Chapter 7
Introduction to Structured Query Language (SQL)

Learning Objectives

- In this chapter, you will learn:
  - Introduction to SQL
  - How to use SQL to query a database for useful information

Introduction to SQL

- Categories of SQL functions:
  - Data definition language (DDL)
  - Data manipulation language (DML)
- Nonprocedural language with basic command vocabulary set of less than 100 words
- Differences in SQL dialects are minor

Table 7.1 - SQL Data Definition Command

TABLE 7.1

SQL DATA DEFINITION COMMANDS

COMMAND OR OPTION	DESCRIPTION
CREATE SCHEMA AUTHORIZATION	Creates a database schema
CREATE TABLE	Creates a new table in the user's database schema
NOT NULL	Ensures that a column will not have null values
UNIQUE	Ensures that a column will not have duplicate values
PRIMARY KEY	Defines a primary key for a table
FOREIGN KEY	Defines a foreign key for a table
DEFAULT	Defines a default value for a column (when no value is given)
CHECK	Validates data in an attribute
CREATE INDEX	Creates an index for a table
CREATE VIEW	Creates a dynamic subset of rows and columns from one or more tables (see Chapter 8, Advanced SQL)
ALTER TABLE	Modifies a table's definition (adds, modifies, or deletes attributes or constraints)
CREATE TABLE AS	Creates a new table based on a query in the user's database schema
DROP TABLE	Permanently deletes a table (and its data)
DROP INDEX	Permanently deletes an index
DROP VIEW	Permanently deletes a view

Table 7.2 - SQL Data Manipulation Commands

TABLE 7.2

SQL DATA MANIPULATION COMMANDS

COMMAND OR OPTION	DESCRIPTION
INSERT	Inserts row(s) into a table
SELECT	Selects attributes from rows in one or more tables or views
WHERE	Restricts the selection of rows based on a conditional expression
GROUP BY	Groups the selected rows based on one or more attributes
HAVING	Restricts the selection of grouped rows based on a condition
ORDER BY	Orders the selected rows based on one or more attributes
UPDATE	Modifies an attribute's values in one or more table's rows
DELETE	Deletes one or more rows from a table
COMMIT	Permanently saves data changes
ROLLBACK	Restores data to its original values
Comparison operators	
=, <, >, <=, >=, <>, !=	Used in conditional expressions
Logical operators	
AND/OR/NOT	Used in conditional expressions
Special operators	Used in conditional expressions
BETWEEN	Checks whether an attribute value is within a range
IS NULL	Checks whether an attribute value is null
LIKE	Checks whether an attribute value matches a given string pattern
IN	Checks whether an attribute value matches any value within a value list
EXISTS	Checks whether a subquery returns any rows
DISTINCT	Limits values to unique values
Aggregate functions	Used with SELECT to return mathematical summaries on columns
COUNT	Returns the number of rows with non-null values for a given column
MIN	Returns the minimum attribute value found in a given column
MAX	Returns the maximum attribute value found in a given column
SUM	Returns the sum of all values for a given column
AVG	Returns the average of all values for a given column

Creating the Database

- Create database structure
  - RDBMS creates physical files that will hold database
  - Differs from one RDBMS to another
- **Authentication** is the process DBMS uses to verify that only registered users access the database
  - Required for the creation tables
  - User should log on to RDBMS using user ID and password created by database administrator

Listing Table Rows

SELECT: Command to list the contents

- Syntax - SELECT *columnlist* FROM *tablename*;

- **Wildcard character**(*): Substitute for other characters/command
- Columnlist represents one or more attributes, separated by commas
- Asterisk can be used as wildcard character to list all attributes

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

SELECT ALL yields

```
SELECT * FROM
P;
```

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

SELECT only PRICE less than $2.00 yields

```
SELECT * FROM P WHERE PRICE <
2.00;
```

P_CODE	P_DESCRIPT	PRICE
213345	9v battery	1.92
254467	100W bulb	1.47

SEL

```
SELECT * FROM P WHERE P_CODE =
311452;
```

P_CODE	P_DESCRIPT	PRICE
311452	Powerdrill	34.99

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

PROJECT PRICE yields

SELECT PRICE FROM P;

PRICE
5.26
25.15
10.99
1.92
1.47
34.99

PROJECT P_DESCRIPT and PRICE yields

SELECT P_DESCRIPT, PRICE FROM P;

P_DESCRIPT	PRICE
Flashlight	5.26
Lamp	25.15
Box Fan	10.99
9v battery	1.92
100W bulb	1.47
Powerdrill	34.99

PROJECT P_CODE and PRICE yields

SELECT P_CODE, PRICE FROM P;

P_CODE	PRICE
123456	5.26
123457	25.15
123458	10.99
213345	1.92
254467	1.47
311452	34.99

Inserting Table Rows with a SELECT Subquery

- Syntax
  - INSERT INTO *tablename1* SELECT *columnlist* FROM *tablename2*
- Used to add multiple rows using another table as source
- SELECT command - Acts as a subquery and is executed first
  - **Subquery**: Query embedded/nested inside another query

Selecting Rows Using Conditional Restrictions

- Can select partial table contents by placing restrictions on rows to be included
- Syntax enables to specify which rows to select:
  - SELECT *columnlist*
  - FROM *tablelist*
  - [WHERE *conditionlist*];
- WHERE clause adds conditional restrictions to the SELECT statement

Table 7.6 - Comparison Operators

- Adds conditional restrictions on selected character attributes and dates

TABLE 7.6

COMPARISON OPERATORS

SYMBOL	MEANING
=	Equal to
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
<> or !=	Not equal to

Comparison Operators: Computed Columns and Column Aliases

- SQL accepts any valid expressions/formulas in the computed columns
- **Alias**: Alternate name given to a column or table in any SQL statement to improve the readability
- Computed column, an alias, and date arithmetic can be used in a single query

```
SELECT    P_DESCRIPT, P_QOH, P_PRICE, P_QOH *
          P_PRICE
FROM      PRODUCT;
```

P_DESCRIPT	P_QOH	P_PRICE	Expr1
Power painter, 15 psi., 3-nozzle	8	109.99	879.92
7.25-in. pwr. saw blade	32	14.99	479.68
9.00-in. pwr. saw blade	18	17.49	314.82
Hrd. cloth, 1/4-in., 2x50	15	39.95	599.25
Hrd. cloth, 1/2-in., 3x50	23	43.99	1011.77
B&D jigsaw, 12-in. blade	8	109.92	879.36

```
SELECT    P_DESCRIPT, P_QOH, P_PRICE, P_QOH * P_PRICE AS TOTVALUE
FROM      PRODUCT;
```

P_DESCRIPT	P_QOH	P_PRICE	TOTVALUE
Power painter, 15 psi., 3-nozzle	8	109.99	879.92
7.25-in. pwr. saw blade	32	14.99	479.68
9.00-in. pwr. saw blade	18	17.49	314.82
Hrd. cloth, 1/4-in., 2x50	15	39.95	599.25
Hrd. cloth, 1/2-in., 3x50	23	43.99	1011.77
B&D jigsaw, 12-in. blade	8	109.92	879.36

Arithmetic Operators

- **The Rule of Precedence**: Establish the order in which computations are completed
- Performed in this order:
  - Operations within parentheses
  - Power operations
  - Multiplications and divisions
  - Additions and subtractions

Table 7.7 - The Arithmetic Operators

TABLE 7.7

THE ARITHMETIC OPERATORS

OPERATOR	DESCRIPTION
+	Add
-	Subtract
*	Multiply
/	Divide
^	Raise to the power of (some applications use ** instead of ^)

Logical Operators: AND, OR and NOT

- **OR** and **AND**: Used to link multiple conditional expressions in a WHERE or HAVING clause
  - **OR** requires only one of the conditional expressions to be true
  - **AND** requires all of the conditional expressions to be true
- **NOT** is used to negate the result of a conditional expression
- **Boolean algebra** is dedicated to the use to logical operations

The Logical OR

```
SELECT    P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM      PRODUCT
WHERE     V_CODE = 21344 OR V_CODE = 24288;
```

FIGURE 7.12 SELECTED PRODUCT TABLE ATTRIBUTES: THE LOGICAL OR

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
7.25-in. pwr. saw blade	13-Dec-15	14.99	21344
9.00-in. pwr. saw blade	13-Nov-15	17.49	21344
B&D jigsaw, 12-in. blade	30-Dec-15	109.92	24288
B&D jigsaw, 8-in. blade	24-Dec-15	99.87	24288
Rat-tail file, 1/8-in. fine	15-Dec-15	4.99	21344
Hicut chain saw, 16 in.	07-Feb-16	256.99	24288

The Logical AND

```
SELECT    P_DESCRIPT,    P_INDATE,    P_PRICE,
          V_CODE
FROM      PRODUCT
WHERE     P_PRICE < 50
AND       P_INDATE > '15-Jan-2010';
```

FIGURE 7.13 SELECTED PRODUCT TABLE ATTRIBUTES: THE LOGICAL AND

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
B&D cordless drill, 1/2-in.	20-Jan-16	38.95	25595
Claw hammer	20-Jan-16	9.95	21225
PVC pipe, 3.5-in., 8-ft	20-Feb-16	5.87	
1.25-in. metal screw, 25	01-Mar-16	6.99	21225
2.5-in. wd. screw, 50	24-Feb-16	8.45	21231

The Logical AND and OR

```
SELECT    P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM      PRODUCT
WHERE     (P_PRICE < 50 AND P_INDATE > '15-Jan-2010')
OR        V_CODE = 24288;
```

FIGURE 7.14 SELECTED PRODUCT TABLE ATTRIBUTES: THE LOGICAL AND AND OR

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
B&D jigsaw, 12-in. blade	30-Dec-15	109.92	24288
B&D jigsaw, 8-in. blade	24-Dec-15	99.87	24288
B&D cordless drill, 1/2-in.	20-Jan-16	38.95	25595
Claw hammer	20-Jan-16	9.95	21225
Hicut chain saw, 16 in.	07-Feb-16	256.99	24288
PVC pipe, 3.5-in., 8-ft	20-Feb-16	5.87	
1.25-in. metal screw, 25	01-Mar-16	6.99	21225
2.5-in. wd. screw, 50	24-Feb-16	8.45	21231

Special Operators

BETWEEN

- Checks whether attribute value is within a range

IS NULL

- Checks whether attribute value is null

LIKE

- Checks whether attribute value matches given string pattern

IN

- Checks whether attribute value matches any value within a value list

EXISTS

- Checks if subquery returns any rows

```
SELECT      *
FROM        PRODUCT
WHERE       P_PRICE BETWEEN 50.00 AND 100.00;
```

```
SELECT      P_CODE, P_DESCRIPT, V_CODE
FROM        PRODUCT
WHERE       V_CODE IS NULL;
```

```
SELECT      *
FROM        PRODUCT
WHERE       V_CODE IN (21344, 24288);
```

```
SELECT      V_CODE, V_NAME
FROM        VENDOR
WHERE       V_CODE IN (SELECT V_CODE FROM PRODUCT);
```

```
SELECT      *
FROM        VENDOR
WHERE       EXISTS (SELECT * FROM PRODUCT WHERE P_QOH <= P_MIN);
```

```
SELECT      V_NAME, V_CONTACT, V_AREACODE, V_PHONE
FROM        VENDOR
WHERE       V_CONTACT LIKE 'Smith%';
```

- % means any and all *following* or preceding characters are eligible. For example,
  'J%' includes Johnson, Jones, Jernigan, July, and J-231Q.
  'Jo%' includes Johnson and Jones.
  '%n' includes Johnson and Jernigan.
- _ means any *one* character may be substituted for the underscore. For example,
  '_23-456-6789' includes 123-456-6789, 223-456-6789, and 323-456-6789.
  '_23-_56-678_' includes 123-156-6781, 123-256-6782, and 823-956-6788.
  '_o_es' includes Jones, Cones, Cokes, totes, and roles.

Ordering a Listing

- **ORDER BY** clause is useful when listing order is important
- Syntax - SELECT *columnlist*

  FROM *tablelist*

  [WHERE *conditionlist*]

  [ORDER BY *columnlist* [ASC | DESC]];
- **Cascading order sequence**: Multilevel ordered sequence
  - Created by listing several attributes after the ORDER BY clause

```
SELECT      P_CODE, P_DESCRIPT, P_INDATE, P_PRICE
FROM        PRODUCT
ORDER BY    P_PRICE;
```

FIGURE 7.17 Selected PRODUCT table attributes: ordered by (ascending) P_PRICE

P_CODE	P_DESCRIPT	P_INDATE	P_PRICE
54778-2T	Rat-tail file, 1/8-in. fine	15-Dec-09	4.99
PVC23DRT	PVC pipe, 3.5-in., 8-ft	20-Feb-10	5.87
SM-18277	1.25-in. metal screw, 25	01-Mar-10	6.99
SW-23116	2.5-in. wd. screw, 50	24-Feb-10	8.45
23109-HB	Claw hammer	20-Jan-10	9.95
23114-AA	Sledge hammer, 12 lb.	02-Jan-10	14.40
13-Q2/P2	7.25-in. pwr. saw blade	13-Dec-09	14.99
14-Q1/L3	9.00-in. pwr. saw blade	13-Nov-09	17.49
2238/QPD	B&D cordless drill, 1/2-in.	20-Jan-10	38.95
1546-QQ2	Hrd. cloth, 1/4-in., 2x50	15-Jan-10	39.95
1558-QW1	Hrd. cloth, 1/2-in., 3x50	15-Jan-10	43.99
2232/QWE	B&D jigsaw, 8-in. blade	24-Dec-09	99.87
2232/QTY	B&D jigsaw, 12-in. blade	30-Dec-09	109.92
11QER/31	Power painter, 15 psi., 3-nozzle	03-Nov-09	109.99
WR3/TT3	Steel matting, 4'x8'x1/6", .5" mesh	17-Jan-10	119.95
89-WRE-Q	Hicut chain saw, 16 in.	07-Feb-10	256.99

```
SELECT      EMP_LNAME, EMP_FNAME, EMP_INITIAL, EMP_AREACODE, EMP_PHONE
FROM        EMPLOYEE
ORDER BY    EMP_LNAME, EMP_FNAME, EMP_INITIAL;
```

EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_AREACODE	EMP_PHONE
Brandon	Marie	G	901	882-0845
Diante	Jorge	D	615	890-4567
Genkazi	Leighla	W	901	569-0093
Johnson	Edward	E	615	898-4387
Jones	Anne	M	615	898-3456
Kolmycz	George	D	615	324-5456
Lange	John	P	901	504-4430
Lewis	Rhonda	G	615	324-4472
Saranda	Hermine	R	615	324-5505
Smith	George	A	615	890-2984
Smith	George	K	901	504-3339
Smith	Jeanine	K	615	324-7883
Smythe	Melanie	P	615	324-9006
Vandam	Rhett		901	675-8993
Washington	Rupert	E	615	890-4925
Wiesenbach	Paul	R	615	897-4358
Williams	Robert	D	615	890-3220

Listing Unique Values

- **DISTINCT** clause: Produces list of values that are unique
- Syntax - SELECT DISTINCT *columnlist*

  FROM *tablelist*;
- Placement of nulls does not affect list contents
  - In Oracle can place nulls at top of list