Chapter 12 Introduction to Java Database Programming

1. A JDBC application loads an appropriate driver using the Driver interface, connects to the database using the Connection interface, creates and executes SQL statements using the Statement interface, and processes the result using the ResultSet interface if the statements return results.

2. Use the Class.for(driverName) method to load the driver with its full name.

3. To expose properties whose get and set methods do not follow standard naming patterns, create a new BeanInfo class that extends the SimpleBeanInfo class and implement the getPropertyDescriptors method. To create a JDBC connection, use DriverManager.getConnection(url), or DriverManager.getConnection(url, username, password). The syntax for a JDBC URL is jdbc:<subprotocol>:<datasource>.

4. To create an instance of Statement, use connection.createStatement(). To execute a statement, use the methods executeQuery(…) and executeUpdate(…). executeQuery(…) returns a result set, but executeUpdate(…) does not return a result set.

5. To retrieve values in a ResultSet, use the getXxx(number) or getXxx(columnName) method.

6. The four types of JDBC drivers are Type 1: JDBC-ODBC Bridge, Type 2: Native-API, Type 3: Middle-tier, and Type 4: Native-protocol. Type 2 drivers require the DBMS client library? Type 3 and 4 drivers can be downloaded and maintained on the server side.