Chapter 3

How to retrieve data   
from a single table

Exercises

Enter and run your own SELECT statements

In these exercises, you’ll enter and run your own SELECT statements.

1. Write a SELECT statement that returns four columns from the Products table: product\_code, product\_name, list\_price, and discount\_percent. Then, run this statement to make sure it works correctly.

Add an ORDER BY clause to this statement that sorts the result set by list price in descending sequence. Then, run this statement again to make sure it works correctly. This is a good way to build and test a statement, one clause at a time.

1. Write a SELECT statement that returns one column from the Customers table named full\_name that joins the last\_name and first\_name columns.

Format this column with the last name, a comma, a space, and the first name like this:

Doe, John

Sort the result set by the last\_name column in ascending sequence.

Return only the customers whose last name begins with letters from M to Z.

NOTE: When comparing strings of characters, ‘M’ comes before any string of characters that begins with ‘M’. For example, ‘M’ comes before ‘Murach’.

1. Write a SELECT statement that returns these columns from the Products table:

product\_name The product\_name column

list\_price The list\_price column

date\_added The date\_added column

Return only the rows with a list price that’s greater than 500 and less than 2000.

Sort the result set by the date\_added column in descending sequence.

1. Write a SELECT statement that returns these column names and data from the Products table:

product\_name The product\_name column

list\_price The list\_price column

discount\_percent The discount\_percent column

discount\_amount A column that’s calculated from the previous two columns

discount\_price A column that’s calculated from the previous three columns

Round the discount\_amount and discount\_price columns to 2 decimal places.

Sort the result set by the discount\_price column in descending sequence.

Use the LIMIT clause so the result set contains only the first 5 rows.

1. Write a SELECT statement that returns these column names and data from the Order\_Items table:

item\_id The item\_id column

item\_price The item\_price column

discount\_amount The discount\_amount column

quantity The quantity column

price\_total A column that’s calculated by multiplying the item price by the quantity

discount\_total A column that’s calculated by multiplying the discount amount by the quantity

item\_total A column that’s calculated by subtracting the discount amount from the item price and then multiplying by the quantity

Only return rows where the item\_total is greater than 500.

Sort the result set by the item\_total column in descending sequence.

Work with nulls and test expressions

1. Write a SELECT statement that returns these columns from the Orders table:

order\_id The order\_id column

order\_date The order\_date column

ship\_date The ship\_date column

Return only the rows where the ship\_date column contains a null value.