        )
        IN ts_O_006
        INDEX IN is_O2_006
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3740,
        O_W_ID STARTING FROM 25561 ENDING AT 30672,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS7.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS7;
CREATE TABLE ORDERS7
        (
        O_C_ID          INTEGER     NOT NULL,
        O_ENTRY_D       TIMESTAMP   NOT NULL,
        O_CARRIER_ID    SMALLINT    NOT NULL,
        O_OL_CNT        SMALLINT    NOT NULL,
        O_ALL_LOCAL     SMALLINT    NOT NULL,
        O_ID            INTEGER     NOT NULL,
        O_W_ID          INTEGER     NOT NULL,
        O_D_ID          SMALLINT    NOT NULL
        )
        IN ts_O_007
        INDEX IN is_O2_007
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3740,
        O_W_ID STARTING FROM 30673 ENDING AT 35784,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS8.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS8;
CREATE TABLE ORDERS8
        (
        O_C_ID          INTEGER     NOT NULL,
        O_ENTRY_D       TIMESTAMP   NOT NULL,
        O_CARRIER_ID    SMALLINT    NOT NULL,
        O_OL_CNT        SMALLINT    NOT NULL,
        O_ALL_LOCAL     SMALLINT    NOT NULL,
        O_ID            INTEGER     NOT NULL,
        O_W_ID          INTEGER     NOT NULL,
        O_D_ID          SMALLINT    NOT NULL
        )
        IN ts_O_008
        INDEX IN is_O2_008
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3740,
        O_W_ID STARTING FROM 35785 ENDING AT 40896,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_ORDERS9.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS9;
CREATE TABLE ORDERS9
        (
        O_C_ID          INTEGER     NOT NULL,
        O_ENTRY_D       TIMESTAMP   NOT NULL,
        O_CARRIER_ID    SMALLINT    NOT NULL,
        O_OL_CNT        SMALLINT    NOT NULL,
        O_ALL_LOCAL     SMALLINT    NOT NULL,
        O_ID            INTEGER     NOT NULL,
        O_W_ID          INTEGER     NOT NULL,
        O_D_ID          SMALLINT    NOT NULL
        )
        IN ts_O_009
        INDEX IN is_O2_009
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3740,
        O_W_ID STARTING FROM 40897 ENDING AT 46008,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK100.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK100;
CREATE TABLE STOCK100
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_100
        INDEX IN ts_S_100
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 126523 ENDING AT 127800
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK101.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK101;
CREATE TABLE STOCK101
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_101
        INDEX IN ts_S_101
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 127801 ENDING AT 129078
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK102.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK102;
CREATE TABLE STOCK102
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_102
        INDEX IN ts_S_102
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 129079 ENDING AT 130356
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK103.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK103;
CREATE TABLE STOCK103
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_103
        INDEX IN ts_S_103
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 130357 ENDING AT 131634
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK104.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK104;
CREATE TABLE STOCK104
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_104
        INDEX IN ts_S_104
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 131635 ENDING AT 132912
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK105.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK105;
CREATE TABLE STOCK105
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_105
        INDEX IN ts_S_105
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 132913 ENDING AT 134190
        )
        ALLOW OVERFLOW;
```

```
connect reset;
```

CRTB_STOCK106.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK106;
CREATE TABLE STOCK106
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_106
        INDEX IN ts_S_106
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 134191 ENDING AT 135468
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK107.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK107;
CREATE TABLE STOCK107
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_107
        INDEX IN ts_S_107
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 135469 ENDING AT 136746
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK108.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK108;
CREATE TABLE STOCK108
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_108
        INDEX IN ts_S_108
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 136747 ENDING AT 138024
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK109.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK109;
CREATE TABLE STOCK109
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_109
        INDEX IN ts_S_109
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 138025 ENDING AT 139302
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK10.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK10;
CREATE TABLE STOCK10
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_010
        INDEX IN ts_S_010
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 11503 ENDING AT 12780
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK110.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK110;
CREATE TABLE STOCK110
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_110
```

```
        INDEX IN ts_S_110
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 139303 ENDING AT 140580
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK111.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK111;
CREATE TABLE STOCK111
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_111
        INDEX IN ts_S_111
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 140581 ENDING AT 141858
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK112.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK112;
CREATE TABLE STOCK112
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_112
        INDEX IN ts_S_112
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 141859 ENDING AT 143136
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK113.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK113;
CREATE TABLE STOCK113
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_113
        INDEX IN ts_S_113
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 143137 ENDING AT 144414
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK114.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK114;
CREATE TABLE STOCK114
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_114
        INDEX IN ts_S_114
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 144415 ENDING AT 145692
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK115.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK115;
CREATE TABLE STOCK115
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_115
        INDEX IN ts_S_115
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 145693 ENDING AT 146970
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK116.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK116;
CREATE TABLE STOCK116
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
```

```
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_S_116
        INDEX IN ts_S_116
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 146971 ENDING AT 148248
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK117.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK117;
CREATE TABLE STOCK117
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_117
        INDEX IN ts_S_117
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 148249 ENDING AT 149526
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK118.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK118;
CREATE TABLE STOCK118
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_118
        INDEX IN ts_S_118
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 149527 ENDING AT 150804
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK119.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK119;
CREATE TABLE STOCK119
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_119
        INDEX IN ts_S_119
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 150805 ENDING AT 152082
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK11.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK11;
CREATE TABLE STOCK11
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_011
        INDEX IN ts_S_011
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 12781 ENDING AT 14058
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK120.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK120;
CREATE TABLE STOCK120
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_120
        INDEX IN ts_S_120
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 152083 ENDING AT 153360
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK121.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK121;
CREATE TABLE STOCK121
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
```

```
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
       )
       IN ts_S_121
       INDEX IN ts_S_121
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 153361 ENDING AT 154638
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK122.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK122;
CREATE TABLE STOCK122
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_122
       INDEX IN ts_S_122
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 154639 ENDING AT 155916
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK123.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK123;
CREATE TABLE STOCK123
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_123
       INDEX IN ts_S_123
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 155917 ENDING AT 157194
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK124.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK124;
CREATE TABLE STOCK124
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_124
       INDEX IN ts_S_124
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 157195 ENDING AT 158472
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK125.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK125;
CREATE TABLE STOCK125
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_125
       INDEX IN ts_S_125
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 158473 ENDING AT 159750
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK126.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK126;
CREATE TABLE STOCK126
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_126
       INDEX IN ts_S_126
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 159751 ENDING AT 161028
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK127.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK127;
CREATE TABLE STOCK127
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
```

```
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_127
       INDEX IN ts_S_127
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 161029 ENDING AT 162306
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK128.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK128;
CREATE TABLE STOCK128
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_128
       INDEX IN ts_S_128
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 162307 ENDING AT 163584
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK129.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK129;
CREATE TABLE STOCK129
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_129
       INDEX IN ts_S_129
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 163585 ENDING AT 164862
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK12.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK12;
CREATE TABLE STOCK12
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_012
       INDEX IN ts_S_012
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 14059 ENDING AT 15336
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK130.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK130;
CREATE TABLE STOCK130
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_130
       INDEX IN ts_S_130
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 164863 ENDING AT 166140
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK131.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK131;
CREATE TABLE STOCK131
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_131
       INDEX IN ts_S_131
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 166141 ENDING AT 167418
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK132.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK132;
CREATE TABLE STOCK132
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_132
        INDEX IN ts_S_132
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 167419 ENDING AT 168696
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK133.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK133;
CREATE TABLE STOCK133
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_133
        INDEX IN ts_S_133
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 168697 ENDING AT 169974
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK134.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK134;
CREATE TABLE STOCK134
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_134
        INDEX IN ts_S_134
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 169975 ENDING AT 171252
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK135.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK135;
CREATE TABLE STOCK135
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_135
        INDEX IN ts_S_135
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 171253 ENDING AT 172530
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK136.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK136;
CREATE TABLE STOCK136
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_136
        INDEX IN ts_S_136
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 172531 ENDING AT 173808
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK137.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK137;
CREATE TABLE STOCK137
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_137
        INDEX IN ts_S_137
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
```

```
 S_W_ID STARTING FROM 173809 ENDING AT 175086
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK138.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK138;
CREATE TABLE STOCK138
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_138
INDEX IN ts_S_138
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 175087 ENDING AT 176364
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK139.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK139;
CREATE TABLE STOCK139
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_139
INDEX IN ts_S_139
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 176365 ENDING AT 177642
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK13.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK13;
CREATE TABLE STOCK13
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_013
INDEX IN ts_S_013
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 15337 ENDING AT 16614
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK140.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK140;
CREATE TABLE STOCK140
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_140
INDEX IN ts_S_140
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 177643 ENDING AT 178920
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK141.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK141;
CREATE TABLE STOCK141
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_141
INDEX IN ts_S_141
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 178921 ENDING AT 180198
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK142.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK142;
CREATE TABLE STOCK142
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
```

```
 S_W_ID         INTEGER     NOT NULL
)
IN ts_S_142
INDEX IN ts_S_142
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 180199 ENDING AT 181476
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK143.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK143;
CREATE TABLE STOCK143
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_143
INDEX IN ts_S_143
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 181477 ENDING AT 182754
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK144.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK144;
CREATE TABLE STOCK144
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_144
INDEX IN ts_S_144
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 182755 ENDING AT 184032
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK14.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK14;
CREATE TABLE STOCK14
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_014
INDEX IN ts_S_014
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 16615 ENDING AT 17892
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK15.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK15;
CREATE TABLE STOCK15
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_015
INDEX IN ts_S_015
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 17893 ENDING AT 19170
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK16.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK16;
CREATE TABLE STOCK16
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_016
INDEX IN ts_S_016
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 19171 ENDING AT 20448
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK17.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK17;
CREATE TABLE STOCK17
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
```

```
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID        INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_S_017
        INDEX IN ts_S_017
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 20449 ENDING AT 21726
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK18.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK18;
CREATE TABLE STOCK18
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_S_018
        INDEX IN ts_S_018
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 21727 ENDING AT 23004
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK19.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK19;
CREATE TABLE STOCK19
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_S_019
        INDEX IN ts_S_019
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 23005 ENDING AT 24282
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK1.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK1;
CREATE TABLE STOCK1
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_S_001
        INDEX IN ts_S_001
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 1 ENDING AT 1278
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK20.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK20;
CREATE TABLE STOCK20
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_S_020
        INDEX IN ts_S_020
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 24283 ENDING AT 25560
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK21.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK21;
CREATE TABLE STOCK21
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_S_021
        INDEX IN ts_S_021
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 25561 ENDING AT 26838
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK22.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK22;
CREATE TABLE STOCK22
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
```

```
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_022
       INDEX IN ts_S_022
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 26839 ENDING AT 28116
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK23.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK23;
CREATE TABLE STOCK23
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_023
       INDEX IN ts_S_023
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 28117 ENDING AT 29394
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK24.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK24;
CREATE TABLE STOCK24
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_024
       INDEX IN ts_S_024
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 29395 ENDING AT 30672
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK25.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK25;
CREATE TABLE STOCK25
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_025
       INDEX IN ts_S_025
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 30673 ENDING AT 31950
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK26.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK26;
CREATE TABLE STOCK26
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_026
       INDEX IN ts_S_026
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 31951 ENDING AT 33228
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK27.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK27;
CREATE TABLE STOCK27
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_027
       INDEX IN ts_S_027
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 33229 ENDING AT 34506
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK28.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK28;
```

```
CREATE TABLE STOCK28
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_028
        INDEX IN ts_S_028
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 34507 ENDING AT 35784
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK29.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK29;
CREATE TABLE STOCK29
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_029
        INDEX IN ts_S_029
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 35785 ENDING AT 37062
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK2.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK2;
CREATE TABLE STOCK2
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_002
        INDEX IN ts_S_002
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 1279 ENDING AT 2556
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK30.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK30;
CREATE TABLE STOCK30
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_030
        INDEX IN ts_S_030
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 37063 ENDING AT 38340
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK31.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK31;
CREATE TABLE STOCK31
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_031
        INDEX IN ts_S_031
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 38341 ENDING AT 39618
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK32.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK32;
CREATE TABLE STOCK32
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_032
        INDEX IN ts_S_032
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 39619 ENDING AT 40896
        )
        ALLOW OVERFLOW;
```

```
connect reset;
```

CRTB_STOCK33.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK33;
CREATE TABLE STOCK33
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_033
        INDEX IN ts_S_033
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 40897 ENDING AT 42174
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK34.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK34;
CREATE TABLE STOCK34
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_034
        INDEX IN ts_S_034
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 42175 ENDING AT 43452
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK35.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK35;
CREATE TABLE STOCK35
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_035
        INDEX IN ts_S_035
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 43453 ENDING AT 44730
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK36.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK36;
CREATE TABLE STOCK36
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_036
        INDEX IN ts_S_036
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 44731 ENDING AT 46008
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK37.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK37;
CREATE TABLE STOCK37
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_037
        INDEX IN ts_S_037
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 46009 ENDING AT 47286
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK38.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK38;
CREATE TABLE STOCK38
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_038
```

```
        INDEX IN ts_S_038
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 47287 ENDING AT 48564
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK39.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK39;
CREATE TABLE STOCK39
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_039
        INDEX IN ts_S_039
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 48565 ENDING AT 49842
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK3.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK3;
CREATE TABLE STOCK3
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_003
        INDEX IN ts_S_003
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 2557 ENDING AT 3834
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK40.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK40;
CREATE TABLE STOCK40
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_040
        INDEX IN ts_S_040
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 49843 ENDING AT 51120
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK41.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK41;
CREATE TABLE STOCK41
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_041
        INDEX IN ts_S_041
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 51121 ENDING AT 52398
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK42.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK42;
CREATE TABLE STOCK42
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_042
        INDEX IN ts_S_042
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 52399 ENDING AT 53676
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK43.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK43;
CREATE TABLE STOCK43
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
```

```
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_S_043
        INDEX IN ts_S_043
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 53677 ENDING AT 54954
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK44.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK44;
CREATE TABLE STOCK44
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_044
        INDEX IN ts_S_044
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 54955 ENDING AT 56232
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK45.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK45;
CREATE TABLE STOCK45
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_045
        INDEX IN ts_S_045
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 56233 ENDING AT 57510
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK46.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK46;
CREATE TABLE STOCK46
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_046
        INDEX IN ts_S_046
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 57511 ENDING AT 58788
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK47.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK47;
CREATE TABLE STOCK47
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_047
        INDEX IN ts_S_047
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 58789 ENDING AT 60066
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK48.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK48;
CREATE TABLE STOCK48
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_048
        INDEX IN ts_S_048
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 60067 ENDING AT 61344
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK49.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK49;
CREATE TABLE STOCK49
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
```

```
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
       )
       IN ts_S_049
       INDEX IN ts_S_049
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 61345 ENDING AT 62622
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK4.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK4;
CREATE TABLE STOCK4
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_004
       INDEX IN ts_S_004
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 3835 ENDING AT 5112
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK50.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK50;
CREATE TABLE STOCK50
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_050
       INDEX IN ts_S_050
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 62623 ENDING AT 63900
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK51.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK51;
CREATE TABLE STOCK51
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_051
       INDEX IN ts_S_051
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 63901 ENDING AT 65178
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK52.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK52;
CREATE TABLE STOCK52
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_052
       INDEX IN ts_S_052
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 65179 ENDING AT 66456
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK53.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK53;
CREATE TABLE STOCK53
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_053
       INDEX IN ts_S_053
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 66457 ENDING AT 67734
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK54.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK54;
CREATE TABLE STOCK54
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
```

```
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_054
        INDEX IN ts_S_054
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 67735 ENDING AT 69012
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK55.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK55;
CREATE TABLE STOCK55
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_055
        INDEX IN ts_S_055
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 69013 ENDING AT 70290
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK56.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK56;
CREATE TABLE STOCK56
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_056
        INDEX IN ts_S_056
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 70291 ENDING AT 71568
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK57.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK57;
CREATE TABLE STOCK57
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_057
        INDEX IN ts_S_057
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 71569 ENDING AT 72846
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK58.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK58;
CREATE TABLE STOCK58
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_058
        INDEX IN ts_S_058
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 72847 ENDING AT 74124
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK59.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK59;
CREATE TABLE STOCK59
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_059
        INDEX IN ts_S_059
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 74125 ENDING AT 75402
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK5.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK5;
CREATE TABLE STOCK5
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_005
        INDEX IN ts_S_005
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 5113 ENDING AT 6390
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK60.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK60;
CREATE TABLE STOCK60
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_060
        INDEX IN ts_S_060
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 75403 ENDING AT 76680
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK61.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK61;
CREATE TABLE STOCK61
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_061
        INDEX IN ts_S_061
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 76681 ENDING AT 77958
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK62.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK62;
CREATE TABLE STOCK62
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_062
        INDEX IN ts_S_062
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 77959 ENDING AT 79236
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK63.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK63;
CREATE TABLE STOCK63
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_063
        INDEX IN ts_S_063
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 79237 ENDING AT 80514
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK64.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK64;
CREATE TABLE STOCK64
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_064
        INDEX IN ts_S_064
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
```

```
 S_W_ID STARTING FROM 80515 ENDING AT 81792
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK65.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK65;
CREATE TABLE STOCK65
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_065
INDEX IN ts_S_065
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 81793 ENDING AT 83070
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK66.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK66;
CREATE TABLE STOCK66
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_066
INDEX IN ts_S_066
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 83071 ENDING AT 84348
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK67.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK67;
CREATE TABLE STOCK67
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_067
INDEX IN ts_S_067
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 84349 ENDING AT 85626
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK68.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK68;
CREATE TABLE STOCK68
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_068
INDEX IN ts_S_068
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 85627 ENDING AT 86904
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK69.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK69;
CREATE TABLE STOCK69
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_S_069
INDEX IN ts_S_069
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 86905 ENDING AT 88182
)
ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK6.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK6;
CREATE TABLE STOCK6
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
```

```
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_006
       INDEX IN ts_S_006
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 6391 ENDING AT 7668
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK70.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK70;
CREATE TABLE STOCK70
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_070
       INDEX IN ts_S_070
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 88183 ENDING AT 89460
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK71.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK71;
CREATE TABLE STOCK71
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_071
       INDEX IN ts_S_071
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 89461 ENDING AT 90738
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK72.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK72;
CREATE TABLE STOCK72
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_072
       INDEX IN ts_S_072
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 90739 ENDING AT 92016
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK73.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK73;
CREATE TABLE STOCK73
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_073
       INDEX IN ts_S_073
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 92017 ENDING AT 93294
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK74.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK74;
CREATE TABLE STOCK74
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_074
       INDEX IN ts_S_074
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 93295 ENDING AT 94572
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK75.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK75;
CREATE TABLE STOCK75
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
```

```
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID        INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
       )
       IN ts_S_075
       INDEX IN ts_S_075
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 94573 ENDING AT 95850
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK76.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK76;
CREATE TABLE STOCK76
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
       )
       IN ts_S_076
       INDEX IN ts_S_076
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 95851 ENDING AT 97128
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK77.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK77;
CREATE TABLE STOCK77
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
       )
       IN ts_S_077
       INDEX IN ts_S_077
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 97129 ENDING AT 98406
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK78.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK78;
CREATE TABLE STOCK78
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
       )
       IN ts_S_078
       INDEX IN ts_S_078
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 98407 ENDING AT 99684
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK79.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK79;
CREATE TABLE STOCK79
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
       )
       IN ts_S_079
       INDEX IN ts_S_079
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 99685 ENDING AT 100962
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK7.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK7;
CREATE TABLE STOCK7
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
       )
       IN ts_S_007
       INDEX IN ts_S_007
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 7669 ENDING AT 8946
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK80.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK80;
CREATE TABLE STOCK80
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD          INTEGER     NOT NULL,
```

```
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_080
       INDEX IN ts_S_080
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 100963 ENDING AT 102240
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK81.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK81;
CREATE TABLE STOCK81
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_081
       INDEX IN ts_S_081
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 102241 ENDING AT 103518
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK82.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK82;
CREATE TABLE STOCK82
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_082
       INDEX IN ts_S_082
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 103519 ENDING AT 104796
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK83.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK83;
CREATE TABLE STOCK83
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_083
       INDEX IN ts_S_083
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 104797 ENDING AT 106074
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK84.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK84;
CREATE TABLE STOCK84
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_084
       INDEX IN ts_S_084
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 106075 ENDING AT 107352
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK85.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK85;
CREATE TABLE STOCK85
       (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
       )
       IN ts_S_085
       INDEX IN ts_S_085
       ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 107353 ENDING AT 108630
       )
       ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK86.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK86;
```

```
CREATE TABLE STOCK86
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_086
        INDEX IN ts_S_086
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 108631 ENDING AT 109908
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK87.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK87;
CREATE TABLE STOCK87
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_087
        INDEX IN ts_S_087
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 109909 ENDING AT 111186
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK88.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK88;
CREATE TABLE STOCK88
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_088
        INDEX IN ts_S_088
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 111187 ENDING AT 112464
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK89.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK89;
CREATE TABLE STOCK89
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_089
        INDEX IN ts_S_089
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 112465 ENDING AT 113742
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK8.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK8;
CREATE TABLE STOCK8
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_008
        INDEX IN ts_S_008
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 8947 ENDING AT 10224
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK90.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK90;
CREATE TABLE STOCK90
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_090
        INDEX IN ts_S_090
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 113743 ENDING AT 115020
        )
        ALLOW OVERFLOW;
```

```
connect reset;
```

CRTB_STOCK91.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK91;
CREATE TABLE STOCK91
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_091
        INDEX IN ts_S_091
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 115021 ENDING AT 116298
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK92.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK92;
CREATE TABLE STOCK92
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_092
        INDEX IN ts_S_092
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 116299 ENDING AT 117576
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK93.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK93;
CREATE TABLE STOCK93
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_093
        INDEX IN ts_S_093
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 117577 ENDING AT 118854
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK94.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK94;
CREATE TABLE STOCK94
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_094
        INDEX IN ts_S_094
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 118855 ENDING AT 120132
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK95.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK95;
CREATE TABLE STOCK95
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_095
        INDEX IN ts_S_095
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 120133 ENDING AT 121410
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK96.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK96;
CREATE TABLE STOCK96
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_096
```

```
        INDEX IN ts_S_096
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 121411 ENDING AT 122688
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK97.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK97;
CREATE TABLE STOCK97
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_097
        INDEX IN ts_S_097
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 122689 ENDING AT 123966
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK98.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK98;
CREATE TABLE STOCK98
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_098
        INDEX IN ts_S_098
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 123967 ENDING AT 125244
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK99.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK99;
CREATE TABLE STOCK99
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_099
        INDEX IN ts_S_099
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 125245 ENDING AT 126522
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_STOCK9.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK9;
CREATE TABLE STOCK9
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
         S_DIST_04       CHAR(24)    NOT NULL,
         S_DIST_05       CHAR(24)    NOT NULL,
         S_DIST_06       CHAR(24)    NOT NULL,
         S_DIST_07       CHAR(24)    NOT NULL,
         S_DIST_08       CHAR(24)    NOT NULL,
         S_DIST_09       CHAR(24)    NOT NULL,
         S_DIST_10       CHAR(24)    NOT NULL,
         S_I_ID          INTEGER     NOT NULL,
         S_W_ID          INTEGER     NOT NULL
        )
        IN ts_S_009
        INDEX IN ts_S_009
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 10225 ENDING AT 11502
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE10.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE10;
CREATE TABLE WAREHOUSE10
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_010
        INDEX IN ts_W_010
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 46009 ENDING AT 51120
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE11.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE11;
CREATE TABLE WAREHOUSE11
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_011
        INDEX IN ts_W_011
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 51121 ENDING AT 56232
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE12.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE12;
```

```
CREATE TABLE WAREHOUSE12
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_012
        INDEX IN ts_W_012
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 56233 ENDING AT 61344
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE13.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE13;
CREATE TABLE WAREHOUSE13
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_013
        INDEX IN ts_W_013
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 61345 ENDING AT 66456
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE14.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE14;
CREATE TABLE WAREHOUSE14
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_014
        INDEX IN ts_W_014
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 66457 ENDING AT 71568
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE15.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE15;
CREATE TABLE WAREHOUSE15
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_015
        INDEX IN ts_W_015
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 71569 ENDING AT 76680
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE16.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE16;
CREATE TABLE WAREHOUSE16
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_016
        INDEX IN ts_W_016
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 76681 ENDING AT 81792
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE17.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE17;
CREATE TABLE WAREHOUSE17
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_017
        INDEX IN ts_W_017
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 81793 ENDING AT 86904
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE18.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE18;
CREATE TABLE WAREHOUSE18
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_018
        INDEX IN ts_W_018
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 86905 ENDING AT 92016
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE19.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE19;
CREATE TABLE WAREHOUSE19
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_019
        INDEX IN ts_W_019
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 92017 ENDING AT 97128
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE1.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE1;
```

```
CREATE TABLE WAREHOUSE1
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_001
        INDEX IN ts_W_001
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 1 ENDING AT 5112
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE20.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE20;
CREATE TABLE WAREHOUSE20
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_020
        INDEX IN ts_W_020
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 97129 ENDING AT 102240
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE21.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE21;
CREATE TABLE WAREHOUSE21
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_021
        INDEX IN ts_W_021
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 102241 ENDING AT 107352
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE22.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE22;
CREATE TABLE WAREHOUSE22
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_022
        INDEX IN ts_W_022
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 107353 ENDING AT 112464
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE23.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE23;
CREATE TABLE WAREHOUSE23
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_023
        INDEX IN ts_W_023
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 112465 ENDING AT 117576
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE24.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE24;
CREATE TABLE WAREHOUSE24
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_024
        INDEX IN ts_W_024
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 117577 ENDING AT 122688
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE25.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE25;
CREATE TABLE WAREHOUSE25
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_025
        INDEX IN ts_W_025
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 122689 ENDING AT 127800
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE26.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE26;
CREATE TABLE WAREHOUSE26
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_026
        INDEX IN ts_W_026
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 127801 ENDING AT 132912
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE27.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE27;
```

```
CREATE TABLE WAREHOUSE27
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_027
        INDEX IN ts_W_027
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 132913 ENDING AT 138024
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE28.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE28;
CREATE TABLE WAREHOUSE28
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_028
        INDEX IN ts_W_028
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 138025 ENDING AT 143136
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE29.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE29;
CREATE TABLE WAREHOUSE29
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_029
        INDEX IN ts_W_029
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 143137 ENDING AT 148248
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE2.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE2;
CREATE TABLE WAREHOUSE2
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_002
        INDEX IN ts_W_002
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 5113 ENDING AT 10224
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE30.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE30;
CREATE TABLE WAREHOUSE30
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_030
        INDEX IN ts_W_030
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 148249 ENDING AT 153360
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE31.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE31;
CREATE TABLE WAREHOUSE31
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_031
        INDEX IN ts_W_031
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 153361 ENDING AT 158472
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE32.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE32;
CREATE TABLE WAREHOUSE32
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_032
        INDEX IN ts_W_032
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 158473 ENDING AT 163584
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE33.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE33;
CREATE TABLE WAREHOUSE33
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_033
        INDEX IN ts_W_033
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 163585 ENDING AT 168696
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE34.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE34;
```

```
CREATE TABLE WAREHOUSE34
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_034
        INDEX IN ts_W_034
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 168697 ENDING AT 173808
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE35.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE35;
CREATE TABLE WAREHOUSE35
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_035
        INDEX IN ts_W_035
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 173809 ENDING AT 178920
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE36.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE36;
CREATE TABLE WAREHOUSE36
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_036
        INDEX IN ts_W_036
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 178921 ENDING AT 184032
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE3.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE3;
CREATE TABLE WAREHOUSE3
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_003
        INDEX IN ts_W_003
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 10225 ENDING AT 15336
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE4.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE4;
CREATE TABLE WAREHOUSE4
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_004
        INDEX IN ts_W_004
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 15337 ENDING AT 20448
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE5.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE5;
CREATE TABLE WAREHOUSE5
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_005
        INDEX IN ts_W_005
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 20449 ENDING AT 25560
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE6.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE6;
CREATE TABLE WAREHOUSE6
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_006
        INDEX IN ts_W_006
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 25561 ENDING AT 30672
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE7.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE7;
CREATE TABLE WAREHOUSE7
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_007
        INDEX IN ts_W_007
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 30673 ENDING AT 35784
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE8.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE8;
```

```
CREATE TABLE WAREHOUSE8
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_008
        INDEX IN ts_W_008
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 35785 ENDING AT 40896
        )
        ALLOW OVERFLOW;
connect reset;
```

CRTB_WAREHOUSE9.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE9;
CREATE TABLE WAREHOUSE9
        (
         W_NAME      CHAR(10)        NOT NULL,
         W_STREET_1  CHAR(20)        NOT NULL,
         W_STREET_2  CHAR(20)        NOT NULL,
         W_CITY      CHAR(20)        NOT NULL,
         W_STATE     CHAR(2)         NOT NULL,
         W_ZIP       CHAR(9)         NOT NULL,
         W_TAX       REAL            NOT NULL,
         W_YTD       DECIMAL(12,2)   NOT NULL,
         W_ID        INTEGER         NOT NULL
        )
        IN ts_W_009
        INDEX IN ts_W_009
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 40897 ENDING AT 46008
        )
        ALLOW OVERFLOW;
connect reset;
```

CRVW_CUSTOMER.ddl

```
connect to TPCC in share mode;
DROP VIEW CUSTOMER;
CREATE VIEW CUSTOMER
        (C_ID,
         C_STATE,
         C_ZIP,
         C_PHONE,
         C_SINCE,
         C_CREDIT_LIM,
         C_MIDDLE,
         C_CREDIT,
         C_DISCOUNT,
         C_DATA,
         C_LAST,
         C_FIRST,
         C_STREET_1,
         C_STREET_2,
         C_CITY,
         C_D_ID,
         C_W_ID,
         C_DELIVERY_CNT,
         C_BALANCE,
         C_YTD_PAYMENT,
         C_PAYMENT_CNT
        ) AS SELECT * FROM CUSTOMER1 UNION ALL
SELECT * FROM CUSTOMER2 UNION ALL
SELECT * FROM CUSTOMER3 UNION ALL
SELECT * FROM CUSTOMER4 UNION ALL
SELECT * FROM CUSTOMER5 UNION ALL
SELECT * FROM CUSTOMER6 UNION ALL
SELECT * FROM CUSTOMER7 UNION ALL
SELECT * FROM CUSTOMER8 UNION ALL
SELECT * FROM CUSTOMER9 UNION ALL
SELECT * FROM CUSTOMER10 UNION ALL
SELECT * FROM CUSTOMER11 UNION ALL
SELECT * FROM CUSTOMER12 UNION ALL
SELECT * FROM CUSTOMER13 UNION ALL
SELECT * FROM CUSTOMER14 UNION ALL
SELECT * FROM CUSTOMER15 UNION ALL
SELECT * FROM CUSTOMER16 UNION ALL
SELECT * FROM CUSTOMER17 UNION ALL
SELECT * FROM CUSTOMER18 UNION ALL
SELECT * FROM CUSTOMER19 UNION ALL
SELECT * FROM CUSTOMER20 UNION ALL
SELECT * FROM CUSTOMER21 UNION ALL
SELECT * FROM CUSTOMER22 UNION ALL
SELECT * FROM CUSTOMER23 UNION ALL
SELECT * FROM CUSTOMER24 UNION ALL
SELECT * FROM CUSTOMER25 UNION ALL
SELECT * FROM CUSTOMER26 UNION ALL
SELECT * FROM CUSTOMER27 UNION ALL
SELECT * FROM CUSTOMER28 UNION ALL
SELECT * FROM CUSTOMER29 UNION ALL
SELECT * FROM CUSTOMER30 UNION ALL
SELECT * FROM CUSTOMER31 UNION ALL
SELECT * FROM CUSTOMER32 UNION ALL
SELECT * FROM CUSTOMER33 UNION ALL
SELECT * FROM CUSTOMER34 UNION ALL
SELECT * FROM CUSTOMER35 UNION ALL
SELECT * FROM CUSTOMER36 UNION ALL
SELECT * FROM CUSTOMER37 UNION ALL
SELECT * FROM CUSTOMER38 UNION ALL
SELECT * FROM CUSTOMER39 UNION ALL
SELECT * FROM CUSTOMER40 UNION ALL
SELECT * FROM CUSTOMER41 UNION ALL
SELECT * FROM CUSTOMER42 UNION ALL
SELECT * FROM CUSTOMER43 UNION ALL
SELECT * FROM CUSTOMER44 UNION ALL
SELECT * FROM CUSTOMER45 UNION ALL
SELECT * FROM CUSTOMER46 UNION ALL
SELECT * FROM CUSTOMER47 UNION ALL
SELECT * FROM CUSTOMER48 UNION ALL
SELECT * FROM CUSTOMER49 UNION ALL
SELECT * FROM CUSTOMER50 UNION ALL
SELECT * FROM CUSTOMER51 UNION ALL
SELECT * FROM CUSTOMER52 UNION ALL
SELECT * FROM CUSTOMER53 UNION ALL
SELECT * FROM CUSTOMER54 UNION ALL
SELECT * FROM CUSTOMER55 UNION ALL
SELECT * FROM CUSTOMER56 UNION ALL
SELECT * FROM CUSTOMER57 UNION ALL
SELECT * FROM CUSTOMER58 UNION ALL
SELECT * FROM CUSTOMER59 UNION ALL
SELECT * FROM CUSTOMER60 UNION ALL
SELECT * FROM CUSTOMER61 UNION ALL
SELECT * FROM CUSTOMER62 UNION ALL
SELECT * FROM CUSTOMER63 UNION ALL
SELECT * FROM CUSTOMER64 UNION ALL
SELECT * FROM CUSTOMER65 UNION ALL
SELECT * FROM CUSTOMER66 UNION ALL
SELECT * FROM CUSTOMER67 UNION ALL
SELECT * FROM CUSTOMER68 UNION ALL
SELECT * FROM CUSTOMER69 UNION ALL
SELECT * FROM CUSTOMER70 UNION ALL
SELECT * FROM CUSTOMER71 UNION ALL
SELECT * FROM CUSTOMER72 UNION ALL
SELECT * FROM CUSTOMER73 UNION ALL
SELECT * FROM CUSTOMER74 UNION ALL
SELECT * FROM CUSTOMER75 UNION ALL
SELECT * FROM CUSTOMER76 UNION ALL
SELECT * FROM CUSTOMER77 UNION ALL
SELECT * FROM CUSTOMER78 UNION ALL
SELECT * FROM CUSTOMER79 UNION ALL
SELECT * FROM CUSTOMER80 UNION ALL
SELECT * FROM CUSTOMER81 UNION ALL
SELECT * FROM CUSTOMER82 UNION ALL
SELECT * FROM CUSTOMER83 UNION ALL
SELECT * FROM CUSTOMER84 UNION ALL
SELECT * FROM CUSTOMER85 UNION ALL
SELECT * FROM CUSTOMER86 UNION ALL
SELECT * FROM CUSTOMER87 UNION ALL
SELECT * FROM CUSTOMER88 UNION ALL
SELECT * FROM CUSTOMER89 UNION ALL
SELECT * FROM CUSTOMER90 UNION ALL
SELECT * FROM CUSTOMER91 UNION ALL
SELECT * FROM CUSTOMER92 UNION ALL
SELECT * FROM CUSTOMER93 UNION ALL
SELECT * FROM CUSTOMER94 UNION ALL
SELECT * FROM CUSTOMER95 UNION ALL
SELECT * FROM CUSTOMER96 UNION ALL
SELECT * FROM CUSTOMER97 UNION ALL
SELECT * FROM CUSTOMER98 UNION ALL
SELECT * FROM CUSTOMER99 UNION ALL
SELECT * FROM CUSTOMER100 UNION ALL
SELECT * FROM CUSTOMER101 UNION ALL
SELECT * FROM CUSTOMER102 UNION ALL
SELECT * FROM CUSTOMER103 UNION ALL
SELECT * FROM CUSTOMER104 UNION ALL
SELECT * FROM CUSTOMER105 UNION ALL
SELECT * FROM CUSTOMER106 UNION ALL
SELECT * FROM CUSTOMER107 UNION ALL
SELECT * FROM CUSTOMER108 UNION ALL
SELECT * FROM CUSTOMER109 UNION ALL
SELECT * FROM CUSTOMER110 UNION ALL
SELECT * FROM CUSTOMER111 UNION ALL
SELECT * FROM CUSTOMER112 UNION ALL
SELECT * FROM CUSTOMER113 UNION ALL
SELECT * FROM CUSTOMER114 UNION ALL
SELECT * FROM CUSTOMER115 UNION ALL
SELECT * FROM CUSTOMER116 UNION ALL
SELECT * FROM CUSTOMER117 UNION ALL
SELECT * FROM CUSTOMER118 UNION ALL
SELECT * FROM CUSTOMER119 UNION ALL
SELECT * FROM CUSTOMER120 UNION ALL
SELECT * FROM CUSTOMER121 UNION ALL
SELECT * FROM CUSTOMER122 UNION ALL
SELECT * FROM CUSTOMER123 UNION ALL
```

```
SELECT * FROM CUSTOMER124 UNION ALL
SELECT * FROM CUSTOMER125 UNION ALL
SELECT * FROM CUSTOMER126 UNION ALL
SELECT * FROM CUSTOMER127 UNION ALL
SELECT * FROM CUSTOMER128 UNION ALL
SELECT * FROM CUSTOMER129 UNION ALL
SELECT * FROM CUSTOMER130 UNION ALL
SELECT * FROM CUSTOMER131 UNION ALL
SELECT * FROM CUSTOMER132 UNION ALL
SELECT * FROM CUSTOMER133 UNION ALL
SELECT * FROM CUSTOMER134 UNION ALL
SELECT * FROM CUSTOMER135 UNION ALL
SELECT * FROM CUSTOMER136 UNION ALL
SELECT * FROM CUSTOMER137 UNION ALL
SELECT * FROM CUSTOMER138 UNION ALL
SELECT * FROM CUSTOMER139 UNION ALL
SELECT * FROM CUSTOMER140 UNION ALL
SELECT * FROM CUSTOMER141 UNION ALL
SELECT * FROM CUSTOMER142 UNION ALL
SELECT * FROM CUSTOMER143 UNION ALL
SELECT * FROM CUSTOMER144
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

CRVW_DISTRICT.ddl

```
connect to TPCC in share mode;
DROP VIEW DISTRICT;
CREATE VIEW DISTRICT
        (D_NEXT_O_ID,
         D_TAX,
         D_YTD,
         D_NAME,
         D_STREET_1,
         D_STREET_2,
         D_CITY,
         D_STATE,
         D_ZIP,
         D_ID,
         D_W_ID
        ) AS SELECT * FROM DISTRICT1 UNION ALL
SELECT * FROM DISTRICT2 UNION ALL
SELECT * FROM DISTRICT3 UNION ALL
SELECT * FROM DISTRICT4 UNION ALL
SELECT * FROM DISTRICT5 UNION ALL
SELECT * FROM DISTRICT6 UNION ALL
SELECT * FROM DISTRICT7 UNION ALL
SELECT * FROM DISTRICT8 UNION ALL
SELECT * FROM DISTRICT9 UNION ALL
SELECT * FROM DISTRICT10 UNION ALL
SELECT * FROM DISTRICT11 UNION ALL
SELECT * FROM DISTRICT12 UNION ALL
SELECT * FROM DISTRICT13 UNION ALL
SELECT * FROM DISTRICT14 UNION ALL
SELECT * FROM DISTRICT15 UNION ALL
SELECT * FROM DISTRICT16 UNION ALL
SELECT * FROM DISTRICT17 UNION ALL
SELECT * FROM DISTRICT18 UNION ALL
SELECT * FROM DISTRICT19 UNION ALL
SELECT * FROM DISTRICT20 UNION ALL
SELECT * FROM DISTRICT21 UNION ALL
SELECT * FROM DISTRICT22 UNION ALL
SELECT * FROM DISTRICT23 UNION ALL
SELECT * FROM DISTRICT24 UNION ALL
SELECT * FROM DISTRICT25 UNION ALL
SELECT * FROM DISTRICT26 UNION ALL
SELECT * FROM DISTRICT27 UNION ALL
SELECT * FROM DISTRICT28 UNION ALL
SELECT * FROM DISTRICT29 UNION ALL
SELECT * FROM DISTRICT30 UNION ALL
SELECT * FROM DISTRICT31 UNION ALL
SELECT * FROM DISTRICT32 UNION ALL
SELECT * FROM DISTRICT33 UNION ALL
SELECT * FROM DISTRICT34 UNION ALL
SELECT * FROM DISTRICT35 UNION ALL
SELECT * FROM DISTRICT36
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

CRVW_HISTORY.ddl

```
connect to TPCC in share mode;
DROP VIEW HISTORY;
CREATE VIEW HISTORY
        (H_C_ID,
         H_C_D_ID,
         H_C_W_ID,
         H_D_ID,
         H_W_ID,
         H_DATE,
         H_AMOUNT,
         H_DATA
        ) AS SELECT * FROM HISTORY1 UNION ALL
SELECT * FROM HISTORY2 UNION ALL
SELECT * FROM HISTORY3 UNION ALL
SELECT * FROM HISTORY4 UNION ALL
SELECT * FROM HISTORY5 UNION ALL
SELECT * FROM HISTORY6 UNION ALL
SELECT * FROM HISTORY7 UNION ALL
SELECT * FROM HISTORY8 UNION ALL
SELECT * FROM HISTORY9 UNION ALL
SELECT * FROM HISTORY10 UNION ALL
SELECT * FROM HISTORY11 UNION ALL
SELECT * FROM HISTORY12 UNION ALL
SELECT * FROM HISTORY13 UNION ALL
SELECT * FROM HISTORY14 UNION ALL
SELECT * FROM HISTORY15 UNION ALL
SELECT * FROM HISTORY16 UNION ALL
SELECT * FROM HISTORY17 UNION ALL
SELECT * FROM HISTORY18 UNION ALL
SELECT * FROM HISTORY19 UNION ALL
SELECT * FROM HISTORY20 UNION ALL
SELECT * FROM HISTORY21 UNION ALL
SELECT * FROM HISTORY22 UNION ALL
SELECT * FROM HISTORY23 UNION ALL
SELECT * FROM HISTORY24 UNION ALL
SELECT * FROM HISTORY25 UNION ALL
SELECT * FROM HISTORY26 UNION ALL
SELECT * FROM HISTORY27 UNION ALL
SELECT * FROM HISTORY28 UNION ALL
SELECT * FROM HISTORY29 UNION ALL
SELECT * FROM HISTORY30 UNION ALL
SELECT * FROM HISTORY31 UNION ALL
SELECT * FROM HISTORY32 UNION ALL
SELECT * FROM HISTORY33 UNION ALL
SELECT * FROM HISTORY34 UNION ALL
SELECT * FROM HISTORY35 UNION ALL
SELECT * FROM HISTORY36
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

CRVW_NEW_ORDER.ddl

```
connect to TPCC in share mode;
DROP VIEW NEW_ORDER;
CREATE VIEW NEW_ORDER
        (NO_O_ID,
         NO_D_ID,
         NO_W_ID
        ) AS SELECT * FROM NEW_ORDER1 UNION ALL
SELECT * FROM NEW_ORDER2 UNION ALL
SELECT * FROM NEW_ORDER3 UNION ALL
SELECT * FROM NEW_ORDER4 UNION ALL
SELECT * FROM NEW_ORDER5 UNION ALL
SELECT * FROM NEW_ORDER6 UNION ALL
SELECT * FROM NEW_ORDER7 UNION ALL
SELECT * FROM NEW_ORDER8 UNION ALL
SELECT * FROM NEW_ORDER9 UNION ALL
SELECT * FROM NEW_ORDER10 UNION ALL
SELECT * FROM NEW_ORDER11 UNION ALL
SELECT * FROM NEW_ORDER12 UNION ALL
SELECT * FROM NEW_ORDER13 UNION ALL
SELECT * FROM NEW_ORDER14 UNION ALL
SELECT * FROM NEW_ORDER15 UNION ALL
SELECT * FROM NEW_ORDER16 UNION ALL
SELECT * FROM NEW_ORDER17 UNION ALL
SELECT * FROM NEW_ORDER18 UNION ALL
SELECT * FROM NEW_ORDER19 UNION ALL
SELECT * FROM NEW_ORDER20 UNION ALL
SELECT * FROM NEW_ORDER21 UNION ALL
SELECT * FROM NEW_ORDER22 UNION ALL
SELECT * FROM NEW_ORDER23 UNION ALL
SELECT * FROM NEW_ORDER24 UNION ALL
SELECT * FROM NEW_ORDER25 UNION ALL
SELECT * FROM NEW_ORDER26 UNION ALL
SELECT * FROM NEW_ORDER27 UNION ALL
SELECT * FROM NEW_ORDER28 UNION ALL
SELECT * FROM NEW_ORDER29 UNION ALL
SELECT * FROM NEW_ORDER30 UNION ALL
SELECT * FROM NEW_ORDER31 UNION ALL
SELECT * FROM NEW_ORDER32 UNION ALL
SELECT * FROM NEW_ORDER33 UNION ALL
SELECT * FROM NEW_ORDER34 UNION ALL
SELECT * FROM NEW_ORDER35 UNION ALL
SELECT * FROM NEW_ORDER36
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

CRVW_ORDER_LINE.ddl

```
connect to TPCC in share mode;
DROP VIEW ORDER_LINE;
CREATE VIEW ORDER_LINE
        (OL_DELIVERY_D,
         OL_AMOUNT,
         OL_I_ID,
         OL_SUPPLY_W_ID,
         OL_QUANTITY,
```

```
            OL_DIST_INFO,
            OL_O_ID,
            OL_D_ID,
            OL_W_ID,
            OL_NUMBER
           ) AS SELECT * FROM ORDER_LINE1 UNION ALL
SELECT * FROM ORDER_LINE2 UNION ALL
SELECT * FROM ORDER_LINE3 UNION ALL
SELECT * FROM ORDER_LINE4 UNION ALL
SELECT * FROM ORDER_LINE5 UNION ALL
SELECT * FROM ORDER_LINE6 UNION ALL
SELECT * FROM ORDER_LINE7 UNION ALL
SELECT * FROM ORDER_LINE8 UNION ALL
SELECT * FROM ORDER_LINE9 UNION ALL
SELECT * FROM ORDER_LINE10 UNION ALL
SELECT * FROM ORDER_LINE11 UNION ALL
SELECT * FROM ORDER_LINE12 UNION ALL
SELECT * FROM ORDER_LINE13 UNION ALL
SELECT * FROM ORDER_LINE14 UNION ALL
SELECT * FROM ORDER_LINE15 UNION ALL
SELECT * FROM ORDER_LINE16 UNION ALL
SELECT * FROM ORDER_LINE17 UNION ALL
SELECT * FROM ORDER_LINE18 UNION ALL
SELECT * FROM ORDER_LINE19 UNION ALL
SELECT * FROM ORDER_LINE20 UNION ALL
SELECT * FROM ORDER_LINE21 UNION ALL
SELECT * FROM ORDER_LINE22 UNION ALL
SELECT * FROM ORDER_LINE23 UNION ALL
SELECT * FROM ORDER_LINE24 UNION ALL
SELECT * FROM ORDER_LINE25 UNION ALL
SELECT * FROM ORDER_LINE26 UNION ALL
SELECT * FROM ORDER_LINE27 UNION ALL
SELECT * FROM ORDER_LINE28 UNION ALL
SELECT * FROM ORDER_LINE29 UNION ALL
SELECT * FROM ORDER_LINE30 UNION ALL
SELECT * FROM ORDER_LINE31 UNION ALL
SELECT * FROM ORDER_LINE32 UNION ALL
SELECT * FROM ORDER_LINE33 UNION ALL
SELECT * FROM ORDER_LINE34 UNION ALL
SELECT * FROM ORDER_LINE35 UNION ALL
SELECT * FROM ORDER_LINE36 UNION ALL
SELECT * FROM ORDER_LINE37 UNION ALL
SELECT * FROM ORDER_LINE38 UNION ALL
SELECT * FROM ORDER_LINE39 UNION ALL
SELECT * FROM ORDER_LINE40 UNION ALL
SELECT * FROM ORDER_LINE41 UNION ALL
SELECT * FROM ORDER_LINE42 UNION ALL
SELECT * FROM ORDER_LINE43 UNION ALL
SELECT * FROM ORDER_LINE44 UNION ALL
SELECT * FROM ORDER_LINE45 UNION ALL
SELECT * FROM ORDER_LINE46 UNION ALL
SELECT * FROM ORDER_LINE47 UNION ALL
SELECT * FROM ORDER_LINE48 UNION ALL
SELECT * FROM ORDER_LINE49 UNION ALL
SELECT * FROM ORDER_LINE50 UNION ALL
SELECT * FROM ORDER_LINE51 UNION ALL
SELECT * FROM ORDER_LINE52 UNION ALL
SELECT * FROM ORDER_LINE53 UNION ALL
SELECT * FROM ORDER_LINE54 UNION ALL
SELECT * FROM ORDER_LINE55 UNION ALL
SELECT * FROM ORDER_LINE56 UNION ALL
SELECT * FROM ORDER_LINE57 UNION ALL
SELECT * FROM ORDER_LINE58 UNION ALL
SELECT * FROM ORDER_LINE59 UNION ALL
SELECT * FROM ORDER_LINE60 UNION ALL
SELECT * FROM ORDER_LINE61 UNION ALL
SELECT * FROM ORDER_LINE62 UNION ALL
SELECT * FROM ORDER_LINE63 UNION ALL
SELECT * FROM ORDER_LINE64 UNION ALL
SELECT * FROM ORDER_LINE65 UNION ALL
SELECT * FROM ORDER_LINE66 UNION ALL
SELECT * FROM ORDER_LINE67 UNION ALL
SELECT * FROM ORDER_LINE68 UNION ALL
SELECT * FROM ORDER_LINE69 UNION ALL
SELECT * FROM ORDER_LINE70 UNION ALL
SELECT * FROM ORDER_LINE71 UNION ALL
SELECT * FROM ORDER_LINE72
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

CRVW_ORDERS.ddl

```
connect to TPCC in share mode;
DROP VIEW ORDERS;
CREATE VIEW ORDERS
           (O_C_ID,
            O_ENTRY_D,
            O_CARRIER_ID,
            O_OL_CNT,
            O_ALL_LOCAL,
            O_ID,
            O_W_ID,
            O_D_ID
           ) AS SELECT * FROM ORDERS1 UNION ALL
SELECT * FROM ORDERS2 UNION ALL
SELECT * FROM ORDERS3 UNION ALL
SELECT * FROM ORDERS4 UNION ALL
SELECT * FROM ORDERS5 UNION ALL
SELECT * FROM ORDERS6 UNION ALL
SELECT * FROM ORDERS7 UNION ALL
SELECT * FROM ORDERS8 UNION ALL
SELECT * FROM ORDERS9 UNION ALL
SELECT * FROM ORDERS10 UNION ALL
SELECT * FROM ORDERS11 UNION ALL
SELECT * FROM ORDERS12 UNION ALL
SELECT * FROM ORDERS13 UNION ALL
SELECT * FROM ORDERS14 UNION ALL
SELECT * FROM ORDERS15 UNION ALL
SELECT * FROM ORDERS16 UNION ALL
SELECT * FROM ORDERS17 UNION ALL
SELECT * FROM ORDERS18 UNION ALL
SELECT * FROM ORDERS19 UNION ALL
SELECT * FROM ORDERS20 UNION ALL
SELECT * FROM ORDERS21 UNION ALL
SELECT * FROM ORDERS22 UNION ALL
SELECT * FROM ORDERS23 UNION ALL
SELECT * FROM ORDERS24 UNION ALL
SELECT * FROM ORDERS25 UNION ALL
SELECT * FROM ORDERS26 UNION ALL
SELECT * FROM ORDERS27 UNION ALL
SELECT * FROM ORDERS28 UNION ALL
SELECT * FROM ORDERS29 UNION ALL
SELECT * FROM ORDERS30 UNION ALL
SELECT * FROM ORDERS31 UNION ALL
SELECT * FROM ORDERS32 UNION ALL
SELECT * FROM ORDERS33 UNION ALL
SELECT * FROM ORDERS34 UNION ALL
SELECT * FROM ORDERS35 UNION ALL
SELECT * FROM ORDERS36
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

CRVW_STOCK.ddl

```
connect to TPCC in share mode;
DROP VIEW STOCK;
CREATE VIEW STOCK
           (S_REMOTE_CNT,
            S_QUANTITY,
            S_ORDER_CNT,
            S_YTD,
            S_DATA,
            S_DIST_01,
            S_DIST_02,
            S_DIST_03,
            S_DIST_04,
            S_DIST_05,
            S_DIST_06,
            S_DIST_07,
            S_DIST_08,
            S_DIST_09,
            S_DIST_10,
            S_I_ID,
            S_W_ID
           ) AS SELECT * FROM STOCK1 UNION ALL
SELECT * FROM STOCK2 UNION ALL
SELECT * FROM STOCK3 UNION ALL
SELECT * FROM STOCK4 UNION ALL
SELECT * FROM STOCK5 UNION ALL
SELECT * FROM STOCK6 UNION ALL
SELECT * FROM STOCK7 UNION ALL
SELECT * FROM STOCK8 UNION ALL
SELECT * FROM STOCK9 UNION ALL
SELECT * FROM STOCK10 UNION ALL
SELECT * FROM STOCK11 UNION ALL
SELECT * FROM STOCK12 UNION ALL
SELECT * FROM STOCK13 UNION ALL
SELECT * FROM STOCK14 UNION ALL
SELECT * FROM STOCK15 UNION ALL
SELECT * FROM STOCK16 UNION ALL
SELECT * FROM STOCK17 UNION ALL
SELECT * FROM STOCK18 UNION ALL
SELECT * FROM STOCK19 UNION ALL
SELECT * FROM STOCK20 UNION ALL
SELECT * FROM STOCK21 UNION ALL
SELECT * FROM STOCK22 UNION ALL
SELECT * FROM STOCK23 UNION ALL
SELECT * FROM STOCK24 UNION ALL
SELECT * FROM STOCK25 UNION ALL
SELECT * FROM STOCK26 UNION ALL
SELECT * FROM STOCK27 UNION ALL
SELECT * FROM STOCK28 UNION ALL
SELECT * FROM STOCK29 UNION ALL
SELECT * FROM STOCK30 UNION ALL
SELECT * FROM STOCK31 UNION ALL
SELECT * FROM STOCK32 UNION ALL
SELECT * FROM STOCK33 UNION ALL
SELECT * FROM STOCK34 UNION ALL
SELECT * FROM STOCK35 UNION ALL
SELECT * FROM STOCK36 UNION ALL
SELECT * FROM STOCK37 UNION ALL
SELECT * FROM STOCK38 UNION ALL
```

```
SELECT * FROM STOCK39 UNION ALL
SELECT * FROM STOCK40 UNION ALL
SELECT * FROM STOCK41 UNION ALL
SELECT * FROM STOCK42 UNION ALL
SELECT * FROM STOCK43 UNION ALL
SELECT * FROM STOCK44 UNION ALL
SELECT * FROM STOCK45 UNION ALL
SELECT * FROM STOCK46 UNION ALL
SELECT * FROM STOCK47 UNION ALL
SELECT * FROM STOCK48 UNION ALL
SELECT * FROM STOCK49 UNION ALL
SELECT * FROM STOCK50 UNION ALL
SELECT * FROM STOCK51 UNION ALL
SELECT * FROM STOCK52 UNION ALL
SELECT * FROM STOCK53 UNION ALL
SELECT * FROM STOCK54 UNION ALL
SELECT * FROM STOCK55 UNION ALL
SELECT * FROM STOCK56 UNION ALL
SELECT * FROM STOCK57 UNION ALL
SELECT * FROM STOCK58 UNION ALL
SELECT * FROM STOCK59 UNION ALL
SELECT * FROM STOCK60 UNION ALL
SELECT * FROM STOCK61 UNION ALL
SELECT * FROM STOCK62 UNION ALL
SELECT * FROM STOCK63 UNION ALL
SELECT * FROM STOCK64 UNION ALL
SELECT * FROM STOCK65 UNION ALL
SELECT * FROM STOCK66 UNION ALL
SELECT * FROM STOCK67 UNION ALL
SELECT * FROM STOCK68 UNION ALL
SELECT * FROM STOCK69 UNION ALL
SELECT * FROM STOCK70 UNION ALL
SELECT * FROM STOCK71 UNION ALL
SELECT * FROM STOCK72 UNION ALL
SELECT * FROM STOCK73 UNION ALL
SELECT * FROM STOCK74 UNION ALL
SELECT * FROM STOCK75 UNION ALL
SELECT * FROM STOCK76 UNION ALL
SELECT * FROM STOCK77 UNION ALL
SELECT * FROM STOCK78 UNION ALL
SELECT * FROM STOCK79 UNION ALL
SELECT * FROM STOCK80 UNION ALL
SELECT * FROM STOCK81 UNION ALL
SELECT * FROM STOCK82 UNION ALL
SELECT * FROM STOCK83 UNION ALL
SELECT * FROM STOCK84 UNION ALL
SELECT * FROM STOCK85 UNION ALL
SELECT * FROM STOCK86 UNION ALL
SELECT * FROM STOCK87 UNION ALL
SELECT * FROM STOCK88 UNION ALL
SELECT * FROM STOCK89 UNION ALL
SELECT * FROM STOCK90 UNION ALL
SELECT * FROM STOCK91 UNION ALL
SELECT * FROM STOCK92 UNION ALL
SELECT * FROM STOCK93 UNION ALL
SELECT * FROM STOCK94 UNION ALL
SELECT * FROM STOCK95 UNION ALL
SELECT * FROM STOCK96 UNION ALL
SELECT * FROM STOCK97 UNION ALL
SELECT * FROM STOCK98 UNION ALL
SELECT * FROM STOCK99 UNION ALL
SELECT * FROM STOCK100 UNION ALL
SELECT * FROM STOCK101 UNION ALL
SELECT * FROM STOCK102 UNION ALL
SELECT * FROM STOCK103 UNION ALL
SELECT * FROM STOCK104 UNION ALL
SELECT * FROM STOCK105 UNION ALL
SELECT * FROM STOCK106 UNION ALL
SELECT * FROM STOCK107 UNION ALL
SELECT * FROM STOCK108 UNION ALL
SELECT * FROM STOCK109 UNION ALL
SELECT * FROM STOCK110 UNION ALL
SELECT * FROM STOCK111 UNION ALL
SELECT * FROM STOCK112 UNION ALL
SELECT * FROM STOCK113 UNION ALL
SELECT * FROM STOCK114 UNION ALL
SELECT * FROM STOCK115 UNION ALL
SELECT * FROM STOCK116 UNION ALL
SELECT * FROM STOCK117 UNION ALL
SELECT * FROM STOCK118 UNION ALL
SELECT * FROM STOCK119 UNION ALL
SELECT * FROM STOCK120 UNION ALL
SELECT * FROM STOCK121 UNION ALL
SELECT * FROM STOCK122 UNION ALL
SELECT * FROM STOCK123 UNION ALL
SELECT * FROM STOCK124 UNION ALL
SELECT * FROM STOCK125 UNION ALL
SELECT * FROM STOCK126 UNION ALL
SELECT * FROM STOCK127 UNION ALL
SELECT * FROM STOCK128 UNION ALL
SELECT * FROM STOCK129 UNION ALL
SELECT * FROM STOCK130 UNION ALL
SELECT * FROM STOCK131 UNION ALL
SELECT * FROM STOCK132 UNION ALL
SELECT * FROM STOCK133 UNION ALL
SELECT * FROM STOCK134 UNION ALL
SELECT * FROM STOCK135 UNION ALL
SELECT * FROM STOCK136 UNION ALL
SELECT * FROM STOCK137 UNION ALL
SELECT * FROM STOCK138 UNION ALL
SELECT * FROM STOCK139 UNION ALL
SELECT * FROM STOCK140 UNION ALL
SELECT * FROM STOCK141 UNION ALL
SELECT * FROM STOCK142 UNION ALL
SELECT * FROM STOCK143 UNION ALL
SELECT * FROM STOCK144
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

CRVW_WAREHOUSE.ddl

```
connect to TPCC in share mode;
DROP VIEW WAREHOUSE;
CREATE VIEW WAREHOUSE
            (W_NAME,
             W_STREET_1,
             W_STREET_2,
             W_CITY,
             W_STATE,
             W_ZIP,
             W_TAX,
             W_YTD,
             W_ID
            ) AS SELECT * FROM WAREHOUSE1 UNION ALL
SELECT * FROM WAREHOUSE2 UNION ALL
SELECT * FROM WAREHOUSE3 UNION ALL
SELECT * FROM WAREHOUSE4 UNION ALL
SELECT * FROM WAREHOUSE5 UNION ALL
SELECT * FROM WAREHOUSE6 UNION ALL
SELECT * FROM WAREHOUSE7 UNION ALL
SELECT * FROM WAREHOUSE8 UNION ALL
SELECT * FROM WAREHOUSE9 UNION ALL
SELECT * FROM WAREHOUSE10 UNION ALL
SELECT * FROM WAREHOUSE11 UNION ALL
SELECT * FROM WAREHOUSE12 UNION ALL
SELECT * FROM WAREHOUSE13 UNION ALL
SELECT * FROM WAREHOUSE14 UNION ALL
SELECT * FROM WAREHOUSE15 UNION ALL
SELECT * FROM WAREHOUSE16 UNION ALL
SELECT * FROM WAREHOUSE17 UNION ALL
SELECT * FROM WAREHOUSE18 UNION ALL
SELECT * FROM WAREHOUSE19 UNION ALL
SELECT * FROM WAREHOUSE20 UNION ALL
SELECT * FROM WAREHOUSE21 UNION ALL
SELECT * FROM WAREHOUSE22 UNION ALL
SELECT * FROM WAREHOUSE23 UNION ALL
SELECT * FROM WAREHOUSE24 UNION ALL
SELECT * FROM WAREHOUSE25 UNION ALL
SELECT * FROM WAREHOUSE26 UNION ALL
SELECT * FROM WAREHOUSE27 UNION ALL
SELECT * FROM WAREHOUSE28 UNION ALL
SELECT * FROM WAREHOUSE29 UNION ALL
SELECT * FROM WAREHOUSE30 UNION ALL
SELECT * FROM WAREHOUSE31 UNION ALL
SELECT * FROM WAREHOUSE32 UNION ALL
SELECT * FROM WAREHOUSE33 UNION ALL
SELECT * FROM WAREHOUSE34 UNION ALL
SELECT * FROM WAREHOUSE35 UNION ALL
SELECT * FROM WAREHOUSE36
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

GEN__1.sh

```
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t  -r 1 5112 -f1
/autobench/sources/db2_tpcc-1/flat_files/flat_001/warehouse_001_1.dat
```

GEN__2.sh

```
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t  -r 5113 10224 -f1
/autobench/sources/db2_tpcc-1/flat_files/flat_002/warehouse_002_1.dat
```

GEN__3.sh

```
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t  -r 10225 15336 -f1
/autobench/sources/db2_tpcc-1/flat_files/flat_003/warehouse_003_1.dat
```

GEN__4.sh

```
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t  -r 15337 20448 -f1
/autobench/sources/db2_tpcc-1/flat_files/flat_004/warehouse_004_1.dat
```

GEN_CUSTOMER_100.sh

```
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 126523
127800 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_100/customer_100_1.dat
```

GEN_CUSTOMER_101.sh

```
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 127801
129078 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_101/customer_101_1.dat
```

GEN_CUSTOMER_102.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 129079 130356 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_102/customer_102_1.dat

GEN_CUSTOMER_103.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 130357 131634 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_103/customer_103_1.dat

GEN_CUSTOMER_104.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 131635 132912 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_104/customer_104_1.dat

GEN_CUSTOMER_105.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 132913 134190 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_105/customer_105_1.dat

GEN_CUSTOMER_106.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 134191 135468 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_106/customer_106_1.dat

GEN_CUSTOMER_107.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 135469 136746 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_107/customer_107_1.dat

GEN_CUSTOMER_108.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 136747 138024 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_108/customer_108_1.dat

GEN_CUSTOMER_109.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 138025 139302 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_109/customer_109_1.dat

GEN_CUSTOMER_10.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 11503 12780 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_010/customer_010_1.dat

GEN_CUSTOMER_110.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 139303 140580 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_110/customer_110_1.dat

GEN_CUSTOMER_111.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 140581 141858 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_111/customer_111_1.dat

GEN_CUSTOMER_112.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 141859 143136 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_112/customer_112_1.dat

GEN_CUSTOMER_113.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 143137 144414 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_113/customer_113_1.dat

GEN_CUSTOMER_114.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 144415 145692 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_114/customer_114_1.dat

GEN_CUSTOMER_115.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 145693 146970 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_115/customer_115_1.dat

GEN_CUSTOMER_116.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 146971 148248 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_116/customer_116_1.dat

GEN_CUSTOMER_117.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 148249 149526 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_117/customer_117_1.dat

GEN_CUSTOMER_118.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 149527 150804 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_118/customer_118_1.dat

GEN_CUSTOMER_119.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 150805 152082 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_119/customer_119_1.dat

GEN_CUSTOMER_11.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 12781 14058 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_011/customer_011_1.dat

GEN_CUSTOMER_120.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 152083 153360 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_120/customer_120_1.dat

GEN_CUSTOMER_121.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 153361 154638 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_121/customer_121_1.dat

GEN_CUSTOMER_122.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 154639 155916 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_122/customer_122_1.dat

GEN_CUSTOMER_123.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 155917 157194 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_123/customer_123_1.dat

GEN_CUSTOMER_124.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 157195 158472 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_124/customer_124_1.dat

GEN_CUSTOMER_125.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 158473 159750 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_125/customer_125_1.dat

GEN_CUSTOMER_126.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 159751 161028 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_126/customer_126_1.dat

GEN_CUSTOMER_127.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 161029 162306 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_127/customer_127_1.dat

GEN_CUSTOMER_128.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 162307 163584 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_128/customer_128_1.dat

GEN_CUSTOMER_129.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 163585 164862 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_129/customer_129_1.dat

GEN_CUSTOMER_12.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 14059 15336 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_012/customer_012_1.dat

GEN_CUSTOMER_130.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 164863 166140 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_130/customer_130_1.dat

GEN_CUSTOMER_131.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 166141 167418 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_131/customer_131_1.dat

GEN_CUSTOMER_132.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 167419 168696 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_132/customer_132_1.dat

GEN_CUSTOMER_133.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 168697 169974 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_133/customer_133_1.dat

GEN_CUSTOMER_134.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 169975 171252 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_134/customer_134_1.dat

GEN_CUSTOMER_135.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 171253 172530 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_135/customer_135_1.dat

GEN_CUSTOMER_136.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 172531 173808 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_136/customer_136_1.dat

GEN_CUSTOMER_137.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 173809 175086 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_137/customer_137_1.dat

GEN_CUSTOMER_138.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 175087 176364 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_138/customer_138_1.dat

GEN_CUSTOMER_139.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 176365 177642 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_139/customer_139_1.dat

GEN_CUSTOMER_13.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 15337 16614 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_013/customer_013_1.dat

GEN_CUSTOMER_140.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 177643 178920 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_140/customer_140_1.dat

GEN_CUSTOMER_141.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 178921 180198 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_141/customer_141_1.dat

GEN_CUSTOMER_142.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 180199 181476 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_142/customer_142_1.dat

GEN_CUSTOMER_143.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 181477 182754 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_143/customer_143_1.dat

GEN_CUSTOMER_144.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 182755 184032 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_144/customer_144_1.dat

GEN_CUSTOMER_14.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 16615 17892 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_014/customer_014_1.dat

GEN_CUSTOMER_15.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 17893 19170 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_015/customer_015_1.dat

GEN_CUSTOMER_16.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 19171 20448 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_016/customer_016_1.dat

GEN_CUSTOMER_17.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 20449 21726 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_017/customer_017_1.dat

GEN_CUSTOMER_18.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 21727 23004 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_018/customer_018_1.dat

GEN_CUSTOMER_19.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 23005 24282 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_019/customer_019_1.dat

GEN_CUSTOMER_1.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 1 1278 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_001/customer_001_1.dat

GEN_CUSTOMER_20.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 24283 25560 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_020/customer_020_1.dat

GEN_CUSTOMER_21.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 25561 26838 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_021/customer_021_1.dat

GEN_CUSTOMER_22.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 26839 28116 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_022/customer_022_1.dat

GEN_CUSTOMER_23.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 28117 29394 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_023/customer_023_1.dat

GEN_CUSTOMER_24.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 29395 30672 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_024/customer_024_1.dat

GEN_CUSTOMER_25.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 30673 31950 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_025/customer_025_1.dat

GEN_CUSTOMER_26.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 31951 33228 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_026/customer_026_1.dat

GEN_CUSTOMER_27.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 33229 34506 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_027/customer_027_1.dat

GEN_CUSTOMER_28.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 34507 35784 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_028/customer_028_1.dat

GEN_CUSTOMER_29.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 35785 37062 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_029/customer_029_1.dat

GEN_CUSTOMER_2.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 1279 2556 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_002/customer_002_1.dat

GEN_CUSTOMER_30.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 37063 38340 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_030/customer_030_1.dat

GEN_CUSTOMER_31.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 38341 39618 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_031/customer_031_1.dat

GEN_CUSTOMER_32.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 39619 40896 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_032/customer_032_1.dat

GEN_CUSTOMER_33.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 40897 42174 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_033/customer_033_1.dat

GEN_CUSTOMER_34.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 42175 43452 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_034/customer_034_1.dat

GEN_CUSTOMER_35.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 43453 44730 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_035/customer_035_1.dat

GEN_CUSTOMER_36.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 44731 46008 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_036/customer_036_1.dat

GEN_CUSTOMER_37.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 46009 47286 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_037/customer_037_1.dat

GEN_CUSTOMER_38.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 47287 48564 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_038/customer_038_1.dat

GEN_CUSTOMER_39.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 48565 49842 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_039/customer_039_1.dat

GEN_CUSTOMER_3.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 2557 3834 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_003/customer_003_1.dat

GEN_CUSTOMER_40.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 49843 51120 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_040/customer_040_1.dat

GEN_CUSTOMER_41.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 51121 52398 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_041/customer_041_1.dat

GEN_CUSTOMER_42.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 52399 53676 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_042/customer_042_1.dat

GEN_CUSTOMER_43.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 53677 54954 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_043/customer_043_1.dat

GEN_CUSTOMER_44.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 54955 56232 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_044/customer_044_1.dat

GEN_CUSTOMER_45.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 56233 57510 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_045/customer_045_1.dat

GEN_CUSTOMER_46.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 57511 58788 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_046/customer_046_1.dat

GEN_CUSTOMER_47.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 58789 60066 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_047/customer_047_1.dat

GEN_CUSTOMER_48.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 60067 61344 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_048/customer_048_1.dat

GEN_CUSTOMER_49.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 61345 62622 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_049/customer_049_1.dat

GEN_CUSTOMER_4.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 3835 5112 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_004/customer_004_1.dat

GEN_CUSTOMER_50.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 62623 63900 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_050/customer_050_1.dat

GEN_CUSTOMER_51.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 63901 65178 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_051/customer_051_1.dat

GEN_CUSTOMER_52.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 65179 66456 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_052/customer_052_1.dat

GEN_CUSTOMER_53.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 66457 67734 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_053/customer_053_1.dat

GEN_CUSTOMER_54.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 67735 69012 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_054/customer_054_1.dat

GEN_CUSTOMER_55.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 69013 70290 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_055/customer_055_1.dat

GEN_CUSTOMER_56.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 70291 71568 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_056/customer_056_1.dat

GEN_CUSTOMER_57.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 71569 72846 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_057/customer_057_1.dat

GEN_CUSTOMER_58.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 72847 74124 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_058/customer_058_1.dat

GEN_CUSTOMER_59.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 74125 75402 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_059/customer_059_1.dat

GEN_CUSTOMER_5.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 5113 6390 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_005/customer_005_1.dat

GEN_CUSTOMER_60.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 75403 76680 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_060/customer_060_1.dat

GEN_CUSTOMER_61.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 76681 77958 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_061/customer_061_1.dat

GEN_CUSTOMER_62.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 77959 79236 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_062/customer_062_1.dat

GEN_CUSTOMER_63.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 79237 80514 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_063/customer_063_1.dat

GEN_CUSTOMER_64.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 80515 81792 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_064/customer_064_1.dat

GEN_CUSTOMER_65.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 81793 83070 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_065/customer_065_1.dat

GEN_CUSTOMER_66.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 83071 84348 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_066/customer_066_1.dat

GEN_CUSTOMER_67.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 84349 85626 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_067/customer_067_1.dat

GEN_CUSTOMER_68.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 85627 86904 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_068/customer_068_1.dat

GEN_CUSTOMER_69.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 86905 88182 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_069/customer_069_1.dat

GEN_CUSTOMER_6.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 6391 7668 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_006/customer_006_1.dat

GEN_CUSTOMER_70.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 88183 89460 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_070/customer_070_1.dat

GEN_CUSTOMER_71.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 89461 90738 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_071/customer_071_1.dat

GEN_CUSTOMER_72.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 90739 92016 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_072/customer_072_1.dat

GEN_CUSTOMER_73.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 92017 93294 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_073/customer_073_1.dat

GEN_CUSTOMER_74.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 93295 94572 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_074/customer_074_1.dat

GEN_CUSTOMER_75.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 94573 95850 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_075/customer_075_1.dat

GEN_CUSTOMER_76.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 95851 97128 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_076/customer_076_1.dat

GEN_CUSTOMER_77.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 97129 98406 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_077/customer_077_1.dat

GEN_CUSTOMER_78.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 98407 99684 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_078/customer_078_1.dat

GEN_CUSTOMER_79.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 99685 100962 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_079/customer_079_1.dat

GEN_CUSTOMER_7.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 7669 8946 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_007/customer_007_1.dat

GEN_CUSTOMER_80.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 100963 102240 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_080/customer_080_1.dat

GEN_CUSTOMER_81.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 102241 103518 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_081/customer_081_1.dat

GEN_CUSTOMER_82.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 103519 104796 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_082/customer_082_1.dat

GEN_CUSTOMER_83.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 104797 106074 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_083/customer_083_1.dat

GEN_CUSTOMER_84.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 106075 107352 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_084/customer_084_1.dat

GEN_CUSTOMER_85.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 107353 108630 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_085/customer_085_1.dat

GEN_CUSTOMER_86.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 108631 109908 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_086/customer_086_1.dat

GEN_CUSTOMER_87.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 109909 111186 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_087/customer_087_1.dat

GEN_CUSTOMER_88.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 111187 112464 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_088/customer_088_1.dat

GEN_CUSTOMER_89.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 112465 113742 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_089/customer_089_1.dat

GEN_CUSTOMER_8.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 8947 10224 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_008/customer_008_1.dat

GEN_CUSTOMER_90.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 113743 115020 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_090/customer_090_1.dat

GEN_CUSTOMER_91.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 115021 116298 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_091/customer_091_1.dat

GEN_CUSTOMER_92.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 116299 117576 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_092/customer_092_1.dat

GEN_CUSTOMER_93.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 117577 118854 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_093/customer_093_1.dat

GEN_CUSTOMER_94.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 118855 120132 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_094/customer_094_1.dat

GEN_CUSTOMER_95.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 120133 121410 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_095/customer_095_1.dat

GEN_CUSTOMER_96.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 121411 122688 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_096/customer_096_1.dat

GEN_CUSTOMER_97.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 122689 123966 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_097/customer_097_1.dat

GEN_CUSTOMER_98.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 123967 125244 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_098/customer_098_1.dat

GEN_CUSTOMER_99.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 125245 126522 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_099/customer_099_1.dat

GEN_CUSTOMER_9.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 7 -r 10225 11502 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_009/customer_009_1.dat

GEN_DISTRICT_10.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 46009 51120 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_010/district_010_1.dat

GEN_DISTRICT_11.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 51121 56232 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_011/district_011_1.dat

GEN_DISTRICT_12.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 56233 61344 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_012/district_012_1.dat

GEN_DISTRICT_13.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 61345 66456 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_013/district_013_1.dat

GEN_DISTRICT_14.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 66457 71568 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_014/district_014_1.dat

GEN_DISTRICT_15.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 71569 76680 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_015/district_015_1.dat

GEN_DISTRICT_16.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 76681 81792 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_016/district_016_1.dat

GEN_DISTRICT_17.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 81793 86904 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_017/district_017_1.dat

GEN_DISTRICT_18.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 86905 92016 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_018/district_018_1.dat

GEN_DISTRICT_19.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 92017 97128 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_019/district_019_1.dat

GEN_DISTRICT_1.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 1 5112 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_001/district_001_1.dat

GEN_DISTRICT_20.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 97129 102240 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_020/district_020_1.dat

GEN_DISTRICT_21.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 102241 107352 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_021/district_021_1.dat

GEN_DISTRICT_22.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 107353 112464 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_022/district_022_1.dat

GEN_DISTRICT_23.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 112465 117576 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_023/district_023_1.dat

GEN_DISTRICT_24.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 117577 122688 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_024/district_024_1.dat

GEN_DISTRICT_25.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 122689 127800 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_025/district_025_1.dat

GEN_DISTRICT_26.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 127801 132912 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_026/district_026_1.dat

GEN_DISTRICT_27.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 132913 138024 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_027/district_027_1.dat

GEN_DISTRICT_28.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 138025 143136 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_028/district_028_1.dat

GEN_DISTRICT_29.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 143137 148248 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_029/district_029_1.dat

GEN_DISTRICT_2.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 5113 10224 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_002/district_002_1.dat

GEN_DISTRICT_30.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 148249 153360 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_030/district_030_1.dat

GEN_DISTRICT_31.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 153361 158472 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_031/district_031_1.dat

GEN_DISTRICT_32.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 158473 163584 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_032/district_032_1.dat

GEN_DISTRICT_33.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 163585 168696 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_033/district_033_1.dat

GEN_DISTRICT_34.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 168697 173808 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_034/district_034_1.dat

GEN_DISTRICT_35.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 173809 178920 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_035/district_035_1.dat

GEN_DISTRICT_36.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 178921 184032 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_036/district_036_1.dat

GEN_DISTRICT_3.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 10225 15336 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_003/district_003_1.dat

GEN_DISTRICT_4.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 15337 20448 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_004/district_004_1.dat

GEN_DISTRICT_5.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 20449 25560 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_005/district_005_1.dat

GEN_DISTRICT_6.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 25561 30672 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_006/district_006_1.dat

GEN_DISTRICT_7.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 30673 35784 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_007/district_007_1.dat

GEN_DISTRICT_8.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 35785 40896 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_008/district_008_1.dat

GEN_DISTRICT_9.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 4 -r 40897 46008 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_009/district_009_1.dat

GEN_HISTORY_10.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 46009 51120 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_010/history_010_1.dat

GEN_HISTORY_11.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 51121 56232 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_011/history_011_1.dat

GEN_HISTORY_12.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 56233 61344 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_012/history_012_1.dat

GEN_HISTORY_13.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 61345 66456 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_013/history_013_1.dat

GEN_HISTORY_14.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 66457 71568 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_014/history_014_1.dat

GEN_HISTORY_15.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 71569 76680 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_015/history_015_1.dat

GEN_HISTORY_16.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 76681 81792 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_016/history_016_1.dat

GEN_HISTORY_17.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 81793 86904 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_017/history_017_1.dat

GEN_HISTORY_18.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 86905 92016 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_018/history_018_1.dat

GEN_HISTORY_19.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 92017 97128 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_019/history_019_1.dat

GEN_HISTORY_1.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 1 5112 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_001/history_001_1.dat

GEN_HISTORY_20.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 97129 102240 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_020/history_020_1.dat

GEN_HISTORY_21.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 102241 107352 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_021/history_021_1.dat

GEN_HISTORY_22.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 107353 112464 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_022/history_022_1.dat

GEN_HISTORY_23.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 112465 117576 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_023/history_023_1.dat

GEN_HISTORY_24.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 117577 122688 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_024/history_024_1.dat

GEN_HISTORY_25.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 122689 127800 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_025/history_025_1.dat

GEN_HISTORY_26.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 127801 132912 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_026/history_026_1.dat

GEN_HISTORY_27.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 132913 138024 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_027/history_027_1.dat

GEN_HISTORY_28.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 138025 143136 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_028/history_028_1.dat

GEN_HISTORY_29.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 143137 148248 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_029/history_029_1.dat

GEN_HISTORY_2.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 5113 10224 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_002/history_002_1.dat

GEN_HISTORY_30.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 148249 153360 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_030/history_030_1.dat

GEN_HISTORY_31.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 153361 158472 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_031/history_031_1.dat

GEN_HISTORY_32.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 158473 163584 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_032/history_032_1.dat

GEN_HISTORY_33.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 163585 168696 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_033/history_033_1.dat

GEN_HISTORY_34.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 168697 173808 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_034/history_034_1.dat

GEN_HISTORY_35.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 173809 178920 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_035/history_035_1.dat

GEN_HISTORY_36.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 178921 184032 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_036/history_036_1.dat

GEN_HISTORY_3.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 10225 15336 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_003/history_003_1.dat

GEN_HISTORY_4.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 15337 20448 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_004/history_004_1.dat

GEN_HISTORY_5.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 20449 25560 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_005/history_005_1.dat

GEN_HISTORY_6.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 25561 30672 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_006/history_006_1.dat

GEN_HISTORY_7.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 30673 35784 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_007/history_007_1.dat

GEN_HISTORY_8.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 35785 40896 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_008/history_008_1.dat

GEN_HISTORY_9.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 8 -r 40897 46008 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_009/history_009_1.dat

GEN_ITEM_1.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 5 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat/item_1.dat

GEN_NEW_ORDER_10.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 46009 51120 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_010/neworder_010_1.dat

GEN_NEW_ORDER_11.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 51121 56232 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_011/neworder_011_1.dat

GEN_NEW_ORDER_12.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 56233 61344 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_012/neworder_012_1.dat

GEN_NEW_ORDER_13.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 61345 66456 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_013/neworder_013_1.dat

GEN_NEW_ORDER_14.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 66457 71568 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_014/neworder_014_1.dat

GEN_NEW_ORDER_15.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 71569 76680 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_015/neworder_015_1.dat

GEN_NEW_ORDER_16.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 76681 81792 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_016/neworder_016_1.dat

GEN_NEW_ORDER_17.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 81793 86904 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_017/neworder_017_1.dat

GEN_NEW_ORDER_18.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 86905 92016 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_018/neworder_018_1.dat

GEN_NEW_ORDER_19.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 92017 97128 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_019/neworder_019_1.dat

GEN_NEW_ORDER_1.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 1 5112 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_001/neworder_001_1.dat

GEN_NEW_ORDER_20.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 97129 102240 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_020/neworder_020_1.dat

GEN_NEW_ORDER_21.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 102241 107352 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_021/neworder_021_1.dat

GEN_NEW_ORDER_22.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 107353 112464 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_022/neworder_022_1.dat

GEN_NEW_ORDER_23.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 112465 117576 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_023/neworder_023_1.dat

GEN_NEW_ORDER_24.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 117577 122688 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_024/neworder_024_1.dat

GEN_NEW_ORDER_25.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 122689 127800 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_025/neworder_025_1.dat

GEN_NEW_ORDER_26.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 127801 132912 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_026/neworder_026_1.dat

GEN_NEW_ORDER_27.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 132913 138024 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_027/neworder_027_1.dat

GEN_NEW_ORDER_28.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 138025 143136 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_028/neworder_028_1.dat

GEN_NEW_ORDER_29.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 143137 148248 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_029/neworder_029_1.dat

GEN_NEW_ORDER_2.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 5113 10224 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_002/neworder_002_1.dat

GEN_NEW_ORDER_30.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 148249 153360 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_030/neworder_030_1.dat

GEN_NEW_ORDER_31.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 153361 158472 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_031/neworder_031_1.dat

GEN_NEW_ORDER_32.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 158473 163584 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_032/neworder_032_1.dat

GEN_NEW_ORDER_33.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 163585 168696 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_033/neworder_033_1.dat

GEN_NEW_ORDER_34.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 168697 173808 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_034/neworder_034_1.dat

GEN_NEW_ORDER_35.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 173809 178920 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_035/neworder_035_1.dat

GEN_NEW_ORDER_36.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 178921 184032 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_036/neworder_036_1.dat

GEN_NEW_ORDER_3.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 10225 15336 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_003/neworder_003_1.dat

GEN_NEW_ORDER_4.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 15337 20448 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_004/neworder_004_1.dat

GEN_NEW_ORDER_5.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 20449 25560 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_005/neworder_005_1.dat

GEN_NEW_ORDER_6.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 25561 30672 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_006/neworder_006_1.dat

GEN_NEW_ORDER_7.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 30673 35784 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_007/neworder_007_1.dat

GEN_NEW_ORDER_8.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 35785 40896 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_008/neworder_008_1.dat

GEN_NEW_ORDER_9.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 11 -r 40897 46008 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_009/neworder_009_1.dat

GEN_ORDERS_10.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 23005 25560 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_005/orders_005_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_010/orderline_010_1.dat

GEN_ORDERS_11.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 25561 28116 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_006/orders_006_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_011/orderline_011_1.dat

GEN_ORDERS_12.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 28117 30672 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_006/orders_006_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_012/orderline_012_1.dat

GEN_ORDERS_13.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 30673 33228 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_007/orders_007_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_013/orderline_013_1.dat

GEN_ORDERS_14.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 33229 35784 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_007/orders_007_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_014/orderline_014_1.dat

GEN_ORDERS_15.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 35785 38340 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_008/orders_008_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_015/orderline_015_1.dat

GEN_ORDERS_16.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 38341 40896 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_008/orders_008_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_016/orderline_016_1.dat

GEN_ORDERS_17.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 40897 43452 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_009/orders_009_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_017/orderline_017_1.dat

GEN_ORDERS_18.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 43453 46008 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_009/orders_009_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_018/orderline_018_1.dat

GEN_ORDERS_19.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 46009 48564 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_010/orders_010_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_019/orderline_019_1.dat

GEN_ORDERS_1.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 1 2556 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_001/orders_001_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_001/orderline_001_1.dat

GEN_ORDERS_20.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 48565 51120 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_010/orders_010_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_020/orderline_020_1.dat

GEN_ORDERS_21.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 51121 53676 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_011/orders_011_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_021/orderline_021_1.dat

GEN_ORDERS_22.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 53677 56232 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_011/orders_011_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_022/orderline_022_1.dat

GEN_ORDERS_23.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 56233 58788 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_012/orders_012_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_023/orderline_023_1.dat

GEN_ORDERS_24.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 58789 61344 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_012/orders_012_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_024/orderline_024_1.dat

GEN_ORDERS_25.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 61345 63900 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_013/orders_013_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_025/orderline_025_1.dat

GEN_ORDERS_26.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 63901 66456 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_013/orders_013_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_026/orderline_026_1.dat

GEN_ORDERS_27.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 66457 69012 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_014/orders_014_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_027/orderline_027_1.dat

GEN_ORDERS_28.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 69013 71568 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_014/orders_014_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_028/orderline_028_1.dat

GEN_ORDERS_29.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 71569 74124 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_015/orders_015_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_029/orderline_029_1.dat

GEN_ORDERS_2.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 2557 5112 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_001/orders_001_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_002/orderline_002_1.dat

GEN_ORDERS_30.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 74125 76680 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_015/orders_015_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_030/orderline_030_1.dat

GEN_ORDERS_31.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 76681 79236 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_016/orders_016_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_031/orderline_031_1.dat

GEN_ORDERS_32.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 79237 81792 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_016/orders_016_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_032/orderline_032_1.dat

GEN_ORDERS_33.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 81793 84348 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_017/orders_017_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_033/orderline_033_1.dat

GEN_ORDERS_34.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 84349 86904 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_017/orders_017_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_034/orderline_034_1.dat

GEN_ORDERS_35.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 86905 89460 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_018/orders_018_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_035/orderline_035_1.dat

GEN_ORDERS_36.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 89461 92016 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_018/orders_018_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_036/orderline_036_1.dat

GEN_ORDERS_37.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 92017 94572 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_019/orders_019_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_037/orderline_037_1.dat

GEN_ORDERS_38.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 94573 97128 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_019/orders_019_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_038/orderline_038_1.dat

GEN_ORDERS_39.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 97129 99684 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_020/orders_020_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_039/orderline_039_1.dat

GEN_ORDERS_3.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 5113 7668 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_002/orders_002_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_003/orderline_003_1.dat

GEN_ORDERS_40.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 99685 102240 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_020/orders_020_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_040/orderline_040_1.dat

GEN_ORDERS_41.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 102241 104796 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_021/orders_021_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_041/orderline_041_1.dat

GEN_ORDERS_42.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 104797 107352 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_021/orders_021_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_042/orderline_042_1.dat

GEN_ORDERS_43.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 107353 109908 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_022/orders_022_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_043/orderline_043_1.dat

GEN_ORDERS_44.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 109909 112464 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_022/orders_022_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_044/orderline_044_1.dat

GEN_ORDERS_45.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 112465 115020 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_023/orders_023_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_045/orderline_045_1.dat

GEN_ORDERS_46.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 115021 117576 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_023/orders_023_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_046/orderline_046_1.dat

GEN_ORDERS_47.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 117577 120132 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_024/orders_024_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_047/orderline_047_1.dat

GEN_ORDERS_48.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 120133 122688 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_024/orders_024_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_048/orderline_048_1.dat

GEN_ORDERS_49.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 122689 125244 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_025/orders_025_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_049/orderline_049_1.dat

GEN_ORDERS_4.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 7669 10224 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_002/orders_002_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_004/orderline_004_1.dat

GEN_ORDERS_50.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 125245 127800 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_025/orders_025_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_050/orderline_050_1.dat

GEN_ORDERS_51.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 127801 130356 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_026/orders_026_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_051/orderline_051_1.dat

GEN_ORDERS_52.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 130357 132912 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_026/orders_026_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_052/orderline_052_1.dat

GEN_ORDERS_53.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 132913 135468 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_027/orders_027_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_053/orderline_053_1.dat

GEN_ORDERS_54.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 135469 138024 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_027/orders_027_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_054/orderline_054_1.dat

GEN_ORDERS_55.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 138025 140580 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_028/orders_028_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_055/orderline_055_1.dat

GEN_ORDERS_56.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 140581 143136 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_028/orders_028_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_056/orderline_056_1.dat

GEN_ORDERS_57.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 143137 145692 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_029/orders_029_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_057/orderline_057_1.dat

GEN_ORDERS_58.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 145693 148248 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_029/orders_029_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_058/orderline_058_1.dat

GEN_ORDERS_59.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 148249 150804 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_030/orders_030_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_059/orderline_059_1.dat

GEN_ORDERS_5.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 10225 12780 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_003/orders_003_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_005/orderline_005_1.dat

GEN_ORDERS_60.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 150805 153360 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_030/orders_030_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_060/orderline_060_1.dat

GEN_ORDERS_61.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 153361 155916 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_031/orders_031_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_061/orderline_061_1.dat

GEN_ORDERS_62.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 155917 158472 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_031/orders_031_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_062/orderline_062_1.dat

GEN_ORDERS_63.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 158473 161028 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_032/orders_032_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_063/orderline_063_1.dat

GEN_ORDERS_64.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 161029 163584 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_032/orders_032_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_064/orderline_064_1.dat

GEN_ORDERS_65.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 163585 166140 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_033/orders_033_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_065/orderline_065_1.dat

GEN_ORDERS_66.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 166141 168696 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_033/orders_033_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_066/orderline_066_1.dat

GEN_ORDERS_67.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 168697 171252 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_034/orders_034_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_067/orderline_067_1.dat

GEN_ORDERS_68.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 171253 173808 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_034/orders_034_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_068/orderline_068_1.dat

GEN_ORDERS_69.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 173809 176364 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_035/orders_035_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_069/orderline_069_1.dat

GEN_ORDERS_6.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 12781 15336 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_003/orders_003_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_006/orderline_006_1.dat

GEN_ORDERS_70.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 176365 178920 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_035/orders_035_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_070/orderline_070_1.dat

GEN_ORDERS_71.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 178921 181476 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_036/orders_036_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_071/orderline_071_1.dat

GEN_ORDERS_72.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 181477 184032 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_036/orders_036_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_072/orderline_072_1.dat

GEN_ORDERS_7.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 15337 17892 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_004/orders_004_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_007/orderline_007_1.dat

GEN_ORDERS_8.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 17893 20448 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_004/orders_004_2.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_008/orderline_008_1.dat

GEN_ORDERS_9.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 9 -r 20449 23004 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_005/orders_005_1.dat -f2 /autobench/sources/db2_tpcc-1/flat_files/flat_009/orderline_009_1.dat

GEN_STOCK_100.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 126523 127800 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_100/stock_100_1.dat

GEN_STOCK_101.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 127801 129078 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_101/stock_101_1.dat

GEN_STOCK_102.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 129079 130356 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_102/stock_102_1.dat

GEN_STOCK_103.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 130357 131634 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_103/stock_103_1.dat

GEN_STOCK_104.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 131635 132912 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_104/stock_104_1.dat

GEN_STOCK_105.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 132913 134190 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_105/stock_105_1.dat

GEN_STOCK_106.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 134191 135468 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_106/stock_106_1.dat

GEN_STOCK_107.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 135469 136746 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_107/stock_107_1.dat

GEN_STOCK_108.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 136747 138024 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_108/stock_108_1.dat

GEN_STOCK_109.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 138025 139302 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_109/stock_109_1.dat

GEN_STOCK_10.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 11503 12780 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_010/stock_010_1.dat

GEN_STOCK_110.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 139303 140580 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_110/stock_110_1.dat

GEN_STOCK_111.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 140581 141858 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_111/stock_111_1.dat

GEN_STOCK_112.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 141859 143136 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_112/stock_112_1.dat

GEN_STOCK_113.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 143137 144414 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_113/stock_113_1.dat

GEN_STOCK_114.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 144415 145692 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_114/stock_114_1.dat

GEN_STOCK_115.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 145693 146970 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_115/stock_115_1.dat

GEN_STOCK_116.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 146971 148248 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_116/stock_116_1.dat

GEN_STOCK_117.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 148249 149526 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_117/stock_117_1.dat

GEN_STOCK_118.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 149527 150804 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_118/stock_118_1.dat

GEN_STOCK_119.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 150805 152082 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_119/stock_119_1.dat

GEN_STOCK_11.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 12781 14058 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_011/stock_011_1.dat

GEN_STOCK_120.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 152083 153360 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_120/stock_120_1.dat

GEN_STOCK_121.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 153361 154638 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_121/stock_121_1.dat

GEN_STOCK_122.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 154639 155916 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_122/stock_122_1.dat

GEN_STOCK_123.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 155917 157194 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_123/stock_123_1.dat

GEN_STOCK_124.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 157195 158472 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_124/stock_124_1.dat

GEN_STOCK_125.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 158473 159750 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_125/stock_125_1.dat

GEN_STOCK_126.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 159751 161028 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_126/stock_126_1.dat

GEN_STOCK_127.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 161029 162306 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_127/stock_127_1.dat

GEN_STOCK_128.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 162307 163584 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_128/stock_128_1.dat

GEN_STOCK_129.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 163585 164862 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_129/stock_129_1.dat

GEN_STOCK_12.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 14059 15336 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_012/stock_012_1.dat

GEN_STOCK_130.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 164863 166140 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_130/stock_130_1.dat

GEN_STOCK_131.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 166141 167418 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_131/stock_131_1.dat

GEN_STOCK_132.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 167419 168696 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_132/stock_132_1.dat

GEN_STOCK_133.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 168697 169974 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_133/stock_133_1.dat

GEN_STOCK_134.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 169975 171252 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_134/stock_134_1.dat

GEN_STOCK_135.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 171253 172530 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_135/stock_135_1.dat

GEN_STOCK_136.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 172531 173808 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_136/stock_136_1.dat

GEN_STOCK_137.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 173809 175086 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_137/stock_137_1.dat

GEN_STOCK_138.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 175087 176364 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_138/stock_138_1.dat

GEN_STOCK_139.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 176365 177642 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_139/stock_139_1.dat

GEN_STOCK_13.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 15337 16614 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_013/stock_013_1.dat

GEN_STOCK_140.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 177643 178920 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_140/stock_140_1.dat

GEN_STOCK_141.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 178921 180198 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_141/stock_141_1.dat

GEN_STOCK_142.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 180199 181476 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_142/stock_142_1.dat

GEN_STOCK_143.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 181477 182754 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_143/stock_143_1.dat

GEN_STOCK_144.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 182755 184032 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_144/stock_144_1.dat

GEN_STOCK_14.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 16615 17892 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_014/stock_014_1.dat

GEN_STOCK_15.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 17893 19170 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_015/stock_015_1.dat

GEN_STOCK_16.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 19171 20448 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_016/stock_016_1.dat

GEN_STOCK_17.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 20449 21726 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_017/stock_017_1.dat

GEN_STOCK_18.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 21727 23004 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_018/stock_018_1.dat

GEN_STOCK_19.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 23005 24282 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_019/stock_019_1.dat

GEN_STOCK_1.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 1 1278 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_001/stock_001_1.dat

GEN_STOCK_20.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 24283 25560 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_020/stock_020_1.dat

GEN_STOCK_21.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 25561 26838 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_021/stock_021_1.dat

GEN_STOCK_22.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 26839 28116 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_022/stock_022_1.dat

GEN_STOCK_23.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 28117 29394 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_023/stock_023_1.dat

GEN_STOCK_24.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 29395 30672 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_024/stock_024_1.dat

GEN_STOCK_25.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 30673 31950 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_025/stock_025_1.dat

GEN_STOCK_26.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 31951 33228 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_026/stock_026_1.dat

GEN_STOCK_27.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 33229 34506 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_027/stock_027_1.dat

GEN_STOCK_28.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 34507 35784 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_028/stock_028_1.dat

GEN_STOCK_29.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 35785 37062 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_029/stock_029_1.dat

GEN_STOCK_2.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 1279 2556 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_002/stock_002_1.dat

GEN_STOCK_30.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 37063 38340 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_030/stock_030_1.dat

GEN_STOCK_31.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 38341 39618 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_031/stock_031_1.dat

GEN_STOCK_32.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 39619 40896 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_032/stock_032_1.dat

GEN_STOCK_33.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 40897 42174 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_033/stock_033_1.dat

GEN_STOCK_34.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 42175 43452 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_034/stock_034_1.dat

GEN_STOCK_35.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 43453 44730 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_035/stock_035_1.dat

GEN_STOCK_36.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 44731 46008 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_036/stock_036_1.dat

GEN_STOCK_37.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 46009 47286 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_037/stock_037_1.dat

GEN_STOCK_38.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 47287 48564 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_038/stock_038_1.dat

GEN_STOCK_39.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 48565 49842 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_039/stock_039_1.dat

GEN_STOCK_3.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 2557 3834 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_003/stock_003_1.dat

GEN_STOCK_40.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 49843 51120 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_040/stock_040_1.dat

GEN_STOCK_41.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 51121 52398 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_041/stock_041_1.dat

GEN_STOCK_42.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 52399 53676 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_042/stock_042_1.dat

GEN_STOCK_43.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 53677 54954 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_043/stock_043_1.dat

GEN_STOCK_44.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 54955 56232 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_044/stock_044_1.dat

GEN_STOCK_45.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 56233 57510 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_045/stock_045_1.dat

GEN_STOCK_46.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 57511 58788 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_046/stock_046_1.dat

GEN_STOCK_47.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 58789 60066 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_047/stock_047_1.dat

GEN_STOCK_48.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 60067 61344 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_048/stock_048_1.dat

GEN_STOCK_49.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 61345 62622 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_049/stock_049_1.dat

GEN_STOCK_4.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 3835 5112 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_004/stock_004_1.dat

GEN_STOCK_50.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 62623 63900 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_050/stock_050_1.dat

GEN_STOCK_51.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 63901 65178 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_051/stock_051_1.dat

GEN_STOCK_52.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 65179 66456 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_052/stock_052_1.dat

GEN_STOCK_53.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 66457 67734 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_053/stock_053_1.dat

GEN_STOCK_54.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 67735 69012 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_054/stock_054_1.dat

GEN_STOCK_55.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 69013 70290 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_055/stock_055_1.dat

GEN_STOCK_56.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 70291 71568 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_056/stock_056_1.dat

GEN_STOCK_57.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 71569 72846 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_057/stock_057_1.dat

GEN_STOCK_58.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 72847 74124 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_058/stock_058_1.dat

GEN_STOCK_59.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 74125 75402 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_059/stock_059_1.dat

GEN_STOCK_5.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 5113 6390 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_005/stock_005_1.dat

GEN_STOCK_60.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 75403 76680 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_060/stock_060_1.dat

GEN_STOCK_61.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 76681 77958 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_061/stock_061_1.dat

GEN_STOCK_62.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 77959 79236 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_062/stock_062_1.dat

GEN_STOCK_63.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 79237 80514 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_063/stock_063_1.dat

GEN_STOCK_64.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 80515 81792 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_064/stock_064_1.dat

GEN_STOCK_65.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 81793 83070 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_065/stock_065_1.dat

GEN_STOCK_66.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 83071 84348 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_066/stock_066_1.dat

GEN_STOCK_67.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 84349 85626 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_067/stock_067_1.dat

GEN_STOCK_68.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 85627 86904 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_068/stock_068_1.dat

GEN_STOCK_69.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 86905 88182 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_069/stock_069_1.dat

GEN_STOCK_6.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 6391 7668 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_006/stock_006_1.dat

GEN_STOCK_70.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 88183 89460 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_070/stock_070_1.dat

GEN_STOCK_71.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 89461 90738 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_071/stock_071_1.dat

GEN_STOCK_72.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 90739 92016 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_072/stock_072_1.dat

GEN_STOCK_73.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 92017 93294 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_073/stock_073_1.dat

GEN_STOCK_74.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 93295 94572 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_074/stock_074_1.dat

GEN_STOCK_75.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 94573 95850 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_075/stock_075_1.dat

GEN_STOCK_76.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 95851 97128 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_076/stock_076_1.dat

GEN_STOCK_77.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 97129 98406 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_077/stock_077_1.dat

GEN_STOCK_78.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 98407 99684 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_078/stock_078_1.dat

GEN_STOCK_79.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 99685 100962 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_079/stock_079_1.dat

GEN_STOCK_7.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 7669 8946 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_007/stock_007_1.dat

GEN_STOCK_80.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 100963 102240 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_080/stock_080_1.dat

GEN_STOCK_81.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 102241 103518 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_081/stock_081_1.dat

GEN_STOCK_82.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 103519 104796 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_082/stock_082_1.dat

GEN_STOCK_83.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 104797 106074 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_083/stock_083_1.dat

GEN_STOCK_84.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 106075 107352 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_084/stock_084_1.dat

GEN_STOCK_85.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 107353 108630 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_085/stock_085_1.dat

GEN_STOCK_86.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 108631 109908 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_086/stock_086_1.dat

GEN_STOCK_87.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 109909 111186 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_087/stock_087_1.dat

GEN_STOCK_88.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 111187 112464 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_088/stock_088_1.dat

GEN_STOCK_89.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 112465 113742 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_089/stock_089_1.dat

GEN_STOCK_8.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 8947 10224 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_008/stock_008_1.dat

GEN_STOCK_90.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 113743 115020 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_090/stock_090_1.dat

GEN_STOCK_91.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 115021 116298 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_091/stock_091_1.dat

GEN_STOCK_92.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 116299 117576 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_092/stock_092_1.dat

GEN_STOCK_93.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 117577 118854 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_093/stock_093_1.dat

GEN_STOCK_94.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 118855 120132 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_094/stock_094_1.dat

GEN_STOCK_95.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 120133 121410 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_095/stock_095_1.dat

GEN_STOCK_96.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 121411 122688 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_096/stock_096_1.dat

GEN_STOCK_97.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 122689 123966 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_097/stock_097_1.dat

GEN_STOCK_98.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 123967 125244 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_098/stock_098_1.dat

GEN_STOCK_99.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 125245 126522 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_099/stock_099_1.dat

GEN_STOCK_9.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 6 -r 10225 11502 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_009/stock_009_1.dat

GEN_WAREHOUSE_10.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 46009 51120 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_010/warehouse_010_1.dat

GEN_WAREHOUSE_11.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 51121 56232 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_011/warehouse_011_1.dat

GEN_WAREHOUSE_12.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 56233 61344 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_012/warehouse_012_1.dat

GEN_WAREHOUSE_13.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 61345 66456 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_013/warehouse_013_1.dat

GEN_WAREHOUSE_14.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 66457 71568 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_014/warehouse_014_1.dat

GEN_WAREHOUSE_15.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 71569 76680 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_015/warehouse_015_1.dat

GEN_WAREHOUSE_16.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 76681 81792 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_016/warehouse_016_1.dat

GEN_WAREHOUSE_17.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 81793 86904 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_017/warehouse_017_1.dat

GEN_WAREHOUSE_18.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 86905 92016 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_018/warehouse_018_1.dat

GEN_WAREHOUSE_19.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 92017 97128 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_019/warehouse_019_1.dat

GEN_WAREHOUSE_1.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 1 5112 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_001/warehouse_001_1.dat

GEN_WAREHOUSE_20.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 97129 102240 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_020/warehouse_020_1.dat

GEN_WAREHOUSE_21.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 102241 107352 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_021/warehouse_021_1.dat

GEN_WAREHOUSE_22.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 107353 112464 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_022/warehouse_022_1.dat

GEN_WAREHOUSE_23.sh

/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 112465 117576 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_023/warehouse_023_1.dat

GEN_WAREHOUSE_24.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 117577 122688 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_024/warehouse_024_1.dat

GEN_WAREHOUSE_25.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 122689 127800 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_025/warehouse_025_1.dat

GEN_WAREHOUSE_26.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 127801 132912 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_026/warehouse_026_1.dat

GEN_WAREHOUSE_27.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 132913 138024 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_027/warehouse_027_1.dat

GEN_WAREHOUSE_28.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 138025 143136 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_028/warehouse_028_1.dat

GEN_WAREHOUSE_29.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 143137 148248 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_029/warehouse_029_1.dat

GEN_WAREHOUSE_2.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 5113 10224 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_002/warehouse_002_1.dat

GEN_WAREHOUSE_30.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 148249 153360 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_030/warehouse_030_1.dat

GEN_WAREHOUSE_31.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 153361 158472 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_031/warehouse_031_1.dat

GEN_WAREHOUSE_32.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 158473 163584 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_032/warehouse_032_1.dat

GEN_WAREHOUSE_33.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 163585 168696 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_033/warehouse_033_1.dat

GEN_WAREHOUSE_34.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 168697 173808 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_034/warehouse_034_1.dat

GEN_WAREHOUSE_35.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 173809 178920 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_035/warehouse_035_1.dat

GEN_WAREHOUSE_36.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 178921 184032 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_036/warehouse_036_1.dat

GEN_WAREHOUSE_3.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 10225 15336 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_003/warehouse_003_1.dat

GEN_WAREHOUSE_4.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 15337 20448 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_004/warehouse_004_1.dat

GEN_WAREHOUSE_5.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 20449 25560 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_005/warehouse_005_1.dat

GEN_WAREHOUSE_6.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 25561 30672 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_006/warehouse_006_1.dat

GEN_WAREHOUSE_7.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 30673 35784 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_007/warehouse_007_1.dat

GEN_WAREHOUSE_8.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 35785 40896 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_008/warehouse_008_1.dat

GEN_WAREHOUSE_9.sh
/autobench/sources/db2_tpcc-1/users/tpcc/tpc-c.ibm/dbgen/gendata -t 3 -r 40897 46008 -f1 /autobench/sources/db2_tpcc-1/flat_files/flat_009/warehouse_009_1.dat

LOAD_CUSTOMER100_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_100/customer_100_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER100;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER101_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER101 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_101/customer_101_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER101;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER10_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/customer_010_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER10;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER102_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER102 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_102/customer_102_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER102;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER103_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER103 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_103/customer_103_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER103;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER104_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER104 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_104/customer_104_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER104;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER105_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER105 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_105/customer_105_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER105;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER106_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER106 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_106/customer_106_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER106;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER107_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER107 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_107/customer_107_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER107;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER108_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER108 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_108/customer_108_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER108;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER109_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER109 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_109/customer_109_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER109;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER110_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER110 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_110/customer_110_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER110;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER111_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER111 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_111/customer_111_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER111;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER11_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/customer_011_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER11;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER112_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER112 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_112/customer_112_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER112;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER113_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER113 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_113/customer_113_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER113;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER114_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER114 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_114/customer_114_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER114;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER115_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER115 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_115/customer_115_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER115;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER116_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER116 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_116/customer_116_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER116;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER117_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER117 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_117/customer_117_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER117;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER118_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER118 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_118/customer_118_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER118;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER119_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER119 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_119/customer_119_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER119;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER1_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/customer_001_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER1;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER120_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER120 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_120/customer_120_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER120;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER121_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER121 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_121/customer_121_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER121;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER12_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/customer_012_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER12;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER122_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER122 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_122/customer_122_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER122;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER123_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER123 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_123/customer_123_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER123;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER124_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER124 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_124/customer_124_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER124;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER125_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER125 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_125/customer_125_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER125;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER126_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER126 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_126/customer_126_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER126;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER127_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER127 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_127/customer_127_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER127;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER128_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER128 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_128/customer_128_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER128;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER129_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER129 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_129/customer_129_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER129;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER130_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER130 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_130/customer_130_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER130;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER131_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER131 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_131/customer_131_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER131;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER13_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/customer_013_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER13;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER132_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER132 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_132/customer_132_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER132;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER133_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER133 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_133/customer_133_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER133;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER134_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER134 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_134/customer_134_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER134;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER135_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER135 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_135/customer_135_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER135;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER136_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER136 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_136/customer_136_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER136;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER137_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER137 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_137/customer_137_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER137;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER138_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER138 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_138/customer_138_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER138;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER139_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER139 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_139/customer_139_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER139;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER140_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER140 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_140/customer_140_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER140;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER141_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER141 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_141/customer_141_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER141;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER14_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/customer_014_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER14;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER142_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER142 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_142/customer_142_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER142;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER143_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER143 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_143/customer_143_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER143;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER144_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER144 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_144/customer_144_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER144;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER15_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/customer_015_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER15;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER16_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/customer_016_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER16;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER17_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/customer_017_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER17;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER18_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/customer_018_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER18;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER19_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/customer_019_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER19;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER20_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/customer_020_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER20;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER21_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/customer_021_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER21;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER2_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/customer_002_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER2;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER22_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/customer_022_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER22;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER23_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/customer_023_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER23;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER24_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/customer_024_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER24;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER25_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/customer_025_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER25;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER26_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/customer_026_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER26;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER27_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/customer_027_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER27;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER28_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/customer_028_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER28;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER29_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/customer_029_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER29;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER30_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/customer_030_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER30;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER31_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/customer_031_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER31;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER3_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/customer_003_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER3;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER32_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/customer_032_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER32;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER33_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/customer_033_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER33;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER34_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/customer_034_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER34;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER35_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/customer_035_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER35;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER36_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER36 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/customer_036_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER36;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER37_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER37 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_037/customer_037_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER37;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER38_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER38 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_038/customer_038_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER38;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER39_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER39 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_039/customer_039_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER39;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER40_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER40 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_040/customer_040_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER40;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER41_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER41 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_041/customer_041_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER41;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER4_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/customer_004_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER4;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER42_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER42 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_042/customer_042_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER42;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER43_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER43 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_043/customer_043_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER43;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER44_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER44 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_044/customer_044_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER44;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER45_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER45 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_045/customer_045_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER45;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER46_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER46 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_046/customer_046_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER46;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER47_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER47 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_047/customer_047_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER47;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER48_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER48 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_048/customer_048_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER48;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER49_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER49 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_049/customer_049_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER49;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER50_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER50 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_050/customer_050_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER50;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER51_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER51 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_051/customer_051_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER51;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER5_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/customer_005_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER5;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER52_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER52 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_052/customer_052_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER52;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER53_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER53 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_053/customer_053_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER53;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER54_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER54 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_054/customer_054_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER54;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER55_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER55 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_055/customer_055_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER55;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER56_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER56 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_056/customer_056_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER56;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER57_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER57 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_057/customer_057_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER57;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER58_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER58 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_058/customer_058_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER58;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER59_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER59 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_059/customer_059_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER59;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER60_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER60 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_060/customer_060_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER60;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER61_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER61 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_061/customer_061_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER61;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER6_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/customer_006_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER6;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER62_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER62 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_062/customer_062_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER62;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER63_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER63 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_063/customer_063_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER63;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER64_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER64 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_064/customer_064_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER64;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER65_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER65 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_065/customer_065_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER65;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER66_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER66 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_066/customer_066_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER66;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER67_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER67 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_067/customer_067_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER67;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER68_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER68 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_068/customer_068_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER68;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER69_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER69 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_069/customer_069_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER69;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER70_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER70 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_070/customer_070_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER70;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER71_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER71 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_071/customer_071_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER71;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER7_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/customer_007_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER7;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER72_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER72 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_072/customer_072_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER72;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER73_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER73 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_073/customer_073_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER73;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER74_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER74 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_074/customer_074_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER74;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER75_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER75 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_075/customer_075_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER75;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER76_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER76 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_076/customer_076_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER76;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER77_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER77 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_077/customer_077_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER77;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER78_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER78 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_078/customer_078_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER78;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER79_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER79 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_079/customer_079_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER79;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER80_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER80 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_080/customer_080_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER80;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER81_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER81 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_081/customer_081_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER81;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER8_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/customer_008_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER8;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER82_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER82 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_082/customer_082_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER82;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER83_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER83 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_083/customer_083_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER83;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER84_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER84 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_084/customer_084_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER84;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER85_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER85 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_085/customer_085_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER85;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER86_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER86 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_086/customer_086_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER86;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER87_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER87 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_087/customer_087_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER87;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER88_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER88 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_088/customer_088_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER88;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER89_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER89 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_089/customer_089_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER89;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER90_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER90 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_090/customer_090_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER90;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER91_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER91 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_091/customer_091_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER91;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER9_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/customer_009_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER9;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER92_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER92 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_092/customer_092_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER92;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER93_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER93 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_093/customer_093_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER93;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER94_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER94 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_094/customer_094_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER94;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER95_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER95 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_095/customer_095_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER95;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER96_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER96 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_096/customer_096_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER96;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER97_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER97 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_097/customer_097_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER97;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER98_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER98 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_098/customer_098_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER98;
COMMIT WORK;
CONNECT RESET;
```

LOAD_CUSTOMER99_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER99 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_099/customer_099_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 38340000 INSERT INTO
CUSTOMER99;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT10_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/district_010_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT10;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT11_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/district_011_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT11;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT1_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/district_001_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT1;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT12_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/district_012_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT12;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT13_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/district_013_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT13;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT14_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/district_014_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT14;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT15_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/district_015_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT15;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT16_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/district_016_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT16;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT17_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/district_017_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
```

```
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT17;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT18_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/district_018_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT18;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT19_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/district_019_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT19;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT20_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/district_020_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT20;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT21_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/district_021_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT21;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT2_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/district_002_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT2;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT22_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/district_022_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT22;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT23_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/district_023_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT23;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT24_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/district_024_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT24;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT25_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/district_025_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT25;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT26_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/district_026_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT26;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT27_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/district_027_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT27;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT28_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/district_028_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT28;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT29_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/district_029_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT29;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT30_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/district_030_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT30;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT31_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/district_031_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT31;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT3_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/district_003_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT3;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT32_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/district_032_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT32;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT33_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/district_033_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT33;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT34_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/district_034_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT34;
```

```
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT35_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/district_035_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT35;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT36_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/district_036_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT36;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT4_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/district_004_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT4;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT5_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/district_005_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT5;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT6_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/district_006_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT6;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT7_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/district_007_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT7;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT8_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/district_008_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT8;
COMMIT WORK;
CONNECT RESET;
```

LOAD_DISTRICT9_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/district_009_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT9;
COMMIT WORK;
CONNECT RESET;
```

LOAD_HISTORY10.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/history_010_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY10
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY11.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/history_011_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY11
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY12.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/history_012_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY12
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY13.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/history_013_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY13
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY14.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/history_014_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY14
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY15.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/history_015_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY15
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY16.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/history_016_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY16
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY17.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/history_017_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY17
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY18.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/history_018_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY18
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY19.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/history_019_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY19
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY1.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/history_001_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY1
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY20.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/history_020_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY20
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY21.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/history_021_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY21
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY22.ddl

```
connect to TPCC in share mode;
```

```
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/history_022_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY22
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY23.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/history_023_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY23
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY24.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/history_024_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY24
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY25.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/history_025_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY25
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY26.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/history_026_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY26
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY27.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/history_027_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY27
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY28.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/history_028_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY28
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY29.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/history_029_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY29
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY2.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/history_002_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY2
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY30.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/history_030_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY30
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY31.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/history_031_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY31
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY32.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/history_032_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY32
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY33.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/history_033_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY33
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY34.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/history_034_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY34
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY35.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/history_035_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY35
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY36.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/history_036_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY36
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY3.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/history_003_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY3
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY4.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/history_004_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY4
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY5.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/history_005_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY5
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY6.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/history_006_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY6
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY7.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/history_007_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY7
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY8.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/history_008_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY8
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_HISTORY9.ddl

```
connect to TPCC in share mode;
LOAD FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/history_009_1.dat OF
DEL MODIFIED BY COLDEL| KEEPBLANKS FASTPARSE REPLACE INTO HISTORY9
NONRECOVERABLE DATA BUFFER 16000 CPU_PARALLELISM 4 ;
connect reset;
```

LOAD_ITEM_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat/item_1.dat OF DEL
MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 1000 INSERT INTO ITEM;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER10_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/neworder_010_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER10;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER11_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/neworder_011_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER11;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER1_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/neworder_001_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER1;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER12_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/neworder_012_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER12;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER13_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/neworder_013_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER13;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER14_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/neworder_014_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER14;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER15_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/neworder_015_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER15;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER16_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/neworder_016_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER16;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER17_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/neworder_017_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER17;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER18_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/neworder_018_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER18;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER19_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/neworder_019_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER19;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER20_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/neworder_020_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER20;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER21_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/neworder_021_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER21;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER2_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/neworder_002_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER2;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER22_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/neworder_022_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER22;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER23_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/neworder_023_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER23;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER24_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/neworder_024_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER24;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER25_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/neworder_025_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER25;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER26_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/neworder_026_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER26;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER27_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
```

```
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/neworder_027_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER27;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER28_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/neworder_028_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER28;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER29_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/neworder_029_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER29;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER30_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/neworder_030_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER30;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER31_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/neworder_031_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER31;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER3_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/neworder_003_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER3;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER32_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/neworder_032_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER32;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER33_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/neworder_033_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER33;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER34_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/neworder_034_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER34;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER35_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/neworder_035_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER35;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER36_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/neworder_036_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER36;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER4_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/neworder_004_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER4;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER5_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/neworder_005_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER5;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER6_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/neworder_006_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER6;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER7_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/neworder_007_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER7;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER8_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/neworder_008_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER8;
COMMIT WORK;
CONNECT RESET;
```

LOAD_NEW_ORDER9_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/neworder_009_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDER9;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE10_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/orderline_010_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE10;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE11_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/orderline_011_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE11;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE1_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
```

```
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/orderline_001_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE1;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE12_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/orderline_012_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE12;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE13_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/orderline_013_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE13;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE14_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/orderline_014_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE14;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE15_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/orderline_015_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE15;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE16_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/orderline_016_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE16;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE17_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/orderline_017_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE17;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE18_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/orderline_018_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE18;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE19_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/orderline_019_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE19;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE20_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/orderline_020_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE20;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE21_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/orderline_021_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE21;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE2_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/orderline_002_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE2;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE22_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/orderline_022_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE22;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE23_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/orderline_023_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE23;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE24_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/orderline_024_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE24;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE25_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/orderline_025_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE25;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE26_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
```

```
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/orderline_026_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE26;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE27_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/orderline_027_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE27;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE28_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/orderline_028_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE28;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE29_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/orderline_029_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE29;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE30_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/orderline_030_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE30;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE31_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/orderline_031_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE31;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE3_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/orderline_003_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE3;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE32_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/orderline_032_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE32;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE33_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/orderline_033_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE33;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE34_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/orderline_034_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE34;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE35_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/orderline_035_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE35;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE36_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE36 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/orderline_036_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE36;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE37_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE37 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_037/orderline_037_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE37;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE38_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE38 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_038/orderline_038_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE38;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE39_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE39 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_039/orderline_039_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE39;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE40_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE40 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_040/orderline_040_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE40;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE41_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
```

```
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE41 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_041/orderline_041_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE41;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE4_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/orderline_004_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE4;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE42_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE42 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_042/orderline_042_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE42;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE43_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE43 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_043/orderline_043_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE43;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE44_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE44 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_044/orderline_044_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE44;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE45_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE45 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_045/orderline_045_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE45;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE46_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE46 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_046/orderline_046_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE46;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE47_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE47 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_047/orderline_047_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE47;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE48_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE48 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_048/orderline_048_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE48;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE49_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE49 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_049/orderline_049_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE49;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE50_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE50 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_050/orderline_050_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE50;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE51_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE51 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_051/orderline_051_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE51;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE5_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/orderline_005_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE5;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE52_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE52 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_052/orderline_052_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE52;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE53_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE53 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_053/orderline_053_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE53;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE54_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE54 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_054/orderline_054_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE54;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE55_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
```

```
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE55 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_055/orderline_055_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE55;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE56_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE56 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_056/orderline_056_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE56;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE57_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE57 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_057/orderline_057_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE57;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE58_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE58 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_058/orderline_058_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE58;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE59_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE59 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_059/orderline_059_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE59;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE60_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE60 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_060/orderline_060_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE60;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE61_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE61 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_061/orderline_061_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE61;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE6_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/orderline_006_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE6;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE62_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE62 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_062/orderline_062_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE62;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE63_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE63 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_063/orderline_063_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE63;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE64_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE64 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_064/orderline_064_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE64;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE65_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE65 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_065/orderline_065_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE65;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE66_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE66 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_066/orderline_066_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE66;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE67_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE67 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_067/orderline_067_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE67;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE68_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE68 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_068/orderline_068_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE68;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE69_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE69 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_069/orderline_069_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE69;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE70_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
```

```
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE70 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_070/orderline_070_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE70;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE71_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE71 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_071/orderline_071_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE71;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE7_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/orderline_007_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE7;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE72_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE72 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_072/orderline_072_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE72;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE8_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/orderline_008_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE8;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDER_LINE9_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/orderline_009_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 843480000 INSERT INTO
ORDER_LINE9;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS10_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/orders_010_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS10;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS10_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/orders_010_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS10;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS11_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/orders_011_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS11;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS11_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/orders_011_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS11;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS1_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/orders_001_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS1;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS12_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/orders_012_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS12;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS12_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/orders_012_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS12;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS1_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/orders_001_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS1;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS13_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/orders_013_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS13;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS13_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/orders_013_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS13;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS14_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS14 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/orders_014_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS14;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS14_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/orders_014_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS14;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS15_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/orders_015_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS15;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS15_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/orders_015_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS15;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS16_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/orders_016_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS16;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS16_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/orders_016_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS16;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS17_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/orders_017_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS17;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS17_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/orders_017_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS17;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS18_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/orders_018_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS18;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS18_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/orders_018_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS18;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS19_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/orders_019_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS19;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS19_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/orders_019_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS19;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS20_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/orders_020_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS20;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS20_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/orders_020_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS20;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS21_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/orders_021_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS21;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS21_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/orders_021_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS21;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS2_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS2 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/orders_002_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS2;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS22_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/orders_022_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS22;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS22_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/orders_022_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS22;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS2_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/orders_002_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS2;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS23_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/orders_023_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS23;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS23_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/orders_023_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS23;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS24_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/orders_024_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS24;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS24_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/orders_024_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS24;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS25_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/orders_025_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS25;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS25_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/orders_025_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS25;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS26_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/orders_026_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS26;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS26_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/orders_026_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS26;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS27_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/orders_027_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS27;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS27_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/orders_027_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS27;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS28_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/orders_028_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS28;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS28_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/orders_028_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS28;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS29_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/orders_029_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
```

```
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS29;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS29_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/orders_029_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS29;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS30_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/orders_030_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS30;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS30_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/orders_030_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS30;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS31_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/orders_031_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS31;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS31_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/orders_031_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS31;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS3_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/orders_003_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS3;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS32_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/orders_032_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS32;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS32_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/orders_032_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS32;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS3_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/orders_003_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS3;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS33_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/orders_033_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS33;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS33_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/orders_033_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS33;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS34_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/orders_034_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS34;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS34_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/orders_034_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS34;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS35_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/orders_035_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS35;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS35_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/orders_035_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS35;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS36_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS36 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/orders_036_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
```

```
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS36;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS36_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS36 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/orders_036_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO
ORDERS36;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS4_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/orders_004_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS4;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS4_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/orders_004_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS4;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS5_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/orders_005_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS5;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS5_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/orders_005_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS5;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS6_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/orders_006_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS6;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS6_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/orders_006_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS6;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS7_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/orders_007_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS7;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS7_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/orders_007_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS7;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS8_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/orders_008_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS8;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS8_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/orders_008_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS8;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS9_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/orders_009_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS9;
COMMIT WORK;
CONNECT RESET;
```

LOAD_ORDERS9_2.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/orders_009_2.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 76680000 INSERT INTO ORDERS9;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK100_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_100/stock_100_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK100;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK101_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK101 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_101/stock_101_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK101;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK10_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/stock_010_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK10;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK102_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK102 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_102/stock_102_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
```

```
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK102;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK103_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK103 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_103/stock_103_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK103;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK104_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK104 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_104/stock_104_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK104;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK105_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK105 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_105/stock_105_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK105;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK106_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK106 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_106/stock_106_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK106;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK107_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK107 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_107/stock_107_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK107;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK108_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK108 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_108/stock_108_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK108;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK109_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK109 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_109/stock_109_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK109;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK110_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK110 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_110/stock_110_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK110;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK111_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK111 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_111/stock_111_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK111;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK11_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/stock_011_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK11;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK112_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK112 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_112/stock_112_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK112;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK113_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK113 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_113/stock_113_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK113;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK114_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK114 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_114/stock_114_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK114;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK115_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK115 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_115/stock_115_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK115;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK116_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK116 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_116/stock_116_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK116;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK117_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK117 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_117/stock_117_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
```

```
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK117;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK118_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK118 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_118/stock_118_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK118;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK119_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK119 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_119/stock_119_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK119;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK1_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/stock_001_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK1;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK120_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK120 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_120/stock_120_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK120;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK121_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK121 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_121/stock_121_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK121;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK12_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/stock_012_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK12;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK122_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK122 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_122/stock_122_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK122;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK123_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK123 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_123/stock_123_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK123;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK124_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK124 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_124/stock_124_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK124;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK125_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK125 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_125/stock_125_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK125;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK126_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK126 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_126/stock_126_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK126;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK127_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK127 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_127/stock_127_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK127;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK128_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK128 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_128/stock_128_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK128;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK129_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK129 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_129/stock_129_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK129;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK130_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK130 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_130/stock_130_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK130;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK131_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK131 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_131/stock_131_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK131;
```

```
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK13_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/stock_013_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK13;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK132_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK132 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_132/stock_132_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK132;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK133_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK133 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_133/stock_133_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK133;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK134_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK134 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_134/stock_134_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK134;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK135_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK135 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_135/stock_135_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK135;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK136_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK136 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_136/stock_136_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK136;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK137_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK137 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_137/stock_137_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK137;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK138_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK138 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_138/stock_138_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK138;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK139_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK139 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_139/stock_139_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK139;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK140_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK140 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_140/stock_140_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK140;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK141_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK141 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_141/stock_141_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK141;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK14_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/stock_014_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK14;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK142_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK142 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_142/stock_142_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK142;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK143_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK143 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_143/stock_143_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK143;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK144_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK144 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_144/stock_144_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO
STOCK144;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK15_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/stock_015_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK15;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK16_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/stock_016_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK16;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK17_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/stock_017_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK17;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK18_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/stock_018_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK18;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK19_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/stock_019_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK19;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK20_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/stock_020_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK20;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK21_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/stock_021_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK21;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK2_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/stock_002_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK2;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK22_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/stock_022_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK22;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK23_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/stock_023_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK23;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK24_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/stock_024_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK24;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK25_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/stock_025_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK25;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK26_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/stock_026_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK26;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK27_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/stock_027_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK27;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK28_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/stock_028_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK28;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK29_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/stock_029_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK29;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK30_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/stock_030_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK30;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK31_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/stock_031_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK31;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK3_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK3 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/stock_003_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK3;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK32_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/stock_032_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK32;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK33_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/stock_033_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK33;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK34_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/stock_034_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK34;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK35_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/stock_035_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK35;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK36_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK36 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/stock_036_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK36;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK37_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK37 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_037/stock_037_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK37;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK38_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK38 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_038/stock_038_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK38;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK39_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK39 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_039/stock_039_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK39;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK40_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK40 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_040/stock_040_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK40;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK41_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK41 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_041/stock_041_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK41;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK4_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/stock_004_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK4;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK42_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK42 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_042/stock_042_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK42;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK43_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK43 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_043/stock_043_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK43;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK44_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK44 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_044/stock_044_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK44;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK45_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK45 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_045/stock_045_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK45;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK46_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK46 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_046/stock_046_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK46;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK47_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK47 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_047/stock_047_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK47;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK48_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK48 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_048/stock_048_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK48;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK49_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK49 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_049/stock_049_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK49;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK50_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK50 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_050/stock_050_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK50;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK51_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK51 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_051/stock_051_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK51;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK5_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/stock_005_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK5;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK52_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK52 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_052/stock_052_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK52;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK53_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK53 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_053/stock_053_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK53;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK54_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK54 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_054/stock_054_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK54;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK55_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK55 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_055/stock_055_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK55;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK56_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK56 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_056/stock_056_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK56;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK57_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK57 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_057/stock_057_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK57;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK58_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK58 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_058/stock_058_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK58;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK59_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK59 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_059/stock_059_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK59;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK60_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK60 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_060/stock_060_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK60;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK61_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK61 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_061/stock_061_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK61;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK6_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/stock_006_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK6;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK62_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK62 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_062/stock_062_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK62;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK63_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK63 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_063/stock_063_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK63;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK64_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK64 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_064/stock_064_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK64;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK65_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK65 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_065/stock_065_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK65;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK66_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK66 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_066/stock_066_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK66;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK67_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK67 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_067/stock_067_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK67;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK68_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK68 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_068/stock_068_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK68;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK69_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK69 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_069/stock_069_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK69;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK70_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK70 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_070/stock_070_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK70;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK71_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK71 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_071/stock_071_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK71;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK7_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/stock_007_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK7;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK72_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK72 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_072/stock_072_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK72;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK73_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK73 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_073/stock_073_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK73;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK74_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK74 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_074/stock_074_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK74;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK75_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK75 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_075/stock_075_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK75;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK76_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK76 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_076/stock_076_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK76;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK77_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK77 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_077/stock_077_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK77;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK78_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK78 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_078/stock_078_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK78;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK79_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK79 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_079/stock_079_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK79;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK80_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK80 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_080/stock_080_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK80;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK81_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK81 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_081/stock_081_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK81;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK8_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/stock_008_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK8;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK82_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK82 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_082/stock_082_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK82;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK83_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK83 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_083/stock_083_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK83;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK84_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK84 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_084/stock_084_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK84;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK85_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK85 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_085/stock_085_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK85;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK86_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK86 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_086/stock_086_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK86;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK87_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK87 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_087/stock_087_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK87;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK88_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK88 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_088/stock_088_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK88;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK89_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK89 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_089/stock_089_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK89;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK90_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK90 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_090/stock_090_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK90;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK91_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK91 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_091/stock_091_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK91;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK9_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/stock_009_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK9;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK92_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK92 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_092/stock_092_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK92;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK93_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK93 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_093/stock_093_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK93;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK94_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK94 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_094/stock_094_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK94;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK95_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK95 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_095/stock_095_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK95;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK96_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK96 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_096/stock_096_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK96;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK97_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK97 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_097/stock_097_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK97;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK98_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK98 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_098/stock_098_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK98;
COMMIT WORK;
CONNECT RESET;
```

LOAD_STOCK99_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK99 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_099/stock_099_1.dat OF
DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50  COMMITCOUNT 127800000 INSERT INTO STOCK99;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE10_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_010/warehouse_010_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE10;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE11_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_011/warehouse_011_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE11;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE1_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_001/warehouse_001_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE1;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE12_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_012/warehouse_012_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE12;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE13_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_013/warehouse_013_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE13;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE14_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_014/warehouse_014_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE14;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE15_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_015/warehouse_015_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE15;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE16_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_016/warehouse_016_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE16;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE17_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_017/warehouse_017_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE17;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE18_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_018/warehouse_018_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE18;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE19_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_019/warehouse_019_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE19;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE20_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_020/warehouse_020_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE20;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE21_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_021/warehouse_021_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE21;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE2_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_002/warehouse_002_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE2;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE22_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_022/warehouse_022_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE22;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE23_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_023/warehouse_023_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE23;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE24_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_024/warehouse_024_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE24;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE25_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_025/warehouse_025_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE25;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE26_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_026/warehouse_026_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE26;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE27_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_027/warehouse_027_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE27;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE28_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_028/warehouse_028_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE28;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE29_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_029/warehouse_029_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE29;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE30_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_030/warehouse_030_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE30;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE31_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_031/warehouse_031_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE31;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE3_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_003/warehouse_003_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE3;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE32_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_032/warehouse_032_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE32;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE33_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_033/warehouse_033_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE33;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE34_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_034/warehouse_034_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE34;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE35_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_035/warehouse_035_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE35;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE36_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_036/warehouse_036_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE36;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE4_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_004/warehouse_004_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE4;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE5_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_005/warehouse_005_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE5;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE6_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_006/warehouse_006_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE6;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE7_1.ddl
```
CONNECT TO TPCC IN SHARE MODE;
```

```
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_007/warehouse_007_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE7;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE8_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_008/warehouse_008_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE8;
COMMIT WORK;
CONNECT RESET;
```

LOAD_WAREHOUSE9_1.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /autobench/sources/db2_tpcc-1/flat_files/flat_009/warehouse_009_1.dat
OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS"
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE9;
COMMIT WORK;
CONNECT RESET;
```

RNST_CUSTOMER100.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER100 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER101.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER101 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER102.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER102 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER103.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER103 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER104.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER104 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER105.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER105 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER106.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER106 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER107.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER107 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER108.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER108 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER109.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER109 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER10.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER110.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER110 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER111.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER111 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER112.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER112 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER113.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER113 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER114.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER114 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER115.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER115 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER116.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER116 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER117.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER117 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER118.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER118 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER119.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER119 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER11.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER120.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER120 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER121.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER121 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER122.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER122 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER123.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER123 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER124.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER124 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER125.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER125 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER126.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER126 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER127.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER127 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER128.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER128 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER129.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER129 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER12.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER130.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER130 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER131.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER131 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER132.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER132 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER133.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER133 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER134.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER134 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER135.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER135 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER136.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER136 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER137.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER137 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER138.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER138 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER139.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER139 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER13.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER140.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER140 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER141.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER141 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER142.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER142 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER143.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER143 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER144.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER144 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER14.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER15.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER16.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER17.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER18.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER19.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER1.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER20.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER21.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER22.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER23.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER24.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER25.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER26.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER27.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER28.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER29.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER2.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER30.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER31.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER32.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER33.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER34.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER35.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER36.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER37.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER37 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER38.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER38 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER39.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER39 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER3.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER40.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER40 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER41.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER41 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER42.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER42 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER43.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER43 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER44.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER44 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER45.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER45 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER46.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER46 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER47.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER47 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER48.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER48 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER49.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER49 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER4.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER50.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER50 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER51.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER51 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER52.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER52 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER53.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER53 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER54.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER54 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER55.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER55 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER56.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER56 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER57.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER57 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER58.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER58 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER59.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER59 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER5.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER60.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER60 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER61.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER61 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER62.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER62 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER63.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER63 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER64.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER64 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER65.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER65 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER66.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER66 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER67.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER67 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER68.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER68 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER69.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER69 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER6.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER70.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER70 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER71.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER71 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER72.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER72 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER73.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER73 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER74.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER74 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER75.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER75 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER76.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER76 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER77.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER77 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER78.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER78 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER79.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER79 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER7.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER80.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER80 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER81.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER81 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER82.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER82 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER83.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER83 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER84.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER84 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER85.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER85 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER86.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER86 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER87.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER87 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER88.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER88 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER89.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER89 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER8.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER90.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER90 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER91.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER91 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER92.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER92 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER93.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER93 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER94.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER94 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER95.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER95 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER96.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER96 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER97.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER97 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER98.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER98 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER99.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER99 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_CUSTOMER9.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.CUSTOMER9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT10.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT11.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT12.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT13.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT14.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT15.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT16.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT17.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT18.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT19.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT1.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT20.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT21.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT22.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT23.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT24.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT25.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT26.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT27.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT28.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT29.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT2.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT30.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT31.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT32.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT33.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT34.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT35.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT36.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT3.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT4.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT5.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT6.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT7.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT8.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_DISTRICT9.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.DISTRICT9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY10.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY11.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY12.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY13.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY14.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY15.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY16.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY17.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY18.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY19.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY1.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY20.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY21.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY22.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY23.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY24.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY25.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY26.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY27.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY28.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY29.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY2.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY30.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY31.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY32.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY33.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY34.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY35.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY36.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY3.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY4.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY5.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY6.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY7.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY8.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_HISTORY9.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.HISTORY9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ITEM.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ITEM AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER10.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER11.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER12.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER13.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER14.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER15.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER16.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER17.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER18.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER19.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER1.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER20.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER21.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER22.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER23.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER24.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER25.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER26.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER27.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER28.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER29.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER2.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER30.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER31.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER32.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER33.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER34.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER35.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER36.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER3.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER4.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER5.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER6.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER7.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER8.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER9.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.NEW_ORDER9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE10.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE11.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE12.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE13.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE14.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE15.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE16.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE17.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE18.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE19.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE1.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE20.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE21.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE22.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE23.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE24.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE25.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE26.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE27.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE28.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE29.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE2.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE30.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE31.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE32.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE33.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE34.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE35.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE36.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE37.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE37 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE38.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE38 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE39.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE39 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE3.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE40.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE40 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE41.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE41 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE42.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE42 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE43.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE43 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE44.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE44 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE45.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE45 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE46.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE46 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE47.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE47 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE48.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE48 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE49.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE49 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE4.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE50.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE50 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE51.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE51 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE52.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE52 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE53.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE53 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE54.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE54 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE55.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE55 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE56.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE56 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE57.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE57 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE58.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE58 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE59.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE59 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE5.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE60.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE60 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE61.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE61 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE62.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE62 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE63.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE63 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE64.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE64 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE65.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE65 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE66.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE66 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE67.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE67 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE68.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE68 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE69.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE69 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE6.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE70.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE70 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE71.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE71 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE72.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE72 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE7.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE8.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDER_LINE9.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDER_LINE9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS10.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS11.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS12.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS13.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS14.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS15.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS16.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS17.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS18.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS19.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS1.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS20.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS21.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS22.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS23.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS24.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS25.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS26.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS27.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS28.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS29.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS2.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS30.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS31.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS32.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS33.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS34.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS35.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS36.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS3.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS4.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS5.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS6.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS7.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS8.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ORDERS9.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK100.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK100 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK101.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK101 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK102.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK102 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK103.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK103 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK104.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK104 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK105.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK105 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK106.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK106 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK107.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK107 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK108.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK108 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK109.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK109 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK10.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK110.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK110 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK111.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK111 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK112.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK112 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK113.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK113 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK114.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK114 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK115.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK115 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK116.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK116 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK117.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK117 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK118.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK118 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK119.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK119 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK11.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK120.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK120 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK121.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK121 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK122.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK122 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK123.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK123 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK124.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK124 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK125.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK125 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK126.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK126 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK127.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK127 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK128.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK128 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK129.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK129 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK12.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK130.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK130 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK131.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK131 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK132.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK132 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK133.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK133 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK134.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK134 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK135.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK135 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK136.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK136 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK137.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK137 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK138.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK138 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK139.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK139 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK13.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK140.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK140 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK141.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK141 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK142.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK142 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK143.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK143 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK144.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK144 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK14.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK15.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK16.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK17.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK18.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK19.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK1.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK20.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK21.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK22.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK23.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK24.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK25.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK26.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK27.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK28.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK29.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK2.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK30.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK31.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK32.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK33.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK34.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK35.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK36.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK37.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK37 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK38.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK38 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK39.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK39 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK3.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK40.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK40 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK41.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK41 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK42.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK42 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK43.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK43 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK44.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK44 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK45.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK45 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK46.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK46 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK47.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK47 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK48.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK48 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK49.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK49 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK4.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK50.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK50 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK51.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK51 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK52.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK52 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK53.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK53 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK54.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK54 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK55.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK55 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK56.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK56 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK57.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK57 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK58.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK58 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK59.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK59 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK5.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK60.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK60 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK61.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK61 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK62.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK62 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK63.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK63 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK64.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK64 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK65.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK65 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK66.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK66 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK67.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK67 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK68.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK68 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK69.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK69 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK6.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK70.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK70 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK71.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK71 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK72.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK72 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK73.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK73 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK74.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK74 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK75.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK75 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK76.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK76 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK77.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK77 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK78.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK78 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK79.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK79 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK7.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK80.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK80 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK81.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK81 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK82.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK82 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK83.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK83 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK84.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK84 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK85.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK85 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK86.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK86 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK87.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK87 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK88.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK88 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK89.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK89 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK8.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK90.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK90 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK91.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK91 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK92.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK92 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK93.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK93 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK94.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK94 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK95.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK95 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK96.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK96 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK97.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK97 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK98.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK98 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK99.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK99 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_STOCK9.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.STOCK9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE10.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE11.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE12.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE13.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE14.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE15.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE16.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE17.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE18.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE19.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE1.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE20.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE21.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE22.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE23.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE24.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE25.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE26.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE27.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE28.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE29.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE2.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE30.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE31.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE32.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE33.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE34.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE35.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE36.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE3.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE4.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE5.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE6.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE7.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE8.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_WAREHOUSE9.ddl
```
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.WAREHOUSE9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

CREATE_BUFFERPOOL_SPLIT.ddl
```
CREATE BUFFERPOOL WAR1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR5 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR6 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR7 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR8 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR9 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR10 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR11 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR12 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR13 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR14 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR15 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR16 SIZE 100 PAGESIZE 4096;

CREATE BUFFERPOOL DIS1  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS2  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS3  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS4  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS5  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS6  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS7  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS8  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS9  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS10 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS11 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS12 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS13 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS14 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS15 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS16 SIZE 100 PAGESIZE 4096;

CREATE BUFFERPOOL ITM SIZE 100 PAGESIZE 8192;

CREATE BUFFERPOOL HST1  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST2  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST3  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST4  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST5  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST6  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST7  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST8  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST9  SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST10 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST11 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST12 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST13 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST14 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST15 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST16 SIZE 100 PAGESIZE 16384;

CREATE BUFFERPOOL NEW1  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW2  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW3  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW4  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW5  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW6  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW7  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW8  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW9  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW10 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW11 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW12 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW13 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW14 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW15 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW16 SIZE 100 PAGESIZE 4096;

CREATE BUFFERPOOL ORD1  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD2  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD3  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD4  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD5  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD6  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD7  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD8  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD9  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD10 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD11 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD12 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD13 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD14 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD15 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD16 SIZE 100 PAGESIZE 8192;

CREATE BUFFERPOOL ORD_I1  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I2  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I3  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I4  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I5  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I6  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I7  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I8  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I9  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I10 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I11 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I12 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I13 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I14 SIZE 100 PAGESIZE 8192;
```

```
CREATE BUFFERPOOL ORD_I15 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I16 SIZE 100 PAGESIZE 8192;

CREATE BUFFERPOOL OLN1  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN2  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN3  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN4  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN5  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN6  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN7  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN8  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN9  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN10 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN11 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN12 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN13 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN14 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN15 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN16 SIZE 100 PAGESIZE 8192;

CREATE BUFFERPOOL CST1  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST2  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST3  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST4  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST5  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST6  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST7  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST8  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST9  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST10 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST11 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST12 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST13 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST14 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST15 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST16 SIZE 100 PAGESIZE 4096;

CREATE BUFFERPOOL CST_I1  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I2  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I3  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I4  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I5  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I6  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I7  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I8  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I9  SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I10 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I11 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I12 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I13 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I14 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I15 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I16 SIZE 100 PAGESIZE 8192;

CREATE BUFFERPOOL STK1  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK2  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK3  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK4  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK5  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK6  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK7  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK8  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK9  SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK10 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK11 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK12 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK13 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK14 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK15 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK16 SIZE 100 PAGESIZE 4096;
```

ALTER_TABLESPACE_SPLIT.ddl

```
ALTER TABLESPACE ts_W_001 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_W_002 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_W_003 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_W_004 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_W_005 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_W_006 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_W_007 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_W_008 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_W_009 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_W_010 BUFFERPOOL WAR5;
ALTER TABLESPACE ts_W_011 BUFFERPOOL WAR5;
ALTER TABLESPACE ts_W_012 BUFFERPOOL WAR5;
ALTER TABLESPACE ts_W_013 BUFFERPOOL WAR6;
ALTER TABLESPACE ts_W_014 BUFFERPOOL WAR6;
ALTER TABLESPACE ts_W_015 BUFFERPOOL WAR7;
ALTER TABLESPACE ts_W_016 BUFFERPOOL WAR7;
ALTER TABLESPACE ts_W_017 BUFFERPOOL WAR8;
ALTER TABLESPACE ts_W_018 BUFFERPOOL WAR8;
ALTER TABLESPACE ts_W_019 BUFFERPOOL WAR9;
ALTER TABLESPACE ts_W_020 BUFFERPOOL WAR9;
ALTER TABLESPACE ts_W_021 BUFFERPOOL WAR9;
ALTER TABLESPACE ts_W_022 BUFFERPOOL WAR10;
ALTER TABLESPACE ts_W_023 BUFFERPOOL WAR10;
ALTER TABLESPACE ts_W_024 BUFFERPOOL WAR11;
ALTER TABLESPACE ts_W_025 BUFFERPOOL WAR11;
ALTER TABLESPACE ts_W_026 BUFFERPOOL WAR12;
ALTER TABLESPACE ts_W_027 BUFFERPOOL WAR12;
ALTER TABLESPACE ts_W_028 BUFFERPOOL WAR13;
ALTER TABLESPACE ts_W_029 BUFFERPOOL WAR13;
ALTER TABLESPACE ts_W_030 BUFFERPOOL WAR13;
ALTER TABLESPACE ts_W_031 BUFFERPOOL WAR14;
ALTER TABLESPACE ts_W_032 BUFFERPOOL WAR14;
ALTER TABLESPACE ts_W_033 BUFFERPOOL WAR15;
ALTER TABLESPACE ts_W_034 BUFFERPOOL WAR15;
ALTER TABLESPACE ts_W_035 BUFFERPOOL WAR16;
ALTER TABLESPACE ts_W_036 BUFFERPOOL WAR16;
ALTER TABLESPACE ts_D_001 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_D_002 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_D_003 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_D_004 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_D_005 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_D_006 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_D_007 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_D_008 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_D_009 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_D_010 BUFFERPOOL DIS5;
ALTER TABLESPACE ts_D_011 BUFFERPOOL DIS5;
ALTER TABLESPACE ts_D_012 BUFFERPOOL DIS5;
ALTER TABLESPACE ts_D_013 BUFFERPOOL DIS6;
ALTER TABLESPACE ts_D_014 BUFFERPOOL DIS6;
ALTER TABLESPACE ts_D_015 BUFFERPOOL DIS7;
ALTER TABLESPACE ts_D_016 BUFFERPOOL DIS7;
ALTER TABLESPACE ts_D_017 BUFFERPOOL DIS8;
ALTER TABLESPACE ts_D_018 BUFFERPOOL DIS8;
ALTER TABLESPACE ts_D_019 BUFFERPOOL DIS9;
ALTER TABLESPACE ts_D_020 BUFFERPOOL DIS9;
ALTER TABLESPACE ts_D_021 BUFFERPOOL DIS9;
ALTER TABLESPACE ts_D_022 BUFFERPOOL DIS10;
ALTER TABLESPACE ts_D_023 BUFFERPOOL DIS10;
ALTER TABLESPACE ts_D_024 BUFFERPOOL DIS11;
ALTER TABLESPACE ts_D_025 BUFFERPOOL DIS11;
ALTER TABLESPACE ts_D_026 BUFFERPOOL DIS12;
ALTER TABLESPACE ts_D_027 BUFFERPOOL DIS12;
ALTER TABLESPACE ts_D_028 BUFFERPOOL DIS13;
ALTER TABLESPACE ts_D_029 BUFFERPOOL DIS13;
ALTER TABLESPACE ts_D_030 BUFFERPOOL DIS13;
ALTER TABLESPACE ts_D_031 BUFFERPOOL DIS14;
ALTER TABLESPACE ts_D_032 BUFFERPOOL DIS14;
ALTER TABLESPACE ts_D_033 BUFFERPOOL DIS15;
ALTER TABLESPACE ts_D_034 BUFFERPOOL DIS15;
ALTER TABLESPACE ts_D_035 BUFFERPOOL DIS16;
ALTER TABLESPACE ts_D_036 BUFFERPOOL DIS16;
ALTER TABLESPACE ts_I BUFFERPOOL ITM;
ALTER TABLESPACE ts_H_001 BUFFERPOOL HST1;
ALTER TABLESPACE ts_H_002 BUFFERPOOL HST1;
ALTER TABLESPACE ts_H_003 BUFFERPOOL HST1;
ALTER TABLESPACE ts_H_004 BUFFERPOOL HST2;
ALTER TABLESPACE ts_H_005 BUFFERPOOL HST2;
ALTER TABLESPACE ts_H_006 BUFFERPOOL HST3;
ALTER TABLESPACE ts_H_007 BUFFERPOOL HST3;
ALTER TABLESPACE ts_H_008 BUFFERPOOL HST4;
ALTER TABLESPACE ts_H_009 BUFFERPOOL HST4;
ALTER TABLESPACE ts_H_010 BUFFERPOOL HST5;
ALTER TABLESPACE ts_H_011 BUFFERPOOL HST5;
ALTER TABLESPACE ts_H_012 BUFFERPOOL HST5;
ALTER TABLESPACE ts_H_013 BUFFERPOOL HST6;
ALTER TABLESPACE ts_H_014 BUFFERPOOL HST6;
ALTER TABLESPACE ts_H_015 BUFFERPOOL HST7;
ALTER TABLESPACE ts_H_016 BUFFERPOOL HST7;
ALTER TABLESPACE ts_H_017 BUFFERPOOL HST8;
ALTER TABLESPACE ts_H_018 BUFFERPOOL HST8;
ALTER TABLESPACE ts_H_019 BUFFERPOOL HST9;
ALTER TABLESPACE ts_H_020 BUFFERPOOL HST9;
ALTER TABLESPACE ts_H_021 BUFFERPOOL HST9;
ALTER TABLESPACE ts_H_022 BUFFERPOOL HST10;
ALTER TABLESPACE ts_H_023 BUFFERPOOL HST10;
ALTER TABLESPACE ts_H_024 BUFFERPOOL HST11;
ALTER TABLESPACE ts_H_025 BUFFERPOOL HST11;
ALTER TABLESPACE ts_H_026 BUFFERPOOL HST12;
ALTER TABLESPACE ts_H_027 BUFFERPOOL HST12;
ALTER TABLESPACE ts_H_028 BUFFERPOOL HST13;
ALTER TABLESPACE ts_H_029 BUFFERPOOL HST13;
ALTER TABLESPACE ts_H_030 BUFFERPOOL HST13;
ALTER TABLESPACE ts_H_031 BUFFERPOOL HST14;
ALTER TABLESPACE ts_H_032 BUFFERPOOL HST14;
ALTER TABLESPACE ts_H_033 BUFFERPOOL HST15;
ALTER TABLESPACE ts_H_034 BUFFERPOOL HST15;
ALTER TABLESPACE ts_H_035 BUFFERPOOL HST16;
ALTER TABLESPACE ts_H_036 BUFFERPOOL HST16;
ALTER TABLESPACE ts_N_001 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_N_002 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_N_003 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_N_004 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_N_005 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_N_006 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_N_007 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_N_008 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_N_009 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_N_010 BUFFERPOOL NEW5;
ALTER TABLESPACE ts_N_011 BUFFERPOOL NEW5;
ALTER TABLESPACE ts_N_012 BUFFERPOOL NEW5;
ALTER TABLESPACE ts_N_013 BUFFERPOOL NEW6;
```

```
ALTER TABLESPACE ts_N_014 BUFFERPOOL NEW6;
ALTER TABLESPACE ts_N_015 BUFFERPOOL NEW7;
ALTER TABLESPACE ts_N_016 BUFFERPOOL NEW7;
ALTER TABLESPACE ts_N_017 BUFFERPOOL NEW8;
ALTER TABLESPACE ts_N_018 BUFFERPOOL NEW8;
ALTER TABLESPACE ts_N_019 BUFFERPOOL NEW9;
ALTER TABLESPACE ts_N_020 BUFFERPOOL NEW9;
ALTER TABLESPACE ts_N_021 BUFFERPOOL NEW9;
ALTER TABLESPACE ts_N_022 BUFFERPOOL NEW10;
ALTER TABLESPACE ts_N_023 BUFFERPOOL NEW10;
ALTER TABLESPACE ts_N_024 BUFFERPOOL NEW11;
ALTER TABLESPACE ts_N_025 BUFFERPOOL NEW11;
ALTER TABLESPACE ts_N_026 BUFFERPOOL NEW12;
ALTER TABLESPACE ts_N_027 BUFFERPOOL NEW12;
ALTER TABLESPACE ts_N_028 BUFFERPOOL NEW13;
ALTER TABLESPACE ts_N_029 BUFFERPOOL NEW13;
ALTER TABLESPACE ts_N_030 BUFFERPOOL NEW13;
ALTER TABLESPACE ts_N_031 BUFFERPOOL NEW14;
ALTER TABLESPACE ts_N_032 BUFFERPOOL NEW14;
ALTER TABLESPACE ts_N_033 BUFFERPOOL NEW15;
ALTER TABLESPACE ts_N_034 BUFFERPOOL NEW15;
ALTER TABLESPACE ts_N_035 BUFFERPOOL NEW16;
ALTER TABLESPACE ts_N_036 BUFFERPOOL NEW16;
ALTER TABLESPACE ts_O_001 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_O_002 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_O_003 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_O_004 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_O_005 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_O_006 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_O_007 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_O_008 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_O_009 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_O_010 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_O_011 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_O_012 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_O_013 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_O_014 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_O_015 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_O_016 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_O_017 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_O_018 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_O_019 BUFFERPOOL ORD9;
ALTER TABLESPACE ts_O_020 BUFFERPOOL ORD9;
ALTER TABLESPACE ts_O_021 BUFFERPOOL ORD9;
ALTER TABLESPACE ts_O_022 BUFFERPOOL ORD10;
ALTER TABLESPACE ts_O_023 BUFFERPOOL ORD10;
ALTER TABLESPACE ts_O_024 BUFFERPOOL ORD11;
ALTER TABLESPACE ts_O_025 BUFFERPOOL ORD11;
ALTER TABLESPACE ts_O_026 BUFFERPOOL ORD12;
ALTER TABLESPACE ts_O_027 BUFFERPOOL ORD12;
ALTER TABLESPACE ts_O_028 BUFFERPOOL ORD13;
ALTER TABLESPACE ts_O_029 BUFFERPOOL ORD13;
ALTER TABLESPACE ts_O_030 BUFFERPOOL ORD13;
ALTER TABLESPACE ts_O_031 BUFFERPOOL ORD14;
ALTER TABLESPACE ts_O_032 BUFFERPOOL ORD14;
ALTER TABLESPACE ts_O_033 BUFFERPOOL ORD15;
ALTER TABLESPACE ts_O_034 BUFFERPOOL ORD15;
ALTER TABLESPACE ts_O_035 BUFFERPOOL ORD16;
ALTER TABLESPACE ts_O_036 BUFFERPOOL ORD16;
ALTER TABLESPACE is_O2_001 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_O2_002 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_O2_003 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_O2_004 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_O2_005 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_O2_006 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_O2_007 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_O2_008 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_O2_009 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_O2_010 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_O2_011 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_O2_012 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_O2_013 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_O2_014 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_O2_015 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_O2_016 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_O2_017 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_O2_018 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_O2_019 BUFFERPOOL ORD_I9;
ALTER TABLESPACE is_O2_020 BUFFERPOOL ORD_I9;
ALTER TABLESPACE is_O2_021 BUFFERPOOL ORD_I9;
ALTER TABLESPACE is_O2_022 BUFFERPOOL ORD_I10;
ALTER TABLESPACE is_O2_023 BUFFERPOOL ORD_I10;
ALTER TABLESPACE is_O2_024 BUFFERPOOL ORD_I11;
ALTER TABLESPACE is_O2_025 BUFFERPOOL ORD_I11;
ALTER TABLESPACE is_O2_026 BUFFERPOOL ORD_I12;
ALTER TABLESPACE is_O2_027 BUFFERPOOL ORD_I12;
ALTER TABLESPACE is_O2_028 BUFFERPOOL ORD_I13;
ALTER TABLESPACE is_O2_029 BUFFERPOOL ORD_I13;
ALTER TABLESPACE is_O2_030 BUFFERPOOL ORD_I13;
ALTER TABLESPACE is_O2_031 BUFFERPOOL ORD_I14;
ALTER TABLESPACE is_O2_032 BUFFERPOOL ORD_I14;
ALTER TABLESPACE is_O2_033 BUFFERPOOL ORD_I15;
ALTER TABLESPACE is_O2_034 BUFFERPOOL ORD_I15;
ALTER TABLESPACE is_O2_035 BUFFERPOOL ORD_I16;
ALTER TABLESPACE is_O2_036 BUFFERPOOL ORD_I16;
ALTER TABLESPACE ts_OL_001 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_OL_002 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_OL_003 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_OL_004 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_OL_005 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_OL_006 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_OL_007 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_OL_008 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_OL_009 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_OL_010 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_OL_011 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_OL_012 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_OL_013 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_OL_014 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_OL_015 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_OL_016 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_OL_017 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_OL_018 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_OL_019 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_OL_020 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_OL_021 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_OL_022 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_OL_023 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_OL_024 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_OL_025 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_OL_026 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_OL_027 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_OL_028 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_OL_029 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_OL_030 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_OL_031 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_OL_032 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_OL_033 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_OL_034 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_OL_035 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_OL_036 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_OL_037 BUFFERPOOL OLN9;
ALTER TABLESPACE ts_OL_038 BUFFERPOOL OLN9;
ALTER TABLESPACE ts_OL_039 BUFFERPOOL OLN9;
ALTER TABLESPACE ts_OL_040 BUFFERPOOL OLN9;
ALTER TABLESPACE ts_OL_041 BUFFERPOOL OLN9;
ALTER TABLESPACE ts_OL_042 BUFFERPOOL OLN10;
ALTER TABLESPACE ts_OL_043 BUFFERPOOL OLN10;
ALTER TABLESPACE ts_OL_044 BUFFERPOOL OLN10;
ALTER TABLESPACE ts_OL_045 BUFFERPOOL OLN10;
ALTER TABLESPACE ts_OL_046 BUFFERPOOL OLN11;
ALTER TABLESPACE ts_OL_047 BUFFERPOOL OLN11;
ALTER TABLESPACE ts_OL_048 BUFFERPOOL OLN11;
ALTER TABLESPACE ts_OL_049 BUFFERPOOL OLN11;
ALTER TABLESPACE ts_OL_050 BUFFERPOOL OLN11;
ALTER TABLESPACE ts_OL_051 BUFFERPOOL OLN12;
ALTER TABLESPACE ts_OL_052 BUFFERPOOL OLN12;
ALTER TABLESPACE ts_OL_053 BUFFERPOOL OLN12;
ALTER TABLESPACE ts_OL_054 BUFFERPOOL OLN12;
ALTER TABLESPACE ts_OL_055 BUFFERPOOL OLN13;
ALTER TABLESPACE ts_OL_056 BUFFERPOOL OLN13;
ALTER TABLESPACE ts_OL_057 BUFFERPOOL OLN13;
ALTER TABLESPACE ts_OL_058 BUFFERPOOL OLN13;
ALTER TABLESPACE ts_OL_059 BUFFERPOOL OLN13;
ALTER TABLESPACE ts_OL_060 BUFFERPOOL OLN14;
ALTER TABLESPACE ts_OL_061 BUFFERPOOL OLN14;
ALTER TABLESPACE ts_OL_062 BUFFERPOOL OLN14;
ALTER TABLESPACE ts_OL_063 BUFFERPOOL OLN14;
ALTER TABLESPACE ts_OL_064 BUFFERPOOL OLN15;
ALTER TABLESPACE ts_OL_065 BUFFERPOOL OLN15;
ALTER TABLESPACE ts_OL_066 BUFFERPOOL OLN15;
ALTER TABLESPACE ts_OL_067 BUFFERPOOL OLN15;
ALTER TABLESPACE ts_OL_068 BUFFERPOOL OLN15;
ALTER TABLESPACE ts_OL_069 BUFFERPOOL OLN16;
ALTER TABLESPACE ts_OL_070 BUFFERPOOL OLN16;
ALTER TABLESPACE ts_OL_071 BUFFERPOOL OLN16;
ALTER TABLESPACE ts_OL_072 BUFFERPOOL OLN16;
ALTER TABLESPACE ts_C_001 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_002 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_003 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_004 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_005 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_006 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_007 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_008 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_009 BUFFERPOOL CST1;
ALTER TABLESPACE ts_C_010 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_011 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_012 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_013 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_014 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_015 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_016 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_017 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_018 BUFFERPOOL CST2;
ALTER TABLESPACE ts_C_019 BUFFERPOOL CST3;
ALTER TABLESPACE ts_C_020 BUFFERPOOL CST3;
ALTER TABLESPACE ts_C_021 BUFFERPOOL CST3;
ALTER TABLESPACE ts_C_022 BUFFERPOOL CST3;
ALTER TABLESPACE ts_C_023 BUFFERPOOL CST3;
ALTER TABLESPACE ts_C_024 BUFFERPOOL CST3;
ALTER TABLESPACE ts_C_025 BUFFERPOOL CST3;
ALTER TABLESPACE ts_C_026 BUFFERPOOL CST3;
ALTER TABLESPACE ts_C_027 BUFFERPOOL CST3;
```

```
ALTER TABLESPACE ts_C_028 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_029 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_030 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_031 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_032 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_033 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_034 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_035 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_036 BUFFERPOOL CST4;
ALTER TABLESPACE ts_C_037 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_038 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_039 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_040 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_041 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_042 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_043 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_044 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_045 BUFFERPOOL CST5;
ALTER TABLESPACE ts_C_046 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_047 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_048 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_049 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_050 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_051 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_052 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_053 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_054 BUFFERPOOL CST6;
ALTER TABLESPACE ts_C_055 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_056 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_057 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_058 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_059 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_060 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_061 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_062 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_063 BUFFERPOOL CST7;
ALTER TABLESPACE ts_C_064 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_065 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_066 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_067 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_068 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_069 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_070 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_071 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_072 BUFFERPOOL CST8;
ALTER TABLESPACE ts_C_073 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_074 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_075 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_076 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_077 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_078 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_079 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_080 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_081 BUFFERPOOL CST9;
ALTER TABLESPACE ts_C_082 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_083 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_084 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_085 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_086 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_087 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_088 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_089 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_090 BUFFERPOOL CST10;
ALTER TABLESPACE ts_C_091 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_092 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_093 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_094 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_095 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_096 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_097 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_098 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_099 BUFFERPOOL CST11;
ALTER TABLESPACE ts_C_100 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_101 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_102 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_103 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_104 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_105 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_106 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_107 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_108 BUFFERPOOL CST12;
ALTER TABLESPACE ts_C_109 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_110 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_111 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_112 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_113 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_114 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_115 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_116 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_117 BUFFERPOOL CST13;
ALTER TABLESPACE ts_C_118 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_119 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_120 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_121 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_122 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_123 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_124 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_125 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_126 BUFFERPOOL CST14;
ALTER TABLESPACE ts_C_127 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_128 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_129 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_130 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_131 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_132 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_133 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_134 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_135 BUFFERPOOL CST15;
ALTER TABLESPACE ts_C_136 BUFFERPOOL CST16;
ALTER TABLESPACE ts_C_137 BUFFERPOOL CST16;
ALTER TABLESPACE ts_C_138 BUFFERPOOL CST16;
ALTER TABLESPACE ts_C_139 BUFFERPOOL CST16;
ALTER TABLESPACE ts_C_140 BUFFERPOOL CST16;
ALTER TABLESPACE ts_C_141 BUFFERPOOL CST16;
ALTER TABLESPACE ts_C_142 BUFFERPOOL CST16;
ALTER TABLESPACE ts_C_143 BUFFERPOOL CST16;
ALTER TABLESPACE ts_C_144 BUFFERPOOL CST16;
ALTER TABLESPACE is_C2_001 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_002 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_003 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_004 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_005 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_006 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_007 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_008 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_009 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_C2_010 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_011 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_012 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_013 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_014 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_015 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_016 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_017 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_018 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_C2_019 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_020 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_021 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_022 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_023 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_024 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_025 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_026 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_027 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_C2_028 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_029 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_030 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_031 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_032 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_033 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_034 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_035 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_036 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_C2_037 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_038 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_039 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_040 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_041 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_042 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_043 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_044 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_045 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_C2_046 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_047 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_048 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_049 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_050 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_051 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_052 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_053 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_054 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_C2_055 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_056 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_057 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_058 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_059 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_060 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_061 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_062 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_063 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_C2_064 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_065 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_066 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_067 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_068 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_069 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_070 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_071 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_072 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_C2_073 BUFFERPOOL CST_I9;
ALTER TABLESPACE is_C2_074 BUFFERPOOL CST_I9;
ALTER TABLESPACE is_C2_075 BUFFERPOOL CST_I9;
ALTER TABLESPACE is_C2_076 BUFFERPOOL CST_I9;
ALTER TABLESPACE is_C2_077 BUFFERPOOL CST_I9;
```

```
ALTER TABLESPACE is_C2_078 BUFFERPOOL CST_I9;
ALTER TABLESPACE is_C2_079 BUFFERPOOL CST_I9;
ALTER TABLESPACE is_C2_080 BUFFERPOOL CST_I9;
ALTER TABLESPACE is_C2_081 BUFFERPOOL CST_I9;
ALTER TABLESPACE is_C2_082 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_083 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_084 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_085 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_086 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_087 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_088 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_089 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_090 BUFFERPOOL CST_I10;
ALTER TABLESPACE is_C2_091 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_092 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_093 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_094 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_095 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_096 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_097 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_098 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_099 BUFFERPOOL CST_I11;
ALTER TABLESPACE is_C2_100 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_101 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_102 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_103 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_104 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_105 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_106 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_107 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_108 BUFFERPOOL CST_I12;
ALTER TABLESPACE is_C2_109 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_110 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_111 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_112 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_113 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_114 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_115 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_116 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_117 BUFFERPOOL CST_I13;
ALTER TABLESPACE is_C2_118 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_119 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_120 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_121 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_122 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_123 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_124 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_125 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_126 BUFFERPOOL CST_I14;
ALTER TABLESPACE is_C2_127 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_128 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_129 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_130 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_131 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_132 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_133 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_134 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_135 BUFFERPOOL CST_I15;
ALTER TABLESPACE is_C2_136 BUFFERPOOL CST_I16;
ALTER TABLESPACE is_C2_137 BUFFERPOOL CST_I16;
ALTER TABLESPACE is_C2_138 BUFFERPOOL CST_I16;
ALTER TABLESPACE is_C2_139 BUFFERPOOL CST_I16;
ALTER TABLESPACE is_C2_140 BUFFERPOOL CST_I16;
ALTER TABLESPACE is_C2_141 BUFFERPOOL CST_I16;
ALTER TABLESPACE is_C2_142 BUFFERPOOL CST_I16;
ALTER TABLESPACE is_C2_143 BUFFERPOOL CST_I16;
ALTER TABLESPACE is_C2_144 BUFFERPOOL CST_I16;
ALTER TABLESPACE ts_S_001 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_002 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_003 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_004 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_005 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_006 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_007 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_008 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_009 BUFFERPOOL STK1;
ALTER TABLESPACE ts_S_010 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_011 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_012 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_013 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_014 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_015 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_016 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_017 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_018 BUFFERPOOL STK2;
ALTER TABLESPACE ts_S_019 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_020 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_021 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_022 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_023 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_024 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_025 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_026 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_027 BUFFERPOOL STK3;
ALTER TABLESPACE ts_S_028 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_029 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_030 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_031 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_032 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_033 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_034 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_035 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_036 BUFFERPOOL STK4;
ALTER TABLESPACE ts_S_037 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_038 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_039 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_040 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_041 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_042 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_043 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_044 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_045 BUFFERPOOL STK5;
ALTER TABLESPACE ts_S_046 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_047 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_048 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_049 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_050 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_051 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_052 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_053 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_054 BUFFERPOOL STK6;
ALTER TABLESPACE ts_S_055 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_056 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_057 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_058 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_059 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_060 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_061 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_062 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_063 BUFFERPOOL STK7;
ALTER TABLESPACE ts_S_064 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_065 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_066 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_067 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_068 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_069 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_070 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_071 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_072 BUFFERPOOL STK8;
ALTER TABLESPACE ts_S_073 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_074 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_075 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_076 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_077 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_078 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_079 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_080 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_081 BUFFERPOOL STK9;
ALTER TABLESPACE ts_S_082 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_083 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_084 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_085 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_086 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_087 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_088 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_089 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_090 BUFFERPOOL STK10;
ALTER TABLESPACE ts_S_091 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_092 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_093 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_094 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_095 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_096 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_097 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_098 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_099 BUFFERPOOL STK11;
ALTER TABLESPACE ts_S_100 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_101 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_102 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_103 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_104 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_105 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_106 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_107 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_108 BUFFERPOOL STK12;
ALTER TABLESPACE ts_S_109 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_110 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_111 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_112 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_113 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_114 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_115 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_116 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_117 BUFFERPOOL STK13;
ALTER TABLESPACE ts_S_118 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_119 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_120 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_121 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_122 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_123 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_124 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_125 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_126 BUFFERPOOL STK14;
ALTER TABLESPACE ts_S_127 BUFFERPOOL STK15;
```

```
ALTER TABLESPACE ts_S_128 BUFFERPOOL STK15;
ALTER TABLESPACE ts_S_129 BUFFERPOOL STK15;
ALTER TABLESPACE ts_S_130 BUFFERPOOL STK15;
ALTER TABLESPACE ts_S_131 BUFFERPOOL STK15;
ALTER TABLESPACE ts_S_132 BUFFERPOOL STK15;
ALTER TABLESPACE ts_S_133 BUFFERPOOL STK15;
ALTER TABLESPACE ts_S_134 BUFFERPOOL STK15;
ALTER TABLESPACE ts_S_135 BUFFERPOOL STK15;
ALTER TABLESPACE ts_S_136 BUFFERPOOL STK16;
ALTER TABLESPACE ts_S_137 BUFFERPOOL STK16;
ALTER TABLESPACE ts_S_138 BUFFERPOOL STK16;
ALTER TABLESPACE ts_S_139 BUFFERPOOL STK16;
ALTER TABLESPACE ts_S_140 BUFFERPOOL STK16;
ALTER TABLESPACE ts_S_141 BUFFERPOOL STK16;
ALTER TABLESPACE ts_S_142 BUFFERPOOL STK16;
ALTER TABLESPACE ts_S_143 BUFFERPOOL STK16;
ALTER TABLESPACE ts_S_144 BUFFERPOOL STK16;
```

GROW_BUFFERPOOL_SPLIT.ddl

```
ALTER BUFFERPOOL WAR1 DEFERRED SIZE  500;
ALTER BUFFERPOOL WAR2 DEFERRED SIZE  334;
ALTER BUFFERPOOL WAR3 DEFERRED SIZE  334;
ALTER BUFFERPOOL WAR4 DEFERRED SIZE  334;
ALTER BUFFERPOOL WAR5 DEFERRED SIZE  500;
ALTER BUFFERPOOL WAR6 DEFERRED SIZE  334;
ALTER BUFFERPOOL WAR7 DEFERRED SIZE  334;
ALTER BUFFERPOOL WAR8 DEFERRED SIZE  334;
ALTER BUFFERPOOL WAR9 DEFERRED SIZE  500;
ALTER BUFFERPOOL WAR10 DEFERRED SIZE 334;
ALTER BUFFERPOOL WAR11 DEFERRED SIZE 334;
ALTER BUFFERPOOL WAR12 DEFERRED SIZE 334;
ALTER BUFFERPOOL WAR13 DEFERRED SIZE 500;
ALTER BUFFERPOOL WAR14 DEFERRED SIZE 334;
ALTER BUFFERPOOL WAR15 DEFERRED SIZE 334;
ALTER BUFFERPOOL WAR16 DEFERRED SIZE 334;
ALTER BUFFERPOOL DIS1 DEFERRED SIZE  4766;
ALTER BUFFERPOOL DIS2 DEFERRED SIZE  3178;
ALTER BUFFERPOOL DIS3 DEFERRED SIZE  3178;
ALTER BUFFERPOOL DIS4 DEFERRED SIZE  3178;
ALTER BUFFERPOOL DIS5 DEFERRED SIZE  4766;
ALTER BUFFERPOOL DIS6 DEFERRED SIZE  3178;
ALTER BUFFERPOOL DIS7 DEFERRED SIZE  3178;
ALTER BUFFERPOOL DIS8 DEFERRED SIZE  3178;
ALTER BUFFERPOOL DIS9 DEFERRED SIZE  4766;
ALTER BUFFERPOOL DIS10 DEFERRED SIZE 3178;
ALTER BUFFERPOOL DIS11 DEFERRED SIZE 3178;
ALTER BUFFERPOOL DIS12 DEFERRED SIZE 3178;
ALTER BUFFERPOOL DIS13 DEFERRED SIZE 4766;
ALTER BUFFERPOOL DIS14 DEFERRED SIZE 3178;
ALTER BUFFERPOOL DIS15 DEFERRED SIZE 3178;
ALTER BUFFERPOOL DIS16 DEFERRED SIZE 3178;
ALTER BUFFERPOOL ITM DEFERRED SIZE 1260;
ALTER BUFFERPOOL HST1 DEFERRED SIZE  800;
ALTER BUFFERPOOL HST2 DEFERRED SIZE  534;
ALTER BUFFERPOOL HST3 DEFERRED SIZE  534;
ALTER BUFFERPOOL HST4 DEFERRED SIZE  534;
ALTER BUFFERPOOL HST5 DEFERRED SIZE  800;
ALTER BUFFERPOOL HST6 DEFERRED SIZE  534;
ALTER BUFFERPOOL HST7 DEFERRED SIZE  534;
ALTER BUFFERPOOL HST8 DEFERRED SIZE  534;
ALTER BUFFERPOOL HST9 DEFERRED SIZE  800;
ALTER BUFFERPOOL HST10 DEFERRED SIZE 534;
ALTER BUFFERPOOL HST11 DEFERRED SIZE 534;
ALTER BUFFERPOOL HST12 DEFERRED SIZE 534;
ALTER BUFFERPOOL HST13 DEFERRED SIZE 800;
ALTER BUFFERPOOL HST14 DEFERRED SIZE 534;
ALTER BUFFERPOOL HST15 DEFERRED SIZE 534;
ALTER BUFFERPOOL HST16 DEFERRED SIZE 534;
ALTER BUFFERPOOL NEW1 DEFERRED SIZE  408368;
ALTER BUFFERPOOL NEW2 DEFERRED SIZE  272245;
ALTER BUFFERPOOL NEW3 DEFERRED SIZE  272245;
ALTER BUFFERPOOL NEW4 DEFERRED SIZE  272245;
ALTER BUFFERPOOL NEW5 DEFERRED SIZE  408368;
ALTER BUFFERPOOL NEW6 DEFERRED SIZE  272245;
ALTER BUFFERPOOL NEW7 DEFERRED SIZE  272245;
ALTER BUFFERPOOL NEW8 DEFERRED SIZE  272245;
ALTER BUFFERPOOL NEW9 DEFERRED SIZE  408368;
ALTER BUFFERPOOL NEW10 DEFERRED SIZE 272245;
ALTER BUFFERPOOL NEW11 DEFERRED SIZE 272245;
ALTER BUFFERPOOL NEW12 DEFERRED SIZE 272245;
ALTER BUFFERPOOL NEW13 DEFERRED SIZE 408368;
ALTER BUFFERPOOL NEW14 DEFERRED SIZE 272245;
ALTER BUFFERPOOL NEW15 DEFERRED SIZE 272245;
ALTER BUFFERPOOL NEW16 DEFERRED SIZE 272245;
ALTER BUFFERPOOL ORD1 DEFERRED SIZE  332883;
ALTER BUFFERPOOL ORD2 DEFERRED SIZE  221922;
ALTER BUFFERPOOL ORD3 DEFERRED SIZE  221922;
ALTER BUFFERPOOL ORD4 DEFERRED SIZE  221922;
ALTER BUFFERPOOL ORD5 DEFERRED SIZE  332883;
ALTER BUFFERPOOL ORD6 DEFERRED SIZE  221922;
ALTER BUFFERPOOL ORD7 DEFERRED SIZE  221922;
ALTER BUFFERPOOL ORD8 DEFERRED SIZE  221922;
ALTER BUFFERPOOL ORD9 DEFERRED SIZE  332883;
ALTER BUFFERPOOL ORD10 DEFERRED SIZE 221922;
ALTER BUFFERPOOL ORD11 DEFERRED SIZE 221922;
ALTER BUFFERPOOL ORD12 DEFERRED SIZE 221922;
ALTER BUFFERPOOL ORD13 DEFERRED SIZE 332883;
ALTER BUFFERPOOL ORD14 DEFERRED SIZE 221922;
ALTER BUFFERPOOL ORD15 DEFERRED SIZE 221922;
ALTER BUFFERPOOL ORD16 DEFERRED SIZE 221922;
ALTER BUFFERPOOL ORD_I1 DEFERRED SIZE  949542;
ALTER BUFFERPOOL ORD_I2 DEFERRED SIZE  633028;
ALTER BUFFERPOOL ORD_I3 DEFERRED SIZE  633028;
ALTER BUFFERPOOL ORD_I4 DEFERRED SIZE  633028;
ALTER BUFFERPOOL ORD_I5 DEFERRED SIZE  949542;
ALTER BUFFERPOOL ORD_I6 DEFERRED SIZE  633028;
ALTER BUFFERPOOL ORD_I7 DEFERRED SIZE  633028;
ALTER BUFFERPOOL ORD_I8 DEFERRED SIZE  633028;
ALTER BUFFERPOOL ORD_I9 DEFERRED SIZE  949542;
ALTER BUFFERPOOL ORD_I10 DEFERRED SIZE 633028;
ALTER BUFFERPOOL ORD_I11 DEFERRED SIZE 633028;
ALTER BUFFERPOOL ORD_I12 DEFERRED SIZE 633028;
ALTER BUFFERPOOL ORD_I13 DEFERRED SIZE 949542;
ALTER BUFFERPOOL ORD_I14 DEFERRED SIZE 633028;
ALTER BUFFERPOOL ORD_I15 DEFERRED SIZE 633028;
ALTER BUFFERPOOL ORD_I16 DEFERRED SIZE 633028;
ALTER BUFFERPOOL OLN1 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN2 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN3 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN4 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN5 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN6 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN7 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN8 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN9 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN10 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN11 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN12 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN13 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN14 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN15 DEFERRED SIZE 435035;
ALTER BUFFERPOOL OLN16 DEFERRED SIZE 435035;
ALTER BUFFERPOOL CST1 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST2 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST3 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST4 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST5 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST6 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST7 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST8 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST9 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST10 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST11 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST12 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST13 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST14 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST15 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST16 DEFERRED SIZE 978828;
ALTER BUFFERPOOL CST_I1 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I2 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I3 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I4 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I5 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I6 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I7 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I8 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I9 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I10 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I11 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I12 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I13 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I14 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I15 DEFERRED SIZE 391531;
ALTER BUFFERPOOL CST_I16 DEFERRED SIZE 391531;
ALTER BUFFERPOOL STK1 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK2 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK3 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK4 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK5 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK6 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK7 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK8 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK9 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK10 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK11 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK12 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK13 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK14 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK15 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL STK16 DEFERRED SIZE 18021172;
ALTER BUFFERPOOL IBMDEFAULTBP  SIZE 1000;
ALTER BUFFERPOOL IBMDEFAULT8K  SIZE 1000;
ALTER BUFFERPOOL IBMDEFAULT16K SIZE 1000;
```

11.2. Data Generation

Makefile.config

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# Makefile.config - Linux 64-bit
#
#


# Make Configuration
MAKE=make

# Compiler Configuration.
# CFLAGS_DEBUG may be set to "-g", "-DDEBUGIT" "-g -DDEBUGIT" or left blank
CC=cc
CFLAGS_OS=-DSQLUNIX -DSQLLinux -O2 -fpic -m64
CFLAGS_OUT=-o
CFLAGS_DEBUG=

# Linker Configuration
LD_EXEC=gcc
LD_STORP=gcc
LDFLAGS_EXEC=
LDFLAGS_SHLIB=-shared
LDFLAGS_STORP=$(LDFLAGS_SHLIB)
LDFLAGS_LIB=-L$(TPCC_SQLLIB)/lib -ldb2 -m64
LDFLAGS_OUT=-o

# Library Configuration
AR=ar
ARFLAGS=-rv
ARFLAGS_LIB=
ARFLAGS_OUT=

# OS Commands
ERASE=rm -f
ERASEDIR=$(ERASE) -R
MOVE=mv
COPY=cp

# OS File Extensions & Path Separators
OBJEXT=.o
LIBEXT=.a
SHLIBEXT=.so
BINEXT=
SLASH=/
CMDSEP=;
```

Src.Common/Makefile

```
##########################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################
#######

#
# Makefile - Makefile for Src.Common
#
#

include $(TPCC_ROOT)/Makefile.config

#
##########################################################################
#
# Preprocessor, Compiler and LInker Flags
#
##########################################################################
#

PRP_OPTS =          PACKAGE \
                    OPTLEVEL 1 \
                    ISOLATION RR \
                    MESSAGES $*.prep.msg \
                    LEVEL $(TPCC_VERSION) \
                    NOLINEMACRO

INCLUDE =   -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

CFLAGS =    $(CFLAGS_OS) $(CFLAGS_DEBUG) $(INCLUDE) \
                    -DSQLA_NOLINES -D$(DB2EDITION) -D$(TPCC_SPTYPE)

UTIL_OBJ_DBG =      tpccdbg$(OBJEXT)
UTIL_OBJ_GEN =      tpccmisc$(OBJEXT)
UTIL_OBJ_DB2 =      tpccctx$(OBJEXT)

#
##########################################################################
#
# User Targets
#
##########################################################################
#

all:        $(UTIL_OBJ_DBG) $(UTIL_OBJ_GEN) connect $(UTIL_OBJ_DB2)
disconnect

dbgen:      $(UTIL_OBJ_GEN)

clean:
            - $(ERASE) *$(OBJEXT) *.bnd *.msg tpccctx.c

#
##########################################################################
#
# Helper Targets
#
##########################################################################
#

connect:
            - db2 connect to $(TPCC_DBNAME)

disconnect:
            - db2 connect reset
            - db2 terminate

#
##########################################################################
#
# Build Rules
#
##########################################################################
#

.SUFFIXES:
.SUFFIXES: $(OBJEXT) .c .sqc

.sqc.c:
            @echo "Prepping $*.sqc"
            db2 prep $*.sqc $(PRP_OPTS) bindfile
            db2 grant execute on package TPCCCTX to public

#
##########################################################################
#
# Dependencies
#
##########################################################################
#

# Source
tpccdbg$(OBJEXT):       tpccdbg.c
tpccctx$(OBJEXT):       tpccctx.c
tpccmisc$(OBJEXT):      tpccmisc.c

# Headers
tpccdbg.c:  $(TPCC_ROOT)/include/db2tpcc.h
```

Src.Common/tpccmisc.c

```
/**************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
**************************************************************************/

/*
 *
 * tpccmisc.c - Miscellaneous routines
 *
 */
```

```
#include <stdlib.h>
#include <sys/types.h>
#include <sys/time.h>

double current_time_ms(void);
double current_time(void);


/* Current time in SECONDS, precision SECONDS */
double current_time(void)
{
  /* use time() to get seconds */
  return(time(NULL));
}

/* Current time in SECONDS, precision MILLISECONDS */
double current_time_ms(void)
{
   /* gettimeofday() returns seconds and microseconds */
   /* convert to fractional seconds */
  struct timeval t;
  gettimeofday(&t,NULL);
  return (t.tv_sec + (double)t.tv_usec/(1000*1000));
}
```

dbgen/Makefile

```
##############################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##############################################################################
#######

# Makefile - Build gendata tool
#

include $(TPCC_ROOT)/Makefile.config

#
##############################################################################
#
# Preprocessor, Compiler and Linker Flags
#
##############################################################################
#

INCLUDE =       -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

CFLAGS =    $(INCLUDE) $(CFLAGS_OS) -DLINT_ARGS -DSQLA_NOLINES \
                        -D$(DB2EDITION) $(CFLAGS_DEBUG)

LDFLAGS =  $(LDFLAGS_EXEC) $(LDFLAGS_LIB)

#
##############################################################################
#
# File Collections
#
##############################################################################
#

OBJS =                  tpccrnd$(OBJEXT) \
                        $(TPCC_ROOT)/Src.Common/tpccmisc$(OBJEXT)
OBJ_EEE =  $(TPCC_ROOT)/Src.Common/tpcclwh$(OBJEXT)

EXEC =                  gendata$(BINEXT)

#
##############################################################################
#
# End-User Targets
#
##############################################################################
#

all:            $(EXEC)

clean:
                - $(ERASE) *$(OBJEXT) $(EXEC)

#
##############################################################################
#
# Build Rules
#
##############################################################################
#

.SUFFIXES:
.SUFFIXES:      $(OBJEXT) .c

# We use $@$(OBJEXT) here so that the UNIX makefiles work with both
# 'traditional' make and GNU make
$(EXEC):
                $(LD_EXEC) $(LDFLAGS) $(OBJS) $@$(OBJEXT) $(LDFLAGS_OUT)$@

#
##############################################################################
#
# Dependencies
#
##############################################################################
#

# Link Dependencies
gendata$(BINEXT):       $(OBJS) gendata$(OBJEXT)

# Build Dependencies
# Source
gendata$(OBJEXT):       gendata.c

# Headers
gendata.c:    $(TPCC_ROOT)/include/tpccrnd.h $(TPCC_ROOT)/include/lval.h
```

dbgen/gendata.c

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * gendata.c - Generate data for TPC-C database
 *
 */

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <sqlutil.h>
/* UNIX named pipe support */
#include <sys/stat.h>
#include <errno.h>
#include <fcntl.h>
#include <ctype.h>
#include <time.h>

#include "platform.h"
#include "db2tpcc.h"
#include "tpccrnd.h"
#include "tpccmisc.h"
#include "lval.h"


/*  PROTOTYPES.  */
void gen_dist_tbl( void );
void gen_cust_tbl( void );
void gen_hist_tbl( void );
void gen_nu_ord_tbl( void );
void gen_ordr_tbl( void );
void gen_item_tbl( void );
void gen_stock_tbl( void );
void gen_ware_tbl( void );

int i, j;
double timestamp1, timestamp2, elapse;
int rc, rc1, rc2;

int using_range = 0;
int using_npipe = 0;
int using_rctload = 0;
int quiet_mode = 0;
sqlint32 ware_start=-1, ware_end=-1;

char fmtWare[]    = "%s|%s|%s|%s|%s|%s|%04.4f|%.2f|%d\n";
char fmtDist[]    = "%d|%04.4f|%.2f|%s|%s|%s|%s|%s|%s|%d|%d\n";
char fmtItem[]    = "%s|%.2f|%s|%d|%d\n";
char fmtStock[]   = "%d|%d|%d|%d|%s|%s|%s|%s|%s|%s|%s|%s|%s|%s|%d|%d\n";
char fmtCust[]    =
"%d|%s|%s|%s|%s|%.2f|%s|%s|%04.4f|%s|%s|%s|%s|%s|%s|%d|%d|%d|%.2f|%.2f|%d\n";
char fmtHist[]    = "%d|%d|%d|%d|%d|%d|%s|%.2f|%s\n";
char fmtOrdr[]    = "%d|%s|%d|%d|%d|%d|%d|%d\n";
```

```
char fmtOLine[]   = "%s|%.2f|%d|%d|%d|%s|%d|%d|%d|%d\n";
char fmtNewOrd[]  = "%d|%d|%d\n";
void InitFormatStrings(char delim);
void ScalingReport(void);

int outtype1 = 0;
int outtype2 = 0;
char *outname1 = NULL;
char *outname2 = NULL;

/*-----------------------------------------------------------------------*/
/*      main                                                             */
/*-----------------------------------------------------------------------*/
int main (int argc, char *argv[])
{
   int option = -1;
   char *delim = NULL;

   /* ******************************************************************* */
   /* Compute Warehouse Ranges                                            */
   /* ******************************************************************* */
   ware_start = 1;
   ware_end = WAREHOUSES;

   /* ******************************************************************* */
   /* Process Command Line Arguments                                      */
   /* ******************************************************************* */

   /* Valid Command Line Options
    * --------------------------
    * Table Option:            -t <table>         (-t 3 for warehouse)
    * Output Column Delimiter:  -d <char>          (-d ' ', -d '|', etc)
    * Output to File:          -f[n] <file>       (-f customer.dat)
    * Output to Pipe:          -p[n] <pipename>   (-p tpccpipe.000)
    * Warehouse Range:         -r <start> <end>   (-r 1 100)
    * Scaling Report:          -s
    * Quiet Mode:              -q
    *
    * The -f[n] and/or -p[n] options are required.
    * The -t, -d, -r, -s and -q options are optional.
    *
    * If -d is omitted, the vertical bar (pipe) symbol ('|') will be used.
    * If -r is omitted, the range [1..WAREHOUSES] will be used.
    *
    * Due to the TPC-C spec requiring that orders and orderline be
    * generated at the same time, there is an extension to the -f and -p
    * options to specify one of the two output streams for each argument.
    *
    * -f1 orders.dat -f2 orderline.dat will output to two files
    * -f1 orders.dat -p2 tpccpipe.000 will output to a file and a pipe
    *
    * -f1/-p1 specifies the destination for the orders table
    * -f2/-p2 specifies the destination for the orderline table
    *
    */

   /* Read Arguments */
   for (i=1; i<argc; i++)
   {
      if (strcmp(argv[i], "-t") == 0) {
         option = atoi(argv[i+1]);
         i++;
      } else if (strcmp(argv[i], "-r") == 0) {
         ware_start = atoi(argv[i+1]);
         ware_end = atoi(argv[i+2]);
         i += 2;
      } else if (strcmp(argv[i], "-d") == 0) {
         delim = argv[i+1];
         i++;
      } else if ((strcmp(argv[i], "-f")  == 0) ||
                 (strcmp(argv[i], "-f1") == 0)) {
         outtype1 = IOH_FILE;
         outname1 = argv[i+1];
         i++;
      } else if (strcmp(argv[i], "-f2") == 0) {
         outtype2 = IOH_FILE;
         outname2 = argv[i+1];
         i++;
      } else if ((strcmp(argv[i], "-p")  == 0) ||
                 (strcmp(argv[i], "-p1") == 0)) {
         outtype1 = IOH_PIPE;
         outname1 = argv[i+1];
         i++;
      } else if (strcmp(argv[i], "-p2") == 0) {
         outtype2 = IOH_PIPE;
         outname2 = argv[i+1];
         i++;
      } else if (strcmp(argv[i], "-s") == 0) {
         ScalingReport();
         exit(0);
      } else if (strcmp(argv[i], "-q") == 0) {
         quiet_mode = 1;
      } else {
         fprintf(stderr, "gendata: Don't understand argument: %s\n",argv[i]);
         exit(-1);
      }
   }

   /* ******************************************************************* */
   /* Validate Command Line Arguments                                     */
   /* ******************************************************************* */

   /* Validate Table Argument */
   if (option < 3 || option > 11 || option == 10)
   {
      fprintf(stderr,"gendata: Invalid table selected: %d\n",option);
      exit(-1);
   }

   /* Validate Delimiter Argument */
   if (delim == NULL) {
     // default delimiter is used for IMPORT & LOAD, no changes neccessary
     using_rctload = 0;
   } else if (strlen(delim) == 1 && !isalnum(delim[0]) &&
              delim[0] != '.' && delim[0] != '%')
   {
      // user-supplied delimiter used for rctload
      InitFormatStrings(delim[0]);
      using_rctload = 1;
   } else {
      fprintf(stderr,"gendata: Invalid delimiter specified: %s\n",delim);
      exit(-1);
   }

   /* Validate File/Pipe Arguments */
   if (option != 9 && outtype1 > 0 && outtype2 > 0)
   {
      fprintf(stderr,"gendata: Specifying two output file/pipes allowed only when
generating\norders/orderline.\n");
      exit(-1);
   }
   if (option == 9 && ((outtype1 == 0) || (outtype2 == 0)))
   {
      fprintf(stderr,"gendata: Must specify two output file/pipes when generating
orders/orderline.\n");
      exit(-1);
   }
   if (outtype1 == 0 || outname1 == NULL || strcmp(outname1,"") == 0)
   {
      fprintf(stderr,"gendata: Invalid 1st output file/pipe specified.\n");
      exit(-1);
   }
   if (option == 9 && (outtype2 == 0 || outname2 == NULL || strcmp(outname2,"") == 0))
   {
      fprintf(stderr,"gendata: Invalid 2nd output file/pipe specified.\n");
      exit(-1);
   }
   /* Ensure O/OL flat files are opened in append mode.  This is required */
   /* because we generate O/OL concurrently.  See comments in genload.pl  */
   /* for further details on why this is neccessary.                      */
   if (option == 9)
   {
      if (outtype1 == IOH_FILE) outtype1 = IOH_FILE_APPEND;
      if (outtype2 == IOH_FILE) outtype2 = IOH_FILE_APPEND;
   }

   /* Validate Range Arguments */
   if (ware_start <= 0 || ware_start > WAREHOUSES) {
      fprintf(stderr,"gendata: Invalid range starting value: %d\n",ware_start);
       exit(-1);
   }
   if (ware_end <= 0 || ware_end > WAREHOUSES || ware_end < ware_start) {
      fprintf(stderr,"gendata: Invalid range ending value: %d\n",ware_end);
      exit(-1);
   }

   initialize_random();

   /* ******************************************************************* */
   /* Generate Data                                                       */
   /* ******************************************************************* */
   switch (option) {
    case 3: /* WAREHOUSE */
      gen_ware_tbl();
      break;
    case 4: /* DISTRICT  */
      gen_dist_tbl();
      break;
    case 5: /* ITEM  */
      gen_item_tbl();
      break;
    case 6: /* STOCK     */
      gen_stock_tbl();
      break;
    case 7: /* CUSTOMER  */
      gen_cust_tbl();
      break;
    case 8: /* HISTORY   */
      gen_hist_tbl();
      break;
    case 9: /* ORDERS + ORDER_LINE */
      gen_ordr_tbl();
      break;
    case 11: /* NEW_ORDER */
```

```
      gen_nu_ord_tbl();
      break;
    case 2:
    case 10:
    default:
      fprintf(stderr, "Error: invalid option = %d \n",(option));
      break;
  }
  return 0;
}

/*----------------------------------------------------------------------*/
/* generate item table                                                  */
/*----------------------------------------------------------------------*/

void gen_item_tbl( void )
{
  sqlint32  item_num = 0 ;
  sqlint32  item_im_id ;
  char  item_name[25] ;
  double  item_price ;
  char  item_data[51] ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto item_done; }

  for(item_num = 1; item_num <= ITEMS; item_num++)
  {
    /* create image id field */
    item_im_id = rand_integer( 1, 10000 ) ;
    /* create name field */
    create_random_a_string( item_name, 14, 24);
    /* create price field */
    item_price = rand_decimal( 100, 10000, 2 ) ;
    /* create ORIGINAL field */
    create_a_string_with_original( item_data, 26, 50, 10) ;

    numBytes = sprintf(Buffer, fmtItem,
                       item_name,
                       item_price,
                       item_data,
                       item_im_id,
                       item_num);

    rc = GenericWrite(&hnd, Buffer, numBytes);
    if (rc != 0) { goto item_done; }

  } /* end for... */

  rc = GenericClose(&hnd);

item_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nITEM table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nITEM table FAILED at (I %d) after %8.2f
seconds.\n\n",item_num,elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------------*/
/* generate stock table                                                 */
/*----------------------------------------------------------------------*/
void gen_stock_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 stock_num = 0 ;
  sqlint32 stock_quant;
  sqlint32 s_ytd;
  sqlint32 s_order_cnt, s_remote_cnt;
  char  stock_dist_01[25] ;
  char  stock_dist_02[25] ;
  char  stock_dist_03[25] ;
  char  stock_dist_04[25] ;
  char  stock_dist_05[25] ;
  char  stock_dist_06[25] ;
  char  stock_dist_07[25] ;
  char  stock_dist_08[25] ;
  char  stock_dist_09[25] ;
  char  stock_dist_10[25] ;
  char  stock_data[51] ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto stock_done; }

  for (stock_num = 1; stock_num <= STOCK_PER_WAREHOUSE; stock_num++)
  {
    if (!quiet_mode && (stock_num%500 == 0))
    {
      fprintf(stdout, "STOCK for Item #%d\n",stock_num);
      fflush(stdout);
    }
    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
      stock_quant = rand_integer( 10, 100 ) ;
      create_random_a_string( stock_dist_01, 24, 24);
      create_random_a_string( stock_dist_02, 24, 24);
      create_random_a_string( stock_dist_03, 24, 24);
      create_random_a_string( stock_dist_04, 24, 24);
      create_random_a_string( stock_dist_05, 24, 24);
      create_random_a_string( stock_dist_06, 24, 24);
      create_random_a_string( stock_dist_07, 24, 24);
      create_random_a_string( stock_dist_08, 24, 24);
      create_random_a_string( stock_dist_09, 24, 24);
      create_random_a_string( stock_dist_10, 24, 24);

      /* create ORIGINAL field */
      create_a_string_with_original( stock_data, 26, 50, 10 );
      s_ytd = s_order_cnt = s_remote_cnt = 0;

      numBytes = sprintf(Buffer, fmtStock,
                         s_remote_cnt,
                         stock_quant,
                         s_order_cnt,
                         s_ytd,
                         stock_data,
                         stock_dist_01,
                         stock_dist_02,
                         stock_dist_03,
                         stock_dist_04,
                         stock_dist_05,
                         stock_dist_06,
                         stock_dist_07,
                         stock_dist_08,
                         stock_dist_09,
                         stock_dist_10,
                         stock_num,
                         ware_num);

      rc = GenericWrite(&hnd, Buffer, numBytes);
      if (rc != 0) { goto stock_done; }

    } /* end for... */
  } /* end for... */

  rc = GenericClose(&hnd);

stock_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nSTOCK table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nSTOCK table FAILED at (S %d W %d) after %8.2f
seconds.\n\n",stock_num, ware_num, elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------------*/
/* generate warehouse table                                             */
/*----------------------------------------------------------------------*/
void gen_ware_tbl( void )
{
  sqlint32 ware_num = 0 ;
  char ware_name[11] ;
  char ware_street_1[21] ;
  char ware_street_2[21] ;
  char ware_city[21] ;
  char ware_state[3]  ;
  char ware_zip[10]  ;
  double ware_tax ;
  double ware_YTD ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
```

```
  if (rc != 0) { goto ware_done; }

  for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
  {
    if (!quiet_mode && ((ware_num % 500) == 0)) {
      fprintf(stdout, "Warehouse #%d\n", ware_num);
      fflush(stdout);
    }

    create_random_a_string( ware_name,      6,10) ; /* create name */
    create_random_a_string( ware_street_1, 10,20) ; /* create street 1 */
    create_random_a_string( ware_street_2, 10,20) ; /* create street 2 */
    create_random_a_string( ware_city,     10,20) ; /* create city */
    create_random_a_string( ware_state,     2,2) ;  /* create state */
    create_random_n_string( ware_zip,       4,4) ;  /* create zip */
    strcat(ware_zip, "11111");

    ware_tax = rand_decimal(0, 2000,4);
    ware_YTD = 300000.00;

    numBytes = sprintf(Buffer, fmtWare,
                       ware_name,
                       ware_street_1,
                       ware_street_2,
                       ware_city,
                       ware_state,
                       ware_zip,
                                          ware_tax,
                                          ware_YTD,
                       ware_num);

    rc = GenericWrite(&hnd, Buffer, numBytes);
    if (rc != 0) { goto ware_done; }

  } /* end for */

  rc = GenericClose(&hnd);

ware_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nWAREHOUSE table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nWAREHOUSE table FAILED at (W %d) after %8.2f
seconds.\n\n",ware_num, elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------------*/
/* generate dist table                                                  */
/*----------------------------------------------------------------------*/
void gen_dist_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 dist_num = 0 ;
  char dist_name[11];
  char dist_street_1[21];
  char dist_street_2[21];
  char dist_city[21];
  char dist_state[3];
  char dist_zip[10];
  double dist_tax;
  sqlint32 next_o_id;
  double dist_YTD;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  next_o_id = CUSTOMERS_PER_DISTRICT + 1;
  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto dist_done; }

  for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
  {
    if (!quiet_mode) {
      fprintf(stdout, "DISTRICT for Warehouse #%d\n", ware_num);
      fflush(stdout);
    }
    for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
    {
      create_random_a_string( dist_name,      6,10) ; /* create name */
      create_random_a_string( dist_street_1, 10,20) ; /* create street 1 */
      create_random_a_string( dist_street_2, 10,20) ; /* create street 2 */
      create_random_a_string( dist_city,     10,20) ; /* create city */
      create_random_a_string( dist_state,     2,2) ;  /* create state */
      create_random_n_string( dist_zip,       4,4) ;  /* create zip */
      strcat(dist_zip, "11111");
      dist_tax = rand_decimal(0, 2000,4);
      dist_YTD = 30000.00;

      numBytes = sprintf(Buffer, fmtDist,
                         next_o_id,
                         dist_tax,
                         dist_YTD,
                         dist_name,
                         dist_street_1,
                         dist_street_2,
                         dist_city,
                         dist_state,
                         dist_zip,
                         dist_num,
                         ware_num);

      rc = GenericWrite(&hnd, Buffer, numBytes);
      if (rc != 0) { goto dist_done; }

    } /* end for... */
  } /* end for... */

  rc = GenericClose(&hnd);

dist_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nDISTRICT table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nDISTRICT table FAILED at (W %d D %d) after %8.2f
seconds.\n\n",ware_num,dist_num,elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------------*/
/* generate customer table                                              */
/*----------------------------------------------------------------------*/
void gen_cust_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 dist_num = 0 ;
  sqlint32 cust_num = 0 ;
  char cust_last[17];
  char cust_middle[3];
  char cust_first[17];
  char cust_street_1[21];
  char cust_street_2[21];
  char cust_city[21];
  char cust_state[3];
  char cust_zip[10];
  char cust_phone[17];
  char cust_credit[3];
  char cust_data[501];
  char cust_since[27];
  double cust_discount;
  double cust_balance;
  double cust_YTD_payment;
  double cust_credit_lim;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];
  int len, pos;

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto cust_done; }

  strcpy(cust_middle, "OE");

  createTimestampString(cust_since);


  for (cust_num = 1; cust_num <= CUSTOMERS_PER_DISTRICT; cust_num++)
  {
    if (!quiet_mode) {
      fprintf(stdout, "CUSTOMER #%d:\n", cust_num);
      fflush(stdout);
    }

    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
      for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
      {
        if (cust_num <= 1000)                    /* create last name */
          create_random_last_name( cust_last, cust_num);
        else                                     /* create last name */
          create_random_last_name( cust_last, 0);
        create_random_a_string( cust_first,     8,16) ; /* create first name */
        create_random_a_string( cust_street_1, 10,20) ; /* create street 1 */
        create_random_a_string( cust_street_2, 10,20) ; /* create street 2 */
```

```
        create_random_a_string( cust_city,     10,20) ; /* create city */
        create_random_a_string( cust_state,     2,2) ;  /* create state */
        create_random_n_string( cust_zip,       4,4) ;  /* create zip */
        strcat(cust_zip, "11111");

        /* create phone number */
        create_random_n_string( cust_phone, 16,16) ;
        if ( rand_integer( 1, 100 ) <= 10 )
          strcpy( cust_credit, "BC" ) ;
        else
          strcpy( cust_credit, "GC" ) ;

        /* create discount rate */
        cust_discount = rand_decimal(0,5000,4);

        /* create customer data */
        create_random_a_string(cust_data, 300, 500);

        /* pad customer data (only for non-rctload) */
        if (using_rctload == 0) {
          for (pos=strlen(cust_data); pos<500; pos++)
            cust_data[pos] = ' ';
          cust_data[500] = '\0';
        }

        cust_credit_lim = 50000.00;
        cust_balance = -10.00;
        cust_YTD_payment = 10.00;

        if (cust_num == 1 && dist_num == 1 && ware_num == 1)
        {
          sprintf(cust_first,"C_LAST_LOAD=%d",C_C_LAST_LOAD);
        }

        numBytes = sprintf(Buffer, fmtCust,
                    cust_num,
                    cust_state,
                    cust_zip,
                    cust_phone,
                    cust_since,
                    cust_credit_lim,
                    cust_middle,
                    cust_credit,
                    cust_discount,
                    cust_data,
                    cust_last,
                    cust_first,
                    cust_street_1,
                    cust_street_2,
                    cust_city,
                    dist_num,
                    ware_num,
                    0,
                    cust_balance,
                    cust_YTD_payment,
                    1);

        rc = GenericWrite(&hnd, Buffer, numBytes);
        if (rc != 0) { goto cust_done; }

      } /* end for district... */
    } /* end for warehouse... */
  } /* end for customer... */

  rc = GenericClose(&hnd);

cust_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nCUSTOMER table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nCUSTOMER table FAILED at (W %d D %d C %d) after %8.2f
seconds.\n\n",ware_num, dist_num, cust_num, elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------------*/
/* generate hist table                                                  */
/*----------------------------------------------------------------------*/
void gen_hist_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 dist_num = 0 ;
  sqlint32 cust_num = 0 ;
  char hist_data[25] ;
  char h_date[27] ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto hist_done; }

  createTimestampString(h_date);

  for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
  {
    if (!quiet_mode) {
      fprintf(stdout, "HISTORY for Warehouse #%d:\n", ware_num);
      fflush(stdout);
    }
    for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
    {
      for (cust_num = 1; cust_num <= CUSTOMERS_PER_DISTRICT; cust_num++)
      {
        /* create history data */
        create_random_a_string( hist_data, 12,24) ;

        numBytes = sprintf(Buffer, fmtHist,
                    cust_num,
                    dist_num,
                    ware_num,
                    dist_num,
                    ware_num,
                    h_date,                    //@d37088nob
                                               10.00,                    //@d37088nob
                    hist_data);

        rc = GenericWrite(&hnd, Buffer, numBytes);
        if (rc != 0) { goto hist_done; }

      } /* end for customer... */
    } /* end for district... */
  } /* end for warehouse... */

  rc = GenericClose(&hnd);

hist_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nHISTORY table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nHISTORY table FAILED at (W %d D %d C %d) after %8.2f
seconds.\n\n",ware_num, dist_num, cust_num, elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------------*/
/* generate nu_ord table                                                */
/*----------------------------------------------------------------------*/
void gen_nu_ord_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 dist_num = 0 ;
  sqlint32 nu_ord_id = 0 ;
  int nu_ord_hi ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  /* compute maximum and minimum
    order numbers for this
    district */
  nu_ord_hi = CUSTOMERS_PER_DISTRICT - NU_ORDERS_PER_DISTRICT + 1;
  if (nu_ord_hi < 0) {
    nu_ord_hi = CUSTOMERS_PER_DISTRICT - (CUSTOMERS_PER_DISTRICT / 3) +
1;
    fprintf(stderr,"\n**** WARNING **** NU_ORDERS_PER_DISTRICT is >
CUSTOMERS_PER_DISTRICT\n");
    fprintf(stderr,"                Check the values in file lval.h\n");
    fprintf(stderr,"                Loading New-Order with 1/3 of
CUSTOMERS_PER_DISTRICT\n");
  }

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto neword_done; }

  /* We generate in O/W/D order for non-RCT tables.  With the
   * data clustered on O_ID, this gives us good bufferpool
   * characteristics.  We also create a btree index in W/D/O
   * order, to satisfy MIN(O_ID) queries.
   *
   * For RCT tables *with* RCT Jump Cache, we *should* generate
   * the data in W/D/O order (to match the table definition.)
   * We don't since it would push schema decisions into flat file
   * generation (and I don't want to do that.)  It's just as easy
```

```
    * to sort the flat files afterwards.
    */

   for (nu_ord_id = nu_ord_hi;
        nu_ord_id <= CUSTOMERS_PER_DISTRICT;
        nu_ord_id++)
   {
     if (!quiet_mode) {
       fprintf(stdout, "NEW_ORDER for Customer #%d:\n", nu_ord_id);
       fflush(stdout);
     }
     for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
     {
       for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
       {
         numBytes = sprintf(Buffer, fmtNewOrd,
                            nu_ord_id,
                            dist_num,
                            ware_num);

         rc = GenericWrite(&hnd, Buffer, numBytes);
         if (rc != 0) { goto neword_done; }

       } /* end for... */
     } /* end for... */
   } /* end for... */

   rc = GenericClose(&hnd);

neword_done:

   timestamp2 = current_time();
   elapse = timestamp2 - timestamp1;
   if (rc == 0) {
     if (!quiet_mode) {
       fprintf(stdout,"\nNEW_ORDER table generated in %8.2f seconds.\n\n",elapse);
       fflush(stdout);
     }
   } else {
     fprintf(stderr,"\nNEW_ORDER table FAILED at (W %d D %d O %d) after %8.2f
seconds.\n\n",ware_num, dist_num, nu_ord_id, elapse);
     fflush(stderr);
   }
}

/*-----------------------------------------------------------------------*/
/* generate order and order_line tables                                  */
/*-----------------------------------------------------------------------*/
void gen_ordr_tbl( void )
{
   sqlint32 ware_num = 0 ;
   sqlint32 dist_num = 0 ;
   sqlint32 cust_num = 0 ;
   sqlint32 ord_num = 0 ;
   sqlint32 ordr_carrier_id;
   sqlint32 ordr_ol_cnt;
   sqlint32 oline_ol_num;
   sqlint32 oline_item_num;

   double oline_amount;                                  //@d37088nob
   char oline_dist_info[25];

   IOH_NUM numBytes;
   ioHandle hnd1, hnd2;
   char Buffer[1024];

   char currtmstmp[27];
   char nulltmstmp[27] = "0001-01-01 00:00:00";

   oline_dist_info[24] = '\0';

   timestamp1 = current_time();

   rc1 = GenericOpen(&hnd1, outtype1, outname1);
   if (rc1 != 0) { goto ool_done; }
   rc2 = GenericOpen(&hnd2, outtype2, outname2);
   if (rc2 != 0) { goto ool_done; }

   createTimestampString(currtmstmp);

   for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
   {
     if (!quiet_mode) {
       fprintf(stdout, "ORDERS & ORDER_LINE for Warehouse #%d\n", ware_num);
       fflush(stdout);
     }
     for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
     {
       if (!quiet_mode) {
         fprintf(stdout, "District #%d\t", dist_num);
         fflush(stdout);
       }

       seed_1_3000();

       for (ord_num = 1; ord_num <= CUSTOMERS_PER_DISTRICT; ord_num++)
       {
         if (ord_num < 2101)
           ordr_carrier_id = rand_integer( 1, 10 ) ;
         else
           ordr_carrier_id = 0;

         cust_num = random_1_3000();
         ordr_ol_cnt = rand_integer(MIN_OL_PER_ORDER,MAX_OL_PER_ORDER);

         numBytes = sprintf(Buffer, fmtOrdr,
                            cust_num,
                            currtmstmp,
                            ordr_carrier_id,
                            ordr_ol_cnt,
                            1,
                            ord_num,
                            ware_num,
                            dist_num);

         rc1 = GenericWrite(&hnd1, Buffer, numBytes);
         if (rc1 != 0) { goto ool_done; }

         for ( oline_ol_num = 1; oline_ol_num <= ordr_ol_cnt; oline_ol_num++ )
         {
           oline_item_num = rand_integer(1, ITEMS) ;
           create_random_a_string( oline_dist_info, 24, 24) ;

           numBytes = sprintf(Buffer, fmtOLine,
                              ((ord_num < 2101) ? currtmstmp : nulltmstmp),
                              ((ord_num < 2101) ? 0.00 : rand_decimal(1,999999,2)),
//@d37088nob
                              oline_item_num,
                              ware_num,
                              5,
                              oline_dist_info,
                              ord_num,
                              dist_num,
                              ware_num,
                              oline_ol_num);

           rc2 = GenericWrite(&hnd2, Buffer, numBytes);
           if (rc2 != 0) { goto ool_done; }

         } /* for order_line */
       } /* for order */
     } /* for dist */
   } /* for ware */

   rc1 = GenericClose(&hnd2);
   rc2 = GenericClose(&hnd1);

ool_done:

   timestamp2 = current_time();
   elapse = timestamp2 - timestamp1;
   if (rc1 == 0 && rc2 == 0) {
     if (!quiet_mode) {
       fprintf(stdout,"\nORDERS & ORDER_LINE tables generated in %8.2f
seconds.\n\n",elapse);
       fflush(stdout);
     }
   } else {
     fprintf(stderr,"\nORDERS & ORDER_LINE tables FAILED at (W %d D %d O %d
OL %d) after %8.2f seconds.\n\n",ware_num, dist_num, ord_num, oline_ol_num, elapse);
     fflush(stderr);
   }
}

// This routine will initalize the printf format strings and replace the
// delimiter with the one provided.  The pipe symbol is the default.
void InitFormatStrings(char delim)
{
  char *p;

  // Check if Using Default Delimiter
  if (delim == '|') return;

  // Replace Delimiters
  while (p = strchr(fmtWare,'|'))   { *p = delim; }
  while (p = strchr(fmtDist,'|'))   { *p = delim; }
  while (p = strchr(fmtItem,'|'))   { *p = delim; }
  while (p = strchr(fmtStock,'|'))  { *p = delim; }
  while (p = strchr(fmtCust,'|'))   { *p = delim; }
  while (p = strchr(fmtHist,'|'))   { *p = delim; }
  while (p = strchr(fmtOrdr,'|'))   { *p = delim; }
  while (p = strchr(fmtOLine,'|'))  { *p = delim; }
  while (p = strchr(fmtNewOrd,'|')) { *p = delim; }
}

void ScalingReport(void)
{
   /* Print Scaling Values */
   fprintf(stdout,"Scaling Values in Use\n");
   fprintf(stdout,"-------------------------------\n");
   fprintf(stdout,"Warehouses:            %d\n", WAREHOUSES);
   fprintf(stdout,"Districts/Warehouse:   %d\n", DISTRICTS_PER_WAREHOUSE);
   fprintf(stdout,"Customers/District:    %d\n", CUSTOMERS_PER_DISTRICT);
   fprintf(stdout,"Items:                 %d\n", ITEMS);
```

```
   fprintf(stdout,"Stock/Warehouse:       %d\n", STOCK_PER_WAREHOUSE);
   fprintf(stdout,"Min Order Lines/Order: %d\n", MIN_OL_PER_ORDER);
   fprintf(stdout,"Max Order Lines/Order: %d\n", MAX_OL_PER_ORDER);
   fprintf(stdout,"New Orders/District:   %d\n", NU_ORDERS_PER_DISTRICT);
   fprintf(stdout,"-------------------------------\n");
}
```

dbgen/tpccrnd.c

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tpccrnd.c - Random generation functions for TPC-C
 *
 */

#include <stdio.h>
#include <string.h>
#include <math.h>
#include "db2tpcc.h"
#include "tpccmisc.h"
#include "lval.h"

#include <stdlib.h>

static char tbl_cust[CUSTOMERS_PER_DISTRICT];

static char alnum[] =
   "0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ";

static char *last_name_parts[] =
{
   "BAR",
   "OUGHT",
   "ABLE",
   "PRI",
   "PRES",
   "ESE",
   "ANTI",
   "CALLY",
   "ATION",
   "EING"
};

/*
*****************************************************************************
*   rand_integer
*
*   create a uniform random numeric value of type integer, of random
*   value between lo and hi. Number is NOT placed in BUFFER, and IS
*   simply RETURNED.
*
*   Routine RETURNS the VALUE.
*
*   parameters
*   ----------
*   lo end of acceptable value range
*   hi end of acceptable value range
*
*   output
*   ------
*   random integer value  RETURNED
*
*****************************************************************************
*/

int rand_integer ( int val_lo, int val_hi )
{
  return((random()%(val_hi-val_lo+1))+val_lo);
}

/*
*****************************************************************************
*   rand_decimal
*
*   create a uniform random numeric value of type double, of random
*   value between lo and hi with val_dec fractional digits.
*   Number is NOT placed in BUFFER, and IS simply RETURNED.
*
*   Routine RETURNS the VALUE.
*
*   parameters
*   ----------
*   lo end of acceptable value range
*   hi end of acceptable value range
*   number of fractional digits
*
*   output
*   ------
*   random double value  RETURNED
*
*****************************************************************************
*/

double rand_decimal ( int val_lo, int val_hi, int val_dec )
{
  return(rand_integer(val_lo,val_hi)/pow(10.0,(double)val_dec));
}

/*
*****************************************************************************
*   seed_1_3000
*
*
*****************************************************************************
*/

void seed_1_3000( void )
{
  int i;

  for (i = 0; i < CUSTOMERS_PER_DISTRICT; i++) {
    tbl_cust[i] = 0;
  }
}

/*
*****************************************************************************
*   random_1_3000
*
*
*****************************************************************************
*/

int random_1_3000( void )
{
  static int i;
  static int x;

  x = rand_integer(0, CUSTOMERS_PER_DISTRICT - 1);

  for (i = 0; i < CUSTOMERS_PER_DISTRICT; i++)
  {
    if (tbl_cust[x] == 0)
    {
      tbl_cust[x] = 1;
      return(x+1);
    } else {
      x++;
    }
    if (x == CUSTOMERS_PER_DISTRICT)
      x=0;
  }

  printf("\nfatal error in random_1_3000 \n");
  abort();
}


/*
*****************************************************************************
*   initialize_random
*****************************************************************************
*/

void initialize_random(void)
{
  int t = current_time();

  srand(t);
  srandom(t);
}


/*
*****************************************************************************
*   create_random_a_string
*
*   create a random alphanumeric string, of random length between lo and
*   hi and place them in designated buffer. Routine returns the actual
*   length.
*
*   parameters
*   ----------
*   lo end of acceptable length range
*   hi end of acceptable length range
*
*   output
*   ------
*   actual length
*   random alphanumeric string
*
```

```
*****************************************************************************
*/

int create_random_a_string( char *out_buffer, int length_lo, int length_hi )
{
  int i, actual_length ;

  actual_length = rand_integer( length_lo, length_hi ) ;

  for (i = 0; i < actual_length; i++ )
  {
    out_buffer[i] = alnum[rand_integer( 0, 61 )] ;
  }
  out_buffer[actual_length] = '\0' ;

  return (actual_length);
}


/*
*****************************************************************************
*   create_random_n_string
*
*   create a random numeric string, of random length between lo and
*   hi and place them in designated buffer. Routine returns the actual
*   length.
*
*   parameters
*   ----------
*   lo end of acceptable length range
*   hi end of acceptable length range
*
*   output
*   ------
*   actual length
*   random numeric string
*
*****************************************************************************
*/

int create_random_n_string( char *out_buffer, int length_lo, int length_hi )
{
  int i, actual_length ;

  actual_length = rand_integer( length_lo, length_hi ) ;

  for (i = 0; i < actual_length; i++ )
  {
    out_buffer[i] = (char)rand_integer( 48,57 ) ;
  }
  out_buffer[actual_length] = '\0' ;

  return (actual_length);
}

/*
*****************************************************************************
*   NUrand_val
*
*   create a non-uniform random numeric value of type integer, of random
*   value between lo and hi. Number is NOT placed in BUFFER, and IS
*   simply RETURNED.
*
*   Routine RETURNS the VALUE.
*
*   parameters
*   ----------
*   lo end of acceptable value range
*   hi end of acceptable value range
*
*   output
*   ------
*   random integer value  RETURNED
*
*****************************************************************************
*/

int NUrand_val ( int A, int x, int y, int C )
{
  return((((rand_integer(0,A)|rand_integer(x,y))+C)%(y-x+1))+x);
}

/*
*****************************************************************************
*   create_a_string_with_original
*
*   create a random alphanumeric string, of random length between lo and
*   hi and place them in designated buffer. Routine returns the actual
*   length.
*
*   the word "ORIGINAL" is placed at a random location in the buffer at
*   random, for a given percent of the records.
*
*   percent_to_set must be an integer value from 0 to 100.
*   if 0, no records will be set. If 100, all records will be set.
*
*   CANNOT USE ON STRINGS OF LENGTH LESS THAN 8 ! LOWER LIMIT MUST BE >
8 !
*
*   parameters
*   ----------
*   lo end of acceptable length range
*   hi end of acceptable length range
*   percentage of records to set to ORIGINAL
*
*   output
*   ------
*   actual length
*   random alphanumeric string with the word "ORIGINAL" is placed at a
*       random location
*
*****************************************************************************
*/

int create_a_string_with_original( char *out_buffer, int length_lo,
                                   int length_hi, int percent_to_set )
{
  int actual_length, start_pos ;

  actual_length = create_random_a_string( out_buffer, length_lo, length_hi ) ;

  if ( rand_integer( 1, 100 ) <= percent_to_set )
  {
    start_pos = rand_integer( 0, actual_length-8 ) ;
    strncpy(out_buffer+start_pos,"ORIGINAL",8) ;
  }

  return (actual_length);
}


/*****************************************************************************
 *
 * create_random_last_name
 *
 * parameters:
 *    out_buffer - target buffer for the generated last name
 *
 * description:
 *    create_random_last_name generates a random number from 0 to 999
 *    inclusive. a random name is generated by associating a random string
 *    with each digit of the generated number. the three strings are
 *    concatenated to generate the name
 *
 *****************************************************************************/

int create_random_last_name(char *out_buffer, int cust_num)
{
 int random_num;

 if (cust_num == 0)
   random_num = NUrand_val( A_C_LAST, 0, 999, C_C_LAST_LOAD );
 else
   random_num = cust_num - 1;

 strcpy(out_buffer, last_name_parts[random_num / 100]);
 random_num %= 100;
 strcat(out_buffer, last_name_parts[random_num / 10]);
 random_num %= 10;
 strcat(out_buffer, last_name_parts[random_num]);

 return(strlen(out_buffer));
}
```

include/lval.h

```
/* lval.h - generated automatically at 20101007.1242 */


#ifndef __LVAL_H

#define __LVAL_H

#define WAREHOUSES 184032

#define DISTRICTS_PER_WAREHOUSE 10

#define CUSTOMERS_PER_DISTRICT 3000

#define ITEMS 100000

#define STOCK_PER_WAREHOUSE 100000

#define MIN_OL_PER_ORDER 5

#define MAX_OL_PER_ORDER 15

#define NU_ORDERS_PER_DISTRICT 900
```

```
#endif // __LVAL_H
```

include/platform.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * platform.h - Platform Isolation Layer
 *
 */

#ifndef __PLATFORM_H
#define __PLATFORM_H

/* ***************************************************************************/
/* Generic Macros                                                            */
/* ***************************************************************************/
#define GEN_ERRCODE   errno

/* ***************************************************************************/
/* Windows I/O Macros                                                        */
/* ***************************************************************************/

/* ***************************************************************************/
/* UNIX I/O Macros                                                           */
/* ***************************************************************************/
#include <fcntl.h>

#define IOH_INIT(hnd, type, name)                                   \
   hnd->fd = -1;                                                    \
   hnd->type = type;                                                \
   hnd->name = name;

#define IOH_CREATE(hnd)                                             \
   if (hnd->type == IOH_PIPE) {                                     \
      rc = mkfifo(hnd->name, 0666);                                 \
   } else {                                                         \
      rc = 0;                                                       \
   }

#define IOH_OPEN(hnd)                                               \
   if (hnd->type == IOH_FILE_APPEND) {                              \
     hnd->fd = open(hnd->name, O_WRONLY | O_CREAT | O_APPEND, 0666);      \
   } else {                                                         \
     hnd->fd = open(hnd->name, O_WRONLY | O_CREAT | O_TRUNC, 0666);       \
   }                                                                \
   if (hnd->fd == -1) {                                             \
      rc = -1;                                                      \
   } else {                                                         \
      rc = 0;                                                       \
   }

#define IOH_WRITE(hnd, buff, num, num2)                             \
   rc = write(hnd->fd, buff, num);                                  \
   if (rc >= 0) {                                                   \
      num2 = rc;                                                    \
      rc = 0;                                                       \
   }

#define IOH_FLUSH(hnd)  rc = 0;
#define IOH_CLOSE(hnd) rc = close(hnd->fd);
#define IOH_DELETE(hnd)                 if (hnd->type == IOH_PIPE) { rc = unlink(hnd-
>name); }

typedef unsigned int IOH_NUM;
typedef int IOH_HND;

/*****************************************************************************/
/* UNIX Semaphore Macros                                                     */
/*****************************************************************************/
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/sem.h>
#include <unistd.h>

union semun {
  int val;
  struct semid_ds *buf;
  unsigned short int *array;
} semUnion;

struct sembuf semBuf;

#define SEM_HANDLE int

#define SEM_INIT(hnd, x, name)                                      \
   if ( (hnd = semget(IPC_PRIVATE, 1 , IPC_CREAT | IPC_EXCL | S_IRUSR | S_IWUSR |
S_IRGRP | S_IWGRP | S_IROTH | S_IWOTH)) == -1)                      \
      API_ERROR(__LINE__, "semget", (rc=GEN_ERRCODE));              \
   semUnion.val = x;                                                \
   if ( semctl(hnd, 0, SETVAL, semUnion) < 0 )                      \
      API_ERROR(__LINE__, "semctl SETVAL", (rc=GEN_ERRCODE));

#define SEM_WAIT(hnd)                                               \
   semBuf.sem_num = 0;                                              \
   semBuf.sem_op = -1;                                              \
   semBuf.sem_flg = SEM_UNDO;                                       \
   if ( semop(hnd, &semBuf, 1) < 0 )                                \
      API_ERROR(__LINE__, "semop wait", (rc=GEN_ERRCODE));

#define SEM_FREE(hnd)                                               \
   semBuf.sem_num = 0;                                              \
   semBuf.sem_op = 1;                                               \
   semBuf.sem_flg = SEM_UNDO;                                       \
   if ( semop(hnd, &semBuf, 1) < 0 )                                \
      API_ERROR(__LINE__, "semop free", (rc=GEN_ERRCODE));

#define SEM_DESTROY(hnd)                                            \
   if ( semctl(hnd, 0, IPC_RMID, 0) )                               \
      API_ERROR(__LINE__, "semctl IPC_RMID", (rc=GEN_ERRCODE));


/* ***************************************************************************/
/* Common I/O Macros and Definitions                                         */
/* ***************************************************************************/
#define IOH_FILE 1
#define IOH_PIPE 2
#define IOH_FILE_APPEND 3

#define IOH_ERRMSG(hnd, msg)                                        \
  if (rc != 0) {                                                    \
    fprintf(stderr, "Error %d %s fd %d (%d, %s)\n", GEN_ERRCODE, msg,      \
                hnd->fd, hnd->type, hnd->name);                     \
    return rc;                                                      \
  }

struct _ioh {
  IOH_HND fd;
  int type;
  char *name;
};

typedef struct _ioh ioHandle;

/* ***************************************************************************/
/* Generic I/O Routine Prototypes                                            */
/* ***************************************************************************/
int GenericOpen(ioHandle *hnd, int type, char *name);
int GenericWrite(ioHandle *hnd, char *Buffer, unsigned int numBytes);
int GenericClose(ioHandle *hnd);

/* ***************************************************************************/
/* Generic I/O Routines                                                      */
/* ***************************************************************************/
int GenericOpen(ioHandle *hnd, int type, char *name)
{
   int rc = 0;

   IOH_INIT(hnd, type, name)

   IOH_CREATE(hnd)
   IOH_ERRMSG(hnd, "creating")

   IOH_OPEN(hnd)
   IOH_ERRMSG(hnd, "opening")

   return rc;
}

int GenericWrite(ioHandle *hnd, char *Buffer, unsigned int numBytes)
{
   int rc = 0;
   int numBytesWritten = -1;

   IOH_WRITE(hnd, Buffer, numBytes, numBytesWritten)
   IOH_ERRMSG(hnd, "writing")
   if (numBytes != numBytesWritten) {
      fprintf(stderr, "Truncated data writing to fd %d (%d, %s)\n", hnd->fd, hnd->type, hnd-
>name);
      rc = -1;
   }
   return rc;
}

int GenericClose(ioHandle *hnd)
{
   int rc = 0;

   IOH_FLUSH(hnd)
   IOH_ERRMSG(hnd, "flushing")

   IOH_CLOSE(hnd)
```

```
  IOH_ERRMSG(hnd, "closing")

  IOH_DELETE(hnd)
  IOH_ERRMSG(hnd, "deleting")

  return rc;
}

#endif // __PLATFORM_H
```

include/tpccmisc.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tpccmisc.h - Miscellanouse Routines
 *
 */

#ifndef __TPCCMISC_H
#define __TPCCMISC_H


extern double current_time_ms(void);
extern double current_time(void);

#include <time.h>
#define createTimestampString(buf)                    \
{                                                                                  \
  time_t now;                                  \
  struct tm *tm;                               \
  time(&now);                                  \
  tm = localtime(&now);                        \
  sprintf(buf,                                 \
        "%4.4d-%2.2d-%2.2d %2.2d:%2.2d:%2.2d",       \
        tm->tm_year + 1900, tm->tm_mon + 1, tm->tm_mday, \
        tm->tm_hour, tm->tm_min, tm->tm_sec);        \
}

#endif // __TPCCMISC_H
```

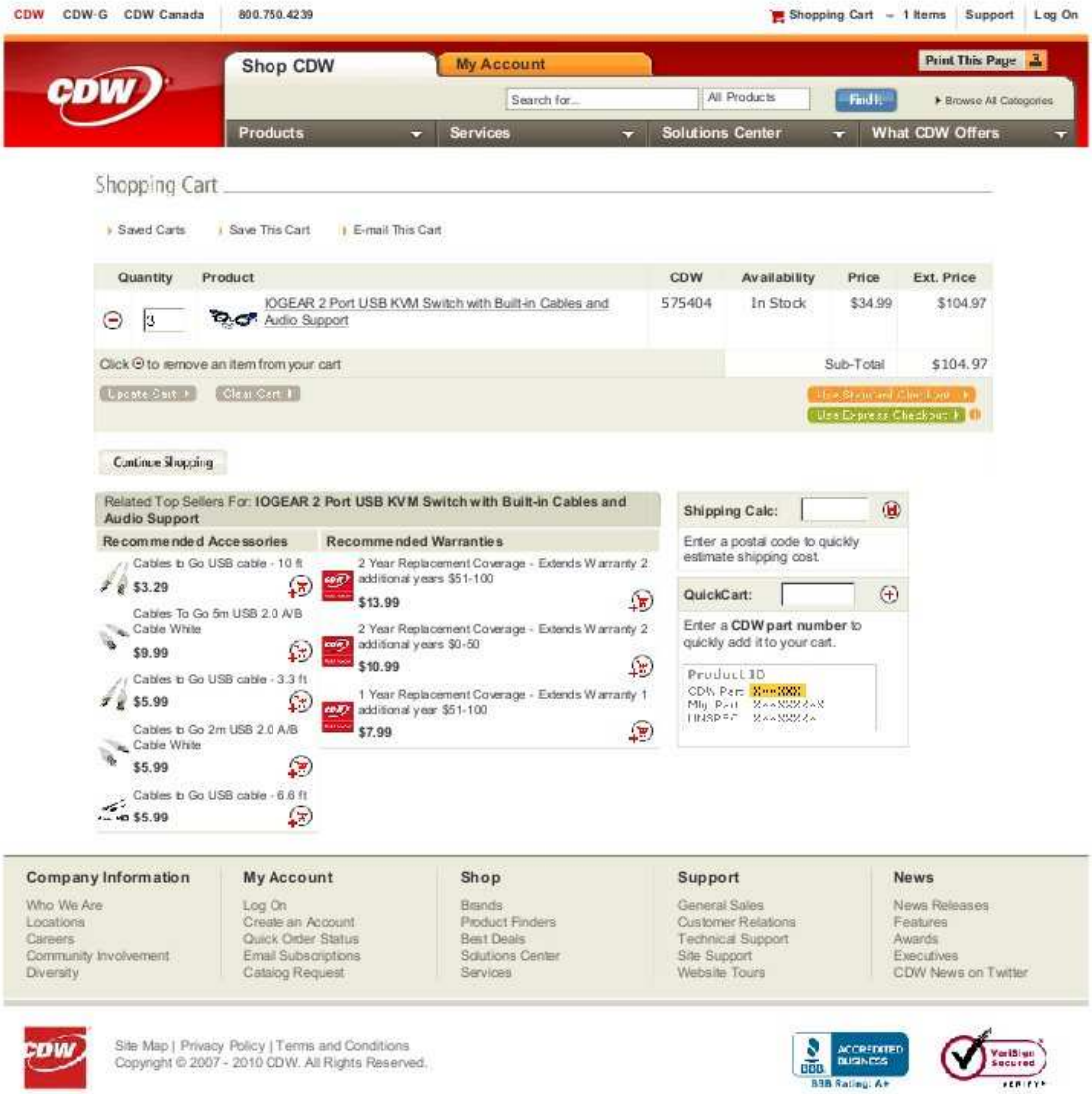
include/tpccrnd.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tpccrnd.h - Random generation functions for TPC-C
 *
 */

#ifndef __TPCCRND_H
#define __TPCCRND_H

void initialize_random(void);
int rand_integer( int val_lo, int val_hi ) ;
double rand_decimal( int val_lo, int val_hi, int val_dec ) ;
int NUrand_val( int A, int val_lo, int val_hi, int C ) ;

void seed_1_3000( void );
int random_1_3000( void );

int create_random_a_string( char *out_buffer,
                        int length_lo,
                        int length_hi );
int create_random_n_string( char *out_buffer,
                        int length_lo,
                        int length_hi );
int create_a_string_with_original( char *out_buffer,
                            int length_lo,
                            int length_hi,
                            int percent_to_set ) ;
int create_random_last_name(char *out_buffer, int cust_num);

#endif // __TPCCRND_H
```

tpccenv.sh

```
#######################################################################
######
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#######################################################################
#######

#
# tpccenv.sh - UNIX Environment Setup
#

# The Kit Version
export TPCC_VERSION=CK080718

# The DB2 Instance Name (for DB2)
export DB2INSTANCE=${USER}

# The OS being used (i.e. "UNIX", "LINUX", "WINDOWS")
export PLATFORM=LINUX
export SERVER_PLATFORM=UNIX

# The type of make command and slash used by the OS.
# (i.e. UNIX - "/", WINDOWS - "\").
# These are referenced all over the kit.
export SLASH="/";
export MAKE=make

# Specifies whether or not to use dari stored proc's for the TPC-C driver.  Set to either
DARIVERSION or NONDARI;
#export TPCC_SPTYPE=NOSP
#export TPCC_SPTYPE=SPGENERAL2
export TPCC_SPTYPE=SPGENERAL
#export TPCC_SPTYPE=DARI2SQLDA

# The schema name is typically the SQL autorization ID (or username).
# This is required for runstats and EEE.
export TPCC_SCHEMA=${USER}
export SERVER_TPCC_SCHEMA=${USER}

# DB2 EE/EEE Configuration
export DB2EDITION=EE
#export DB2EDITION=EEE
export DB2NODE=0
export DB2NODES=1;                    # set to the number of nodes you have.  Set to 1
for EE.

# TPCC General Configuration
export TPCC_DBNAME=TPCC
export TPCC_ROOT=${HOME}/tpc-c.ibm
export TPCC_SQLLIB=${HOME}/sqllib
export TPCC_RUNDATA=${HOME}/tpccdata

# TPCC Debug Configuration
# This is the path where all error and debug logs are placed.
# To get debugging from within the stored procedures, you must
# set DB2ENVLIST="TPCC_DEBUGDIR" in tpcc.config.
export TPCC_DEBUGDIR=/tmp

# Specifies where stored procedures should be placed and if they should
# be fenced.
export TPCC_SPDIR=${TPCC_SQLLIB}/function
export TPCC_FENCED=NO

# changes by KMR
# MTE says aaaahhh!  Nooooo!
export EDITOR=emacs
```

12 Appendix D: Pricing



CDW CDW-G CDW Canada | 800.800.4239

Shopping Cart 2 Items | Support | Log On

Shop CDW | My Account | Print This Page

Search for... All Products Find It Browse All Categories

Products | Services | Solutions Center | What CDW Offers

Shopping Cart

Saved Carts | Save This Cart | E-mail This Cart

Quantity	Product	CDW	Availability	Price	Ext. Price
5	NETGEAR GS748T 48-port Gigabit Smart Switch	698653	In Stock	$659.99	$3,299.95
1	Acer A181HLB 18.5" Wide LED	2146072	In Stock	$97.99	$97.99
Click ⊖ to remove an item from your cart				Sub-Total	$3,397.94

Update Cart | Clear Cart | Use Standard Checkout | Use Express Checkout

Continue Shopping

CDW CDW-G CDW Canada | 800.750.4239

Shopping Cart 1 Items | Support | Log On

Shop CDW | My Account | Print This Page

Search for... All Products Find It Browse All Categories

Products | Services | Solutions Center | What CDW Offers

Shopping Cart

Saved Carts | Save This Cart | E-mail This Cart

Quantity	Product	CDW	Availability	Price	Ext. Price
5	NETGEAR GS748T 48-port Gigabit Smart Switch	698653	In Stock	$659.99	$3,299.95
Click ⊖ to remove an item from your cart				Sub-Total	$3,299.95

Update Cart | Clear Cart | Use Standard Checkout | Use Express Checkout

Continue Shopping

Novell

LOGIN

Solutions | Products | Services & Support | Partners & Communities | About Novell | How to Buy

SHOPPING CART — BILLING — SHIPPING — REVIEW ORDER — PLACE ORDER

Reference Number: 8934255421

Qty	Description	Stock Status	Delivery	Unit Price	Subtotal	
5	SUSE Linux Enterprise Server for x86 and x86_64 1 Year Priority Support	Instant Delivery	Per Server Subscription	$1,499.00	$4,497.00	remove
5	SUSE Linux Enterprise Server 11 for x86, AMD64 & Intel64 e-Software Media Kit	Downloadable	e-Software	$0.00	$0.00	remove
5	High Availability Extension SUSE Linux Enterprise Server 11 for x86, AMD64 & Intel64 e-Software Media Kit	Downloadable	e-Software	$0.00	$0.00	remove

Update

Coupon Code apply

Order Total: $4,497.00

continue shopping

Proceed to Checkout

WE ACCEPT: MasterCard Maestro VISA American Express JCB Diners CHECK LINE OF CREDIT WIRE TRANSFER

PLEASE NOTE: If the product you have ordered is In Stock, your order will ship within 2 business days. If your product is Built to Order, you will receive an e-mail notification when your order is dispatched.
UPS - Ground is only available for US lower 48 States
- For UPS Ground shipments, please allow 5 - 7 business days for delivery.
UPS - Air - Next Day Air is only available for US lower 48 States
UPS - Air - Worldwide Express is available for all international shipments
- For UPS - Air - Worldwide Express, please allow 2 - 4 business days for delivery.
International Shipments: Shipping charges do not include any Brokerage Fees, Customs Fees or Taxes that you may be charged.

SECURED BY GeoTrust click to verify

Security

Your personal information is protected on our order form by Secure Socket Layer (SSL) technology.

Digital River Inc. is the authorised reseller and merchant of the products and services offered within this store.

Privacy Policy Terms of Sale

Novell. Making IT Work As One™ Careers | Contact Us | Feedback | Legal | Print

© 2010 Novell

ALLHDD

CALL: 718-766-5269

VeriSign Trusted VERIFY

McAfee SECURE TESTED DAILY 11-NOV

HOME | CATALOG | RETURN | MY ACCOUNT | VIEW CART

Hard Drives | Servers | Memory | Power Supply | Controllers | Networking | Accessories | Processors | Mother Board | Tape Drives | Battery | Enclosures

Search: GO

CATEGORIES

- Hard Drives
- Servers
- Memory
- Power Supply
- Controllers
- Networking
- Accessories
- Processors
- Mother Board
- Tape Drives
- Battery
- Enclosures

MANUFACTURERS

- Select a manufacturer

INFORMATION

- About Us
- Contact Us
- Free Ground Shipping
- Privacy Policy
- Return Policy

Products: 3
Subtotal: $1,569.00

» View cart » Checkout

CART CONTENT

QLE2464 QLogic SanBlade PCI Express 4GB Fibre Channel 4 Ports Host Bus Adapter. New Bulk Pack.

Part Number: QLE2464

Quantity: 3 X $523.00 = $1,569.00

Clear cart | Update

Estimate shipping cost

Subtotal: $1,569.00

Total cost: $1,569.00

CONTINUE SHOPPING | CHECKOUT

McAfee SECURE TESTED DAILY 11-NOV

FREE UPS GROUND SHIPPING

MY ACCOUNT

» Log in / Register
» Orders

Track my order(s):
Order ID/E-mail

SITE NEWS

Submit e-mail address to receive free updates and promotions:
Enter e-mail address

RECENTLY VIEWED

QLE2464 QLogic SanBlade PCI Express 4GB Fibre Channel 4 Ports Host Bus Adapter. New Bulk Pack.

Home > My Shopping Cart

MY SHOPPING CART

My Wish Lists | Print Cart | Email Cart

Update Qtys | Remove Selected | Move Selected To ...

	Qty.	Product Description	Savings	Total Price
	3	powercom KIN-1500AP 1500 VA 900 Watts UPS Item #: N82E16842106115 Return Policy: Standard Return Policy Protect Your Investment (expand for options)	-$15.00 Instant	~~$449.97~~ $404.97 ($134.99 each)

Subtotal: $404.97

Calculate Shipping

Zip Code: UPS Guaranteed 2 Day Service GO

Shipping: $0.00

Redeem Newegg Gift Cards

Card Number: Security Code: Apply

Apply Promo Codes: Enter up to 5 promo codes. Please note that stacking is not permitted.

Promo Code: Apply

(For Example:code1,code2,code3,code4,code5)

Promo Code: $0.00

BillMeLater

No Payments + No Interest if paid in full in 6 Months on order over $250.

Subject to credit approval. See Terms

Grand Total:* $404.97

* Above total does not include shipping or taxes. Please input zip code to calculate your grand total.

Having problems with your cart? Chat Now for help or try emptying your cart to start over.

view important shipping information.

Continue Shopping | Checkout »

Policy & Agreement | Privacy Policy © 2000-2010 Newegg Inc. All rights reserved.

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Tel 425 882 8080
Fax 425 936 7329
http://www.microsoft.com/

Microsoft

November 11, 2010

IBM Corporation
Andrea Davis
11501 Burnet Road
Austin, TX 78758

Here is the information you requested regarding pricing for several Microsoft products to be used in conjunction with your TPC-C benchmark testing.

All pricing shown is in US Dollars ($).

Part Number	Description	Unit Price	Quantity	Price
LWA-00984	**Windows Web Server 2008 R2** *Full License* *No Discounts Applied*	$469	16	$7,504
127-00166	**Microsoft Visual Studio 2008 Professional** *No Discounts Applied*	$799	1	$799
N/A	**Microsoft Problem Resolution Services** *Professional Support* *(1 Incident).*	$259	1	$259

All Microsoft products listed above are currently orderable and available through Microsoft's normal distribution channels. A list of Microsoft's resellers can be found at the Microsoft Product Information Center at
http://www.microsoft.com/products/info/render.aspx?view=22&type=how

Defect support is included in the purchase price. Additional support is available from Microsoft PSS on an incident by incident basis at $259 per call.

This quote is valid for the next 90 days.

Reference ID: TPCC_g3wOpiq6ZAtgdnQQtWllgNjU7f+RiLyr_V1.0.0.

400 Riverpark Drive Suite 300
North Reading, MA 01864
PHONE: 1-800-332-8638
FAX: 1-800-257-0568
www.avnetexpress.com

DATE: 5-12-2011
PAGES: 1

ATTENTION: Ray Venditti
COMPANY: IBM
FROM: Sandra Nocivelli

PLEASE REFER TO QUOTE #:H255 5457560

QTY	*PART #*	*MFG*	*QTD PRICE*	*DELIVERY*	*COMMENTS*
8	LSI00217	LSI	$2041.17/ea	2wks	**Avnet Terms & Conditions**

- PRICE AND DELIVERY ARE SUBJECT TO CHANGE
- ALL ORDERS ARE SUBJECT TO AVNET STANDARD TERMS AND CONDITIONS.
- Quote valid for Thirty(30)Days

CLEARCOMM
TECHNICAL SALES, INC.

SMART®
Modular Technologies

IBM Corporation
Ray Venditti
11401 Burnet Rd.
Austin, TX 78758

May 04, 2011

Dear Ray,

Noted below is the information you requested regarding pricing for the SMART Modular 2.5" SAS Solid State Drive (SSD) to be used in conjunction with your TPC benchmark testing.

All prices shown are in US Dollars ($). Pricing does not include applicable taxes or shipping charges, which may vary.

Part Number	Description	Unit Price [1]	Quantity [2]	Extended Price
SG9XCA2E200GE01	200GB, 2.5" XceedIOPS SAS Solid-State Drive, eMLC	$1,800.00	71	$127,800.00

[1] Unit price includes special product replacement support (7 day replacement TAT). Certain restrictions apply.
[2] The minimum order quantity on this part number is 10 units.

The SMART SG9XCA2E200GE01 is currently available. For price confirmation and order assistance, please contact Ray Eustace at ClearComm Technical Sales at (919) 859-4400, or via email at smart@clearcommsales.com.

Notes:
1. This pricing is valid for thirty (30) days from the date of quotation.
2. All units are covered by a 5 year warranty from the date of shipment from SMART, subject to usage conditions defined in the product data sheet.

Sincerely,

Raymond Eustace

Raymond Eustace
ClearComm Technical Sales (representative of SMART Modular Technologies, Inc.)
919-859-4400
reustace@clearcommsales.com

Ref quote ID: SMART ClearComm 20110504RV SG9XCA2E200GE01

cc: B. Heinze / SMART Modular

1215 Jones Franklin Road, Suite 103, Raleigh, NC 27606
Main Office: (919) 859-4400 ♦ Fax: (888) 334-4031