Chapter 4

How to retrieve data   
from two or more tables

Exercises

1. Write a SELECT statement that joins the Categories table to the Products table and returns these columns: category\_name, product\_name, list\_price.

Sort the result set by the category\_name column and then by the product\_name column in ascending sequence.

1. Write a SELECT statement that joins the Customers table to the Addresses table and returns these columns: first\_name, last\_name, line1, city, state, zip\_code.

Return one row for each address for the customer with an email address of allan.sherwood@yahoo.com.

1. Write a SELECT statement that joins the Customers table to the Addresses table and returns these columns: first\_name, last\_name, line1, city, state, zip\_code.

Return one row for each customer, but only return addresses that are the shipping address for a customer.

1. Write a SELECT statement that joins the Customers, Orders, Order\_Items, and Products tables. This statement should return these columns: last\_name, first\_name, order\_date, product\_name, item\_price, discount\_amount, and quantity.

Use aliases for the tables.

Sort the final result set by the last\_name, order\_date, and product\_name columns.

1. Write a SELECT statement that returns the product\_name and list\_price columns from the Products table.

Return one row for each product that has the same list price as another product.   
*Hint: Use a self-join to check that the product\_id columns aren’t equal but the list\_price columns are equal.*

Sort the result set by the product\_name column.

1. Write a SELECT statement that returns these two columns:

category\_name The category\_name column from the Categories table

product\_id The product\_id column from the Products table

Return one row for each category that has never been used. *Hint: Use an outer join and only return rows where the product\_id column contains a null value.*

1. Use the UNION operator to generate a result set consisting of three columns from the Orders table:

ship\_status A calculated column that contains a value of SHIPPED or NOT SHIPPED

order\_id The order\_id column

order\_date The order\_date column

If the order has a value in the ship\_date column, the ship\_status column should contain a value of SHIPPED. Otherwise, it should contain a value of NOT SHIPPED.

Sort the final result set by the order\_date column.