8.1 a) SQL. b) rows, columns. c) ResultSet. d) primary key. e) WHERE. f) ORDER BY.
g) joining. h) database. i) foreign key. j) java.sql. k) Connection. l) Statement.

EXERCISES

8.2 Using the techniques shown in this chapter, define a complete query application for the books database. Provide a series of predefined queries, with an appropriate name for each query, displayed in a JComboBox. Also allow users to supply their own queries and add them to the JComboBox. Provide the following predefined queries:

   a) Select all authors from the Authors table.
   b) Select all publishers from the Publishers table.
   c) Select a specific author and list all books for that author. Include the title, year and ISBN number. Order the information alphabetically by the author's last name and first name.
   d) Select a specific publisher and list all books published by that publisher. Include the title, year and ISBN number. Order the information alphabetically by title.
   e) Provide any other queries you feel are appropriate.

8.3 Modify Exercise 8.2 to define a complete database manipulation application for the books database. In addition to the querying, the user should be able to edit existing data and add new data to the database (obeying referential and entity integrity constraints). Allow the user to edit the database in the following ways:

   a) Add a new author.
   b) Edit the existing information for an author.
   c) Add a new title for an author. (Remember that the book must have an entry in the AuthorISBN table.) Be sure to specify the publisher of the title.
   d) Add a new publisher.
   e) Edit the existing information for a publisher.
   f) For each of the preceding database manipulations, design an appropriate GUI to allow the user to perform the data manipulation.

8.4 Modify the Search capability in the address book example of Fig. 8.33–Fig. 8.38 to allow the user to scroll through the ResultSet in case there is more than one person with the specified last name in the addressbook database. Provide an appropriate GUI.

8.5 Modify the address book example of Fig. 8.33–Fig. 8.38 to enable each address book entry to have multiple addresses, phone numbers and e-mail addresses. The user of the program should be able to view multiple addresses, phone numbers and e-mail addresses. The user also should be able to add, update or delete individual addresses, phone numbers and e-mail addresses. [Note: This exercise is large and requires substantial modifications to the original classes in the address book example.]

BIBLIOGRAPHY

