

Assignment

Deitel & Deitel Exercises 15.6, 15.9

HW08-1: (Deitel & Deitel Exercise 15.6)

15.6 (*File Browser*) Create a file browser (similar to Windows Explorer) based on the programs in Figs. 15.14, 15.28 and 15.31. The file browser should have a `TreeView`, which allows the user to browse directories. There should also be a `Listview`, which displays the contents (all subdirectories and files) of the directory being browsed. Double clicking a file in the `Listview` should open it, and double clicking a directory in either the `Listview` or the `TreeView` should browse it. If a file or directory cannot be accessed because of its permission settings, notify the user.

HW08-2: (Deitel & Deitel Exercise 15.9)

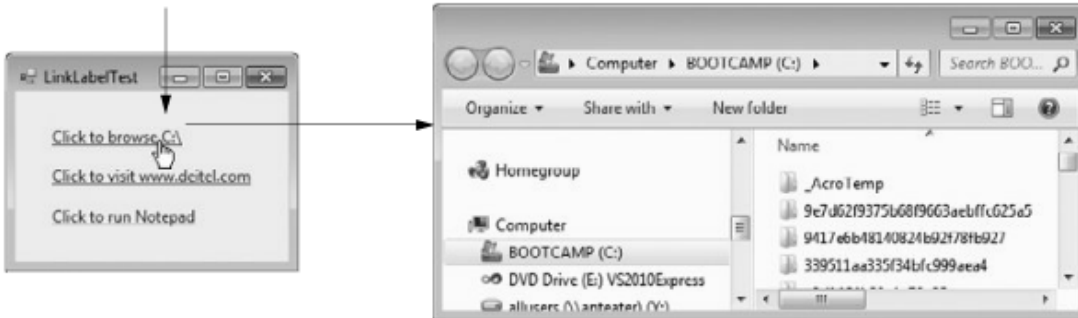
15.9 (*Restaurant Bill Calculator*) A restaurant wants an app that calculates a table's bill. The app should display all the menu items from Fig. 15.53 in four `ComboBoxes`. Each `ComboBox` should contain a category of food offered by the restaurant (Beverage, Appetizer, Main Course and Dessert). The user can choose from one of these `ComboBoxes` to add an item to a table's bill. As each item is selected in the `ComboBoxes`, add the price of that item to the bill. The user can click the **Clear Bill** Button to restore the **Subtotal**, **Tax**, and **Total** fields to \$0.00.

Name	Category	Price	Name	Category	Price
Soda	Beverage	\$1.95	Chicken Alfredo	Main Course	\$13.95
Tea	Beverage	\$1.50	Chicken Picatta	Main Course	\$13.95
Coffee	Beverage	\$1.25	Turkey Club	Main Course	\$11.95
Mineral Water	Beverage	\$2.95	Lobster Pie	Main Course	\$19.95
Juice	Beverage	\$2.50	Prime Rib	Main Course	\$20.95
Milk	Beverage	\$1.50	Shrimp Scampi	Main Course	\$18.95
Buffalo Wings	Appetizer	\$5.95	Turkey Dinner	Main Course	\$13.95
Buffalo Fingers	Appetizer	\$6.95	Stuffed Chicken	Main Course	\$14.95
Potato Skins	Appetizer	\$8.95	Apple Pie	Dessert	\$5.95
Nachos	Appetizer	\$8.95	Sundae	Dessert	\$3.95
Mushroom Caps	Appetizer	\$10.95	Carrot Cake	Dessert	\$5.95
Shrimp Cocktail	Appetizer	\$12.95	Mud Pie	Dessert	\$4.95
Chips and Salsa	Appetizer	\$6.95	Apple Crisp	Dessert	\$5.95
Seafood Alfredo	Main Course	\$15.95			

Fig. 15.53 | Food items and prices. (Part 2 of 2.)

```
1 // Fig. 15.14: LinkLabelTestForm.cs
2 // Using LinkLabels to create hyperlinks.
3 using System;
4 using System.Windows.Forms;
5
6 namespace LinkLabelTest
7 {
8     // Form using LinkLabels to browse the C:\ drive,
9     // load a web page and run Notepad
10    public partial class LinkLabelTestForm : Form
11    {
12        // constructor
13        public LinkLabelTestForm()
14        {
15            InitializeComponent();
16        } // end constructor
17
18        // browse C:\ drive
19        private void cDriveLinkLabel_LinkClicked( object sender,
20            LinkLabelLinkClickedEventArgs e )
21        {
22            // change LinkColor after it has been clicked
23            driveLinkLabel.LinkVisited = true;
24
25            System.Diagnostics.Process.Start( @"C:\" );
26        } // end method cDriveLinkLabel_LinkClicked
27
28        // load www.deitel.com in web browser
29        private void deitelLinkLabel_LinkClicked( object sender,
30            LinkLabelLinkClickedEventArgs e )
31        {
32            // change LinkColor after it has been clicked
33            deitelLinkLabel.LinkVisited = true;
34
35            System.Diagnostics.Process.Start( "http://www.deitel.com" );
36        } // end method deitelLinkLabel_LinkClicked
37
38        // run app Notepad
39        private void notepadLinkLabel_LinkClicked( object sender,
40            LinkLabelLinkClickedEventArgs e )
41        {
42            // change LinkColor after it has been clicked
43            notepadLinkLabel.LinkVisited = true;
44
45            // program called as if in run
46            // menu and full path not needed
47            System.Diagnostics.Process.Start( "notepad" );
48        } // end method notepadLinkLabel_LinkClicked
49    } // end class LinkLabelTestForm
50 } // end namespace LinkLabelTest
```

Click first `LinkLabel1` to look at contents of C: drive



Click second `LinkLabel1` to go to Deitel website



Click on third `LinkLabel1` to open Notepad

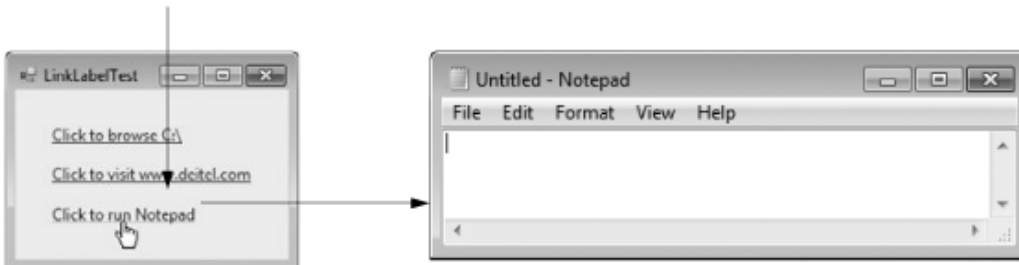


Fig. 15.14 | `LinkLabel1`s used to link to a drive, a web page and an app. (Part 3 of 3.)

```
1 // Fig. 15.28: TreeViewDirectoryStructureForm.cs
2 // Using TreeView to display directory structure.
3 using System;
4 using System.Windows.Forms;
5 using System.IO;
6
7 namespace TreeViewDirectoryStructure
8 {
9     // Form uses TreeView to display directory structure
10    public partial class TreeViewDirectoryStructureForm : Form
11    {
12        string substringDirectory; // store last part of full path name
13
14        // constructor
15        public TreeViewDirectoryStructureForm()
16        {
17            InitializeComponent();
18        } // end constructor
19
20        // populate current node with subdirectories
21        public void PopulateTreeView(
22            string directoryValue, TreeNode parentNode )
23        {
24            // array stores all subdirectories in the directory
25            string[] directoryArray =
26                Directory.GetDirectories( directoryValue );
27
28            // populate current node with subdirectories
29            try
30            {
31                // check to see if any subdirectories are present
32                if ( directoryArray.Length != 0 )
33                {
34                    // for every subdirectory, create new TreeNode,
35                    // add as a child of current node and recursively
36                    // populate child nodes with subdirectories
37                    foreach ( string directory in directoryArray )
38                    {
39                        // obtain last part of path name from the full path
40                        // name by calling the GetFileNameWithoutExtension
41                        // method of class Path
42                        substringDirectory =
43                            Path.GetFileNameWithoutExtension( directory );
44
45                        // create TreeNode for current directory
46                        TreeNode myNode = new TreeNode( substringDirectory );
```

```
47
48         // add current directory node to parent node
49         parentNode.Nodes.Add( myNode );
50
51         // recursively populate every subdirectory
52         PopulateTreeView( directory, myNode );
53     } // end foreach
54 } // end if
55 } //end try
56
57 // catch exception
58 catch ( UnauthorizedAccessException )
59 {
60     parentNode.Nodes.Add( "Access denied" );
61 } // end catch
62 } // end method PopulateTreeView
63
64 // handles enterButton click event
65 private void enterButton_Click( object sender, EventArgs e )
66 {
67     // clear all nodes
68     directoryTreeView.Nodes.Clear();
69
70     // check if the directory entered by user exists
71     // if it does, then fill in the TreeView,
72     // if not, display error MessageBox
73     if ( Directory.Exists( inputTextBox.Text ) )
74     {
75         // add full path name to directoryTreeView
76         directoryTreeView.Nodes.Add( inputTextBox.Text );
77
78         // insert subfolders
79         PopulateTreeView(
80             inputTextBox.Text, directoryTreeView.Nodes[ 0 ] );
81     }
82     // display error MessageBox if directory not found
83     else
84         MessageBox.Show( inputTextBox.Text + " could not be found."
85             "Directory Not Found", MessageBoxButtons.OK,
86             MessageBoxIcon.Error );
87 } // end method enterButton_Click
88 } // end class TreeViewDirectoryStructureForm
89 } // end namespace TreeViewDirectoryStructure
```

ig. 15.28 | TreeView used to display directories. (Part 2 of 3.)

a) GUI after user enters a directory path



b) GUI after the user presses **Enter** to display the directory's contents

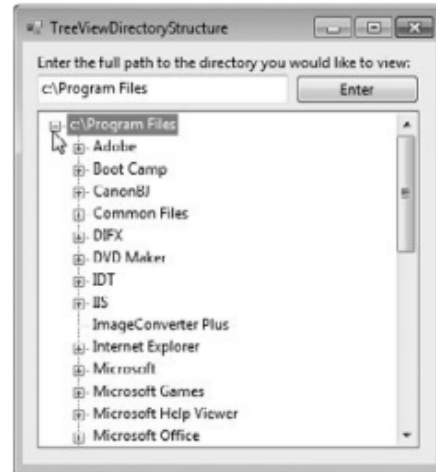


Fig. 15.28 | TreeView used to display directories. (Part 3 of 3.)

```

1 // Fig. 15.31: ListViewTestForm.cs
2 // Displaying directories and their contents in ListView.
3 using System;
4 using System.Windows.Forms;
5 using System.IO;
6
7 namespace ListViewTest
8 {
9     // Form contains a ListView which displays
10    // folders and files in a directory
11    public partial class ListViewTestForm : Form
12    {
13        // store current directory
14        string currentDirectory = Directory.GetCurrentDirectory();
15
16        // constructor
17        public ListViewTestForm()
18        {
19            InitializeComponent();
20        } // end constructor

```

```
21
22 // browse directory user clicked or go up one level
23 private void browserListView_Click( object sender, EventArgs e )
24 {
25     // ensure an item is selected
26     if ( browserListView.SelectedItems.Count != 0 )
27     {
28         // if first item selected, go up one level
29         if ( browserListView.Items[ 0 ].Selected )
30         {
31             // create DirectoryInfo object for directory
32             DirectoryInfo directoryObject =
33                 new DirectoryInfo( currentDirectory );
34
35             // if directory has parent, load it
36             if ( directoryObject.Parent != null )
37             {
38                 LoadFilesInDirectory(
39                     directoryObject.Parent.FullName );
40             } // end if
41         } // end if
42
43         // selected directory or file
44         else
45         {
46             // directory or file chosen
47             string chosen = browserListView.SelectedItems[ 0 ].Text;
48
49             // if item selected is directory, load selected directory
50             if ( Directory.Exists(
51                 Path.Combine( currentDirectory, chosen ) ) )
52             {
53                 LoadFilesInDirectory(
54                     Path.Combine( currentDirectory, chosen ) );
55             } // end if
56         } // end else
57
58         // update displayLabel
59         displayLabel.Text = currentDirectory;
60     } // end if
61 } // end method browserListView_Click
62
```

```
63 // display files/subdirectories of current directory
64 public void LoadFilesInDirectory( string currentDirectoryValue )
65 {
66     // load directory information and display
67     try
68     {
69         // clear ListView and set first item
70         browserListView.Items.Clear();
71         browserListView.Items.Add( "Go Up One Level" );
72
73         // update current directory
74         currentDirectory = currentDirectoryValue;
75         DirectoryInfo newCurrentDirectory =
76             new DirectoryInfo( currentDirectory );
77
78         // put files and directories into arrays
79         DirectoryInfo[] directoryArray =
80             newCurrentDirectory.GetDirectories();
81         FileInfo[] fileArray = newCurrentDirectory.GetFiles();
82
83         // add directory names to ListView
84         foreach ( DirectoryInfo dir in directoryArray )
85         {
86             // add directory to ListView
87             ListViewItem newDirectoryItem =
88                 browserListView.Items.Add( dir.Name );
89
90             newDirectoryItem.ImageIndex = 0; // set directory image
91         } // end foreach
92
93         // add file names to ListView
94         foreach ( FileInfo file in fileArray )
95         {
96             // add file to ListView
97             ListViewItem newFileItem =
98                 browserListView.Items.Add( file.Name );
99
100             newFileItem.ImageIndex = 1; // set file image
101         } // end foreach
102     } // end try
103
104     // access denied
105     catch ( UnauthorizedAccessException )
106     {
107         MessageBox.Show( "Warning: Some files may not be " +
108             "visible due to permission settings",
109             "Attention", 0, MessageBoxIcon.Warning );
110     } // end catch
111 } // end method LoadFilesInDirectory
112
```

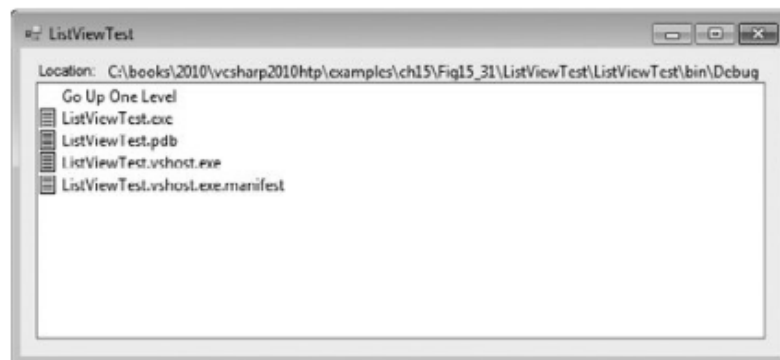


```

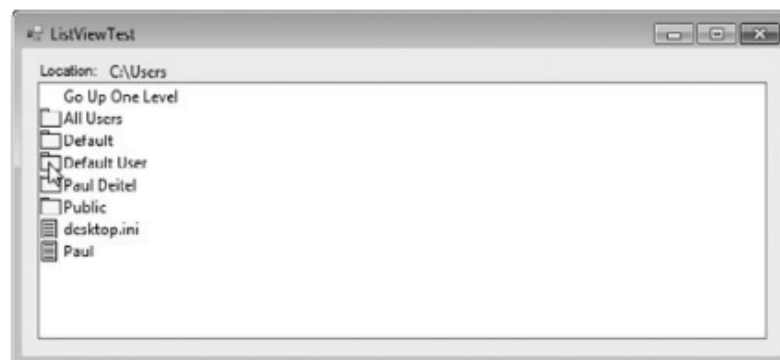
113 // handle load event when Form displayed for first time
114 private void ListViewTestForm_Load( object sender, EventArgs e )
115 {
116     // add icon images to ImageList
117     fileFolderImageList.Images.Add( Properties.Resources.folder );
118     fileFolderImageList.Images.Add( Properties.Resources.file );
119
120     // load current directory into browserListView
121     LoadFilesInDirectory( currentDirectory );
122     displayLabel.Text = currentDirectory;
123 } // end method ListViewTestForm_Load
124 } // end class ListViewTestForm
125 } // end namespace ListViewTest

```

a) GUI showing app's default folder



b) GUI showing the contents of the c:\Users directory



c) Dialog that appears if you try to access a directory for which you do not have permission

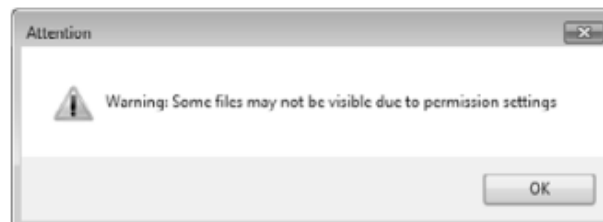


Fig. 15.31 | ListView displaying files and folders. (Part 4 of 4.)

HW08 Problem Set

CS-3020

Grading Rubric

Each problem is worth 10 pts (score will be recorded as a percentage of that amount)

10% Properly submitted

10% Properly named

20% Adequate comments

10% Runs

20% Produces correct output

30% Effort evidenced by the submitted work