C# Programming: From Problem Analysis to Program Design, 4th edition

ISBN 978-1-285-09626-1

Chapter 14

1. d. dataset

2. e. Access

3. c. DataGridView

4. d tables

5. d. DataSet

6. a. namespace

7. b. SELECT \* FROM customer

8. a. @ followed by an identifier

9. b. INNER JOIN

10 a. data providers

11. d. DataReader

12. a. System.Data.OleDb

13. a. DataAdapter

14. a DataAdapter

15. d. Close( )

16. c. TableAdapter

17. c. TableAdapter Fill( ) method

18. a. Data Source Configuration

19. d. Data Sources

20. b. the name of the data source

21. The command class is used to hold the SQL query used to retrieve specific records. The data reader class allows read-only forward retrieval of data from the database. You can sequentially loop through the query results with the data reader object. Dataset used in conjunction with the data adapter class allows you to access the database using a disconnected state. With a dataset you also select records using an SQL statement. The SQL statement is again packaged in the command object. The difference between the data reader and data set object is the data set stores the entire table; data reader stores one row of the table at a time. Adapters are used to exchange data between a database source and a dataset object. You fill the dataset using the data adapter by specifying the name of the table to use as the data source. If you associate the data source with a data grid object, such as the DataGridView, the data can be displayed to the user in a grid like or tabular format. Users can make changes to the data, including inserting, deleting or updating the data on the data grid. Then you can flush the changes back up to the live database using the Update( ) method of the DataAdapter class

22. a. SELECT partNumber, partDescription FROM Parts

b. SELECT partDescription FROM Parts

WHERE (partNumber > 999 AND

partNumber < 2001)

c. INSERT INTO Customer (CNumber, LName, FName)

VALUES (23456, "Osprey", "Teola")

23. The DataGridView class is instantiated and all of the columns from the table are placed in this control structure on the form. You can delete or modify the grid like structure using the control's smart tag. In addition, four other objects are placed in the component tray below the form. These four objects are instances of the TableAdapter, BindingNavigator, BindingSource, and DataSet classes. A strip of buttons below the title bar are placed there when the BindingNavigator object is instantiated. The BindingSource control attaches or provides the connection to the data source and then each of the individual controls placed on the form are bound to the BindingSource control. The DataSet is the in-memory representation of the data from the data source. The TableAdapter fills the dataset from the data source (database) and then its update method is used to send the changes back to the database.

24. In addition to using the DataGridView, which is the default control, you can select Details prior to dragging a column to the form. This is available when you right-mouse click on the column in the Data Sources window. Again each control has a default under the Details view. This can be changed using the pull down menu on the control. One of the pull down options is Customize. Selecting this option displays a dialog box with a much larger list of controls.

25. The TableAdapter provides more functionality. It has a built in connection object and enables you to configure the SELECT, INSERT, UPDATE and DELETE commands manually in the Properties window of the DataSet Designer. Plus, multiple queries can be added to TableAdapter objects and treated like methods.