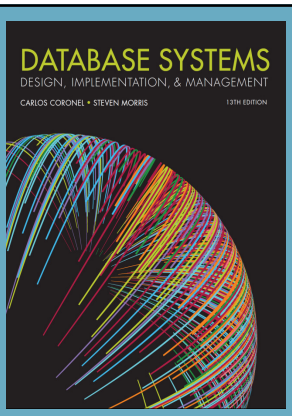

Chapter 7

Introduction to Structured Query Language (SQL)

Learning Objectives

- After completing this chapter, you will be able to:
  - Retrieve specified columns of data from a database
  - Join multiple tables in a single SQL query
  - Restrict data retrievals to rows that match complex criteria
  - Aggregate data across groups of rows
  - Create subqueries to preprocess data for inclusion in other queries
  - Identify and use a variety of SQL functions for string, numeric, and date manipulation
  - Explain the key principles in crafting a SELECT query

CENGAGE

Introduction to SQL

- SQL is relatively easy to learn
  - Nonprocedural language with basic command vocabulary set of less than 100 words
  - Differences in SQL dialects are minor
- Data type: specification about the kinds of data that can be stored in an attribute
  - Influence queries that retrieve data
- Fundamental types of data
  - Character data
  - Numeric data
  - Date data
- At the heart of SQL is the query
  - Covers both questions and actions

Basic SELECT Queries

- Each clause in a SELECT query performs a specific function
  - SELECT: specifies the attributes to be returned by the query
  - FROM: specifies the table(s) from which the data will be retrieved
  - WHERE: filters the rows of data based on provided criteria
  - GROUP BY: groups the rows of data into collections based on sharing the same values in one or more attributes
  - HAVING: filters the groups formed in the GROUP BY clause based on provided criteria
  - ORDER BY: sorts the final query result rows in ascending or descending order based on the values of one or more attributes
- SQL commands can be grouped together on a single line
  - Complex command sequences are best shown on separate lines, with space between the SQL command and the command's components

SELECT Statement Options (1 of 7)

- The SELECT query specifies the columns to be retrieved as a column list
  - Syntax:

    SELECT *columnlist*

    FROM *tablelist;*
  - The columnlist represents one or more attributes, separated by commas
  - A wildcard character is a symbol that can be used as a general substitute for other characters or commands
- Using column aliases
  - Alternative name for a column or table in a SQL statement
- Using computed columns
  - Computed column (also called a calculated column) represents a derived attribute

Comparison Operators

Symbol	Meaning
=	Equal to
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
<> or !=	Not equal to

SQL Queries

- 1. Write a SQL query that will list P_Code, P_description, P_Price, PQOH.
- 2. Write a SQL query that will list all the attributes from the PRODUCT table.
- 3. Write a SQL query that will list P_DESCRIPT, P_INDATE, P_PRICE, V_CODE for only the products with vendor code as 21344.
- 4. Write a SQL query that will list P_DESCRIP, P_QOH, P_MIN, P_Price for those with a price of less than or equal to 10
- 5. Write a SQL query that will list P_DESCRIPT, P_QOH, P_PRICE and the calculated product of P_QOH and P_PRICE.
- 6. Repeat the statement 5. above and rename the calculated column using the alias "TOTVALUE"

WHERE Clause Options

- Logical operators: AND, OR, and NOT
  - SQL allows you to include multiple conditions in a query through the use of these logical operators
  - Boolean algebra is dedicated to the use of logical operators
- Special operators
  - BETWEEN
  - IN
  - LIKE (using wildcard characters % or *)
  - IS NULL
  - NOT

Special and Logical Operators

- 1. Write a SQL query that will list P_DESCRIPT, P_INDATE, P_PRICE, V_CODE for only the products with vendor code as either 21344 or 24288.
- 2. Write a SQL query that will list P_DESCRIPT, P_INDATE, P_PRICE, V_CODE for all the products whose prices are less than 50 with an INDATE of greater than 15-Jan-2006.
- 3. Repeat the statement 1. above for all the products whose vendor is not 21344
- 4. Write a SQL query that will list all the products whose prices are between 50 and 100.
- 5. Write a SQL query that lists P_CODE, P_DESCRIPT, V_CODE for all the products with a null vendor code.
- 6. Write a SQL query that would list V_NAME, V_CONTACT, V_AREACODE, V_PHONE for all vendors whose last names begin with Smith.
- 7. Repeat the statement 6 above for all vendors whose last names do not begin with Smith.

CENGAGE

ORDER BY

- The ORDER BY clause is useful when the listing order is important to you. The default order is ascending.
  - Syntax:

```
SELECT      columnlist
FROM        tablelist
[WHERE      conditionlist ]
[ORDER BY   columnlist [ASC | DESC] ];
```

- The ORDER BY clause is useful when the listing order is important to you. The default order is ascending.
- Write a SQL query that will list P_DESCRIPT, V_CODE, P_PRICE for only the products that have date before 21-Jan-2006 with a price of less than or equal to 50. The results should be ordered in descending manner by the vendor code and price.
- SELECT P_DESCRIPT, V_CODE, P_PRICE
  FROM PRODUCT
  WHERE P_INDATE < '21-Jan-2006' AND P_PRICE <= 50
  ORDER BY V_CODE, P_PRICE DESC

CENGAGE

FROM Clause Options

- JOIN ON syntax
  - Express a join when the tables have no common attribute names
  - Query returns only the rows that meet the indicated join condition
  - Syntax:

    SELECT *column-list* FROM *table1* JOIN table2 ON *join-condition*

FROM Clause Options

- Outer joins
  - Returns not only the rows matching the join condition (rows with matching values in the common columns) and returns the rows with unmatched values
  - ANSI standard defines three types of outer joins: left, right, and full
- A natural join returns all rows with matching values in the matching columns and eliminates duplicate columns
- Write a SQL query that will list P_CODE, V_CODE, and V_NAME showing all vendor rows matching with product rows.
- SELECT P_CODE, VENDOR.V_CODE, V_NAME
  FROM VENDOR
  LEFT JOIN PRODUCT
  ON VENDOR.V_CODE = PRODUCT.V_CODE

Left Join Results

P_CODE	V_CODE	V_NAME
23109-HB	21225	Bryson, Inc.
SM-18277	21225	Bryson, Inc.
	21226	SuperLoo, Inc.
SW-23116	21231	D&E Supply
13-Q2/P2	21344	Gomez Bros.
14-Q1/L3	21344	Gomez Bros.
54778-2T	21344	Gomez Bros.
	22567	Dome Supply
1546-QQ2	23119	Randsets Ltd.
1558-QW1	23119	Randsets Ltd.
	24004	Brackman Bros.
2232/QTY	24288	ORDVA, Inc.
2232/QWE	24288	ORDVA, Inc.
89-WRE-Q	24288	ORDVA, Inc.
	25443	B&K, Inc.
	25501	Damal Supplies
11QER/31	25595	Rubicon Systems
2238/QPD	25595	Rubicon Systems
WR3/TT3	25595	Rubicon Systems

- Write a SQL query that will list P_CODE, V_CODE, and V_NAME showing all product rows matching with vendor rows.
- SELECT P_CODE, VENDOR.V_CODE, V_NAME
  FROM VENDOR
  LEFT JOIN PRODUCT
  ON VENDOR.V_CODE = PRODUCT.V_CODE

Right Outer Join Results

P_CODE	V_CODE	V_NAME
23114-AA		
PVC23DRT		
23109-HB	21225	Bryson, Inc.
SM-18277	21225	Bryson, Inc.
SW-23116	21231	D&E Supply
13-Q2/P2	21344	Gomez Bros.
14-Q1/L3	21344	Gomez Bros.
54778-2T	21344	Gomez Bros.
1546-QQ2	23119	Randsets Ltd.
1558-QW1	23119	Randsets Ltd.
2232/QTY	24288	ORDVA, Inc.
2232/QWE	24288	ORDVA, Inc.
89-WRE-Q	24288	ORDVA, Inc.
11QER/31	25595	Rubicon Systems
2238/QPD	25595	Rubicon Systems
WR3/TT3	25595	Rubicon Systems

CENGAGE

FULL OUTER JOIN Results

P_CODE	V_CODE	V_NAME
	21226	SuperLoo, Inc.
	22567	Dome Supply
	24004	Brackman Bros.
	25443	B&K, Inc.
	25501	Damal Supplies
11QER/31	25595	Rubicon Systems
13-Q2/P2	21344	Gomez Bros.
14-Q1/L3	21344	Gomez Bros.
1546-QQ2	23119	Randsets Ltd.
1558-QW1	23119	Randsets Ltd.
2232/QTY	24288	ORDVA, Inc.
2232/QWE	24288	ORDVA, Inc.
2238/QPD	25595	Rubicon Systems
23109-HB	21225	Bryson, Inc.
23114-AA		
54778-2T	21344	Gomez Bros.
89-WRE-Q	24288	ORDVA, Inc.
PVC23DRT		
SM-18277	21225	Bryson, Inc.
SW-23116	21231	D&E Supply
WR3/TT3	25595	Rubicon Systems

CENGAGE

Natural Join

P_CODE	P_DESCRIPT	V_CODE	V_NAME	V_AREACODE	V_PHONE
23109-HB	Claw hammer	21225	Bryson, Inc.	615	223-3234
SM-18277	1.25-in. metal screw, 25	21225	Bryson, Inc.	615	223-3234
SW-23116	2.5-in. wd. screw, 50	21231	D&E Supply	615	228-3245
13-Q2/P2	7.25-in. pwr. saw blade	21344	Gomez Bros.	615	889-2546
14-Q1/L3	9.00-in. pwr. saw blade	21344	Gomez Bros.	615	889-2546
54778-2T	Rat-tail file, 1/8-in. fine	21344	Gomez Bros.	615	889-2546
1546-QQ2	Hrd. cloth, 1/4-in., 2x50	23119	Randsets Ltd.	901	678-3998
1558-QW1	Hrd. cloth, 1/2-in., 3x50	23119	Randsets Ltd.	901	678-3998
2232/QTY	B&D jigsaw, 12-in. blade	24288	ORDVA, Inc.	615	898-1234
2232/QWE	B&D jigsaw, 8-in. blade	24288	ORDVA, Inc.	615	898-1234
89-WRE-Q	Hicut chain saw, 16 in.	24288	ORDVA, Inc.	615	898-1234
11QER/31	Power painter, 15 psi., 3-nozzle	25595	Rubicon Systems	904	456-0092
2238/QPD	B&D cordless drill, 1/2-in.	25595	Rubicon Systems	904	456-0092
WR3/TT3	Steel matting, 4'x8'x1/6", .5" mesh	25595	Rubicon Systems	904	456-0092

Crafting SELECT Queries (1 of 2)

- Know your data
  - The importance of understanding the data model that you are working in cannot be overstated
  - Real-world databases are messy; most database systems remain in service in an organization for decades
- Know the problem
  - Understand the question you are attempting to answer
  - Information reporting requests will come from a range of sources; may be one-time events or ongoing operations within an application

Crafting SELECT Queries (2 of 2)

- Build one clause at a time
  - FROM
  - WHERE
  - SELECT
  - ORDER BY

Summary

- The SELECT statement is the main data retrieval command in SQL
- The column list represents one or more column names separated by commas
- Operations that join tables can be classified as inner joins and outer joins
- A natural join returns all rows with matching values in the matching columns and eliminates duplicate columns
- The ORDER BY clause is used to sort the output of a SELECT statement

Solutions

```
1.
SELECT      P_Code, P_description, P_Price, PQOH
FROM        PRODUCT
2.
SELECT      *
FROM        PRODUCT
3.
SELECT      P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM        PRODUCT
WHERE       V_CODE = 21344
4.
SELECT      P_DESCRIP, P_QOH, P_MIN, P_Price
FROM        PRODUCT
WHERE       P_PRICE <= 10
5.
SELECT      P_DESCRIPT, P_QOH, P_PRICE, P_QOH*P_PRICE
FROM        PRODUCT
6.
SELECT      P_DESCRIPT, P_QOH, P_PRICE, P_QOH*P_PRICE AS TOTVALUE
FROM        PRODUCT
```

CENGAGE

Solutions

```
1.
SELECT      P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM        PRODUCT
WHERE       V_CODE = 21344 OR V_CODE = 24288
2.
SELECT      P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM        PRODUCT
WHERE       P_PRICE < 50 AND P_INDATE > '15-Jan-2006'
3.
SELECT      P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM        PRODUCT
WHERE       NOT (V_CODE = 21344)
4.
SELECT      *
FROM        PRODUCT
WHERE       P_PRICE BETWEEN 50.00 AND 100.00
5.
SELECT      P_CODE, P_DESCRIPT, V_CODE
FROM        PRODUCT
WHERE       V_CODE IS NULL
6.
SELECT      V_NAME, V_CONTACT, V_AREACODE, V_PHONE
FROM        VENDOR
WHERE       V_CONTACT LIKE 'Smith%'
7.
SELECT      V_NAME, V_CONTACT, V_AREACODE, V_PHONE
FROM        VENDOR
WHERE       V_CONTACT NOT LIKE 'Smith%'
```

CENGAGE