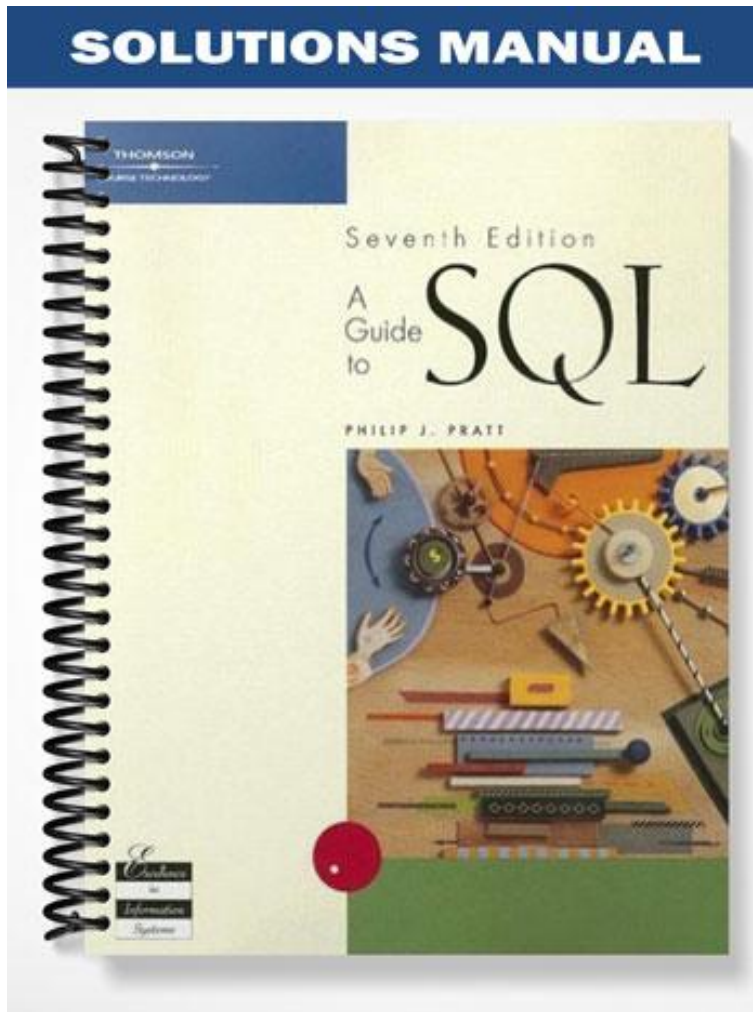


# SOLUTIONS MANUAL





## Chapter 2

# An Introduction to SQL

## Solutions

### Answers to Review Questions

1. A relation is a two-dimensional table in which the entries in the table are single-valued (each location in the table contains a single entry), each column has a distinct name (or attribute name), all values in a column are values of the same attribute, the order of the rows and columns is immaterial, and each row contains unique values.
2. A relational database is a collection of relations.
3. Rows in a relation are also called records or tuples. Columns are also called fields or attributes.
4. Use the CREATE TABLE command to create a table by typing the table name and then listing, in parentheses, the columns in the table.
5. The primary key of a table is the column, or collection of columns, that uniquely identifies a given row in the table. You usually identify primary keys by underlining the column name(s).
6. Use the DROP TABLE command.
7. The available data types are INTEGER, SMALLINT, DECIMAL, CHAR, and DATE.
8. A null data value (or null) is a special value that is used when the actual value for a column is unknown or unavailable. Use the NOT NULL clause to identify columns that cannot accept null values.
9. Use the INSERT command.
10. Use the SELECT command.
11. Use the UPDATE command.
12. Use the DELETE command.
13. You can use the DESCRIBE command to display or print the layout of a table in SQL\*Plus or SQL\*Plus Worksheet. You can use the SHOW COLUMNS command in MySQL to display or print the layout. You can use the Documenter in Access to display or print the layout.

**Exercise Note1:** In Access, students could create the tables and add data using Design view and Datasheet windows. They also could do the assignment in SQL view using the commands given in this chapter.

**Exercise Note2:** The various CREATE TABLE and INSERT commands are included in the script files on the data disk. Because these exercises can require a large amount of coding, you may want the students to create one table and add a few records. This gives students an understanding of the process. Then, use the use the appropriate script to drop the tables in the database. Finally, use the appropriate script to add all the tables and data. This also ensures that students have the correct descriptions and data for future exercises.

## Answers to Premiere Products Exercises

1. Use the CREATE TABLE commands in Figures 2.3, 2.29, 2.31, 2.33, and 2.35.

2.

```
INSERT INTO REP
VALUES
('20','Kaiser','Valerie','624 Randall','Grove','FL','33321',20542.50,0.05);
INSERT INTO REP
VALUES
('35','Hull','Richard','532 Jackson','Sheldon','FL','33553',39216.00,0.07);
INSERT INTO REP
VALUES
('65','Perez','Juan','1626 Taylor','Fillmore','FL','33336',23487.00,0.05);
```

3.

```
INSERT INTO CUSTOMER
VALUES
('148','Al's Appliance and Sport','2837 Greenway','Fillmore','FL','33336',6550.00,7500.00,'20');
INSERT INTO CUSTOMER
VALUES
('282','Brookings Direct','3827 Devon','Grove','FL','33321',431.50,10000.00,'35');
INSERT INTO CUSTOMER
VALUES
('356','Ferguson's','382 Wildwood','Northfield','FL','33146',5785.00,7500.00,'65');
INSERT INTO CUSTOMER
VALUES
('408','The Everything Shop','1828 Raven','Crystal','FL','33503',5285.25,5000.00,'35');
INSERT INTO CUSTOMER
VALUES
('462','Bargains Galore','3829 Central','Grove','FL','33321',3412.00,10000.00,'65');
INSERT INTO CUSTOMER
VALUES
('524','Kline's','838 Ridgeland','Fillmore','FL','33336',12762.00,15000.00,'20');
INSERT INTO CUSTOMER
VALUES
('608','Johnson's Department Store','372 Oxford','Sheldon','FL','33553',2106.00,10000.00,'65');
INSERT INTO CUSTOMER
VALUES
('687','Lee's Sport and Appliance','282 Evergreen','Altonville','FL','32543',2851.00,5000.00,'35');
INSERT INTO CUSTOMER
VALUES
('725','Deerfield's Four Seasons','282 Columbia','Sheldon','FL','33553',248.00,7500.00,'35');
INSERT INTO CUSTOMER
VALUES
('842','All Season','28 Lakeview','Grove','FL','33321',8221.00,7500.00,'20');
```

4.

```
INSERT INTO ORDERS
VALUES
('21608','20-OCT-2007','148');
INSERT INTO ORDERS
VALUES
('21610','20-OCT-2007','356');
```

```

VALUES
('21613','21-OCT-2007','408');
INSERT INTO ORDERS
VALUES
('21614','21-OCT-2007','282');
INSERT INTO ORDERS
VALUES
('21617','23-OCT-2007','608');
INSERT INTO ORDERS
VALUES
('21619','23-OCT-2007','148');
INSERT INTO ORDERS
VALUES
('21623','23-OCT-2007','608');

```

## 5.

```

INSERT INTO PART
VALUES
('AT94','Iron',50,'HW','3',24.95);
INSERT INTO PART
VALUES
('BV06','Home Gym',45,'SG','2',794.95);
INSERT INTO PART
VALUES
('CD52','Microwave Oven',32,'AP','1',165.00);
INSERT INTO PART
VALUES
('DL71','Cordless Drill',21,'HW','3',129.95);
INSERT INTO PART
VALUES
('DR93','Gas Range',8,'AP','2',495.00);
INSERT INTO PART
VALUES
('DW11','Washer',12,'AP','3',399.99);
INSERT INTO PART
VALUES
('FD21','Stand Mixer',22,'HW','3',159.95);
INSERT INTO PART
VALUES
('KL62','Dryer',12,'AP','1',349.95);
INSERT INTO PART
VALUES
('KT03','Dishwasher',8,'AP','3',595.00);
INSERT INTO PART
VALUES
('KV29','Treadmill',9,'SG','2',1390.00);

```

## 6.

```

INSERT INTO ORDER_LINE
VALUES
('21608','AT94',11,21.95);
INSERT INTO ORDER_LINE
VALUES
('21610','DR93',1,495.00);
INSERT INTO ORDER_LINE
VALUES
('21610','DW11',1,399.99);
INSERT INTO ORDER_LINE
VALUES
('21613','KL62',4,329.95);

```

```

INSERT INTO ORDER_LINE
VALUES
('21614','KT03',2,595.00);
INSERT INTO ORDER_LINE
VALUES
('21617','BV06',2,794.95);
INSERT INTO ORDER_LINE
VALUES
('21617','CD52',4,150.00);
INSERT INTO ORDER_LINE
VALUES
('21619','DR93',1,495.00);
INSERT INTO ORDER_LINE
VALUES
('21623','KV29',2,1290.00);

```

## Answers to Henry Books Case

1.

```

CREATE TABLE AUTHOR
(AUTHOR_NUM DECIMAL(2,0),
AUTHOR_LAST CHAR(12),
AUTHOR_FIRST CHAR(10) )
;
CREATE TABLE BOOK
(BOOK_CODE CHAR(4),
TITLE CHAR(40),
PUBLISHER_CODE CHAR(3),
TYPE CHAR(3),
PRICE DECIMAL(8,2),
PAPERBACK CHAR(1) )
;
CREATE TABLE BRANCH
(BRANCH_NUM DECIMAL(2,0),
BRANCH_NAME CHAR(50),
BRANCH_LOCATION CHAR(50),
NUM_EMPLOYEES DECIMAL(2,0) )
;
CREATE TABLE INVENTORY
(BOOK_CODE CHAR(4),
BRANCH_NUM DECIMAL(2,0),
ON_HAND DECIMAL(2,0) )
;
CREATE TABLE PUBLISHER
(PUBLISHER_CODE CHAR(3),
PUBLISHER_NAME CHAR(25),
CITY CHAR(20) )
;
CREATE TABLE WROTE
(BOOK_CODE CHAR(4),
AUTHOR_NUM DECIMAL(2,0),
SEQUENCE DECIMAL(2,0) )
;

```

2.

```

INSERT INTO BRANCH
VALUES
(1,'Henry Downtown','16 Riverview',10);

```

```

INSERT INTO BRANCH
VALUES
(2,'Henry On The Hill','1289 Bedford',6);
INSERT INTO BRANCH
VALUES
(3,'Henry Brentwood','Brentwood Mall',15);
INSERT INTO BRANCH
VALUES
(4,'Henry Eastshore','Eastshore Mall',9);

```

3.

```

INSERT INTO PUBLISHER
VALUES
('AH','Arkham House','Sauk City WI');
INSERT INTO PUBLISHER
VALUES
('AP','Arcade Publishing','New York');
INSERT INTO PUBLISHER
VALUES
('BA','Basic BOOKs','Boulder CO');
INSERT INTO PUBLISHER
VALUES
('BP','Berkley Publishing','Boston');
INSERT INTO PUBLISHER
VALUES
('BY','Back Bay BOOKs','New York');
INSERT INTO PUBLISHER
VALUES
('CT','Course Technology','Cambridge MA');
INSERT INTO PUBLISHER
VALUES
('FA','Fawcett BOOKs','New York');
INSERT INTO PUBLISHER
VALUES
('FS','Farrar Straus and Giroux','New York');
INSERT INTO PUBLISHER
VALUES
('HC','HarperCollins PUBLISHERs','New York');
INSERT INTO PUBLISHER
VALUES
('JP','Jove Publications','New York');
INSERT INTO PUBLISHER
VALUES
('JT','Jeremy P. Tarcher','Los Angeles');
INSERT INTO PUBLISHER
VALUES
('LB','Lb BOOKs','New York');
INSERT INTO PUBLISHER
VALUES
('MP','McPherson and Co.','Kingston');
INSERT INTO PUBLISHER
VALUES
('PE','Penguin USA','New York');
INSERT INTO PUBLISHER
VALUES
('PL','Plume','New York');
INSERT INTO PUBLISHER
VALUES
('PU','Putnam Publishing Group','New York');
INSERT INTO PUBLISHER
VALUES
('RH','Random House','New York');
INSERT INTO PUBLISHER
VALUES

```

```

('SB','Schoken BOOKs','New York');
INSERT INTO PUBLISHER
VALUES
('SC','Scribner','New York');
INSERT INTO PUBLISHER
VALUES
('SS','Simon and Schuster','New York');
INSERT INTO PUBLISHER
VALUES
('ST','Scholastic Trade','New York');
INSERT INTO PUBLISHER
VALUES
('TA','Taunton Press','Newtown CT');
INSERT INTO PUBLISHER
VALUES
('TB','Tor BOOKs','New York');
INSERT INTO PUBLISHER
VALUES
('TH','Thames and Hudson','New York');
INSERT INTO PUBLISHER
VALUES
('TO','Touchstone BOOKs','Westport CT');
INSERT INTO PUBLISHER
VALUES
('VB','Vintage BOOKs','New York');
INSERT INTO PUBLISHER
VALUES
('WN','W.W. Norton','New York');
INSERT INTO PUBLISHER
VALUES
('WP','Westview Press','Boulder CO');

```

4.

```

INSERT INTO AUTHOR
VALUES
(1,'Morrison','Toni');
INSERT INTO AUTHOR
VALUES
(2,'Solotaroff','Paul');
INSERT INTO AUTHOR
VALUES
(3,'Vintage','Vernor');
INSERT INTO AUTHOR
VALUES
(4,'Francis','Dick');
INSERT INTO AUTHOR
VALUES
(5,'Straub','Peter');
INSERT INTO AUTHOR
VALUES
(6,'King','Stephen');
INSERT INTO AUTHOR
VALUES
(7,'Pratt','Philip');
INSERT INTO AUTHOR
VALUES
(8,'Chase','Truddi');
INSERT INTO AUTHOR
VALUES
(9,'Collins','Bradley');
INSERT INTO AUTHOR
VALUES
(10,'Heller','Joseph');
INSERT INTO AUTHOR

```



```

VALUES
(11,'Wills','Gary');
INSERT INTO AUTHOR
VALUES
(12,'Hofstadter','Douglas R. ');
INSERT INTO AUTHOR
VALUES
(13,'Lee','Harper');
INSERT INTO AUTHOR
VALUES
(14,'Ambrose','Stephen E. ');
INSERT INTO AUTHOR
VALUES
(15,'Rowling','J.K. ');
INSERT INTO AUTHOR
VALUES
(16,'Salinger','J.D. ');
INSERT INTO AUTHOR
VALUES
(17,'Heaney','Seamus');
INSERT INTO AUTHOR
VALUES
(18,'Camus','Albert');
INSERT INTO AUTHOR
VALUES
(19,'Collins, Jr.','Bradley');
INSERT INTO AUTHOR
VALUES
(20,'Steinbeck','John');
INSERT INTO AUTHOR
VALUES
(21,'Castelman','Riva');
INSERT INTO AUTHOR
VALUES
(22,'Owen','Barbara');
INSERT INTO AUTHOR
VALUES
(23,'O'Rourke','Randy');
INSERT INTO AUTHOR
VALUES
(24,'Kidder','Tracy');
INSERT INTO AUTHOR
VALUES
(25,'Schleining','Lon');

```

5.

```

INSERT INTO BOOK
VALUES
('0180','A Deepness in the Sky','TB','SFI',7.19,'Y');
INSERT INTO BOOK
VALUES
('0189','Magic Terror','FA','HOR',7.99,'Y');
INSERT INTO BOOK
VALUES
('0200','The Stranger','VB','FIC',8.00,'Y');
INSERT INTO BOOK
VALUES
('0378','Venice','SS','ART',24.50,'N');
INSERT INTO BOOK
VALUES
('079X','Second Wind','PU','MYS',24.95,'N');
INSERT INTO BOOK
VALUES
('0808','The Edge','JP','MYS',6.99,'Y');

```

```
INSERT INTO BOOK
VALUES
('1351','Dreamcatcher: A Novel','SC','HOR',19.60,'N');
INSERT INTO BOOK
VALUES
('1382','Treasure Chests','TA','ART',24.46,'N');
INSERT INTO BOOK
VALUES
('138X','Beloved','PL','FIC',12.95,'Y');
INSERT INTO BOOK
VALUES
('2226','Harry Potter and the Prisoner of Azkaban','ST','SFI',13.96,'N');
INSERT INTO BOOK
VALUES
('2281','Van Gogh and Gauguin','WP','ART',21.00,'N');
INSERT INTO BOOK
VALUES
('2766','Of Mice and Men','PE','FIC',6.95,'Y');
INSERT INTO BOOK
VALUES
('2908','Electric Light','FS','POE',14.00,'N');
INSERT INTO BOOK
VALUES
('3350','Group: Six People in Search of a Life','BP','PSY',10.40,'Y');
INSERT INTO BOOK
VALUES
('3743','Nine Stories','LB','FIC',5.99,'Y');
INSERT INTO BOOK
VALUES
('3906','The Soul of a New Machine','BY','SCI',11.16,'Y');
INSERT INTO BOOK
VALUES
('5163','Travels with Charley','PE','TRA',7.95,'Y');
INSERT INTO BOOK
VALUES
('5790','Catch-22','SC','FIC',12.00,'Y');
INSERT INTO BOOK
VALUES
('6128','Jazz','PL','FIC',12.95,'Y');
INSERT INTO BOOK
VALUES
('6328','Band of Brothers','TO','HIS',9.60,'Y');
INSERT INTO BOOK
VALUES
('669X','A Guide to SQL','CT','CMP',37.95,'Y');
INSERT INTO BOOK
VALUES
('6908','Franny and Zooey','LB','FIC',5.99,'Y');
INSERT INTO BOOK
VALUES
('7405','East of Eden','PE','FIC',12.95,'Y');
INSERT INTO BOOK
VALUES
('7443','Harry Potter and the Goblet of Fire','ST','SFI',18.16,'N');
INSERT INTO BOOK
VALUES
('7559','The Fall','VB','FIC',8.00,'Y');
INSERT INTO BOOK
VALUES
('8092','Godel, Escher, Bach','BA','PHI',14.00,'Y');
INSERT INTO BOOK
VALUES
('8720','When Rabbit Howls','JP','PSY',6.29,'Y');
INSERT INTO BOOK
VALUES
('9611','Black House','RH','HOR',18.81,'N');
INSERT INTO BOOK
VALUES
```

```

('9627','Song of Solomon','PL','FIC',14.00,'Y');
INSERT INTO BOOK
VALUES
('9701','The Grapes of Wrath','PE','FIC',13.00,'Y');
INSERT INTO BOOK
VALUES
('9882','Slay Ride','JP','MYS',6.99,'Y');
INSERT INTO BOOK
VALUES
('9883','The Catcher in the Rye','LB','FIC',5.99,'Y');
INSERT INTO BOOK
VALUES
('9931','To Kill a Mockingbird','HC','FIC',18.00,'N');

```

6.

```

INSERT INTO WROTE
VALUES
('0180',3,1);
INSERT INTO WROTE
VALUES
('0189',5,1);
INSERT INTO WROTE
VALUES
('0200',18,1);
INSERT INTO WROTE
VALUES
('0378',11,1);
INSERT INTO WROTE
VALUES
('079X',4,1);
INSERT INTO WROTE
VALUES
('0808',4,1);
INSERT INTO WROTE
VALUES
('1351',6,1);
INSERT INTO WROTE
VALUES
('1382',23,2);
INSERT INTO WROTE
VALUES
('1382',25,1);
INSERT INTO WROTE
VALUES
('138X',1,1);
INSERT INTO WROTE
VALUES
('2226',15,1);
INSERT INTO WROTE
VALUES
('2281',9,2);
INSERT INTO WROTE
VALUES
('2281',19,1);
INSERT INTO WROTE
VALUES
('2766',20,1);
INSERT INTO WROTE
VALUES
('2908',17,1);
INSERT INTO WROTE
VALUES
('3350',2,1);
INSERT INTO WROTE

```

```
VALUES
('3743',16,1);
INSERT INTO WROTE
VALUES
('3906',24,1);
INSERT INTO WROTE
VALUES
('5163',20,1);
INSERT INTO WROTE
VALUES
('5790',10,1);
INSERT INTO WROTE
VALUES
('6128',1,1);
INSERT INTO WROTE
VALUES
('6328',14,1);
INSERT INTO WROTE
VALUES
('669X',7,1);
INSERT INTO WROTE
VALUES
('6908',16,1);
INSERT INTO WROTE
VALUES
('7405',20,1);
INSERT INTO WROTE
VALUES
('7443',15,1);
INSERT INTO WROTE
VALUES
('7559',18,1);
INSERT INTO WROTE
VALUES
('8092',12,1);
INSERT INTO WROTE
VALUES
('8720',8,1);
INSERT INTO WROTE
VALUES
('9611',5,2);
INSERT INTO WROTE
VALUES
('9611',6,1);
INSERT INTO WROTE
VALUES
('9627',1,1);
INSERT INTO WROTE
VALUES
('9701',20,1);
INSERT INTO WROTE
VALUES
('9882',4,1);
INSERT INTO WROTE
VALUES
('9883',16,1);
INSERT INTO WROTE
VALUES
('9931',13,1);
```

7.

```
INSERT INTO INVENTORY
VALUES
('0180',1,2);
```

```
INSERT INTO INVENTORY
VALUES
('0189',2,2);
INSERT INTO INVENTORY
VALUES
('0200',1,1);
INSERT INTO INVENTORY
VALUES
('0200',2,3);
INSERT INTO INVENTORY
VALUES
('0378',3,2);
INSERT INTO INVENTORY
VALUES
('079X',2,1);
INSERT INTO INVENTORY
VALUES
('079X',3,2);
INSERT INTO INVENTORY
VALUES
('079X',4,3);
INSERT INTO INVENTORY
VALUES
('0808',2,1);
INSERT INTO INVENTORY
VALUES
('1351',2,4);
INSERT INTO INVENTORY
VALUES
('1351',3,2);
INSERT INTO INVENTORY
VALUES
('1382',2,1);
INSERT INTO INVENTORY
VALUES
('138X',2,3);
INSERT INTO INVENTORY
VALUES
('2226',1,3);
INSERT INTO INVENTORY
VALUES
('2226',3,2);
INSERT INTO INVENTORY
VALUES
('2226',4,1);
INSERT INTO INVENTORY
VALUES
('2281',4,3);
INSERT INTO INVENTORY
VALUES
('2766',3,2);
INSERT INTO INVENTORY
VALUES
('2908',1,3);
INSERT INTO INVENTORY
VALUES
('2908',4,1);
INSERT INTO INVENTORY
VALUES
('3350',1,2);
INSERT INTO INVENTORY
VALUES
('3743',2,1);
INSERT INTO INVENTORY
VALUES
('3906',2,1);
INSERT INTO INVENTORY
VALUES
```

```
('3906',3,2);
INSERT INTO INVENTORY
VALUES
('5163',1,1);
INSERT INTO INVENTORY
VALUES
('5790',4,2);
INSERT INTO INVENTORY
VALUES
('6128',2,4);
INSERT INTO INVENTORY
VALUES
('6128',3,3);
INSERT INTO INVENTORY
VALUES
('6328',2,2);
INSERT INTO INVENTORY
VALUES
('669X',1,1);
INSERT INTO INVENTORY
VALUES
('6908',2,2);
INSERT INTO INVENTORY
VALUES
('7405',3,2);
INSERT INTO INVENTORY
VALUES
('7443',4,1);
INSERT INTO INVENTORY
VALUES
('7559',2,2);
INSERT INTO INVENTORY
VALUES
('8092',3,1);
INSERT INTO INVENTORY
VALUES
('8720',1,3);
INSERT INTO INVENTORY
VALUES
('9611',1,2);
INSERT INTO INVENTORY
VALUES
('9627',3,5);
INSERT INTO INVENTORY
VALUES
('9627',4,2);
INSERT INTO INVENTORY
VALUES
('9701',1,2);
INSERT INTO INVENTORY
VALUES
('9701',2,1);
INSERT INTO INVENTORY
VALUES
('9701',3,3);
INSERT INTO INVENTORY
VALUES
('9701',4,2);
INSERT INTO INVENTORY
VALUES
('9882',3,3);
INSERT INTO INVENTORY
VALUES
('9883',2,3);
INSERT INTO INVENTORY
VALUES
('9883',4,2);
INSERT INTO INVENTORY
```

```
VALUES
('9931',1,2);
```

## Answers to Alexamara Marina Group Exercise

1.

```
CREATE TABLE MARINA
(MARINA_NUM CHAR(4) PRIMARY KEY,
NAME CHAR(20),
ADDRESS CHAR(15),
CITY CHAR(15),
STATE CHAR(2),
ZIP CHAR(5) )
;
CREATE TABLE MARINA_SLIP
(SLIP_ID DECIMAL(4,0) PRIMARY KEY,
MARINA_NUM CHAR(4),
SLIP_NUM CHAR(4),
LENGTH DECIMAL(4,0),
RENTAL_FEE DECIMAL(8,2),
BOAT_NAME CHAR(50),
BOAT_TYPE CHAR(50),
OWNER_NUM CHAR(4) )
;
CREATE TABLE OWNER
(OWNER_NUM CHAR(4) PRIMARY KEY,
LAST_NAME CHAR(50),
FIRST_NAME CHAR(20),
ADDRESS CHAR(15),
CITY CHAR(15),
STATE CHAR(2),
ZIP CHAR(5) )
;
CREATE TABLE SERVICE_CATEGORY
(CATEGORY_NUM DECIMAL(4,0) PRIMARY KEY,
CATEGORY_DESCRIPTION CHAR(255) )
;
CREATE TABLE SERVICE_REQUEST
(SERVICE_ID DECIMAL(4,0) PRIMARY KEY,
SLIP_ID DECIMAL(4,0),
CATEGORY_NUM DECIMAL(4,0),
DESCRIPTION CHAR(255),
STATUS CHAR(255),
EST_HOURS DECIMAL(4,2),
SPENT_HOURS DECIMAL(4,2),
NEXT_SERVICE_DATE DATE )
;
```

2.

```
INSERT INTO MARINA
VALUES
('1','Alexamara East','108 2nd Ave.','Brinman','FL','32273');
INSERT INTO MARINA
VALUES
('2','Alexamara Central','283 Branston','W. Brinman','FL','32274');
```

3.

```

INSERT INTO OWNER
VALUES
('AD57','Adney','Bruce and Jean','208 Citrus','Bowton','FL','31313');
INSERT INTO OWNER
VALUES
('AN75','Anderson','Bill','18 Wilcox','Glander Bay','FL','31044');
INSERT INTO OWNER
VALUES
('BL72','Blake','Mary','2672 Commodore','Bowton','FL','31313');
INSERT INTO OWNER
VALUES
('EL25','Elend','Sandy and Bill','462 Riverside','Rivard','FL','31062');
INSERT INTO OWNER
VALUES
('FE82','Feenstra','Daniel','7822 Coventry','Kaleva','FL','32521');
INSERT INTO OWNER
VALUES
('JU92','Juarez','Maria','8922 Oak','Rivard','FL','31062');
INSERT INTO OWNER
VALUES
('KE22','Kelly','Alyssa','5271 Waters','Bowton','FL','31313');
INSERT INTO OWNER
VALUES
('NO27','Norton','Peter','2811 Lakewood','Lewiston','FL','32765');
INSERT INTO OWNER
VALUES
('SM72','Smeltz','Becky and Dave','922 Garland','Glander Bay','FL','31044');
INSERT INTO OWNER
VALUES
('TR72','Trent','Ashton','922 Crest','Bay Shores','FL','30992');

```

4.

```

INSERT INTO MARINA_SLIP
VALUES
(1,'1','A1',40,3800.00,'Anderson II','Sprite 4000','AN75');
INSERT INTO MARINA_SLIP
VALUES
(2,'1','A2',40,3800.00,'Our Toy','Ray 4025','EL25');
INSERT INTO MARINA_SLIP
VALUES
(3,'1','A3',40,3600.00,'Escape','Sprite 4000','KE22');
INSERT INTO MARINA_SLIP
VALUES
(4,'1','B1',30,2400.00,'Gypsy','Dolphin 28','JU92');
INSERT INTO MARINA_SLIP
VALUES
(5,'1','B2',30,2600.00,'Anderson III','Sprite 3000','AN75');
INSERT INTO MARINA_SLIP
VALUES
(6,'2','1',25,1800.00,'Bravo','Dolphin 25','AD57');
INSERT INTO MARINA_SLIP
VALUES
(7,'2','2',25,1800.00,'Chinook','Dolphin 22','FE82');
INSERT INTO MARINA_SLIP
VALUES
(8,'2','3',25,2000.00,'Listy','Dolphin 25','SM72');
INSERT INTO MARINA_SLIP
VALUES
(9,'2','4',30,2500.00,'Mermaid','Dolphin 28','BL72');
INSERT INTO MARINA_SLIP
VALUES
(10,'2','5',40,4200.00,'Axxon II','Dolphin 40','NO27');

```



```
INSERT INTO MARINA_SLIP
VALUES
(11,'2','6',40,4200.00,'Karvel','Ray 4025','TR72');
```

5.

```
INSERT INTO SERVICE_CATEGORY
VALUES
(1,'Routine engine maintenance');
INSERT INTO SERVICE_CATEGORY
VALUES
(2,'Engine repair');
INSERT INTO SERVICE_CATEGORY
VALUES
(3,'Air conditioning');
INSERT INTO SERVICE_CATEGORY
VALUES
(4,'Electrical systems');
INSERT INTO SERVICE_CATEGORY
VALUES
(5,'Fiberglass repair');
INSERT INTO SERVICE_CATEGORY
VALUES
(6,'Canvas installation');
INSERT INTO SERVICE_CATEGORY
VALUES
(7,'Canvas repair');
INSERT INTO SERVICE_CATEGORY
VALUES
(8,'Electronic systems (radar, GPS, autopilots, etc.)');
```

6. The following commands use the hint given in the text for handling null service dates:

```
INSERT INTO SERVICE_REQUEST
VALUES
(1,1,3,'Air conditioner periodically stops with code indicating low coolant level. Diagnose and
repair.','Technician has verified the problem. Air conditioning specialist has been
called.','4','2','12-JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(2,5,4,'Fuse on port motor blown on two occasions. Diagnose and repair.','Open','2','0','12-JUL-
2007');
INSERT INTO SERVICE_REQUEST
VALUES
(3,4,1,'Oil change and general routine maintenance (check fluid levels, clean sea strainers
etc.)','Service call has been scheduled.','1','0','16-JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(4,1,2,'Engine oil level has been dropping drastically. Diagnose and repair.','Open','2','0','13-
JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(5,3,5,'Open pockets at base of two stantions.','Technician has completed the initial filling of the
open pockets. Will complete the job after the initial fill has had sufficient time to
dry.','4','2','13-JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(6,11,4,'Electric-flush system periodically stops functioning. Diagnose and
repair.','Open','3','0','31-DEC-2010');
INSERT INTO SERVICE_REQUEST
VALUES
(7,6,2,'Engine overheating. Loss of coolant. Diagnose and repair.','Open','2','0','13-JUL-2007');
INSERT INTO SERVICE_REQUEST
```

```
VALUES
(8,6,2,'Heat exchanger not operating correctly.','Technician has determined that the exchanger is
faulty. New exchanger has been ordered.','4','1','17-JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(9,7,6,'Canvas severely damaged in windstorm. Order and install new canvas.','Open','8','0','16-JUL-
2007');
INSERT INTO SERVICE_REQUEST
VALUES
(10,2,8,'Install new GPS and chart plotter','Scheduled','7','0','17-JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(11,2,3,'Air conditioning unit shuts down with HHH showing on the control panel.','Technician not
able to replicate the problem. Air conditioning unit ran fine through multiple tests. Owner to
notify technician if the problem recurs.','1','1','31-DEC-2010');
INSERT INTO SERVICE_REQUEST
VALUES
(12,4,8,'Both speed and depth readings on data unit are significantly less than the owner thinks
they should be.','Technician has scheduled appointment with owner to attempt to verify the
problem.','2','0','16-JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(13,8,2,'Engine seems to be making clattering (customer description) noise.','Technician suspects
problem with either propeller or shaft and has scheduled the boat to be pulled from the water for
further investigation.','5','2','12-JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(14,7,5,'Owner accident caused damage to forward portion of port side.','Technician has scheduled
repair.','6','0','13-JUL-2007');
INSERT INTO SERVICE_REQUEST
VALUES
(15,11,7,'Canvas leaks around zippers in heavy rain. Install overlap around zippers to prevent
leaks.','Overlap has been created. Installation has been scheduled.','8','3','17-JUL-2007');
UPDATE SERVICE_REQUEST
SET NEXT_SERVICE_DATE = Null
WHERE NEXT_SERVICE_DATE = '31-DEC-2010';
```

## Chapter 2

# An Introduction to SQL

### At a Glance

#### Table of Contents

- Overview
- Objectives
- Teaching Tips
- Instructor Notes
- Quick Quizzes
- Classroom Activities
- Discussion Questions
- Key Terms

### Lecture Notes

#### Overview

In this chapter, students learn about the relational model. Students learn how to create a database by describing and defining the tables and columns that make up the database. They also learn how to assign data types and how to use null values. Finally, students learn how to describe a table's layout using SQL.

#### Chapter Objectives

- Understand the concepts and terminology associated relational databases
- Create and run SQL commands in Oracle, Microsoft Access, and MySQL
- Identify and use data types to define columns in SQL tables
- Understand and use nulls
- Add rows to tables
- Describe a table's layout using SQL

<b>Teaching Tips</b>	<ul style="list-style-type: none"> <li>• This text describes how to use SQL in Oracle, Microsoft Access, and MySQL. For more information on Oracle, see <a href="http://www.oracle.com/database/index.html">http://www.oracle.com/database/index.html</a>. For more information on Microsoft Access, see <a href="http://office.microsoft.com">http://office.microsoft.com</a>. For more information on MySQL, see <a href="http://www.mysql.com/">http://www.mysql.com/</a>.</li> <li>• Familiarize your students with the DBMS they will be using in this chapter and subsequent chapters. If there are special issues related to your environment (networking, off-campus access), discuss them before starting the material.</li> <li>• Although SQL is a standard language, each vendor has its own SQL dialect, which may include extensions to the standard. This text attempts to identify those commands that are not part of the standard language and alert you to implementation differences.</li> <li>• For more information on the database concepts presented in this chapter as well as ANSI SQL standards, see <a href="http://www.service-architecture.com/database/articles/">http://www.service-architecture.com/database/articles/</a>. Also, Pratt and Adamski's <i>Concepts of Database Management, Fifth Edition</i> published by Course Technology is an excellent textbook to use with this textbook for a course requiring that students learn both concepts and SQL.</li> <li>• The data disk that accompanies this text contains Oracle and MySQL script files for the three databases. Students do not need to create the tables and add the data for all tables in the databases. You can have them create a small table, for example, the Orders table in the Premiere Products database or the Branch table in the Henry Books database, and then give them the scripts for the remaining tables.</li> </ul>
----------------------	---

## Instructor Notes

### Introduction

The relational model is the database management approach implemented on most personal computer database management systems (DBMSs) and many mainframe systems as well. **SQL (Structured Query Language)** is a popular language for retrieving and manipulating data in a relational database. SQL was developed in the mid-1970's at IBM. Other popular DBMSs that use SQL are Sybase (<http://www.sybase.com/home>) and Microsoft SQL Server (<http://www.microsoft.com/sql/>).

### Relational Databases

A relational database is essentially a collection of tables. Formally, tables are called relations. Use Figure 2.1 to emphasize that the Premiere Products database is a collection of tables.

### Entities, Attributes, and Relationships

Define and discuss the database terms: **entity**, **attribute**, and **relationship**. An entity is a person, place, object, event, or idea for which you want to store and process data. The entities of interest to Premiere Products are customers, orders, parts, and sales reps.

An attribute is a characteristic or property of an entity. The terms **column** and **field** are used as synonyms in many database systems. For Premiere Products, the attributes of interest for the entity "customer" are such things as customer name, street, city, and so on.

Define **relationship** and **one-to-many relationship**. A relationship is an association between entities. There is a one-to-many relationship between sales reps and customers in the Premiere Products database. One sales rep represents many customers but each customer is associated with only one sales rep.

In a relational database, each entity has its own tables and the attributes of the entity are columns in the table. A one-to-many relationship is handled by using common columns in the two tables.

A relation is a two-dimensional table with specific properties. These properties include:

- Entries in the table are single-valued.
- Each column has a distinct name.
- All values in a column are values of the same attribute.
- The order of the columns is immaterial.
- Each row is distinct.
- The order of the rows is immaterial.

If a structure contains entries that are not single-valued (**repeating groups** occur), then the structure is called an unnormalized relation. Use Figure 2.2 to discuss repeating groups and each of the six properties of a relation. Mention that the formal term for a table is relation, and the formal term for a row is **tuple**. A row also is called a record. Define **relational database**. A relational database is a collection of relations.

DBDL (Database Definition Language) is a commonly accepted shorthand notation for showing the structure of a table. After the name of the table, all the columns in the table are listed within a set of parentheses. While each column in a table has a distinct name, the same column name can be used in more than one table within the same database. When two or more tables in a database use the same column name, **qualify** the column name, that is, combine the table name and the column name. To make each row distinct, one or more columns must uniquely identify a given row in a table. This column or collection of columns is called the **primary key**. Review the embedded Question and Answer on page 31 to make sure students understand why a primary key can consist of two or more columns.

## Quick Quizzes

1. A(n) \_\_\_\_\_ is a person, place, object, event, or idea for which you want to store and process data.  
Answer: entity
2. A(n) \_\_\_\_\_ is a characteristic or property of an entity.  
Answer: attribute
3. A(n) \_\_\_\_\_ is the association between entities.  
Answer: relationship

## Database Creation

The layout of each table that the database will contain must be described before data in a table is loaded and accessed. Use the SQL command **CREATE TABLE** to describe the layout of a table. Following the word **TABLE** is the name of the table, and then the names and data types of the columns included in the table. The **data type** indicates the type of data that the column can contain (for example, characters, numbers, or dates) as well as the maximum number of characters or digits that the column can store. Restrictions for naming tables and columns are:

- Names cannot exceed 18 characters
- Names must start with a letter
- Names can contain only letters, numbers, and underscores (\_)
- Names cannot contain spaces.

Review the naming restrictions for tables and columns with students. Use Example 1 (Figure 2.3) to explain the **CREATE TABLE** statement. Point out that the rules for naming tables and column vary slightly from one implementation of SQL to another. The names used in this text should work for any implementation of SQL. Mention that commands are free-format in SQL, that is, no rule says that a particular word must begin in a particular position on the line.

## **Quick Quizzes**

1. The SQL command used to describe the layout of a table is \_\_\_\_\_.  
Answer: CREATE TABLE
2. The \_\_\_\_\_ indicates the type of data that the column can contain as well as the maximum number of characters or digits that the column can store.  
Answer: data type
3. Table and column names must start with a(n) \_\_\_\_\_.  
Answer: letter

## **Running SQL Commands**

The precise manner in which you run SQL commands depends on the program in which you are working.

If you are using Oracle 10g, you can complete your work in SQL\*Plus or SQL\*Plus Worksheet. SQL\*Plus is a program in which you type SQL commands at an SQL> prompt. To end and execute a command, type a semicolon (;) and then press the ENTER key. SQL\*Plus Worksheet is a GUI (graphical user interface) program in which you type commands in the upper pane of the window and then click the Execute button to run the command. Oracle displays the result in the lower pane. Use Figure 2.4 to explain SQL\*Plus and Figure 2.5 to explain SQL\*Plus Worksheet.

Microsoft Access is a DBMS that allows you to use SQL in SQL view. To use SQL in Access, you create a new query, click the View button list arrow and then click SQL View. You then can type SQL commands in the window and click the Run button to execute the SQL command. Use Figure 2.6 to describe using Access to create a table using SQL. Point out that Access does not have a DECIMAL data type. You use either CURRENCY or NUMBER.

MySQL is an open-source DBMS that supports SQL. In MySQL, you enter SQL commands at the mysql> prompt. To end and execute a command, type a semicolon (;) and then press the ENTER key. Use Figure 2.7 to explain using MySQL to create a table.

## **Editing SQL Commands**

When using Oracle SQL\*Plus Worksheet or Access SQL, you can edit your commands or correct errors by using the mouse and the correction techniques you might use in a word processor. In Oracle SQL\*Plus and MySQL, you must edit the on-screen command to make changes.

Use Figures 2.8 through 2.10 to illustrate editing SQL commands with SQL\*Plus. Point out that you must edit commands one line at a time. The most recent command you entered is stored in a special area of memory called the **command buffer**.

Use Figures 2.11 and 2.12 to illustrate editing SQL commands in MySQL. In MySQL, the most recent command you entered is stored in a special area of memory called the **statement history**.

## **Dropping a Table**

Another way of correcting errors in a table is to delete (drop) the table and start over. To delete a table, use the **DROP TABLE** command. The word TABLE is followed by the name of the table you want to delete and a semicolon. Point out that the DROP TABLE command also deletes any data that may have been inserted into the table. Review the embedded Question and Answer.

## **Data Types**

Use Figure 2.13 to describe some of the more common data types. In Microsoft Access, the Text data type corresponds to the CHAR data type. Date/Time corresponds to DATE and Number corresponds to DECIMAL, INTEGER, and SMALLINT. Access also has a Currency data type for monetary values.

## Nulls

There are times when the values for one or more columns for a row being added to a table are unknown or unavailable. To handle this problem, use a special value called a **null data value** or simply a **null**. Null is used to represent situation in which an actual value is unknown, unavailable, or not applicable. When creating a table, you can specify when to allow null in individual columns. Use the embedded Question and Answer to explain why null values are not allowed for the primary key.

### Implementation of Nulls

In SQL, you use the NOT NULL clause in a CREATE TABLE command to columns that cannot contain null values. The default is to allow nulls. Review the CREATE TABLE example on page 43.

### Loading a Table with Data

After tables have been created in a database, use the INSERT command to load data.

### The INSERT Command

The **INSERT** command adds rows to a table. To use this command, type INSERT INTO followed by the name of table. Then, type the VALUES command followed by the specific values to be inserted in parentheses. Point out that character values (values whose data type is CHAR) must be enclosed in single quotation marks. Use Example 2 (Figure 2.14) to illustrate the INSERT command.

Use Example 3 (Figures 2.15 through 2.18) to illustrate editing an INSERT command in SQL\*Plus to easily add more data to a table. Use Figure 2.19 to illustrate editing an INSERT command in MySQL.

### The INSERT Command with Nulls

To enter a null value into a table, use a special format of the INSERT command. In this special format, you identify the names of the columns that will accept non-null values, and then list only these non-null values after the VALUES command. Use Example 4 (Figure 2.20) to illustrate inserting a row containing null values.

## Viewing Table Data

To view the data in a table, use the **SELECT** command. The most basic form of the SELECT statement consists of the word SELECT followed by an asterisk, followed by the word FROM and then the name of the table containing the data you want to view. This command will display all rows and columns in the table. Use Figure 2.21 to illustrate using a SELECT statement in SQL\*Plus. Use Figure 2.22 to illustrate using a SELECT statement in SQL\*Plus Worksheet. Use Figure 2.23 to illustrate using a SELECT statement in Access. Use Figures 2.24 through 2.26 to illustrate using a SELECT statement in MySQL.

## Correcting Errors in the Database

To change the value in a column, use the **UPDATE** command. Use Figure 2.27 to illustrate using an UPDATE command. To delete a record, use the **DELETE** command. Use Figure 2.28 to illustrate the DELETE command. Review the embedded Question and Answer to be sure students understand how to correct errors in the data.

## Quick Quizzes

1. Use the \_\_\_\_\_ command to view data in a table.  
Answer: SELECT
2. Use the \_\_\_\_\_ command to change the value in a column.  
Answer: UPDATE
3. Use the \_\_\_\_\_ command to delete a record.  
Answer: DELETE

## **Saving SQL Commands**

You can re-use SQL command without retyping them by saving SQL commands in a file. This file is called a script file. The exact manner in which you create and use saved files depends on the SQL implementation that you are using. Discuss the way to save SQL commands for the particular implementation that you and your students are using.

## **Creating the Remaining Database Tables**

To create the remaining tables in the Premiere Products database, use the appropriate CREATE TABLE and INSERT commands. Use Figure 2.29 to discuss the CREATE TABLE command for the CUSTOMER table. Use Figure 2.30 to discuss the INSERT commands for the CUSTOMER table. Use Figures 2.31 through 2.36 to discuss the creation of the remaining tables in the Premiere Products database. The data disk includes the script files necessary to create the Premiere Products database tables.

## **Describing a Table**

You may need to examine the structure of a table to see details concerning the columns in the table. Each DBMS provides a method to examine a table's structure.

In SQL\*Plus, use the **DESCRIBE** command to list all the columns in a table and their corresponding data types. Use Figure 2.37 to illustrate using the DESCRIBE command.

In Access, use the Documenter tool to produce documentation about tables and other objects stored in a database. Use Figure 2.38 to illustrate the documentation produced by the Documenter.

In MySQL, use the SHOW COLUMNS command to list all the columns in a table and their corresponding data types. Use Figure 2.39 to illustrate using the SHOW COLUMNS command.

## **Classroom Activities**

Ask students to work in teams and use the CREATE TABLE command to create a table of student grade data. Suggested attributes are: student id, student last name, student first name, GPA, and year in school.

## **Discussion Questions**

This chapter describes four implementations of SQL. Which one do you think would be the easiest to use? Why?

MySQL was described as an open-source DBMS. What is open-source software? What are the advantages of open-source software over proprietary (commercial) software?

## **Key Terms**

<b>attribute</b>	A characteristic or property of an entity
<b>buffer</b>	A special area in which the most recently entered command is stored
<b>column</b>	An attribute
<b>command buffer</b>	A special area in which the most recently entered command is stored
<b>CREATE TABLE</b>	The SQL command used to describe the layout of the table
<b>data type</b>	Type of data that a column can contain as well as the maximum number of characters or digits that the column can store
<b>DELETE</b>	The SQL command used to delete a row in a table
<b>DESCRIBE</b>	The SQL command used to list all the columns in a table and their corresponding data types
<b>Documenter</b>	The Access tool that produces documentation about tables and other objects stored in a database
<b>DROP TABLE</b>	The SQL command used to drop an entire table
<b>entity</b>	A person, place, object, event, or idea for which you want to store and process data
<b>field</b>	An attribute
<b>INSERT</b>	The SQL command used to add rows to a table
<b>Microsoft Access</b>	A database management system (DBMS) that let you create queries in SQL view
<b>MYSQL</b>	An open source database management system that supports the SQL language



<b>NOT NULL</b>	The clause in a CREATE TABLE command used to indicate which columns cannot contain null values
<b>null</b>	A data value used to represent situations in which an actual value is unknown, unavailable or not applicable
<b>null data value</b>	A data value used to represent situations in which an actual value is unknown, unavailable or not applicable
<b>one-to-many relationship</b>	A relationship in which one entity is associated with many other entities
<b>Oracle SQL*Plus</b>	A program in which you type SQL commands at a SQL> prompt
<b>Oracle SQL*Plus Worksheet</b>	A GUI (graphical user interface) program in which you type commands in the upper pane of the window and then click the Execute button to run the command
<b>primary key</b>	The column or collection of columns that uniquely identifies a given row in a table
<b>qualify</b>	To combine a column name with a table name
<b>record</b>	A row in a table
<b>relation</b>	A two dimensional table in which the entries are single valued; each column has a distinct name (or attribute name); all values in a column are values of the same attribute; the order of the rows and columns is immaterial; and each row contains unique values
<b>relational database relationship</b>	A collection of relations The association between entities
<b>repeating group</b>	Multiple entries in an individual location in a table
<b>script file</b>	A file that contains one or more SQL commands
<b>SELECT</b>	The SQL command used to view data in a table
<b>SQL</b>	Structured Query Language
<b>statement history</b>	A special area in which the most recently entered command is stored
<b>Structured Query Language</b>	A language used for retrieving and manipulating database data
<b>tuple</b>	A row in a table
<b>UPDATE</b>	The SQL command used to change a value in a table