Accessing Teradata University for Academics on Walton Enterprise Systems

(8/7/2020)

**Sources**

Ron Freeze, Michael Gibbs, Evelyn Lee

Enterprise Systems, Sam M. Walton College of Business, University of Arkansas, Fayetteville

Teradata Viewpoint 16.50.01.00-b710

Copyright © 2020 *For educational uses only - adapted from sources with permission. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission from the author/presenter.*

# Use Case – Teradata University

The purpose for this document is to understand and demonstrate the processes of requesting Teradata accounts, navigating a remote desktop, executing queries, and batch loading database structures and data. These steps require access to VMware, ViewPoint, and Teradata Studio Express (also referred to as Teradata Studio in this document).

When navigating personal databases, it should be noted that faculty and students are allotted identical permanent space in Teradata to create their own databases, views, and more. Additionally, identical functionality for faculty and students is given such as “SELECT”, “CREATE”, “GRANT”, “INSERT”, “UPDATE”, and “DELETE” SQL functions. This can be done by referring to the given SQL User ID in database commands and queries. Faculty also have additional permissions to view more databases and alter student databases from the students’ perm space for the purposes of modifying and demonstrating Teradata functionality.

Faculty with new textbooks that wish to have their data on the new Teradata system at the University of Arkansas should contact Michael Gibbs and Ron Freeze. They are also available via email for additional support with creating accounts, accessing enterprise data sets, and general questions about the system.

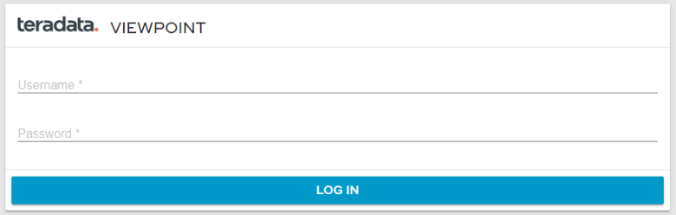
* Michael Gibbs at [mgibbs@walton.uark.edu](mailto:mgibbs@walton.uark.edu)
* Ron Freeze at [rfreeze@walton.uark.edu](mailto:rfreeze@walton.uark.edu)

## Step 1: Request Access to Teradata System

1. Refer to the document “Virtual Access Guide” to view the tutorial on how to request access to the Teradata system.

## Step 2: Accessing ViewPoint

Teradata ViewPoint is a browser-based application whose primary purpose is the use of SQL statements. ViewPoint offers SQL Scratchpad, a program with limited functionality allowing faculty and students to execute simple SQL queries for the datasets for which you have been given access.

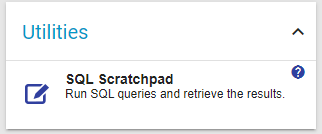
1. Navigate to <http://uatdviewpoint.waltoncollege.uark.edu/login.html>
2. Enter in your University of Arkansas username and password provided by your instructor.
3. Click “Log in”

You will be taken to a blank VIEWPOINT page.

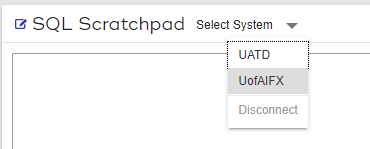
1. Click “Add Content” in the upper right corner.

On this page, you may search for content or click on the content available to you.

1. Find SQL Scratchpad, click it, and add it to your homepage.

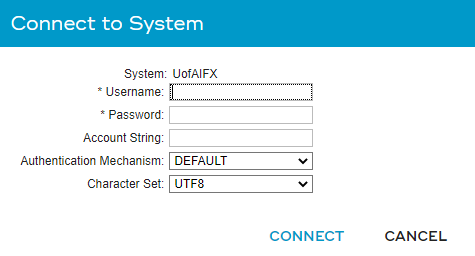


Adding SQL Scratchpad will allow you to execute SQL queries on the Teradata datasets available to you through the University of Arkansas.

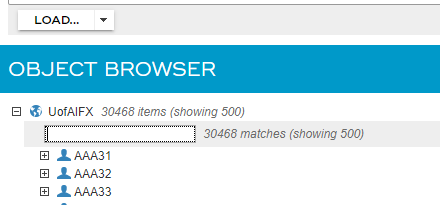


1. Click “Select System” beside the title “SQL Scratchpad” and then click “UofAIFX”.

Selecting this allows you to query from the University of Arkansas Teradata system.

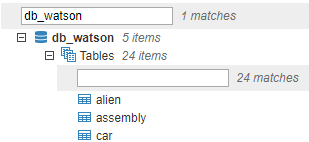
1. Enter your University of Arkansas username and password once more into the pop-up box.

If the pop-up box goes away, your username and password entry is successful. You can ignore the fields “Account String”, “Authentication Mechanism”, and “Character Set”.

1. Click “Load” and then “Insert Object” to find your desired database.

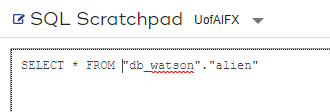
We will be querying from “db\_watson” for this example. Your account may not have access to view db\_watson. Your instructor can provide your authorized datasets that can be used for this tutorial. You can find this using the search function available.

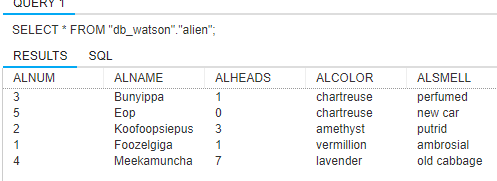
Once you find db\_watson, you can explore the tables that reside under this database and the associated views, if applicable.



1. Click on the table you want to query and select “Insert Object” below.

In this case, we will be querying db\_watson.alien. The above box will be filled with the database name followed by the table name with a “.” in between. We will alter this to be an appropriate SQL statement.

1. Type “SELECT \* FROM” before the populated content and then click “Run”.

SQL Scratchpad will show you the query typed and the results from that query including the column names and data rows. You can also click the “SQL” tab beside “Results” to see the specific SQL statement used to generate those results.

## Step 3: Using VMware

For those students and faculty needed to use Teradata Studio, you will need to gain this access through the Walton Enterprise System virtual desktop – VMware. You can install VMware on your local desktop or use the browser version to access your remote desktop. All accounts using Teradata Viewpoint are also granted access to VMware using the same credentials. Either the downloaded VMware or browser version are acceptable and provide you with the same access. Faculty and students will need to use VMware to access Teradata Studio.

1. Refer to the document “VMware Guide via Windows Client” to view the tutorial on how to download, install, and use VMware on your computer.
2. Refer to the document “VMware Guide via Browser” to view the tutorial on how to access VMware through your browser.

## Step 4: Teradata Studio

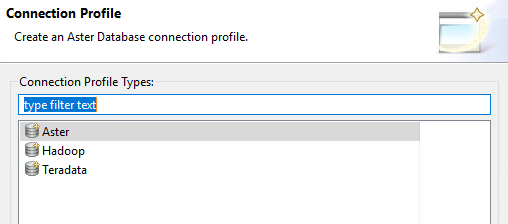
### Accessing

Teradata Studio is a more robust application used to access the Teradata IntelliFlex system. Teradata Studio will be the main faculty and students’ point of access unless using ViewPoint in order to learn SQL. There will be limited support for Teradata SQL Assistant (another application from Teradata) through the end of 2020. Teradata SQL Assistant will be phased out at the end of 2020.

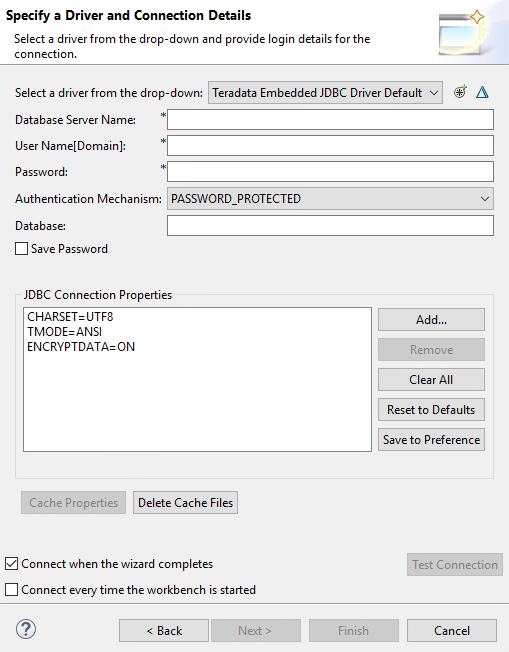
On your desktop, you will see the Teradata Studio icon.

1. Double click the Teradata Studio icon

Teradata Studio will open. If you have logged in before, Teradata Studio may prompt you for your password to the University of Arkansas IntelliFlex system. Please enter your password given by your professor. If you have not logged in before, you will be prompted with a Quick Tour which you may go through if you wish. After clicking out of the Quick Tour, you will be prompted to create a connection profile.

1. Select “Teradata” from the Connection Profile window and click “Next”

After clicking next, you will be prompted with a new window, “Specify a Driver and Connection Details”

Specify the drive and connection details by entering in the database server name, and your username and password given to you by your professor.

**Database Server Name:** uofaifx.walton.uark.edu

User Name(Domain): (your username)

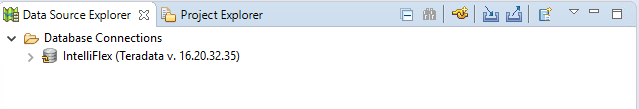
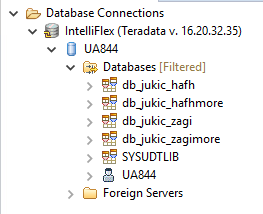
**Password:** (your password)

If you are faculty, your username and password are given by the University of Arkansas through the request system. If you are a student, your professor will give you a dedicated username and password.

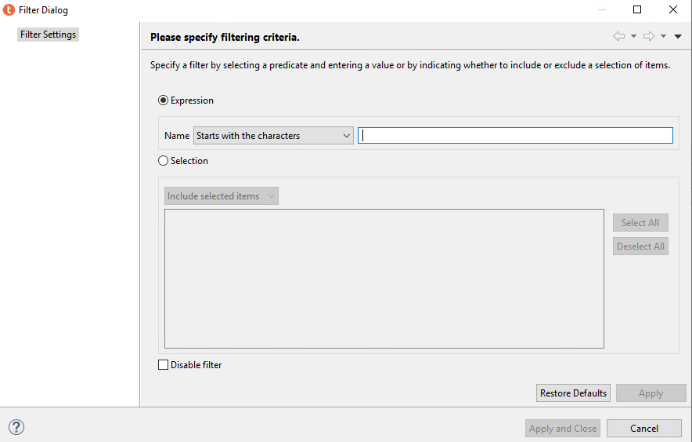
You may leave the rest of the dialog boxes blank. Once in Teradata Studio, you will see several different features of the application with which you can get familiar. The next three sections (Navigating, Executing Queries and Batch Loading) will help you get familiar with Teradata Studio.

### Navigating

This section explains how to navigate around Teradata Studio. We will explore how to manage queries and add and remove databases.

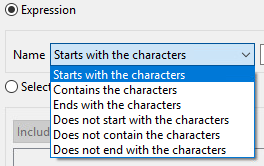
1. View the Data Source Explorer on the left.

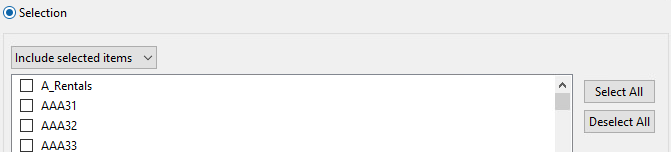
This will show you the databases you can view in the Database Connections folder. You can explore which databases are available to you, and expand the Intelliflex (Teradata v 16.20.32.35) icon to see the databases set up for you. For example, this student (UA844) can see their connections from the IntelliFlex dropdown. This student’s databases are filtered, a feature which is shown in the next step.

1. Right click on the “Databases” folder and select “Filter”.

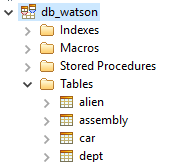
You are free to filter to any database by typing an expression. **This is also how you “add” and “remove” databases from your view in Teradata Studio.** Filter to view the preferred databases and remove the database from the filter to remove it from view and from your dropdown.

There are two methods to filtering. First, you can filter to include only what you type, such as selecting “Starts with the characters” and then type “db\_wat” to return “db\_watson”. Doing this a second time will negate previous filters, not add to your current filters.

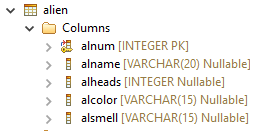
Other methods for the expression include any phrase shown in this picture. These will help narrow down a selection to something specific without going through individual databases.

You may also select the “Selection” circle and select from the generated list in the dialog box below. This may take a while to load.

You may also disable the filter if you do not which to have this folder filtered anymore.

1. Expand your preferred database. For this example, we will use “db\_watson”
2. Expand the tables folder. This shows all of the existing tables in the database you selected.

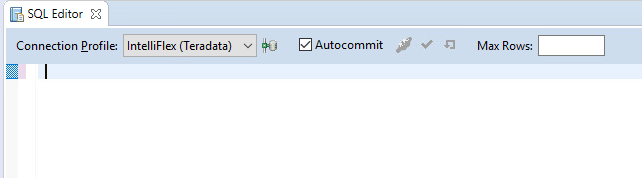
In this example, we can see that the database “db\_watson” has tables titled “alien”, “assembly”, “car”, “dept”, and more that are not listed here. We will now explore individual tables.

1. Expand a table of your choice and expand the Columns folder for that table.

In this case we can see that the “alien” table has five columns that reside in the table. Each of these columns are defined by a data type (e.g., integer, varchar, etc.), a data length (e.g., 20, 15, etc.), and whether the field can be null.

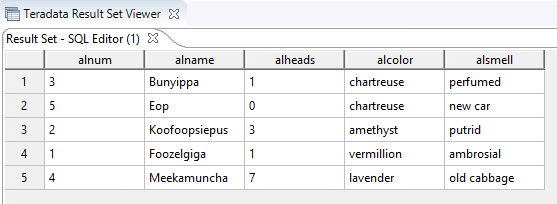
### Executing Queries

Executing SQL queries is the same for faculty and students. We advise that faculty execute queries on their databases to ensure correct data, and student execute queries on their databases to ensure they have SELECT access to the desired databases.

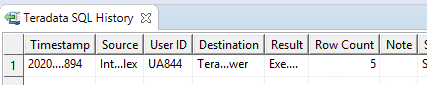
1. Navigate to the SQL Editor in Teradata Studio, and click on the dialog box below.
2. Verify that the Connection Profile says “IntelliFlex (Teradata)”.
3. Define the maximum number of rows of data to be returned, if desired.

Defining the maximum number of rows to be returned can be useful if you would like to execute a query faster if there is a lot of data that will be queried, or to have a quick view of how the data is organized in the table being queried.

1. Type “SELECT \* FROM db\_watson.alien into the SQL Editor
2. Press F5 to execute the command

This SQL statement will select all of the data available in the “alien” table from the database “db\_watson”. The results will be shown in the Teradata Result Set Viewer below the SQL Editor.

To see your SQL history, or all of the SQL commands you have executed during your Teradata Studio session, view the Teradata SQL History area below the Teradata Result Set Viewer.

1. View your SQL history in the Teradata SQL History section of Teradata Studio

This section contains important information about the SQL statements you’ve executed or attempted to execute. In this example, we can see our “UA844” ID executed the above query, the result (successful or not), how many rows were in our result (Row Count), and other information. If a SQL command does not execute properly, the error will be explained in the Result column.

## Step 5: Batch Loading Data Using Database Scripts

Teradata University for Academics provides database scripts to copy the database structure and data into student perm spaces. Students can copy and paste these scripts into Teradata Studio or Viewpoint and execute the queries to create databases and import data quickly.

1. Navigate to the next section of this document (Step 5a), **Textbook Dataset Scripts**.

Locate the textbook that your instructor has picked for your class. Under the textbook name, there may be several dataset names followed by their respective scripts. These can be used to insert the database structure and data into your own database. If your textbook is not listed, your instructor may be using another dataset than those listed. For this example, we will look at “db\_watson”.

1. Scroll down and view “db\_watson”.

The script can be copied and pasted into Teradata Studio or ViewPoint and executed as mentioned above. Below, you can find the scripts that the University of Arkansas supports.

## Step 5a: Textbook Database Scripts

### Elmasri, R. & Navathe, S.

Fundamentals of Database Systems (6e)

#### db\_company\_elmasri

CREATE TABLE DEPARTMENT AS db\_company\_elmasri.DEPARTMENT WITH NO DATA;

CREATE TABLE DEPENDENT AS db\_company\_elmasri.DEPENDENT WITH NO DATA;

CREATE TABLE DEPT\_LOCATIONS AS db\_company\_elmasri.DEPT\_LOCATIONS WITH NO DATA;

CREATE TABLE EMPLOYEE AS db\_company\_elmasri.EMPLOYEE WITH NO DATA;

CREATE TABLE PROJECT AS db\_company\_elmasri.PROJECT WITH NO DATA;

CREATE TABLE WORKS\_ON AS db\_company\_elmasri.WORKS\_ON WITH NO DATA;

INSERT INTO DEPARTMENT SELECT \* FROM db\_company\_elmasri.DEPARTMENT;

INSERT INTO DEPENDENT SELECT \* FROM db\_company\_elmasri.DEPENDENT;

INSERT INTO DEPT\_LOCATIONS SELECT \* FROM db\_company\_elmasri.DEPT\_LOCATIONS;

INSERT INTO EMPLOYEE SELECT \* FROM db\_company\_elmasri.EMPLOYEE;

INSERT INTO PROJECT SELECT \* FROM db\_company\_elmasri.PROJECT;

INSERT INTO WORKS\_ON SELECT \* FROM db\_company\_elmasri.WORKS\_ON;

COLLECT STATISTICS ON DEPARTMENT INDEX (Dnumber);

COLLECT STATISTICS ON DEPENDENT INDEX (Essn);

COLLECT STATISTICS ON DEPT\_LOCATIONS INDEX (Dnumber);

COLLECT STATISTICS ON EMPLOYEE INDEX (Ssn);

COLLECT STATISTICS ON PROJECT INDEX (Pnumber);

COLLECT STATISTICS ON WORKS\_ON INDEX (Essn);

### Gillenson, M.

Fundamentals of Database Management Systems (2e)

#### db\_fdms\_hcl

CREATE TABLE CRUISE AS db\_fdms\_hcl.CRUISE WITH NO DATA;

CREATE TABLE PASSENGER AS db\_fdms\_hcl.PASSENGER WITH NO DATA;

CREATE TABLE PORT AS db\_fdms\_hcl.PORT WITH NO DATA;

CREATE TABLE SHIP AS db\_fdms\_hcl.SHIP WITH NO DATA;

CREATE TABLE VISIT AS db\_fdms\_hcl.VISIT WITH NO DATA;

CREATE TABLE VOYAGE AS db\_fdms\_hcl.VOYAGE WITH NO DATA;

INSERT INTO CRUISE SELECT \* FROM db\_fdms\_hcl.CRUISE;

INSERT INTO PASSENGER SELECT \* FROM db\_fdms\_hcl.PASSENGER;

INSERT INTO PORT SELECT \* FROM db\_fdms\_hcl.PORT;

INSERT INTO SHIP SELECT \* FROM db\_fdms\_hcl.SHIP;

INSERT INTO VISIT SELECT \* FROM db\_fdms\_hcl.VISIT;

INSERT INTO VOYAGE SELECT \* FROM db\_fdms\_hcl.VOYAGE;

COLLECT STATISTICS ON CRUISE INDEX (CRUISENUM);

COLLECT STATISTICS ON PASSENGER INDEX (PASSENGERNUM);

COLLECT STATISTICS ON PORT INDEX (PORTNAME);

COLLECT STATISTICS ON SHIP INDEX (SHIPNUM);

COLLECT STATISTICS ON VISIT INDEX (CRUISENUM);

COLLECT STATISTICS ON VOYAGE INDEX (PASSENGERNUM);

### Pratt, P.

A Guide to SQL (7e)

#### db\_gts\_pp

CREATE TABLE CUSTOMER AS db\_gts\_pp.CUSTOMER WITH NO DATA;

CREATE TABLE ORDERS AS db\_gts\_pp.ORDERS WITH NO DATA;

CREATE TABLE ORDER\_LINE AS db\_gts\_pp.ORDER\_LINE WITH NO DATA;

CREATE TABLE PART AS db\_gts\_pp.PART WITH NO DATA;

CREATE TABLE REP AS db\_gts\_pp.REP WITH NO DATA;

INSERT INTO CUSTOMER SELECT \* FROM db\_gts\_pp.CUSTOMER;

INSERT INTO ORDERS SELECT \* FROM db\_gts\_pp.ORDERS;

INSERT INTO ORDER\_LINE SELECT \* FROM db\_gts\_pp.ORDER\_LINE;

INSERT INTO PART SELECT \* FROM db\_gts\_pp.PART;

INSERT INTO REP SELECT \* FROM db\_gts\_pp.REP;

COLLECT STATISTICS ON CUSTOMER INDEX (CUSTOMER\_NUM);

COLLECT STATISTICS ON ORDERS INDEX (ORDER\_NUM);

COLLECT STATISTICS ON ORDER\_LINE INDEX (ORDER\_NUM, PART\_NUM);

COLLECT STATISTICS ON PART INDEX (PART\_NUM);

COLLECT STATISTICS ON REP INDEX (REP\_NUM);

### Jukic, N., Vrbsky, S., & Nestorov, S.

Database Systems: Introduction to Databases and Data Warehouses (1e)

#### db\_jukic\_hafh

CREATE TABLE manager

( managerid CHAR(4) NOT NULL,

mfname VARCHAR(15) NOT NULL,

mlname VARCHAR(15) NOT NULL,

mbdate DATE NOT NULL,

msalary NUMERIC(9,2) NOT NULL,

mbonus NUMERIC(9,2),

mresbuildingid CHAR(3),

PRIMARY KEY (managerid) );

CREATE TABLE managerphone

( managerid CHAR(4) NOT NULL,

mphone CHAR(11) NOT NULL,

PRIMARY KEY (managerid, mphone),

FOREIGN KEY (managerid) REFERENCES manager(managerid) );

CREATE TABLE building

( buildingid CHAR(3) NOT NULL,

bnooffloors INT NOT NULL,

bmanagerid CHAR(4) NOT NULL,

PRIMARY KEY (buildingid),

FOREIGN KEY (bmanagerid) REFERENCES manager(managerid) );

CREATE TABLE inspector

( insid CHAR(3) NOT NULL,

insname VARCHAR(15) NOT NULL,

PRIMARY KEY (insid) );

CREATE TABLE inspecting

( insid CHAR(3) NOT NULL,

buildingid CHAR(3) NOT NULL,

datelast DATE NOT NULL,

datenext DATE NOT NULL,

PRIMARY KEY (insid, buildingid),

FOREIGN KEY (insid) REFERENCES inspector(insid),

FOREIGN KEY (buildingid) REFERENCES building(buildingid) );

CREATE TABLE corpclient

( ccid CHAR(4) NOT NULL,

ccname VARCHAR(25) NOT NULL,

ccindustry VARCHAR(25) NOT NULL,

cclocation VARCHAR(25) NOT NULL,

ccidreferredby CHAR(4),

PRIMARY KEY (ccid),

UNIQUE (ccname),

FOREIGN KEY (ccidreferredby) REFERENCES corpclient(ccid) );

CREATE TABLE apartment

( buildingid CHAR(3) NOT NULL,

aptno CHAR(5) NOT NULL,

anoofbedrooms INT NOT NULL,

ccid CHAR(4),

PRIMARY KEY (buildingid, aptno),

FOREIGN KEY (buildingid) REFERENCES building(buildingid),

FOREIGN KEY (ccid) REFERENCES corpclient(ccid) );

CREATE TABLE staffmember

( smemberid CHAR(4) NOT NULL,

smembername VARCHAR(15) NOT NULL,

PRIMARY KEY (smemberid) );

CREATE TABLE cleaning

( buildingid CHAR(3) NOT NULL,

aptno CHAR(5) NOT NULL,

smemberid CHAR(4) NOT NULL,

CONSTRAINT cleaningpk PRIMARY KEY (buildingid, aptno, smemberid ),

CONSTRAINT cleaningfk FOREIGN KEY (buildingid, aptno)

REFERENCES apartment(buildingid, aptno) );

INSERT INTO manager VALUES ('M12', 'Boris', 'Grant', '1980-06-20', 60000, null, null);

INSERT INTO manager VALUES ('M23', 'Austin', 'Lee', '1975-10-30', 50000, 5000, null);

INSERT INTO manager VALUES ('M34', 'George', 'Sherman', '1976-01-11', 52000, 2000, null);

INSERT INTO managerphone VALUES ('M12','555-2222');

INSERT INTO managerphone VALUES ('M12','555-3232');

INSERT INTO managerphone VALUES ('M23','555-9988');

INSERT INTO managerphone VALUES ('M34','555-9999');

INSERT INTO building VALUES ('B1', '5', 'M12');

INSERT INTO building VALUES ('B2', '6', 'M23');

INSERT INTO building VALUES ('B3', '4', 'M23');

INSERT INTO building VALUES ('B4', '4', 'M34');

INSERT INTO inspector VALUES ('I11', 'Jane');

INSERT INTO inspector VALUES ('I22', 'Niko');

INSERT INTO inspector VALUES ('I33', 'Mick');

INSERT INTO inspecting VALUES ('I11','B1','2012-05-15','2013-05-14');

INSERT INTO inspecting VALUES ('I11','B2','2013-02-17','2013-05-17');

INSERT INTO inspecting VALUES ('I22','B2','2013-02-17','2013-05-17');

INSERT INTO inspecting VALUES ('I22','B3','2013-01-11','2014-01-11');

INSERT INTO inspecting VALUES ('I33','B3','2013-01-12','2014-01-12');

INSERT INTO inspecting VALUES ('I33','B4','2013-01-11','2014-01-11');

INSERT INTO corpclient VALUES ('C111', 'BlingNotes', 'Music', 'Chicago', null);

INSERT INTO corpclient VALUES ('C222', 'SkyJet', 'Airline', 'Oak Park', 'C111');

INSERT INTO corpclient VALUES ('C777', 'WindyCT', 'Music', 'Chicago', 'C222');

INSERT INTO corpclient VALUES ('C888', 'SouthAlps', 'Sports', 'Rosemont', 'C777');

INSERT INTO apartment VALUES ('B1', '21', 1, 'C111');

INSERT INTO apartment VALUES ('B1', '41', 1, null);

INSERT INTO apartment VALUES ('B2', '11', 2, 'C222');

INSERT INTO apartment VALUES ('B2', '31', 2, null);

INSERT INTO apartment VALUES ('B3', '11', 2, 'C777');

INSERT INTO apartment VALUES ('B4', '11', 2, 'C777');

INSERT INTO staffmember VALUES ('5432', 'Brian');

INSERT INTO staffmember VALUES ('9876', 'Boris');

INSERT INTO staffmember VALUES ('7652', 'Caroline');

INSERT INTO cleaning VALUES ('B1', '21', '5432');

INSERT INTO cleaning VALUES ('B1', '41', '9876');

INSERT INTO cleaning VALUES ('B2', '31', '5432');

INSERT INTO cleaning VALUES ('B2', '11', '9876');

INSERT INTO cleaning VALUES ('B3', '11', '5432');

INSERT INTO cleaning VALUES ('B4', '11', '7652');

UPDATE manager SET mresbuildingid = 'B1' WHERE managerid = 'M12';

UPDATE manager SET mresbuildingid = 'B2' WHERE managerid = 'M23';

UPDATE manager SET mresbuildingid = 'B4' WHERE managerid = 'M34';

#### db\_jukic\_hafhmore

CREATE TABLE manager

( managerid CHAR(4) NOT NULL,

mfname VARCHAR(15) NOT NULL,

mlname VARCHAR(15) NOT NULL,

mbdate DATE NOT NULL,

msalary NUMERIC(9,2) NOT NULL,

mbonus NUMERIC(9,2),

mresbuildingid CHAR(3),

PRIMARY KEY (managerid) );

CREATE TABLE managerphone

( managerid CHAR(4) NOT NULL,

mphone CHAR(11) NOT NULL,

PRIMARY KEY (managerid, mphone),

FOREIGN KEY (managerid) REFERENCES manager(managerid) );

CREATE TABLE building

( buildingid CHAR(3) NOT NULL,

bnooffloors INT NOT NULL,

bmanagerid CHAR(4) NOT NULL,

PRIMARY KEY (buildingid),

FOREIGN KEY (bmanagerid) REFERENCES manager(managerid) );

CREATE TABLE inspector

( insid CHAR(3) NOT NULL,

insname VARCHAR(15) NOT NULL,

PRIMARY KEY (insid) );

CREATE TABLE inspecting

( insid CHAR(3) NOT NULL,

buildingid CHAR(3) NOT NULL,

datelast DATE NOT NULL,

datenext DATE NOT NULL,

PRIMARY KEY (insid, buildingid),

FOREIGN KEY (insid) REFERENCES inspector(insid),

FOREIGN KEY (buildingid) REFERENCES building(buildingid) );

CREATE TABLE corpclient

( ccid CHAR(4) NOT NULL,

ccname VARCHAR(25) NOT NULL,

ccindustry VARCHAR(25) NOT NULL,

cclocation VARCHAR(25) NOT NULL,

ccidreferredby CHAR(4),

PRIMARY KEY (ccid),

UNIQUE (ccname),

FOREIGN KEY (ccidreferredby) REFERENCES corpclient(ccid) );

CREATE TABLE apartment

( buildingid CHAR(3) NOT NULL,

aptno CHAR(5) NOT NULL,

anoofbedrooms INT NOT NULL,

ccid CHAR(4),

PRIMARY KEY (buildingid, aptno),

FOREIGN KEY (buildingid) REFERENCES building(buildingid),

FOREIGN KEY (ccid) REFERENCES corpclient(ccid) );

CREATE TABLE staffmember

( smemberid CHAR(4) NOT NULL,

smembername VARCHAR(15) NOT NULL,

PRIMARY KEY (smemberid) );

CREATE TABLE cleaning

( buildingid CHAR(3) NOT NULL,

aptno CHAR(5) NOT NULL,

smemberid CHAR(4) NOT NULL,

CONSTRAINT cleaningpk PRIMARY KEY (buildingid, aptno, smemberid ),

CONSTRAINT cleaningfk FOREIGN KEY (buildingid, aptno)

REFERENCES apartment(buildingid, aptno) );

INSERT INTO manager VALUES ('M12', 'Boris', 'Grant', '1980-06-20', 60000, null, null);

INSERT INTO manager VALUES ('M23', 'Austin', 'Lee', '1975-10-30', 50000, 5000, null);

INSERT INTO manager VALUES ('M34', 'George', 'Sherman', '1976-01-11', 52000, 2000, null);

INSERT INTO manager VALUES ('M45','Mariana','Gonzalez','1980-12-27',54000,null,null);

INSERT INTO manager VALUES ('M56','Fiona', 'Keane','1977-10-04',57000,2000,null);

INSERT INTO manager VALUES ('M67','Alexander','Sanborn','1953-08-17',62000,3000,null);

INSERT INTO managerphone VALUES ('M12','555-2222');

INSERT INTO managerphone VALUES ('M12','555-3232');

INSERT INTO managerphone VALUES ('M23','555-9988');

INSERT INTO managerphone VALUES ('M34','555-9999');

INSERT INTO managerphone VALUES ('M34','555-1003');

INSERT INTO managerphone VALUES ('M45','555-1216');

INSERT INTO managerphone VALUES ('M56','555-5180');

INSERT INTO managerphone VALUES ('M67','555-6767');

INSERT INTO managerphone VALUES ('M67','555-1327');

INSERT INTO managerphone VALUES ('M67','555-3794');

INSERT INTO building VALUES ('B1','5','M12');

INSERT INTO building VALUES ('B2','6','M23');

INSERT INTO building VALUES ('B3','4','M23');

INSERT INTO building VALUES ('B4','4','M34');

INSERT INTO building VALUES ('B5','3','M45');

INSERT INTO building VALUES ('B6','3','M45');

INSERT INTO building VALUES ('B7','2','M56');

INSERT INTO building VALUES ('B8','4','M67');

INSERT INTO building VALUES ('B9','3','M67');

INSERT INTO inspector VALUES ('I11','Jane');

INSERT INTO inspector VALUES ('I22','Niko');

INSERT INTO inspector VALUES ('I33','Mick');

INSERT INTO inspector VALUES ('I44','Bianca');

INSERT INTO inspector VALUES ('I55','Sergei');

INSERT INTO inspecting VALUES ('I11','B1','2012-05-15','2013-05-14');

INSERT INTO inspecting VALUES ('I11','B2','2013-02-17','2013-05-17');

INSERT INTO inspecting VALUES ('I11','B7','2012-04-08','2013-04-08');

INSERT INTO inspecting VALUES ('I22','B2','2013-02-17','2013-05-17');

INSERT INTO inspecting VALUES ('I22','B3','2013-01-11','2014-01-11');

INSERT INTO inspecting VALUES ('I22','B8','2013-03-19','2014-03-19');

INSERT INTO inspecting VALUES ('I33','B3','2013-01-12','2014-01-12');

INSERT INTO inspecting VALUES ('I33','B4','2013-01-11','2014-01-11');

INSERT INTO inspecting VALUES ('I33','B9','2013-05-12','2014-05-12');

INSERT INTO inspecting VALUES ('I44','B4','2013-01-11','2014-01-11');

INSERT INTO inspecting VALUES ('I44','B5','2013-07-23','2014-07-23');

INSERT INTO inspecting VALUES ('I55','B5','2013-08-15','2014-08-15');

INSERT INTO inspecting VALUES ('I55','B6','2013-07-26','2014-07-26');

INSERT INTO corpclient VALUES ('C111','BlingNotes','Music','Chicago',null);

INSERT INTO corpclient VALUES ('C222','SkyJet','Airline','Oak Park','C111');

INSERT INTO corpclient VALUES ('C333','Xilerate','Sports','Chicago',null);

INSERT INTO corpclient VALUES ('C444','NanoCorp','Broadcasting','Rosemont','C111');

INSERT INTO corpclient VALUES ('C555','EntertainUs','Broadcasting','Oak Brook',null);

INSERT INTO corpclient VALUES ('C666','DelishInc','Food Service','Oak Brook','C444');

INSERT INTO corpclient VALUES ('C777','WindyCT','Music','Chicago','C222');

INSERT INTO corpclient VALUES ('C888','SouthAlps','Sports','Rosemont','C777');

INSERT INTO corpclient VALUES ('C999','CommuteAir','Airline','Oak Brook','C111');

INSERT INTO apartment VALUES ('B1','11',1,'C111');

INSERT INTO apartment VALUES ('B1','21',1,'C111');

INSERT INTO apartment VALUES ('B1','31',1,'C333');

INSERT INTO apartment VALUES ('B1','41',1,null);

INSERT INTO apartment VALUES ('B1','51',1,null);

INSERT INTO apartment VALUES ('B2','11',2,'C222');

INSERT INTO apartment VALUES ('B2','21',2,'C222');

INSERT INTO apartment VALUES ('B2','31',2,null);

INSERT INTO apartment VALUES ('B2','41',2,null);

INSERT INTO apartment VALUES ('B2','51',2,'C111');

INSERT INTO apartment VALUES ('B2','61',2,'C111');

INSERT INTO apartment VALUES ('B3','11',2,'C777');

INSERT INTO apartment VALUES ('B3','21',2,'C777');

INSERT INTO apartment VALUES ('B3','31',2,'C555');

INSERT INTO apartment VALUES ('B3','41',2,'C555');

INSERT INTO apartment VALUES ('B4','11',2,'C777');

INSERT INTO apartment VALUES ('B4','21',2,'C777');

INSERT INTO apartment VALUES ('B4','31',2,'C222');

INSERT INTO apartment VALUES ('B4','41',2,'C222');

INSERT INTO apartment VALUES ('B5','11',3,'C555');

INSERT INTO apartment VALUES ('B5','21',3,null);

INSERT INTO apartment VALUES ('B5','31',3,'C111');

INSERT INTO apartment VALUES ('B6','11',1,'C111');

INSERT INTO apartment VALUES ('B6','12',1,'C111');

INSERT INTO apartment VALUES ('B6','21',1,'C444');

INSERT INTO apartment VALUES ('B6','22',1,'C444');

INSERT INTO apartment VALUES ('B6','31',1,'C555');

INSERT INTO apartment VALUES ('B6','32',1,'C333');

INSERT INTO apartment VALUES ('B7','11',3,'C999');

INSERT INTO apartment VALUES ('B7','12',3,'C999');

INSERT INTO apartment VALUES ('B7','13',3,'C999');

INSERT INTO apartment VALUES ('B7','21',3,null);

INSERT INTO apartment VALUES ('B7','22',3,'C222');

INSERT INTO apartment VALUES ('B7','23',3,'C111');

INSERT INTO apartment VALUES ('B8','11',2,'C777');

INSERT INTO apartment VALUES ('B8','12',2,'C777');

INSERT INTO apartment VALUES ('B8','21',2,'C444');

INSERT INTO apartment VALUES ('B8','22',2,'C444');

INSERT INTO apartment VALUES ('B8','31',2,null);

INSERT INTO apartment VALUES ('B8','32',2,null);

INSERT INTO apartment VALUES ('B8','41',2,'C666');

INSERT INTO apartment VALUES ('B8','42',2,'C666');

INSERT INTO apartment VALUES ('B9','11',2,'C111');

INSERT INTO apartment VALUES ('B9','21',2,'C222');

INSERT INTO apartment VALUES ('B9','31',2,'C222');

INSERT INTO staffmember VALUES ('5432','Brian');

INSERT INTO staffmember VALUES ('9876','Boris');

INSERT INTO staffmember VALUES ('7652','Caroline');

INSERT INTO staffmember VALUES ('2537','Howard');

INSERT INTO staffmember VALUES ('3984','Luis');

INSERT INTO staffmember VALUES ('4196','Arthur');

INSERT INTO staffmember VALUES ('8467','Mariana');

INSERT INTO staffmember VALUES ('1028','Franz');

INSERT INTO cleaning VALUES ('B1','11','5432');

INSERT INTO cleaning VALUES ('B1','21','5432');

INSERT INTO cleaning VALUES ('B1','31','5432');

INSERT INTO cleaning VALUES ('B1','41','9876');

INSERT INTO cleaning VALUES ('B1','51','9876');

INSERT INTO cleaning VALUES ('B2','11','9876');

INSERT INTO cleaning VALUES ('B2','21','9876');

INSERT INTO cleaning VALUES ('B2','31','5432');

INSERT INTO cleaning VALUES ('B2','41','5432');

INSERT INTO cleaning VALUES ('B2','51','1028');

INSERT INTO cleaning VALUES ('B2','61','1028');

INSERT INTO cleaning VALUES ('B3','11','5432');

INSERT INTO cleaning VALUES ('B3','21','5432');

INSERT INTO cleaning VALUES ('B3','31','8467');

INSERT INTO cleaning VALUES ('B3','41','8467');

INSERT INTO cleaning VALUES ('B4','11','7652');

INSERT INTO cleaning VALUES ('B4','21','7652');

INSERT INTO cleaning VALUES ('B4','31','7652');

INSERT INTO cleaning VALUES ('B4','41','7652');

INSERT INTO cleaning VALUES ('B5','11','9876');

INSERT INTO cleaning VALUES ('B5','11','3984');

INSERT INTO cleaning VALUES ('B5','21','9876');

INSERT INTO cleaning VALUES ('B5','21','3984');

INSERT INTO cleaning VALUES ('B5','31','9876');

INSERT INTO cleaning VALUES ('B5','31','3984');

INSERT INTO cleaning VALUES ('B6','11','3984');

INSERT INTO cleaning VALUES ('B6','12','3984');

INSERT INTO cleaning VALUES ('B6','21','2537');

INSERT INTO cleaning VALUES ('B6','22','2537');

INSERT INTO cleaning VALUES ('B6','31','2537');

INSERT INTO cleaning VALUES ('B6','32','2537');

INSERT INTO cleaning VALUES ('B7','11','4196');

INSERT INTO cleaning VALUES ('B7','11','8467');

INSERT INTO cleaning VALUES ('B7','12','4196');

INSERT INTO cleaning VALUES ('B7','12','8467');

INSERT INTO cleaning VALUES ('B7','13','4196');

INSERT INTO cleaning VALUES ('B7','13','8467');

INSERT INTO cleaning VALUES ('B7','21','3984');

INSERT INTO cleaning VALUES ('B7','21','2537');

INSERT INTO cleaning VALUES ('B7','22','3984');

INSERT INTO cleaning VALUES ('B7','22','2537');

INSERT INTO cleaning VALUES ('B7','23','3984');

INSERT INTO cleaning VALUES ('B7','23','2537');

INSERT INTO cleaning VALUES ('B8','11','7652');

INSERT INTO cleaning VALUES ('B8','12','7652');

INSERT INTO cleaning VALUES ('B8','21','1028');

INSERT INTO cleaning VALUES ('B8','22','1028');

INSERT INTO cleaning VALUES ('B8','31','1028');

INSERT INTO cleaning VALUES ('B8','32','1028');

INSERT INTO cleaning VALUES ('B8','41','4196');

INSERT INTO cleaning VALUES ('B8','42','4196');

INSERT INTO cleaning VALUES ('B9','11','8467');

INSERT INTO cleaning VALUES ('B9','21','8467');

INSERT INTO cleaning VALUES ('B9','31','8467');

UPDATE manager SET mresbuildingid = 'B1' WHERE managerid = 'M12';

UPDATE manager SET mresbuildingid = 'B2' WHERE managerid = 'M23';

UPDATE manager SET mresbuildingid = 'B4' WHERE managerid = 'M34';

UPDATE manager SET mresbuildingid = 'B5' WHERE managerid = 'M45';

UPDATE manager SET mresbuildingid = 'B7' WHERE managerid = 'M56';

UPDATE manager SET mresbuildingid = 'B8' WHERE managerid = 'M67';

#### db\_jukic\_zagi

CREATE TABLE vendor

( vendorid CHAR(2) NOT NULL,

vendorname VARCHAR(25) NOT NULL,

PRIMARY KEY (vendorid) );

CREATE TABLE category

( categoryid CHAR(2) NOT NULL,

categoryname VARCHAR(25) NOT NULL,

PRIMARY KEY (categoryid) );

CREATE TABLE product

( productid CHAR(3) NOT NULL,

productname VARCHAR(25) NOT NULL,

productprice NUMERIC (7,2) NOT NULL,

vendorid CHAR(2) NOT NULL,

categoryid CHAR(2) NOT NULL,

PRIMARY KEY (productid),

FOREIGN KEY (vendorid) REFERENCES vendor(vendorid),

FOREIGN KEY (categoryid) REFERENCES category(categoryid) );

CREATE TABLE region

( regionid CHAR NOT NULL,

regionname VARCHAR(25) NOT NULL,

PRIMARY KEY (regionid) );

CREATE TABLE store

( storeid VARCHAR(3) NOT NULL,

storezip CHAR(5) NOT NULL,

regionid CHAR NOT NULL,

PRIMARY KEY (storeid),

FOREIGN KEY (regionid) REFERENCES region(regionid) );

CREATE TABLE customer

( customerid CHAR(7) NOT NULL,

customername VARCHAR(15) NOT NULL,

customerzip CHAR(5) NOT NULL,

PRIMARY KEY (customerid) );

CREATE TABLE salestransaction

( tid VARCHAR(8) NOT NULL,

customerid CHAR(7) NOT NULL,

storeid VARCHAR(3) NOT NULL,

tdate DATE NOT NULL,

PRIMARY KEY (tid),

FOREIGN KEY (customerid) REFERENCES customer(customerid),

FOREIGN KEY (storeid) REFERENCES store(storeid));

CREATE TABLE soldvia

( productid CHAR(3) NOT NULL,

tid VARCHAR(8) NOT NULL,

noofitems INT NOT NULL,

PRIMARY KEY (productid, tid),

FOREIGN KEY (productid) REFERENCES product(productid),

FOREIGN KEY (tid) REFERENCES salestransaction(tid) );

INSERT INTO vendor VALUES ('PG','Pacifica Gear');

INSERT INTO vendor VALUES ('MK','Mountain King');

INSERT INTO category VALUES ('CP','Camping');

INSERT INTO category VALUES ('FW','Footwear');

INSERT INTO product VALUES ('1X1','Zzz Bag',100,'PG','CP');

INSERT INTO product VALUES ('2X2','Easy Boot',70,'MK','FW');

INSERT INTO product VALUES ('3X3','Cosy Sock',15,'MK','FW');

INSERT INTO product VALUES ('4X4','Dura Boot',90,'PG','FW');

INSERT INTO product VALUES ('5X5','Tiny Tent',150,'MK','CP');

INSERT INTO product VALUES ('6X6','Biggy Tent',250,'MK','CP');

INSERT INTO region VALUES ('C','Chicagoland');

INSERT INTO region VALUES ('T','Tristate');

INSERT INTO store VALUES ('S1','60600','C');

INSERT INTO store VALUES ('S2','60605','C');

INSERT INTO store VALUES ('S3','35400','T');

INSERT INTO customer VALUES ('1-2-333','Tina','60137');

INSERT INTO customer VALUES ('2-3-444','Tony','60611');

INSERT INTO customer VALUES ('3-4-555','Pam','35401');

INSERT INTO salestransaction VALUES ('T111','1-2-333','S1','2013-01-01');

INSERT INTO salestransaction VALUES ('T222','2-3-444','S2','2013-01-01');

INSERT INTO salestransaction VALUES ('T333','1-2-333','S3','2013-01-02');

INSERT INTO salestransaction VALUES ('T444','3-4-555','S3','2013-01-02');

INSERT INTO salestransaction VALUES ('T555','2-3-444','S3','2013-01-02');

INSERT INTO soldvia VALUES ('1X1','T111',1);

INSERT INTO soldvia VALUES ('2X2','T222',1);

INSERT INTO soldvia VALUES ('3X3','T333',5);

INSERT INTO soldvia VALUES ('1X1','T333',1);

INSERT INTO soldvia VALUES ('4X4','T444',1);

INSERT INTO soldvia VALUES ('2X2','T444',2);

INSERT INTO soldvia VALUES ('4X4','T555',4);

INSERT INTO soldvia VALUES ('5X5','T555',2);

INSERT INTO soldvia VALUES ('6X6','T555',1);

#### db\_jukic\_zagimore

CREATE TABLE vendor

( vendorid CHAR(2) NOT NULL,

vendorname VARCHAR(25) NOT NULL,

PRIMARY KEY (vendorid) );

CREATE TABLE category

( categoryid CHAR(2) NOT NULL,

categoryname VARCHAR(25) NOT NULL,

PRIMARY KEY (categoryid) );

CREATE TABLE product

( productid CHAR(3) NOT NULL,

productname VARCHAR(25) NOT NULL,

productprice NUMERIC (7,2) NOT NULL,

vendorid CHAR(2) NOT NULL,

categoryid CHAR(2) NOT NULL,

PRIMARY KEY (productid),

FOREIGN KEY (vendorid) REFERENCES vendor(vendorid),

FOREIGN KEY (categoryid) REFERENCES category(categoryid) );

CREATE TABLE region

( regionid CHAR NOT NULL,

regionname VARCHAR(25) NOT NULL,

PRIMARY KEY (regionid) );

CREATE TABLE store

( storeid VARCHAR(3) NOT NULL,

storezip CHAR(5) NOT NULL,

regionid CHAR NOT NULL,

PRIMARY KEY (storeid),

FOREIGN KEY (regionid) REFERENCES region(regionid) );

CREATE TABLE customer

( customerid CHAR(7) NOT NULL,

customername VARCHAR(15) NOT NULL,

customerzip CHAR(5) NOT NULL,

PRIMARY KEY (customerid) );

CREATE TABLE salestransaction

( tid VARCHAR(8) NOT NULL,

customerid CHAR(7) NOT NULL,

storeid VARCHAR(3) NOT NULL,

tdate DATE NOT NULL,

PRIMARY KEY (tid),

FOREIGN KEY (customerid) REFERENCES customer(customerid),

FOREIGN KEY (storeid) REFERENCES store(storeid));

CREATE TABLE soldvia

( productid CHAR(3) NOT NULL,

tid VARCHAR(8) NOT NULL,

noofitems INT NOT NULL,

PRIMARY KEY (productid, tid),

FOREIGN KEY (productid) REFERENCES product(productid),

FOREIGN KEY (tid) REFERENCES salestransaction(tid) );

INSERT INTO vendor VALUES ('PG','Pacifica Gear');

INSERT INTO vendor VALUES ('MK','Mountain King');

INSERT INTO vendor VALUES ('OA','Outdoor Adventures');

INSERT INTO vendor VALUES ('WL','Wilderness Limited');

INSERT INTO category VALUES ('CP','Camping');

INSERT INTO category VALUES ('FW','Footwear');

INSERT INTO category VALUES ('CL','Climbing');

INSERT INTO category VALUES ('EL','Electronics');

INSERT INTO category VALUES ('CY','Cycling');

INSERT INTO product VALUES ('1X1','Zzz Bag',100,'PG','CP');

INSERT INTO product VALUES ('2X2','Easy Boot',70,'MK','FW');

INSERT INTO product VALUES ('3X3','Cosy Sock',15,'MK','FW');

INSERT INTO product VALUES ('4X4','Dura Boot',90,'PG','FW');

INSERT INTO product VALUES ('5X5','Tiny Tent',150,'MK','CP');

INSERT INTO product VALUES ('6X6','Biggy Tent',250,'MK','CP');

INSERT INTO product VALUES ('7X7','Hi-Tec GPS',300,'OA','EL');

INSERT INTO product VALUES ('8X8','Power Pedals',20,'MK','CY');

INSERT INTO product VALUES ('9X9','Trusty Rope',30,'WL','CL');

INSERT INTO product VALUES ('1X2','Comfy Harness',150,'MK','CL');

INSERT INTO product VALUES ('1X3','Sunny Charger',125,'OA','EL');

INSERT INTO product VALUES ('1X4','Safe-T Helmet',40,'PG','CY');

INSERT INTO product VALUES ('2X1','Mmm Stove',80,'WL','CP');

INSERT INTO product VALUES ('2X3','Reflect-o Jacket',35,'PG','CY');

INSERT INTO product VALUES ('2X4','Strongster Carribeaner',20,'MK','CL');

INSERT INTO product VALUES ('3X1','Sleepy Pad',25,'WL','CP');

INSERT INTO product VALUES ('3X2','Bucky Knife',60,'WL','CP');

INSERT INTO product VALUES ('3X4','Treado Tire',30,'OA','CY');

INSERT INTO product VALUES ('4X1','Slicky Tire',25,'OA','CY');

INSERT INTO product VALUES ('4X2','Electra Compass',45,'MK','EL');

INSERT INTO product VALUES ('4X3','Mega Camera',275,'WL','EL');

INSERT INTO product VALUES ('5X1','Simple Sandal',50,'PG','FW');

INSERT INTO product VALUES ('5X2','Action Sandal',70,'PG','FW');

INSERT INTO product VALUES ('5X3','Luxo Tent',500,'OA','CP');

INSERT INTO region VALUES ('C ','Chicagoland');

INSERT INTO region VALUES ('T','Tristate');

INSERT INTO region VALUES ('I','Indiana');

INSERT INTO region VALUES ('N','North');

INSERT INTO store VALUES ('S1','60600','C ');

INSERT INTO store VALUES ('S2','60605','C');

INSERT INTO store VALUES ('S3','35400','T');

INSERT INTO store VALUES ('S4','60640','C');

INSERT INTO store VALUES ('S5','46307','T');

INSERT INTO store VALUES ('S6','47374','I');

INSERT INTO store VALUES ('S7','47401','I');

INSERT INTO store VALUES ('S8','55401','N');

INSERT INTO store VALUES ('S9','54937','N');

INSERT INTO store VALUES ('S10','60602','C');

INSERT INTO store VALUES ('S11','46201','I');

INSERT INTO store VALUES ('S12','55701','N');

INSERT INTO store VALUES ('S13','60085','T');

INSERT INTO store VALUES ('S14','53140','T');

INSERT INTO customer VALUES ('1-2-333','Tina','60137');

INSERT INTO customer VALUES ('2-3-444','Tony','60611');

INSERT INTO customer VALUES ('3-4-555','Pam','35401');

INSERT INTO customer VALUES ('4-5-666','Elly','47374');

INSERT INTO customer VALUES ('5-6-777','Nora','60640');

INSERT INTO customer VALUES ('6-7-888','Miles','60602');

INSERT INTO customer VALUES ('7-8-999','Neil','55403');

INSERT INTO customer VALUES ('8-9-000','Maggie','47401');

INSERT INTO customer VALUES ('9-0-111','Ryan','46202');

INSERT INTO customer VALUES ('0-1-222','Dan','55499');

INSERT INTO salestransaction VALUES ('T111','1-2-333','S1','2013-01-01');

INSERT INTO salestransaction VALUES ('T222','2-3-444','S2','2013-01-01');

INSERT INTO salestransaction VALUES ('T333','1-2-333','S3','2013-01-02');

INSERT INTO salestransaction VALUES ('T444','3-4-555','S3','2013-01-02');

INSERT INTO salestransaction VALUES ('T555','2-3-444','S3','2013-01-02');

INSERT INTO salestransaction VALUES ('T666','5-6-777','S10','2013-01-03');

INSERT INTO salestransaction VALUES ('T777','6-7-888','S13','2013-01-03');

INSERT INTO salestransaction VALUES ('T888','8-9-000','S4','2013-01-04');

INSERT INTO salestransaction VALUES ('T999','4-5-666','S6','2013-01-04');

INSERT INTO salestransaction VALUES ('T101','7-8-999','S12','2013-01-04');

INSERT INTO salestransaction VALUES ('T202','0-1-222','S8','2013-01-04');

INSERT INTO salestransaction VALUES ('T303','4-5-666','S6','2013-01-05');

INSERT INTO salestransaction VALUES ('T404','8-9-000','S6','2013-01-05');

INSERT INTO salestransaction VALUES ('T505','6-7-888','S14','2013-01-05');

INSERT INTO salestransaction VALUES ('T606','0-1-222','S11','2013-01-06');

INSERT INTO salestransaction VALUES ('T707','5-6-777','S4','2013-01-06');

INSERT INTO salestransaction VALUES ('T808','7-8-999','S9','2013-01-06');

INSERT INTO salestransaction VALUES ('T909','5-6-777','S4','2013-01-06');

INSERT INTO salestransaction VALUES ('T011','8-9-000','S7','2013-01-07');

INSERT INTO salestransaction VALUES ('T022','9-0-111','S5','2013-01-07');

INSERT INTO soldvia VALUES ('1X1','T111',1);

INSERT INTO soldvia VALUES ('2X2','T222',1);

INSERT INTO soldvia VALUES ('3X3','T333',5);

INSERT INTO soldvia VALUES ('1X1','T333',1);

INSERT INTO soldvia VALUES ('4X4','T444',1);

INSERT INTO soldvia VALUES ('2X2','T444',2);

INSERT INTO soldvia VALUES ('4X4','T555',4);

INSERT INTO soldvia VALUES ('5X5','T555',2);

INSERT INTO soldvia VALUES ('6X6','T555',1);

INSERT INTO soldvia VALUES ('7X7','T666',1);

INSERT INTO soldvia VALUES ('9X9','T666',1);

INSERT INTO soldvia VALUES ('1X3','T666',2);

INSERT INTO soldvia VALUES ('8X8','T777',1);

INSERT INTO soldvia VALUES ('1X4','T888',4);

INSERT INTO soldvia VALUES ('2X3','T888',3);

INSERT INTO soldvia VALUES ('9X9','T999',1);

INSERT INTO soldvia VALUES ('1X2','T999',5);

INSERT INTO soldvia VALUES ('8X8','T999',3);

INSERT INTO soldvia VALUES ('1X3','T999',1);

INSERT INTO soldvia VALUES ('1X2','T101',3);

INSERT INTO soldvia VALUES ('1X4','T101',1);

INSERT INTO soldvia VALUES ('2X4','T202',4);

INSERT INTO soldvia VALUES ('9X9','T303',3);

INSERT INTO soldvia VALUES ('1X4','T303',2);

INSERT INTO soldvia VALUES ('2X1','T303',2);

INSERT INTO soldvia VALUES ('3X1','T303',2);

INSERT INTO soldvia VALUES ('2X4','T404',1);

INSERT INTO soldvia VALUES ('2X3','T404',2);

INSERT INTO soldvia VALUES ('2X2','T505',3);

INSERT INTO soldvia VALUES ('3X2','T505',1);

INSERT INTO soldvia VALUES ('2X1','T505',4);

INSERT INTO soldvia VALUES ('2X4','T606',7);

INSERT INTO soldvia VALUES ('3X1','T606',4);

INSERT INTO soldvia VALUES ('2X2','T606',3);

INSERT INTO soldvia VALUES ('3X4','T606',2);

INSERT INTO soldvia VALUES ('4X4','T606',2);

INSERT INTO soldvia VALUES ('3X2','T707',1);

INSERT INTO soldvia VALUES ('3X4','T707',4);

INSERT INTO soldvia VALUES ('4X1','T707',2);

INSERT INTO soldvia VALUES ('5X3','T808',1);

INSERT INTO soldvia VALUES ('4X2','T808',1);

INSERT INTO soldvia VALUES ('2X2','T808',1);

INSERT INTO soldvia VALUES ('4X3','T808',1);

INSERT INTO soldvia VALUES ('3X3','T808',4);

INSERT INTO soldvia VALUES ('4X2','T909',3);

INSERT INTO soldvia VALUES ('6X6','T909',1);

INSERT INTO soldvia VALUES ('3X3','T011',3);

INSERT INTO soldvia VALUES ('4X3','T022',3);

INSERT INTO soldvia VALUES ('2X2','T022',3);

INSERT INTO soldvia VALUES ('5X1','T022',2);

### Hoffer, J., Prescott, M., & Topi, H.

Modern Database Management (9e)

#### db\_pvfc9\_std

CREATE TABLE CUSTOMER\_T AS db\_pvfc9\_std.CUSTOMER\_T WITH NO DATA;

CREATE TABLE DOES\_BUSINESS\_IN\_T AS db\_pvfc9\_std.DOES\_BUSINESS\_IN\_T WITH NO DATA;

CREATE TABLE EMPLOYEE\_SKILLS\_T AS db\_pvfc9\_std.EMPLOYEE\_SKILLS\_T WITH NO DATA;

CREATE TABLE EMPLOYEE\_T AS db\_pvfc9\_std.EMPLOYEE\_T WITH NO DATA;

CREATE TABLE ORDER\_LINE\_T AS db\_pvfc9\_std.ORDER\_LINE\_T WITH NO DATA;

CREATE TABLE ORDER\_T AS db\_pvfc9\_std.ORDER\_T WITH NO DATA;

CREATE TABLE PRODUCED\_IN\_T AS db\_pvfc9\_std.PRODUCED\_IN\_T WITH NO DATA;

CREATE TABLE PRODUCT\_LINE\_T AS db\_pvfc9\_std.PRODUCT\_LINE\_T WITH NO DATA;

CREATE TABLE PRODUCT\_T AS db\_pvfc9\_std.PRODUCT\_T WITH NO DATA;

CREATE TABLE RAW\_MATERIAL\_T AS db\_pvfc9\_std.RAW\_MATERIAL\_T WITH NO DATA;

CREATE TABLE SALESPERSON\_T AS db\_pvfc9\_std.SALESPERSON\_T WITH NO DATA;

CREATE TABLE SKILL\_T AS db\_pvfc9\_std.SKILL\_T WITH NO DATA;

CREATE TABLE SUPPLIES\_T AS db\_pvfc9\_std.SUPPLIES\_T WITH NO DATA;

CREATE TABLE TERRITORY\_T AS db\_pvfc9\_std.TERRITORY\_T WITH NO DATA;

CREATE TABLE USES\_T AS db\_pvfc9\_std.USES\_T WITH NO DATA;

CREATE TABLE VENDOR\_T AS db\_pvfc9\_std.VENDOR\_T WITH NO DATA;

CREATE TABLE WORKS\_IN\_T AS db\_pvfc9\_std.WORKS\_IN\_T WITH NO DATA;

CREATE TABLE WORK\_CENTER\_T AS db\_pvfc9\_std.WORK\_CENTER\_T WITH NO DATA;

INSERT INTO CUSTOMER\_T SELECT \* FROM db\_pvfc9\_std.CUSTOMER\_T;

INSERT INTO DOES\_BUSINESS\_IN\_T SELECT \* FROM db\_pvfc9\_std.DOES\_BUSINESS\_IN\_T;

INSERT INTO EMPLOYEE\_SKILLS\_T SELECT \* FROM db\_pvfc9\_std.EMPLOYEE\_SKILLS\_T;

INSERT INTO EMPLOYEE\_T SELECT \* FROM db\_pvfc9\_std.EMPLOYEE\_T;

INSERT INTO ORDER\_LINE\_T SELECT \* FROM db\_pvfc9\_std.ORDER\_LINE\_T;

INSERT INTO ORDER\_T SELECT \* FROM db\_pvfc9\_std.ORDER\_T;

INSERT INTO PRODUCED\_IN\_T SELECT \* FROM db\_pvfc9\_std.PRODUCED\_IN\_T;

INSERT INTO PRODUCT\_LINE\_T SELECT \* FROM db\_pvfc9\_std.PRODUCT\_LINE\_T;

INSERT INTO PRODUCT\_T SELECT \* FROM db\_pvfc9\_std.PRODUCT\_T;

INSERT INTO RAW\_MATERIAL\_T SELECT \* FROM db\_pvfc9\_std.RAW\_MATERIAL\_T;

INSERT INTO SALESPERSON\_T SELECT \* FROM db\_pvfc9\_std.SALESPERSON\_T;

INSERT INTO SKILL\_T SELECT \* FROM db\_pvfc9\_std.SKILL\_T;

INSERT INTO SUPPLIES\_T SELECT \* FROM db\_pvfc9\_std.SUPPLIES\_T;

INSERT INTO TERRITORY\_T SELECT \* FROM db\_pvfc9\_std.TERRITORY\_T;

INSERT INTO USES\_T SELECT \* FROM db\_pvfc9\_std.USES\_T;

INSERT INTO VENDOR\_T SELECT \* FROM db\_pvfc9\_std.VENDOR\_T;

INSERT INTO WORKS\_IN\_T SELECT \* FROM db\_pvfc9\_std.WORKS\_IN\_T;

INSERT INTO WORK\_CENTER\_T SELECT \* FROM db\_pvfc9\_std.WORK\_CENTER\_T;

COLLECT STATISTICS ON CUSTOMER\_T INDEX (Customer\_ID);

COLLECT STATISTICS ON DOES\_BUSINESS\_IN\_T INDEX (Customer\_Id);

COLLECT STATISTICS ON EMPLOYEE\_SKILLS\_T INDEX (Employee\_Id);

COLLECT STATISTICS ON EMPLOYEE\_T INDEX (Employee\_ID);

COLLECT STATISTICS ON ORDER\_LINE\_T INDEX (Order\_ID);

COLLECT STATISTICS ON ORDER\_T INDEX (Order\_ID);

COLLECT STATISTICS ON PRODUCED\_IN\_T INDEX (Product\_Id);

COLLECT STATISTICS ON PRODUCT\_LINE\_T INDEX (Product\_Line\_ID);

COLLECT STATISTICS ON PRODUCT\_T INDEX (Product\_ID);

COLLECT STATISTICS ON RAW\_MATERIAL\_T INDEX (Material\_ID);

COLLECT STATISTICS ON SALESPERSON\_T INDEX (SalesPerson\_ID);

COLLECT STATISTICS ON SKILL\_T INDEX (Skill\_Id);

COLLECT STATISTICS ON SUPPLIES\_T INDEX (Vendor\_Id);

COLLECT STATISTICS ON TERRITORY\_T INDEX (Territory\_ID);

COLLECT STATISTICS ON USES\_T INDEX (Product\_Id);

COLLECT STATISTICS ON VENDOR\_T INDEX (Vendor\_ID);

COLLECT STATISTICS ON WORKS\_IN\_T INDEX (Employee\_Id);

COLLECT STATISTICS ON WORK\_CENTER\_T INDEX (Work\_Center\_ID);

### Hoffer, J., Venkataraman, R., & Topi, H.

Modern Database Management (10e)

#### db\_pvfc10\_big

CREATE TABLE CUSTOMERSHIPADDRESS\_T AS db\_pvfc10\_big.CUSTOMERSHIPADDRESS\_T WITH NO DATA;

CREATE TABLE CUSTOMER\_T AS db\_pvfc10\_big.CUSTOMER\_T WITH NO DATA;

CREATE TABLE DOESBUSINESSIN\_T AS db\_pvfc10\_big.DOESBUSINESSIN\_T WITH NO DATA;

CREATE TABLE EMPLOYEESKILLS\_T AS db\_pvfc10\_big.EMPLOYEESKILLS\_T WITH NO DATA;

CREATE TABLE EMPLOYEE\_T AS db\_pvfc10\_big.EMPLOYEE\_T WITH NO DATA;

CREATE TABLE ORDERLINE\_T AS db\_pvfc10\_big.ORDERLINE\_T WITH NO DATA;

CREATE TABLE ORDER\_T AS db\_pvfc10\_big.ORDER\_T WITH NO DATA;

CREATE TABLE PAYMENT\_T AS db\_pvfc10\_big.PAYMENT\_T WITH NO DATA;

CREATE TABLE PAYMENTTYPE\_T AS db\_pvfc10\_big.PAYMENTTYPE\_T WITH NO DATA;

CREATE TABLE PRODUCEDIN\_T AS db\_pvfc10\_big.PRODUCEDIN\_T WITH NO DATA;

CREATE TABLE PRODUCTLINE\_T AS db\_pvfc10\_big.PRODUCTLINE\_T WITH NO DATA;

CREATE TABLE PRODUCT\_T AS db\_pvfc10\_big.PRODUCT\_T WITH NO DATA;

CREATE TABLE RAWMATERIAL\_T AS db\_pvfc10\_big.RAWMATERIAL\_T WITH NO DATA;

CREATE TABLE SALESPERSON\_T AS db\_pvfc10\_big.SALESPERSON\_T WITH NO DATA;

CREATE TABLE SHIPPED\_T AS db\_pvfc10\_big.SHIPPED\_T WITH NO DATA;

CREATE TABLE SKILL\_T AS db\_pvfc10\_big.SKILL\_T WITH NO DATA;

CREATE TABLE SUPPLIES\_T AS db\_pvfc10\_big.SUPPLIES\_T WITH NO DATA;

CREATE TABLE TERRITORY\_T AS db\_pvfc10\_big.TERRITORY\_T WITH NO DATA;

CREATE TABLE USES\_T AS db\_pvfc10\_big.USES\_T WITH NO DATA;

CREATE TABLE VENDOR\_T AS db\_pvfc10\_big.VENDOR\_T WITH NO DATA;

CREATE TABLE WORKSIN\_T AS db\_pvfc10\_big.WORKSIN\_T WITH NO DATA;

CREATE TABLE WORKCENTER\_T AS db\_pvfc10\_big.WORKCENTER\_T WITH NO DATA;

INSERT INTO CUSTOMERSHIPADDRESS\_T SELECT \* FROM db\_pvfc10\_big.CUSTOMERSHIPADDRESS\_T;

INSERT INTO CUSTOMER\_T SELECT \* FROM db\_pvfc10\_big.CUSTOMER\_T;

INSERT INTO DOESBUSINESSIN\_T SELECT \* FROM db\_pvfc10\_big.DOESBUSINESSIN\_T;

INSERT INTO EMPLOYEESKILLS\_T SELECT \* FROM db\_pvfc10\_big.EMPLOYEESKILLS\_T;

INSERT INTO EMPLOYEE\_T SELECT \* FROM db\_pvfc10\_big.EMPLOYEE\_T;

INSERT INTO ORDERLINE\_T SELECT \* FROM db\_pvfc10\_big.ORDERLINE\_T;

INSERT INTO ORDER\_T SELECT \* FROM db\_pvfc10\_big.ORDER\_T;

INSERT INTO PAYMENT\_T SELECT \* FROM db\_pvfc10\_big.PAYMENT\_T;

INSERT INTO PAYMENTTYPE\_T SELECT \* FROM db\_pvfc10\_big.PAYMENTTYPE\_T;

INSERT INTO PRODUCEDIN\_T SELECT \* FROM db\_pvfc10\_big.PRODUCEDIN\_T;

INSERT INTO PRODUCTLINE\_T SELECT \* FROM db\_pvfc10\_big.PRODUCTLINE\_T;

INSERT INTO PRODUCT\_T SELECT \* FROM db\_pvfc10\_big.PRODUCT\_T;

INSERT INTO RAWMATERIAL\_T SELECT \* FROM db\_pvfc10\_big.RAWMATERIAL\_T;

INSERT INTO SALESPERSON\_T SELECT \* FROM db\_pvfc10\_big.SALESPERSON\_T;

INSERT INTO SHIPPED\_T SELECT \* FROM db\_pvfc10\_big.SHIPPED\_T;

INSERT INTO SKILL\_T SELECT \* FROM db\_pvfc10\_big.SKILL\_T;

INSERT INTO SUPPLIES\_T SELECT \* FROM db\_pvfc10\_big.SUPPLIES\_T;

INSERT INTO TERRITORY\_T SELECT \* FROM db\_pvfc10\_big.TERRITORY\_T;

INSERT INTO USES\_T SELECT \* FROM db\_pvfc10\_big.USES\_T;

INSERT INTO VENDOR\_T SELECT \* FROM db\_pvfc10\_big.VENDOR\_T;

INSERT INTO WORKSIN\_T SELECT \* FROM db\_pvfc10\_big.WORKSIN\_T;

INSERT INTO WORKCENTER\_T SELECT \* FROM db\_pvfc10\_big.WORKCENTER\_T;

COLLECT STATISTICS ON CUSTOMERSHIPADDRESS\_T INDEX (ShipAddressID);

COLLECT STATISTICS ON CUSTOMER\_T INDEX (CustomerID);

COLLECT STATISTICS ON DOESBUSINESSIN\_T INDEX (CustomerID);

COLLECT STATISTICS ON EMPLOYEESKILLS\_T INDEX (EmployeeID);

COLLECT STATISTICS ON EMPLOYEE\_T INDEX (EmployeeID);

COLLECT STATISTICS ON ORDERLINE\_T INDEX (OrderLineID);

COLLECT STATISTICS ON ORDER\_T INDEX (OrderID);

COLLECT STATISTICS ON PAYMENT\_T INDEX (PaymentID);

COLLECT STATISTICS ON PAYMENTTYPE\_T INDEX (PaymentTypeID);

COLLECT STATISTICS ON PRODUCEDIN\_T INDEX (ProductID);

COLLECT STATISTICS ON PRODUCTLINE\_T INDEX (ProductLineID);

COLLECT STATISTICS ON PRODUCT\_T INDEX (ProductID);

COLLECT STATISTICS ON RAWMATERIAL\_T INDEX (MaterialID);

COLLECT STATISTICS ON SALESPERSON\_T INDEX (SalesPersonID);

COLLECT STATISTICS ON SHIPPED\_T INDEX (OrderLineID);

COLLECT STATISTICS ON SKILL\_T INDEX (SkillID);

COLLECT STATISTICS ON SUPPLIES\_T INDEX (VendorID);

COLLECT STATISTICS ON TERRITORY\_T INDEX (TerritoryID);

COLLECT STATISTICS ON USES\_T INDEX (ProductID);

COLLECT STATISTICS ON VENDOR\_T INDEX (VendorID);

COLLECT STATISTICS ON WORKSIN\_T INDEX (EmployeeID);

COLLECT STATISTICS ON WORKCENTER\_T INDEX (WorkCenterID);

#### db\_pvfc10\_std

CREATE TABLE CUSTOMER\_T AS db\_pvfc10\_std.CUSTOMER\_T WITH NO DATA;

CREATE TABLE DOESBUSINESSIN\_T AS db\_pvfc10\_std.DOESBUSINESSIN\_T WITH NO DATA;

CREATE TABLE EMPLOYEESKILLS\_T AS db\_pvfc10\_std.EMPLOYEESKILLS\_T WITH NO DATA;

CREATE TABLE EMPLOYEE\_T AS db\_pvfc10\_std.EMPLOYEE\_T WITH NO DATA;

CREATE TABLE ORDERLINE\_T AS db\_pvfc10\_std.ORDERLINE\_T WITH NO DATA;

CREATE TABLE ORDER\_T AS db\_pvfc10\_std.ORDER\_T WITH NO DATA;

CREATE TABLE PRODUCEDIN\_T AS db\_pvfc10\_std.PRODUCEDIN\_T WITH NO DATA;

CREATE TABLE PRODUCTLINE\_T AS db\_pvfc10\_std.PRODUCTLINE\_T WITH NO DATA;

CREATE TABLE PRODUCT\_T AS db\_pvfc10\_std.PRODUCT\_T WITH NO DATA;

CREATE TABLE RAWMATERIAL\_T AS db\_pvfc10\_std.RAWMATERIAL\_T WITH NO DATA;

CREATE TABLE SALESPERSON\_T AS db\_pvfc10\_std.SALESPERSON\_T WITH NO DATA;

CREATE TABLE SKILL\_T AS db\_pvfc10\_std.SKILL\_T WITH NO DATA;

CREATE TABLE SUPPLIES\_T AS db\_pvfc10\_std.SUPPLIES\_T WITH NO DATA;

CREATE TABLE TERRITORY\_T AS db\_pvfc10\_std.TERRITORY\_T WITH NO DATA;

CREATE TABLE VENDOR\_T AS db\_pvfc10\_std.VENDOR\_T WITH NO DATA;

INSERT INTO CUSTOMER\_T SELECT \* FROM db\_pvfc10\_std.CUSTOMER\_T;

INSERT INTO DOESBUSINESSIN\_T SELECT \* FROM db\_pvfc10\_std.DOESBUSINESSIN\_T;

INSERT INTO EMPLOYEESKILLS\_T SELECT \* FROM db\_pvfc10\_std.EMPLOYEESKILLS\_T;

INSERT INTO EMPLOYEE\_T SELECT \* FROM db\_pvfc10\_std.EMPLOYEE\_T;

INSERT INTO ORDERLINE\_T SELECT \* FROM db\_pvfc10\_std.ORDERLINE\_T;

INSERT INTO ORDER\_T SELECT \* FROM db\_pvfc10\_std.ORDER\_T;

INSERT INTO PRODUCEDIN\_T SELECT \* FROM db\_pvfc10\_std.PRODUCEDIN\_T;

INSERT INTO PRODUCTLINE\_T SELECT \* FROM db\_pvfc10\_std.PRODUCTLINE\_T;

INSERT INTO PRODUCT\_T SELECT \* FROM db\_pvfc10\_std.PRODUCT\_T;

INSERT INTO RAWMATERIAL\_T SELECT \* FROM db\_pvfc10\_std.RAWMATERIAL\_T;

INSERT INTO SALESPERSON\_T SELECT \* FROM db\_pvfc10\_std.SALESPERSON\_T;

INSERT INTO SKILL\_T SELECT \* FROM db\_pvfc10\_std.SKILL\_T;

INSERT INTO SUPPLIES\_T SELECT \* FROM db\_pvfc10\_std.SUPPLIES\_T;

INSERT INTO TERRITORY\_T SELECT \* FROM db\_pvfc10\_std.TERRITORY\_T;

INSERT INTO VENDOR\_T SELECT \* FROM db\_pvfc10\_std.VENDOR\_T;

COLLECT STATISTICS ON CUSTOMER\_T INDEX (CustomerID);

COLLECT STATISTICS ON DOESBUSINESSIN\_T INDEX (CustomerID);

COLLECT STATISTICS ON EMPLOYEESKILLS\_T INDEX (EmployeeID);

COLLECT STATISTICS ON EMPLOYEE\_T INDEX (EmployeeID);

COLLECT STATISTICS ON ORDERLINE\_T INDEX (OrderID);

COLLECT STATISTICS ON ORDER\_T INDEX (OrderID);

COLLECT STATISTICS ON PRODUCEDIN\_T INDEX (ProductID);

COLLECT STATISTICS ON PRODUCTLINE\_T INDEX (ProductLineID);

COLLECT STATISTICS ON PRODUCT\_T INDEX (ProductID);

COLLECT STATISTICS ON RAWMATERIAL\_T INDEX (MaterialID);

COLLECT STATISTICS ON SALESPERSON\_T INDEX (SalesPersonID);

COLLECT STATISTICS ON SKILL\_T INDEX (SkillID);

COLLECT STATISTICS ON SUPPLIES\_T INDEX (VendorID);

COLLECT STATISTICS ON TERRITORY\_T INDEX (TerritoryID);

COLLECT STATISTICS ON VENDOR\_T INDEX (VendorID);

### Hoffer, J., Venkataraman, R., & Topi, H.

Modern Database Management (11e)

#### db\_pvfc11\_big

CREATE TABLE CUSTOMERSHIPADDRESS\_T AS db\_pvfc11\_big.CUSTOMERSHIPADDRESS\_T WITH NO DATA;

CREATE TABLE CUSTOMER\_T AS db\_pvfc11\_big.CUSTOMER\_T WITH NO DATA;

CREATE TABLE DOESBUSINESSIN\_T AS db\_pvfc11\_big.DOESBUSINESSIN\_T WITH NO DATA;

CREATE TABLE EMPLOYEESKILLS\_T AS db\_pvfc11\_big.EMPLOYEESKILLS\_T WITH NO DATA;

CREATE TABLE EMPLOYEE\_T AS db\_pvfc11\_big.EMPLOYEE\_T WITH NO DATA;

CREATE TABLE ORDERLINE\_T AS db\_pvfc11\_big.ORDERLINE\_T WITH NO DATA;

CREATE TABLE ORDER\_T AS db\_pvfc11\_big.ORDER\_T WITH NO DATA;

CREATE TABLE PAYMENT\_T AS db\_pvfc11\_big.PAYMENT\_T WITH NO DATA;

CREATE TABLE PAYMENTTYPE\_T AS db\_pvfc11\_big.PAYMENTTYPE\_T WITH NO DATA;

CREATE TABLE PRODUCEDIN\_T AS db\_pvfc11\_big.PRODUCEDIN\_T WITH NO DATA;

CREATE TABLE PRODUCTLINE\_T AS db\_pvfc11\_big.PRODUCTLINE\_T WITH NO DATA;

CREATE TABLE PRODUCT\_T AS db\_pvfc11\_big.PRODUCT\_T WITH NO DATA;

CREATE TABLE RAWMATERIAL\_T AS db\_pvfc11\_big.RAWMATERIAL\_T WITH NO DATA;

CREATE TABLE SALESPERSON\_T AS db\_pvfc11\_big.SALESPERSON\_T WITH NO DATA;

CREATE TABLE SHIPPED\_T AS db\_pvfc11\_big.SHIPPED\_T WITH NO DATA;

CREATE TABLE SKILL\_T AS db\_pvfc11\_big.SKILL\_T WITH NO DATA;

CREATE TABLE SUPPLIES\_T AS db\_pvfc11\_big.SUPPLIES\_T WITH NO DATA;

CREATE TABLE TERRITORY\_T AS db\_pvfc11\_big.TERRITORY\_T WITH NO DATA;

CREATE TABLE USES\_T AS db\_pvfc10\_big.USES\_T WITH NO DATA;

CREATE TABLE VENDOR\_T AS db\_pvfc11\_big.VENDOR\_T WITH NO DATA;

CREATE TABLE WORKSIN\_T AS db\_pvfc11\_big.WORKSIN\_T WITH NO DATA;

CREATE TABLE WORKCENTER\_T AS db\_pvfc11\_big.WORKCENTER\_T WITH NO DATA;

INSERT INTO CUSTOMERSHIPADDRESS\_T SELECT \* FROM db\_pvfc11\_big.CUSTOMERSHIPADDRESS\_T;

INSERT INTO CUSTOMER\_T SELECT \* FROM db\_pvfc11\_big.CUSTOMER\_T;

INSERT INTO DOESBUSINESSIN\_T SELECT \* FROM db\_pvfc11\_big.DOESBUSINESSIN\_T;

INSERT INTO EMPLOYEESKILLS\_T SELECT \* FROM db\_pvfc11\_big.EMPLOYEESKILLS\_T;

INSERT INTO EMPLOYEE\_T SELECT \* FROM db\_pvfc11\_big.EMPLOYEE\_T;

INSERT INTO ORDERLINE\_T SELECT \* FROM db\_pvfc11\_big.ORDERLINE\_T;

INSERT INTO ORDER\_T SELECT \* FROM db\_pvfc11\_big.ORDER\_T;

INSERT INTO PAYMENT\_T SELECT \* FROM db\_pvfc11\_big.PAYMENT\_T;

INSERT INTO PAYMENTTYPE\_T SELECT \* FROM db\_pvfc11\_big.PAYMENTTYPE\_T;

INSERT INTO PRODUCEDIN\_T SELECT \* FROM db\_pvfc11\_big.PRODUCEDIN\_T;

INSERT INTO PRODUCTLINE\_T SELECT \* FROM db\_pvfc11\_big.PRODUCTLINE\_T;

INSERT INTO PRODUCT\_T SELECT \* FROM db\_pvfc11\_big.PRODUCT\_T;

INSERT INTO RAWMATERIAL\_T SELECT \* FROM db\_pvfc11\_big.RAWMATERIAL\_T;

INSERT INTO SALESPERSON\_T SELECT \* FROM db\_pvfc11\_big.SALESPERSON\_T;

INSERT INTO SHIPPED\_T SELECT \* FROM db\_pvfc11\_big.SHIPPED\_T;

INSERT INTO SKILL\_T SELECT \* FROM db\_pvfc11\_big.SKILL\_T;

INSERT INTO SUPPLIES\_T SELECT \* FROM db\_pvfc11\_big.SUPPLIES\_T;

INSERT INTO TERRITORY\_T SELECT \* FROM db\_pvfc11\_big.TERRITORY\_T;

INSERT INTO USES\_T SELECT \* FROM db\_pvfc11\_big.USES\_T;

INSERT INTO VENDOR\_T SELECT \* FROM db\_pvfc11\_big.VENDOR\_T;

INSERT INTO WORKSIN\_T SELECT \* FROM db\_pvfc11\_big.WORKSIN\_T;

INSERT INTO WORKCENTER\_T SELECT \* FROM db\_pvfc11\_big.WORKCENTER\_T;

COLLECT STATISTICS ON CUSTOMERSHIPADDRESS\_T INDEX (ShipAddressID);

COLLECT STATISTICS ON CUSTOMER\_T INDEX (CustomerID);

COLLECT STATISTICS ON DOESBUSINESSIN\_T INDEX (CustomerID);

COLLECT STATISTICS ON EMPLOYEESKILLS\_T INDEX (EmployeeID);

COLLECT STATISTICS ON EMPLOYEE\_T INDEX (EmployeeID);

COLLECT STATISTICS ON ORDERLINE\_T INDEX (OrderLineID);

COLLECT STATISTICS ON ORDER\_T INDEX (OrderID);

COLLECT STATISTICS ON PAYMENT\_T INDEX (PaymentID);

COLLECT STATISTICS ON PAYMENTTYPE\_T INDEX (PaymentTypeID);

COLLECT STATISTICS ON PRODUCEDIN\_T INDEX (ProductID);

COLLECT STATISTICS ON PRODUCTLINE\_T INDEX (ProductLineID);

COLLECT STATISTICS ON PRODUCT\_T INDEX (ProductID);

COLLECT STATISTICS ON RAWMATERIAL\_T INDEX (MaterialID);

COLLECT STATISTICS ON SALESPERSON\_T INDEX (SalesPersonID);

COLLECT STATISTICS ON SHIPPED\_T INDEX (OrderLineID);

COLLECT STATISTICS ON SKILL\_T INDEX (SkillID);

COLLECT STATISTICS ON SUPPLIES\_T INDEX (VendorID);

COLLECT STATISTICS ON TERRITORY\_T INDEX (TerritoryID);

COLLECT STATISTICS ON USES\_T INDEX (ProductID);

COLLECT STATISTICS ON VENDOR\_T INDEX (VendorID);

COLLECT STATISTICS ON WORKSIN\_T INDEX (EmployeeID);

COLLECT STATISTICS ON WORKCENTER\_T INDEX (WorkCenterID);

#### db\_pvfc11\_std

CREATE TABLE CUSTOMER\_T AS db\_pvfc11\_std.CUSTOMER\_T WITH NO DATA;

CREATE TABLE DOESBUSINESSIN\_T AS db\_pvfc11\_std.DOESBUSINESSIN\_T WITH NO DATA;

CREATE TABLE EMPLOYEESKILLS\_T AS db\_pvfc11\_std.EMPLOYEESKILLS\_T WITH NO DATA;

CREATE TABLE EMPLOYEE\_T AS db\_pvfc11\_std.EMPLOYEE\_T WITH NO DATA;

CREATE TABLE ORDERLINE\_T AS db\_pvfc11\_std.ORDERLINE\_T WITH NO DATA;

CREATE TABLE ORDER\_T AS db\_pvfc11\_std.ORDER\_T WITH NO DATA;

CREATE TABLE PRODUCEDIN\_T AS db\_pvfc11\_std.PRODUCEDIN\_T WITH NO DATA;

CREATE TABLE PRODUCTLINE\_T AS db\_pvfc11\_std.PRODUCTLINE\_T WITH NO DATA;

CREATE TABLE PRODUCT\_T AS db\_pvfc11\_std.PRODUCT\_T WITH NO DATA;

CREATE TABLE RAWMATERIAL\_T AS db\_pvfc11\_std.RAWMATERIAL\_T WITH NO DATA;

CREATE TABLE SALESPERSON\_T AS db\_pvfc11\_std.SALESPERSON\_T WITH NO DATA;

CREATE TABLE SKILL\_T AS db\_pvfc11\_std.SKILL\_T WITH NO DATA;

CREATE TABLE SUPPLIES\_T AS db\_pvfc11\_std.SUPPLIES\_T WITH NO DATA;

CREATE TABLE TERRITORY\_T AS db\_pvfc11\_std.TERRITORY\_T WITH NO DATA;

CREATE TABLE VENDOR\_T AS db\_pvfc11\_std.VENDOR\_T WITH NO DATA;

INSERT INTO CUSTOMER\_T SELECT \* FROM db\_pvfc11\_std.CUSTOMER\_T;

INSERT INTO DOESBUSINESSIN\_T SELECT \* FROM db\_pvfc11\_std.DOESBUSINESSIN\_T;

INSERT INTO EMPLOYEESKILLS\_T SELECT \* FROM db\_pvfc11\_std.EMPLOYEESKILLS\_T;

INSERT INTO EMPLOYEE\_T SELECT \* FROM db\_pvfc11\_std.EMPLOYEE\_T;

INSERT INTO ORDERLINE\_T SELECT \* FROM db\_pvfc11\_std.ORDERLINE\_T;

INSERT INTO ORDER\_T SELECT \* FROM db\_pvfc11\_std.ORDER\_T;

INSERT INTO PRODUCEDIN\_T SELECT \* FROM db\_pvfc11\_std.PRODUCEDIN\_T;

INSERT INTO PRODUCTLINE\_T SELECT \* FROM db\_pvfc11\_std.PRODUCTLINE\_T;

INSERT INTO PRODUCT\_T SELECT \* FROM db\_pvfc11\_std.PRODUCT\_T;

INSERT INTO RAWMATERIAL\_T SELECT \* FROM db\_pvfc11\_std.RAWMATERIAL\_T;

INSERT INTO SALESPERSON\_T SELECT \* FROM db\_pvfc11\_std.SALESPERSON\_T;

INSERT INTO SKILL\_T SELECT \* FROM db\_pvfc11\_std.SKILL\_T;

INSERT INTO SUPPLIES\_T SELECT \* FROM db\_pvfc11\_std.SUPPLIES\_T;

INSERT INTO TERRITORY\_T SELECT \* FROM db\_pvfc11\_std.TERRITORY\_T;

INSERT INTO VENDOR\_T SELECT \* FROM db\_pvfc11\_std.VENDOR\_T;

COLLECT STATISTICS ON CUSTOMER\_T INDEX (CustomerID);

COLLECT STATISTICS ON DOESBUSINESSIN\_T INDEX (CustomerID);

COLLECT STATISTICS ON EMPLOYEESKILLS\_T INDEX (EmployeeID);

COLLECT STATISTICS ON EMPLOYEE\_T INDEX (EmployeeID);

COLLECT STATISTICS ON ORDERLINE\_T INDEX (OrderID);

COLLECT STATISTICS ON ORDER\_T INDEX (OrderID);

COLLECT STATISTICS ON PRODUCEDIN\_T INDEX (ProductID);

COLLECT STATISTICS ON PRODUCTLINE\_T INDEX (ProductLineID);

COLLECT STATISTICS ON PRODUCT\_T INDEX (ProductID);

COLLECT STATISTICS ON RAWMATERIAL\_T INDEX (MaterialID);

COLLECT STATISTICS ON SALESPERSON\_T INDEX (SalesPersonID);

COLLECT STATISTICS ON SKILL\_T INDEX (SkillID);

COLLECT STATISTICS ON SUPPLIES\_T INDEX (VendorID);

COLLECT STATISTICS ON TERRITORY\_T INDEX (TerritoryID);

COLLECT STATISTICS ON VENDOR\_T INDEX (VendorID);

### Hoffer, J., Venkataraman, R., & Topi, H.

Modern Database Management (12e)

#### db\_pvfc12\_big

CREATE TABLE CUSTOMERSHIPADDRESS\_T AS db\_pvfc12\_big.CUSTOMERSHIPADDRESS\_T WITH NO DATA;

CREATE TABLE CUSTOMER\_T AS db\_pvfc12\_big.CUSTOMER\_T WITH NO DATA;

CREATE TABLE DOESBUSINESSIN\_T AS db\_pvfc12\_big.DOESBUSINESSIN\_T WITH NO DATA;

CREATE TABLE EMPLOYEESKILLS\_T AS db\_pvfc12\_big.EMPLOYEESKILLS\_T WITH NO DATA;

CREATE TABLE EMPLOYEE\_T AS db\_pvfc12\_big.EMPLOYEE\_T WITH NO DATA;

CREATE TABLE ORDERLINE\_T AS db\_pvfc12\_big.ORDERLINE\_T WITH NO DATA;

CREATE TABLE ORDER\_T AS db\_pvfc12\_big.ORDER\_T WITH NO DATA;

CREATE TABLE PAYMENT\_T AS db\_pvfc12\_big.PAYMENT\_T WITH NO DATA;

CREATE TABLE PAYMENTTYPE\_T AS db\_pvfc12\_big.PAYMENTTYPE\_T WITH NO DATA;

CREATE TABLE PRODUCEDIN\_T AS db\_pvfc12\_big.PRODUCEDIN\_T WITH NO DATA;

CREATE TABLE PRODUCTLINE\_T AS db\_pvfc12\_big.PRODUCTLINE\_T WITH NO DATA;

CREATE TABLE PRODUCT\_T AS db\_pvfc12\_big.PRODUCT\_T WITH NO DATA;

CREATE TABLE RAWMATERIAL\_T AS db\_pvfc12\_big.RAWMATERIAL\_T WITH NO DATA;

CREATE TABLE SALESPERSON\_T AS db\_pvfc12\_big.SALESPERSON\_T WITH NO DATA;

CREATE TABLE SHIPPED\_T AS db\_pvfc12\_big.SHIPPED\_T WITH NO DATA;

CREATE TABLE SKILL\_T AS db\_pvfc12\_big.SKILL\_T WITH NO DATA;

CREATE TABLE SUPPLIES\_T AS db\_pvfc12\_big.SUPPLIES\_T WITH NO DATA;

CREATE TABLE TERRITORY\_T AS db\_pvfc12\_big.TERRITORY\_T WITH NO DATA;

CREATE TABLE USES\_T AS db\_pvfc10\_big.USES\_T WITH NO DATA;

CREATE TABLE VENDOR\_T AS db\_pvfc12\_big.VENDOR\_T WITH NO DATA;

CREATE TABLE WORKSIN\_T AS db\_pvfc12\_big.WORKSIN\_T WITH NO DATA;

CREATE TABLE WORKCENTER\_T AS db\_pvfc12\_big.WORKCENTER\_T WITH NO DATA;

INSERT INTO CUSTOMERSHIPADDRESS\_T SELECT \* FROM db\_pvfc12\_big.CUSTOMERSHIPADDRESS\_T;

INSERT INTO CUSTOMER\_T SELECT \* FROM db\_pvfc12\_big.CUSTOMER\_T;

INSERT INTO DOESBUSINESSIN\_T SELECT \* FROM db\_pvfc12\_big.DOESBUSINESSIN\_T;

INSERT INTO EMPLOYEESKILLS\_T SELECT \* FROM db\_pvfc12\_big.EMPLOYEESKILLS\_T;

INSERT INTO EMPLOYEE\_T SELECT \* FROM db\_pvfc12\_big.EMPLOYEE\_T;

INSERT INTO ORDERLINE\_T SELECT \* FROM db\_pvfc12\_big.ORDERLINE\_T;

INSERT INTO ORDER\_T SELECT \* FROM db\_pvfc12\_big.ORDER\_T;

INSERT INTO PAYMENT\_T SELECT \* FROM db\_pvfc12\_big.PAYMENT\_T;

INSERT INTO PAYMENTTYPE\_T SELECT \* FROM db\_pvfc12\_big.PAYMENTTYPE\_T;

INSERT INTO PRODUCEDIN\_T SELECT \* FROM db\_pvfc12\_big.PRODUCEDIN\_T;

INSERT INTO PRODUCTLINE\_T SELECT \* FROM db\_pvfc12\_big.PRODUCTLINE\_T;

INSERT INTO PRODUCT\_T SELECT \* FROM db\_pvfc12\_big.PRODUCT\_T;

INSERT INTO RAWMATERIAL\_T SELECT \* FROM db\_pvfc12\_big.RAWMATERIAL\_T;

INSERT INTO SALESPERSON\_T SELECT \* FROM db\_pvfc12\_big.SALESPERSON\_T;

INSERT INTO SHIPPED\_T SELECT \* FROM db\_pvfc12\_big.SHIPPED\_T;

INSERT INTO SKILL\_T SELECT \* FROM db\_pvfc12\_big.SKILL\_T;

INSERT INTO SUPPLIES\_T SELECT \* FROM db\_pvfc12\_big.SUPPLIES\_T;

INSERT INTO TERRITORY\_T SELECT \* FROM db\_pvfc12\_big.TERRITORY\_T;

INSERT INTO USES\_T SELECT \* FROM db\_pvfc12\_big.USES\_T;

INSERT INTO VENDOR\_T SELECT \* FROM db\_pvfc12\_big.VENDOR\_T;

INSERT INTO WORKSIN\_T SELECT \* FROM db\_pvfc12\_big.WORKSIN\_T;

INSERT INTO WORKCENTER\_T SELECT \* FROM db\_pvfc12\_big.WORKCENTER\_T;

COLLECT STATISTICS ON CUSTOMERSHIPADDRESS\_T INDEX (ShipAddressID);

COLLECT STATISTICS ON CUSTOMER\_T INDEX (CustomerID);

COLLECT STATISTICS ON DOESBUSINESSIN\_T INDEX (CustomerID);

COLLECT STATISTICS ON EMPLOYEESKILLS\_T INDEX (EmployeeID);

COLLECT STATISTICS ON EMPLOYEE\_T INDEX (EmployeeID);

COLLECT STATISTICS ON ORDERLINE\_T INDEX (OrderLineID);

COLLECT STATISTICS ON ORDER\_T INDEX (OrderID);

COLLECT STATISTICS ON PAYMENT\_T INDEX (PaymentID);

COLLECT STATISTICS ON PAYMENTTYPE\_T INDEX (PaymentTypeID);

COLLECT STATISTICS ON PRODUCEDIN\_T INDEX (ProductID);

COLLECT STATISTICS ON PRODUCTLINE\_T INDEX (ProductLineID);

COLLECT STATISTICS ON PRODUCT\_T INDEX (ProductID);

COLLECT STATISTICS ON RAWMATERIAL\_T INDEX (MaterialID);

COLLECT STATISTICS ON SALESPERSON\_T INDEX (SalesPersonID);

COLLECT STATISTICS ON SHIPPED\_T INDEX (OrderLineID);

COLLECT STATISTICS ON SKILL\_T INDEX (SkillID);

COLLECT STATISTICS ON SUPPLIES\_T INDEX (VendorID);

COLLECT STATISTICS ON TERRITORY\_T INDEX (TerritoryID);

COLLECT STATISTICS ON USES\_T INDEX (ProductID);

COLLECT STATISTICS ON VENDOR\_T INDEX (VendorID);

COLLECT STATISTICS ON WORKSIN\_T INDEX (EmployeeID);

COLLECT STATISTICS ON WORKCENTER\_T INDEX (WorkCenterID);

#### db\_pvfc12\_std

CREATE TABLE CUSTOMER\_T AS db\_pvfc12\_std.CUSTOMER\_T WITH NO DATA;

CREATE TABLE DOESBUSINESSIN\_T AS db\_pvfc12\_std.DOESBUSINESSIN\_T WITH NO DATA;

CREATE TABLE EMPLOYEESKILLS\_T AS db\_pvfc12\_std.EMPLOYEESKILLS\_T WITH NO DATA;

CREATE TABLE EMPLOYEE\_T AS db\_pvfc12\_std.EMPLOYEE\_T WITH NO DATA;

CREATE TABLE ORDERLINE\_T AS db\_pvfc12\_std.ORDERLINE\_T WITH NO DATA;

CREATE TABLE ORDER\_T AS db\_pvfc12\_std.ORDER\_T WITH NO DATA;

CREATE TABLE PRODUCEDIN\_T AS db\_pvfc12\_std.PRODUCEDIN\_T WITH NO DATA;

CREATE TABLE PRODUCTLINE\_T AS db\_pvfc12\_std.PRODUCTLINE\_T WITH NO DATA;

CREATE TABLE PRODUCT\_T AS db\_pvfc12\_std.PRODUCT\_T WITH NO DATA;

CREATE TABLE RAWMATERIAL\_T AS db\_pvfc12\_std.RAWMATERIAL\_T WITH NO DATA;

CREATE TABLE SALESPERSON\_T AS db\_pvfc12\_std.SALESPERSON\_T WITH NO DATA;

CREATE TABLE SKILL\_T AS db\_pvfc12\_std.SKILL\_T WITH NO DATA;

CREATE TABLE SUPPLIES\_T AS db\_pvfc12\_std.SUPPLIES\_T WITH NO DATA;

CREATE TABLE TERRITORY\_T AS db\_pvfc12\_std.TERRITORY\_T WITH NO DATA;

CREATE TABLE VENDOR\_T AS db\_pvfc12\_std.VENDOR\_T WITH NO DATA;

INSERT INTO CUSTOMER\_T SELECT \* FROM db\_pvfc12\_std.CUSTOMER\_T;

INSERT INTO DOESBUSINESSIN\_T SELECT \* FROM db\_pvfc12\_std.DOESBUSINESSIN\_T;

INSERT INTO EMPLOYEESKILLS\_T SELECT \* FROM db\_pvfc12\_std.EMPLOYEESKILLS\_T;

INSERT INTO EMPLOYEE\_T SELECT \* FROM db\_pvfc12\_std.EMPLOYEE\_T;

INSERT INTO ORDERLINE\_T SELECT \* FROM db\_pvfc12\_std.ORDERLINE\_T;

INSERT INTO ORDER\_T SELECT \* FROM db\_pvfc12\_std.ORDER\_T;

INSERT INTO PRODUCEDIN\_T SELECT \* FROM db\_pvfc12\_std.PRODUCEDIN\_T;

INSERT INTO PRODUCTLINE\_T SELECT \* FROM db\_pvfc12\_std.PRODUCTLINE\_T;

INSERT INTO PRODUCT\_T SELECT \* FROM db\_pvfc12\_std.PRODUCT\_T;

INSERT INTO RAWMATERIAL\_T SELECT \* FROM db\_pvfc12\_std.RAWMATERIAL\_T;

INSERT INTO SALESPERSON\_T SELECT \* FROM db\_pvfc12\_std.SALESPERSON\_T;

INSERT INTO SKILL\_T SELECT \* FROM db\_pvfc12\_std.SKILL\_T;

INSERT INTO SUPPLIES\_T SELECT \* FROM db\_pvfc12\_std.SUPPLIES\_T;

INSERT INTO TERRITORY\_T SELECT \* FROM db\_pvfc12\_std.TERRITORY\_T;

INSERT INTO VENDOR\_T SELECT \* FROM db\_pvfc12\_std.VENDOR\_T;

COLLECT STATISTICS ON CUSTOMER\_T INDEX (CustomerID);

COLLECT STATISTICS ON DOESBUSINESSIN\_T INDEX (CustomerID);

COLLECT STATISTICS ON EMPLOYEESKILLS\_T INDEX (EmployeeID);

COLLECT STATISTICS ON EMPLOYEE\_T INDEX (EmployeeID);

COLLECT STATISTICS ON ORDERLINE\_T INDEX (OrderID);

COLLECT STATISTICS ON ORDER\_T INDEX (OrderID);

COLLECT STATISTICS ON PRODUCEDIN\_T INDEX (ProductID);

COLLECT STATISTICS ON PRODUCTLINE\_T INDEX (ProductLineID);

COLLECT STATISTICS ON PRODUCT\_T INDEX (ProductID);

COLLECT STATISTICS ON RAWMATERIAL\_T INDEX (MaterialID);

COLLECT STATISTICS ON SALESPERSON\_T INDEX (SalesPersonID);

COLLECT STATISTICS ON SKILL\_T INDEX (SkillID);

COLLECT STATISTICS ON SUPPLIES\_T INDEX (VendorID);

COLLECT STATISTICS ON TERRITORY\_T INDEX (TerritoryID);

COLLECT STATISTICS ON VENDOR\_T INDEX (VendorID);

### Watson, R.

Data Management: Databases and Organizations (4e)

#### db\_watson

CREATE TABLE ALIEN AS db\_watson.ALIEN WITH NO DATA;

CREATE TABLE ASSEMBLY AS db\_watson.ASSEMBLY WITH NO DATA;

CREATE TABLE CAR AS db\_watson.CAR WITH NO DATA;

CREATE TABLE DEPT AS db\_watson.DEPT WITH NO DATA;

CREATE TABLE DONOR AS db\_watson.DONOR WITH NO DATA;

CREATE TABLE EMP AS db\_watson.EMP WITH NO DATA;

CREATE TABLE EXPED AS db\_watson.EXPED WITH NO DATA;

CREATE TABLE GIFT AS db\_watson.GIFT WITH NO DATA;

CREATE TABLE ITEM AS db\_watson.ITEM WITH NO DATA;

CREATE TABLE LINEITEM AS db\_watson.LINEITEM WITH NO DATA;

CREATE TABLE MONARCH AS db\_watson.MONARCH WITH NO DATA;

CREATE TABLE NATION AS db\_watson.NATION WITH NO DATA;

CREATE TABLE PERSON AS db\_watson.PERSON WITH NO DATA;

CREATE TABLE PRODUCT AS db\_watson.PRODUCT WITH NO DATA;

CREATE TABLE QDEL AS db\_watson.QDEL WITH NO DATA;

CREATE TABLE QDEPT AS db\_watson.QDEPT WITH NO DATA;

CREATE TABLE QEMP AS db\_watson.QEMP WITH NO DATA;

CREATE TABLE QITEM AS db\_watson.QITEM WITH NO DATA;

CREATE TABLE QSALE AS db\_watson.QSALE WITH NO DATA;

CREATE TABLE QSPL AS db\_watson.QSPL WITH NO DATA;

CREATE TABLE SALE AS db\_watson.SALE WITH NO DATA;

CREATE TABLE SHR AS db\_watson.SHR WITH NO DATA;

CREATE TABLE STOCK AS db\_watson.STOCK WITH NO DATA;

CREATE TABLE YEARR AS db\_watson.YEARR WITH NO DATA;

INSERT INTO ALIEN SELECT \* FROM db\_watson.ALIEN;

INSERT INTO ASSEMBLY SELECT \* FROM db\_watson.ASSEMBLY;

INSERT INTO CAR SELECT \* FROM db\_watson.CAR;

INSERT INTO DEPT SELECT \* FROM db\_watson.DEPT;

INSERT INTO DONOR SELECT \* FROM db\_watson.DONOR;

INSERT INTO EMP SELECT \* FROM db\_watson.EMP;

INSERT INTO EXPED SELECT \* FROM db\_watson.EXPED;

INSERT INTO GIFT SELECT \* FROM db\_watson.GIFT;

INSERT INTO ITEM SELECT \* FROM db\_watson.ITEM;

INSERT INTO LINEITEM SELECT \* FROM db\_watson.LINEITEM;

INSERT INTO MONARCH SELECT \* FROM db\_watson.MONARCH;

INSERT INTO NATION SELECT \* FROM db\_watson.NATION;

INSERT INTO PERSON SELECT \* FROM db\_watson.PERSON;

INSERT INTO PRODUCT SELECT \* FROM db\_watson.PRODUCT;

INSERT INTO QDEL SELECT \* FROM db\_watson.QDEL;

INSERT INTO QDEPT SELECT \* FROM db\_watson.QDEPT;

INSERT INTO QEMP SELECT \* FROM db\_watson.QEMP;

INSERT INTO QITEM SELECT \* FROM db\_watson.QITEM;

INSERT INTO QSALE SELECT \* FROM db\_watson.QSALE;

INSERT INTO QSPL SELECT \* FROM db\_watson.QSPL;

INSERT INTO SALE SELECT \* FROM db\_watson.SALE;

INSERT INTO SHR SELECT \* FROM db\_watson.SHR;

INSERT INTO STOCK SELECT \* FROM db\_watson.STOCK;

INSERT INTO YEARR SELECT \* FROM db\_watson.YEARR;

COLLECT STATISTICS ON ALIEN INDEX (alnum);

COLLECT STATISTICS ON ASSEMBLY INDEX (prodid, subprodid);

COLLECT STATISTICS ON CAR INDEX (carid);

COLLECT STATISTICS ON DEPT INDEX (deptname);

COLLECT STATISTICS ON DONOR INDEX (donorno);

COLLECT STATISTICS ON EMP INDEX (empno);

COLLECT STATISTICS ON EXPED INDEX (transid);

COLLECT STATISTICS ON GIFT INDEX (yearr, donorno);

COLLECT STATISTICS ON ITEM INDEX (itemno);

COLLECT STATISTICS ON LINEITEM INDEX (lineno, saleno);

COLLECT STATISTICS ON MONARCH INDEX (monname, monnum);

COLLECT STATISTICS ON NATION INDEX (natcode);

COLLECT STATISTICS ON PERSON INDEX (personid);

COLLECT STATISTICS ON PRODUCT INDEX (prodid);

COLLECT STATISTICS ON QDEL INDEX (delno);

COLLECT STATISTICS ON QDEPT INDEX (deptname);

COLLECT STATISTICS ON QEMP INDEX (empno);

COLLECT STATISTICS ON QITEM INDEX (itemname);

COLLECT STATISTICS ON QSALE INDEX (saleno);

COLLECT STATISTICS ON QSPL INDEX (splno);

COLLECT STATISTICS ON SALE INDEX (saleno);

COLLECT STATISTICS ON SHR INDEX (shrcode);

COLLECT STATISTICS ON STOCK INDEX (stkcode);

COLLECT STATISTICS ON YEARR INDEX (yearr);