Working with Menus

Menus – provide groups of

- Menu items (commands, grouped by a common theme):
  - *access shortcuts* or *hot keys* (Alt+key)
  - *checkmarks*
  - Submenus (menus within a menu)
- Context menus (commonly used commands, pop up in response to <R>)

Control MainMenu (MenuStrip)

Displays a menu at run time. All submenus of the main menu and individual items are MenuItem (ToolStripMenuItem, ToolStripComboBox, ToolStripSeparator, and ToolStripTextBox) objects. To bind the menu bar to the Form that will display it, assign the MainMenu (MenuStrip) to the Menu (MainMenuStrip) property of the Form. The method Menu.CloneMenu creates a copy of the menu structure and is used to modifications for the menu structure.

Properties

MenuItems	Gets the collection of MenuItem objects associated with the menu
(Items	Gets all the items that belong to a ToolStrip)

Control ContextMenu (ContextMenuStrip)

Represents shortcut menus that can be displayed when the user clicks the right mouse button over a control or area of the form. To bind this control to the control that displays the shortcut menu, assign ContextMenu (ContextMenuStrip) to the ContextMenu (ContextMenuStrip) property of the control.

Properties

SourceControl	Gets the control that is displaying the shortcut menu

Class MenuItem (ToolStripMenuItem)

Represents an individual item that is displayed within a menu.

Properties

(Name)	Gets/sets the name of the menu item
Checked	Gets/sets true/false indicating whether a check mark appears next to the text of the menu item
DefaultItem	Gets/sets true/false indicating whether the menu item is the default menu item
Enabled	Gets/sets true/false indicating whether the menu item is enabled

Index	Gets/sets the position of the menu item in its parent menu
MenuItems (DropDownItems)	Gets the collection of MenuItem objects associated with the menu
RadioCheck	Gets/sets true/false indicating whether the menu item, if checked (Checked=true), displays a radio-button instead of a check mark
Shortcut (ShortcutKeys)	Gets/sets the shortcut key associated with the menu item
Text	Gets/sets the caption of the menu item; & before the character is used as the access key
Visible	Gets/sets true/false indicating whether the menu item is visible

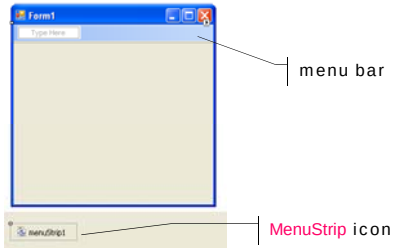
Events

Click	Occurs when the menu item is clicked or selected using a shortcut key or access key defined for the menu item (default)
Popup	Occurs before a menu item's list of menu items is displayed
Select	Occurs when the user places the pointer over a menu item

Creating a Menu using the Designer:

1. Create a menu bar

   Toolbox ⇒ <L²> MenuStrip

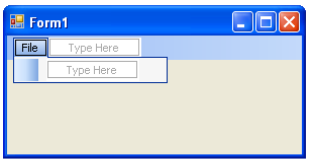


2. Set the menuStrip1 properties

   (Name) MenuStrip_class_name

3. Add menu items to the menu bar

   <L> Type Here ⇒ text for menu_name



4. Set ToolStripMenuItem properties

   (Name) ToolStripMenuItem_class_name
   Text menu_item_name
   (add ... after menu_item_name for a dialog window)

5. Add submenus to a menu

   <L> menu_item_name
   <L> Type Here ⇒ text for submenu_name

6. Set submenu properties

   (Name) ToolStripMenuItem_class_submenu_name
   Text submenu_name

7. Access key

   & before the character
   &File File (Alt+F)
   (Alt is a shortcut key by default.)

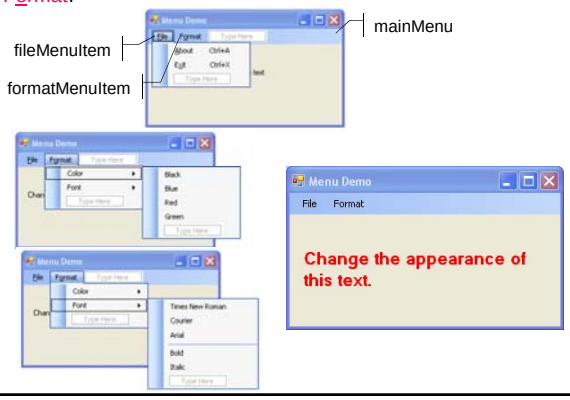
8. Insert a separator

   <R> menu item ⇒ Insert Separator
   or – for the text of the menu-item

9. Delete a menu item

   <L> menu item ⇒ <Delete>

Example: Menu bar with two menus: File and Format.




```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication10
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
```

```
    private void aboutMenuItem_Click(object sender, EventArgs e)
    {
        MessageBox.Show("Example of using menus", "About",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
    }

    private void exitMenuItem_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }

    private void ClearColor()
    {
        blackMenuItem.Checked = false;
        blueMenuItem.Checked  = false;
        redMenuItem.Checked   = false;
        greenMenuItem.Checked = false;
    }
```

```
    private void blackMenuItem_Click(object sender, EventArgs e)
    {
        ClearColor();
        displayLabel.ForeColor = Color.Black;
        blackMenuItem.Checked = true;
    }

    private void blueMenuItem_Click(object sender, EventArgs e)
    {
        ClearColor();
        displayLabel.ForeColor = Color.Blue;
        blueMenuItem.Checked = true;
    }

    private void redMenuItem_Click(object sender, EventArgs e)
    {
        ClearColor();
        displayLabel.ForeColor = Color.Red;
        redMenuItem.Checked = true;
    }
```

```
    private void greenMenuItem_Click(object sender, EventArgs e)
    {
        ClearColor();
        displayLabel.ForeColor = Color.Green;
        greenMenuItem.Checked = true;
    }

    private void ClearFont()
    {
        timesMenuItem.Checked   = false;
        courierMenuItem.Checked = false;
        arialMenuItem.Checked   = false;
    }

    private void timesMenuItem_Click(object sender, EventArgs e)
    {
        ClearFont();
        displayLabel.Font = new Font("Times New Roman", 14,
                                     displayLabel.Font.Style);
        timesMenuItem.Checked = true;
    }
```

```
    private void courierMenuItem_Click(object sender, EventArgs e)
    {
        ClearFont();
        displayLabel.Font = new Font("Courier", 14, displayLabel.Font.Style);
        courierMenuItem.Checked = true;
    }

    private void arialMenuItem_Click(object sender, EventArgs e)
    {
        ClearFont();
        displayLabel.Font = new Font("Arial", 14, displayLabel.Font.Style);
        arialMenuItem.Checked = true;
    }

    private void boldMenuItem_Click(object sender, EventArgs e)
    {
        ClearFont();
        displayLabel.Font = new Font(displayLabel.Font.FontFamily, 14,
                                     displayLabel.Font.Style ^ FontStyle.Bold);
        boldMenuItem.Checked = true;
    }
```

```
    private void italicMenuItem_Click(object sender, EventArgs e)
    {
        ClearFont();
        displayLabel.Font = new Font(displayLabel.Font.FontFamily, 14,
                                     displayLabel.Font.Style ^ FontStyle.Italic);
        italicMenuItem.Checked = true;
    }
  }
}
```

```
namespace WindowsApplication10
{
  partial class Form1
  {  private System.ComponentModel.IContainer components = null;
    protected override void Dispose(bool disposing) { … }
    #region Windows Form Designer generated code
    private void InitializeComponent()
    {
      this.mainMenu.Items.AddRange(new
          System.Windows.Forms.ToolStripItem[] {
          this.fileMenuItem, this.formatMenuItem});
      this.fileMenuItem.DropDownItems.AddRange(new
          System.Windows.Forms.ToolStripItem[] {
          this.aboutMenuItem, this.exitMenuItem});
      this.fileMenuItem.Text = "&File";
      this.aboutMenuItem.Text = "&About";
      this.aboutMenuItem.ShortcutKeys = ((System.Windows.Forms.Keys)
        ((System.Windows.Forms.Keys.Control |
          System.Windows.Forms.Keys.A)));
      this.aboutMenuItem.Click +=
              new System.EventHandler(this.aboutMenuItem_Click);
```

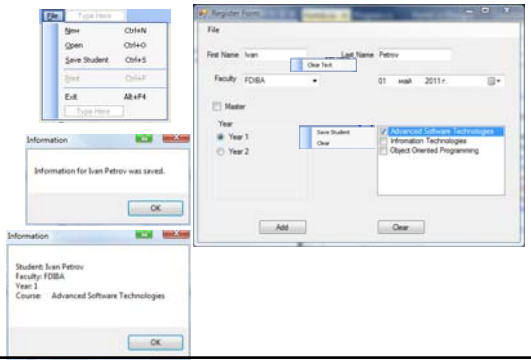
```
      this.exitMenuItem.Text = "E&xit";
      this.exitMenuItem.ShortcutKeys = ((System.Windows.Forms.Keys)
        ((System.Windows.Forms.Keys.Control |
          System.Windows.Forms.Keys.X)));
      this.exitMenuItem.Click +=
              new System.EventHandler(this.exitMenuItem_Click);
      this.formaMenuItem.DropDownItems.AddRange(new
        System.Windows.Forms.ToolStripItem[] {
        this.colorMenuItem, this.fontMenuItem});
      this.formaMenuItem.Text = "F&ormat";
      this.colorMenuItem.DropDownItems.AddRange(new
        System.Windows.Forms.ToolStripItem[] {this.blackMenuItem,
        this.blueMenuItem, this.redMenuItem, this.greenMenuItem});
      this.blackMenuItem.Text = "Black";
      this.blackMenuItem.Click +=
              new System.EventHandler(this.blackMenuItem_Click);
      this.blueMenuItem.Text = "Blue";
      this.blueMenuItem.Click +=
              new System.EventHandler(this.blueMenuItem_Click);
```

```
        this.redMenuItem.Text = "Red";
        this.redMenuItem.Click +=
                new System.EventHandler(this.redMenuItem_Click);

        this.greenMenuItem.Text = "Green";
        this.greenMenuItem.Click +=
                new System.EventHandler(this.greenMenuItem_Click);

        this.fontMenuItem.DropDownItems.AddRange(new
          System.Windows.Forms.ToolStripItem[] {this.timesMenuItem,
          this.courierMenuItem, this.arialMenuItem, this.separatorMenuItem1,
          this.boldMenuItem, this.italicMenuItem});
        this.fontMenuItem.Text = "Font";

        this.timesMenuItem.Text = "Times New Roman";
        this.timesMenuItem.Click +=
                new System.EventHandler(this.timesMenuItem_Click);

        this.courierMenuItem.Text = "Courier";
        this.courierMenuItem.Click +=
                new System.EventHandler(this.courierMenuItem_Click);
```

```
        this.arialMenuItem.Text = "Arial";
        this.arialMenuItem.Click +=
                new System.EventHandler(this.arialMenuItem_Click);

        this.separatorMenuItem1.Name = "separatorMenuItem1";

        this.boldMenuItem.Text = "Bold";
        this.boldMenuItem.Click +=
                new System.EventHandler(this.boldMenuItem_Click);

        this.italicMenuItem.Text = "Italic";
        this.italicMenuItem.Click +=
                new System.EventHandler(this.italicMenuItem_Click);
        this.displayLabel.Text = "Change the appearance of this text.";
        this.Controls.Add(this.displayLabel);
        this.Controls.Add(this.mainMenu);
        this.MainMenuStrip = this.mainMenu;
        this.Name = "Form1";
        this.Text = "Menu Demo";
    }
    #endregion
```

```
        private System.Windows.Forms.MenuStrip mainMenu;
        private System.Windows.Forms.ToolStripMenuItem fileMenuItem;
        private System.Windows.Forms.ToolStripMenuItem aboutMenuItem;
        private System.Windows.Forms.ToolStripMenuItem exitMenuItem;
        private System.Windows.Forms.ToolStripMenuItem formaMenuItem;
        private System.Windows.Forms.ToolStripMenuItem colorMenuItem;
        private System.Windows.Forms.ToolStripMenuItem blackMenuItem;
        private System.Windows.Forms.ToolStripMenuItem blueMenuItem;
        private System.Windows.Forms.ToolStripMenuItem redMenuItem;
        private System.Windows.Forms.ToolStripMenuItem greenMenuItem;
        private System.Windows.Forms.ToolStripMenuItem fontMenuItem;
        private System.Windows.Forms.ToolStripMenuItem timesMenuItem;
        private System.Windows.Forms.ToolStripMenuItem courierMenuItem;
        private System.Windows.Forms.ToolStripMenuItem arialMenuItem;
        private System.Windows.Forms.ToolStripSeparator separatorMenuItem1;
        private System.Windows.Forms.ToolStripMenuItem boldMenuItem;
        private System.Windows.Forms.ToolStripMenuItem italicMenuItem;
        private System.Windows.Forms.Label displayLabel;
    }
}
```

Example: Add Register Form with a menu File and context menus for the text controls (Clear Text) and for the form (Save Student, Clear).



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication11
{
    public partial class Form1 : Form
    {
        private System.Windows.Forms.ContextMenuStrip textBoxMenu;
        private System.Windows.Forms.ToolStripMenuItem textBoxClearItem;
        private System.Windows.Forms.ContextMenuStrip formMenu;
        private System.Windows.Forms.ToolStripMenuItem saveContextItem;
        private System.Windows.Forms.ToolStripMenuItem clearContextItem;
```

```
    public Form1()
    {
        InitializeComponent();
        Reset();
        // Creates a context menu with one item "Clear Text"
        this.textBoxMenu = new System.Windows.Forms.ContextMenuStrip();
        this.textBoxClearItem =
                new System.Windows.Forms.ToolStripMenuItem();
        this.textBoxMenu.Items.AddRange(new
            System.Windows.Forms.ToolStripMenuItem[] {
            this.textBoxClearItem});
        this.textBoxClearItem.Text = "Clear Text";
        // Bind the context menu to the TextBox controls
        this.firstName.ContextMenuStrip = this.textBoxMenu;
        this.lastName.ContextMenuStrip = this.textBoxMenu;
        // Add the Click event handler at run time
        this.textBoxClearItem.Click +=
            new System.EventHandler(this.textBoxClearClick);
```

```
        // Creates a context menu with two items "Save Student" and "Clear"
        formMenu = new System.Windows.Forms.ContextMenuStrip();
        this.saveContextItem =
                    new System.Windows.Forms.ToolStripMenuItem();
        this.clearContextItem =
                    new System.Windows.Forms.ToolStripMenuItem();
        this.formMenu.Items.AddRange(new
           System.Windows.Forms.ToolStripMenuItem[] {
           this.saveContextItem, this.clearContextItem});
        this.saveContextItem.Text = "Save Student";
        this.clearContextItem.Text = "Clear";
        // Bind the context menu to the Form control
        this.ContextMenuStrip = formMenu;
        // Add the Click event handler at run time
        this.saveContextItem.Click += new EventHandler(this.saveItem_Click);
        this.clearContextItem.Click +=
                                    new EventHandler(this.clearButton_Click);
}
    public void Reset() { … }
    public void CheckedBoxReset() { … }
    private void type_CheckedChanged(object sender, EventArgs e) { … }
```

```
    private void addButton_Click(object sender, EventArgs e) { … }

    private void clearButton_Click(object sender, EventArgs e) { … }

    private void memeberFormClosing(object sender,
                                        FormClosingEventArgs e) { … }
    private void newItem_Click(object sender, EventArgs e)
    {
       Reset();
       printItem.Enabled = true;
    }
    private void saveItem_Click(object sender, EventArgs e)
    {
       string details = "Information for " + firstName.Text + " " +
                        lastName.Text + " was saved.";
       MessageBox.Show(details, "Information");
    }
    private void exitItem_Click(object sender, EventArgs e)
    {
       Close();
    }
```

```
    private void textBoxClearClick(object sender, System.EventArgs e)
    {
       if (textBoxMenu.SourceControl.Equals(firstName))
          firstName.Clear();
       else
          lastName.Clear();
    }
  }
}
```

```
    private void InitializeComponent()
    {   …
       this.menuStrip1.Items.AddRange(new
                System.Windows.Forms.ToolStripItem[] { this.fileItem});

       this.fileItem.DropDownItems.AddRange(new
                System.Windows.Forms.ToolStripItem[] {this.newItem,
                this.openItem, this.saveItem, this.toolStripMenuItem1,
                this.printItem, this.toolStripMenuItem2, this.exitItem});
       this.fileItem.ShortcutKeys = ((System.Windows.Forms.Keys)
                ((System.Windows.Forms.Keys.Control |
                  System.Windows.Forms.Keys.O)));
       this.fileItem.Text = "&File";

       this.newItem.ShortcutKeys = ((System.Windows.Forms.Keys)
                ((System.Windows.Forms.Keys.Control |
                  System.Windows.Forms.Keys.N)));
       this.newItem.Text = "&New";
       this.newItem.Click += new System.EventHandler(this.newItem_Click);
```

```
        this.openItem.ShortcutKeys = ((System.Windows.Forms.Keys)
                ((System.Windows.Forms.Keys.Control |
                  System.Windows.Forms.Keys.O)));
        this.openItem.Text = "&Open";

        this.saveItem.ShortcutKeys = ((System.Windows.Forms.Keys)
                ((System.Windows.Forms.Keys.Control |
                  System.Windows.Forms.Keys.S)));
        this.saveItem.Text = "&Save Student";
        this.saveItem.Click +=
                            new System.EventHandler(this.saveItem_Click);

        this.toolStripMenuItem1.Name = "toolStripMenuItem1";

        this.printItem.Enabled = false;
        this.printItem.ShortcutKeys = ((System.Windows.Forms.Keys)
                ((System.Windows.Forms.Keys.Control |
                  System.Windows.Forms.Keys.P)));
        this.printItem.Text = "&Print";

        this.toolStripMenuItem2.Name = "toolStripMenuItem2";
```

```
        this.exitItem.ShortcutKeys = ((System.Windows.Forms.Keys)
                ((System.Windows.Forms.Keys.Alt |
                  System.Windows.Forms.Keys.F4)));
        this.exitItem.Text = "Exit";
        this.exitItem.Click += new System.EventHandler(this.exitItem_Click);

        this.Controls.Add(this.menuStrip1);
        this.MainMenuStrip = this.menuStrip1;
        this.Name = "Form1";
    }
    #endregion
    private System.Windows.Forms.MenuStrip menuStrip1;
    private System.Windows.Forms.ToolStripMenuItem fileItem;
    private System.Windows.Forms.ToolStripMenuItem newItem;
    private System.Windows.Forms.ToolStripMenuItem openItem;
    private System.Windows.Forms.ToolStripMenuItem saveItem;
    private System.Windows.Forms.ToolStripSeparator toolStripMenuItem1;
    private System.Windows.Forms.ToolStripMenuItem printItem;
    private System.Windows.Forms.ToolStripSeparator toolStripMenuItem2;
    private System.Windows.Forms.ToolStripMenuItem exitItem;
  }
}
```

Controls OpenFileDialog and SaveFileDialog

Display a dialog box for file opening/saving.

Properties

AddExtension	Gets/sets true/false indicating whether the dialog box automatically adds an extension to a file name if the user omits the extension
DefaultExt	Gets/sets the default file name extension
FileName	Gets/sets a string containing the file name selected in the file dialog box
InitialDirectory	Gets/sets the initial directory displayed by the file dialog box
Title	Gets/sets the file dialog box title

ValidateNames	Gets/sets true/false indicating whether the dialog box accepts only valid Win32 file names
OverwritePrompt (SaveFileDialog)	Gets/sets true/false indicating whether the Save As dialog box displays a warning if the user specifies a file name that already exists

Methods

OpenFile	Opens the file selected by the user specified by the FileName property, with read-only permission (OpenFileDialog) / with read/write permission (SaveFileDialog) and returns the Stream
ShowDialog	Runs a dialog box and returns DialogResult.OK if the user clicks OK in the dialog box; otherwise, DialogRezult.Cancel

Control RichTextBox

Displays a control for displaying, entering, and manipulating text with formatting. The text is entering directly or is loading form a file including plain text, Unicode plain text, and Rich Text Format (RTF).

Methods

public int Find (string str)

Searches the text for a string str. Returns the location within the control where the search text was found or -1 if the search string is not found or an empty search string is specified in the str parameter.

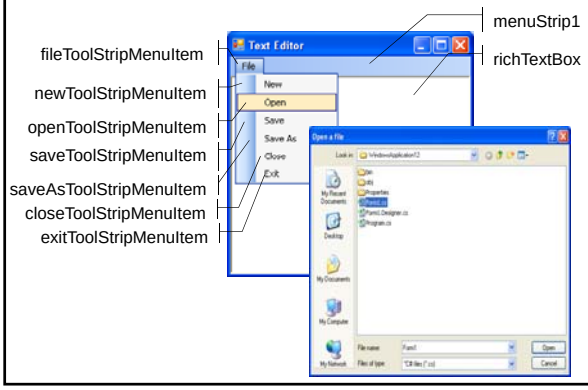
public void LoadFile (Stream data,
RichTextBoxStreamType fileType);

Loads the contents of an existing data stream from fileType (PlainText, RichText, UnicodePlainText) in the control.

public void SaveFile (string path,
RichTextBoxStreamType fileType);

Saves the contents of the control to a file with a name path and specific fileType.

Example: Text editor – menu File with submenus for file manipulation.



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication12
{
   public partial class Form1 : Form
   {
      private bool fileNew;
      public Form1()
      {
         InitializeComponent();
         fileNew = true;
      }
```

```
      private void newToolStripMenuItem_Click(object sender, EventArgs e)
      {
         richTextBox.Text = String.Empty;
         fileNew = true;
      }

      private void openToolStripMenuItem_Click(object sender, EventArgs e)
      {
         if (openFileDialog.ShowDialog() == DialogResult.OK)
         {
            richTextBox.LoadFile(openFileDialog.FileName,
                                 RichTextBoxStreamType.PlainText);
            fileNew = false;
         }
      }
```

```
      private void saveToolStripMenuItem_Click(object sender, EventArgs e)
      {
         if (!fileNew)
         {
            saveFileDialog.FileName = openFileDialog.FileName;
            richTextBox.SaveFile(saveFileDialog.FileName,
                                 RichTextBoxStreamType.PlainText);
         }
         else
         {
            if ((saveFileDialog.ShowDialog() == DialogResult.OK))
               richTextBox.SaveFile(saveFileDialog.FileName,
                                    RichTextBoxStreamType.PlainText);
         }
         fileNew = false;
      }
```

```
      private void saveAsToolStripMenuItem_Click(object sender,
                                                 EventArgs e)
      {
         if ((saveFileDialog.ShowDialog() == DialogResult.OK))
         {
            richTextBox.SaveFile(saveFileDialog.FileName,
                                 RichTextBoxStreamType.PlainText);
         }
      }

      private void closeToolStripMenuItem_Click(object sender, EventArgs e)
      {
         richTextBox.Text = String.Empty;
         fileNew = false;
      }

      private void exitToolStripMenuItem_Click(object sender, EventArgs e)
      {
         Application.Exit();
      }
   }
}
```

```
namespace WindowsApplication12
{
   partial class Form1
   {
      private System.ComponentModel.IContainer components = null;

      protected override void Dispose(bool disposing) { … }

      #region Windows Form Designer generated code

      private void InitializeComponent()
      {  …
         this.menuStrip1.Text = "mainMenuStrip";

         this.fileToolStripMenuItem.Text = "&File";

         this.newToolStripMenuItem.Text = "&New";
         this.newToolStripMenuItem.Click +=
            new System.EventHandler(this.newToolStripMenuItem_Click);
```

```
         this.openToolStripMenuItem.Text = "&Open";
         this.openToolStripMenuItem.Click +=
            new System.EventHandler(this.openToolStripMenuItem_Click);

         this.saveToolStripMenuItem.Text = "&Save";
         this.saveToolStripMenuItem.Click +=
            new System.EventHandler(this.saveToolStripMenuItem_Click);

         this.saveAsToolStripMenuItem.Text = "Save &As";
         this.saveAsToolStripMenuItem.Click +=
            new System.EventHandler(this.saveAsToolStripMenuItem_Click);

         this.closeToolStripMenuItem.Text = "&Close";
         this.closeToolStripMenuItem.Click +=
            new System.EventHandler(this.closeToolStripMenuItem_Click);

         this.exitToolStripMenuItem.Text = "E&xit";
         this.exitToolStripMenuItem.Click +=
            new System.EventHandler(this.exitToolStripMenuItem_Click);

         this.richTextBox.Dock = System.Windows.Forms.DockStyle.Fill;
         this.richTextBox.Text = "";
```

```
        this.openFileDialog.DefaultExt = "*.cs";
        this.openFileDialog.Filter = "\"C# files (*.cs)|*.cs|All files (*.*)|*.*\"";
        this.openFileDialog.Title = "Open a file";

        this.saveFileDialog.DefaultExt = "*.cs";
        this.saveFileDialog.Filter = "\"C# files (*.cs)|*.cs|All files (*.*)|*.*\"";
        this.saveFileDialog.Title = "Save a file";

        this.Controls.Add(this.richTextBox);
        this.Controls.Add(this.menuStrip1);
        this.MainMenuStrip = this.menuStrip1;
        this.Name = "Form1";
        this.Text = "Text Editor";
    }

    #endregion
```

```
    private System.Windows.Forms.MenuStrip menuStrip1;
    private System.Windows.Forms.ToolStripMenuItem
        fileToolStripMenuItem;
    private System.Windows.Forms.ToolStripMenuItem
        newToolStripMenuItem;
    private System.Windows.Forms.ToolStripMenuItem
        openToolStripMenuItem;
    private System.Windows.Forms.ToolStripMenuItem
        saveToolStripMenuItem;
    private System.Windows.Forms.ToolStripMenuItem
        saveAsToolStripMenuItem;
    private System.Windows.Forms.ToolStripMenuItem
        closeToolStripMenuItem;
    private System.Windows.Forms.ToolStripMenuItem
        exitToolStripMenuItem;
    private System.Windows.Forms.RichTextBox richTextBox;
    private System.Windows.Forms.OpenFileDialog openFileDialog;
    private System.Windows.Forms.SaveFileDialog saveFileDialog;
  }
}
```

Control LinkLabel

Displays Web-style links to Windows Forms applications (links to a file, folder, or Web page).

Properties

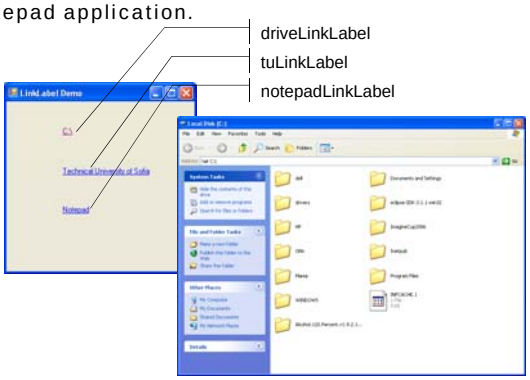
ActiveLinkColor	Gets/sets the color used to display an active link (red)
LinkColor	Gets/sets the color used when displaying a normal link (blue)
LinkVisited	Gets/sets true/false indicating whether a link should be displayed as though it were visited
VisitedLinkColor	Gets/sets the color used when displaying a link that that has been previously visited (purple)

Text	Gets/sets the text associated with the control
UseMnemonic	Gets/sets true/false indicating whether the control interprets an ampersand character (&) in the control's Text property to be an access key prefix character
Links	Gets the collection of LinkLabel.Link objects contained within the LinkLabel

Events

LinkClicked	Occurs when a link is clicked within the control (default)

Example: Uses three LinkLabels, to link to C:\ drive, the Technical University Web page and the Notepad application.



The class System.Diagnostics.Process provides access to local and remote processes and enables to start and stop local system processes.

public static Process Start (string fileName);
public static Process Start (string fileName, string arguments);

Starts a process resource by specifying the name of an application fileName and a set of command-line arguments, and associates the resource with a new Process component.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication13
{
   public partial class Form1 : Form
   {
      public Form1()
      {  InitializeComponent();  }

      private void driveLinkLabel_LinkClicked(object sender,
                                          LinkLabelLinkClickedEventArgs e)
      {
         driveLinkLabel.LinkVisited = true;
         System.Diagnostics.Process.Start("C:\\");
      }
```

```
      private void tuLinkLabel_LinkClicked(object sender,
                                          LinkLabelLinkClickedEventArgs e)
      {
         tuLinkLabel.LinkVisited = true;
         System.Diagnostics.Process.Start
                                          ("IExplore", "http://www.tu-sofia.bg");
      }

      private void notepadLinkLabel_LinkClicked(object sender,
                                          LinkLabelLinkClickedEventArgs e)
      {
         notepadLinkLabel.LinkVisited = true;
         System.Diagnostics.Process.Start("notepad.exe");
      }
   }
}
```

```
namespace WindowsApplication13
{
   partial class Form1
   {
      private System.ComponentModel.IContainer components = null;

      protected override void Dispose(bool disposing) {  …  }

      #region Windows Form Designer generated code

      private void InitializeComponent()
      {
         this.driveLinkLabel = new System.Windows.Forms.LinkLabel();
         this.tuLinkLabel = new System.Windows.Forms.LinkLabel();
         this.notepadLinkLabel = new System.Windows.Forms.LinkLabel();

         this.driveLinkLabel.Text = "C:\\";
         this.driveLinkLabel.LinkClicked += new
          System.Windows.Forms.LinkLabelLinkClickedEventHandler
          (this.driveLinkLabel_LinkClicked);
```

```
         this.tuLinkLabel.Text = "Technical University of Sofia";
         this.tuLinkLabel.LinkClicked += new
          System.Windows.Forms.LinkLabelLinkClickedEventHandler
          (this.tuLinkLabel_LinkClicked);

         this.notepadLinkLabel.Text = "Notepad";
         this.notepadLinkLabel.LinkClicked += new
          System.Windows.Forms.LinkLabelLinkClickedEventHandler
          (this.notepadLinkLabel_LinkClicked);

         this.Controls.Add(this.notepadLinkLabel);
         this.Controls.Add(this.tuLinkLabel);
         this.Controls.Add(this.driveLinkLabel);
         this.Name = "Form1";
         this.Text = "LinkLabel Demo";
         …
      }
      #endregion

      private System.Windows.Forms.LinkLabel driveLinkLabel;
      private System.Windows.Forms.LinkLabel tuLinkLabel;
      private System.Windows.Forms.LinkLabel notepadLinkLabel;
   }
}
```

Control **TreeView**

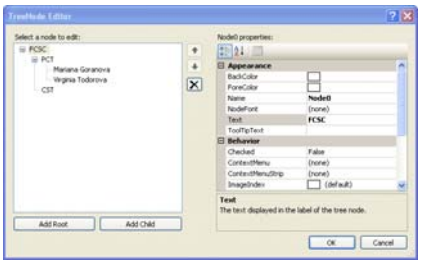
Displays nodes hierarchically on a tree.

A tree (class TreeView) is a collection of nodes, usually organized in a hierarchical manner.

A node (class TreeNode) is an object that contains values and can refer to other node.

Creating a Tree

1. TreeView ⇒ property Nodes ⇒ TreeNode Editor



2. Using a code

- Create and add a tree root

  <myTreeView>.Nodes.Add
  (new TreeNode(<root_label>);
- Select a parent node

  <myTreeView>.Nodes(parent_node_index);
- Add a child node

  <myTreeView>.Nodes(parent_node_index).
  Add(new TreeNode(<child_label>);

Properties of TreeView

CheckBoxes	Gets/sets true/false indicating whether check boxes are displayed next to the tree nodes in the tree view control
ImageList	Gets/sets the ImageList collection that contains the Image objects used by the tree nodes
Nodes	Gets the collection of tree nodes that are assigned to the tree view control. Methods: Add (add a node), Clear (clear the collection) and Remove (delete a node)

Events of the TreeView

AfterSelect	Occurs after the tree node is selected (default)

Properties of TreeNode

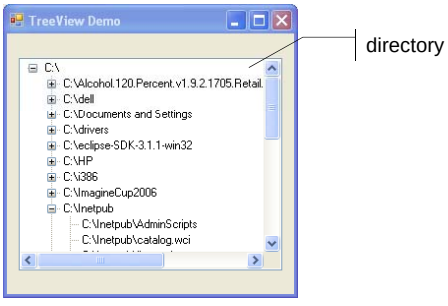
Checked	Gets/sets true/false indicating whether the node's check box is selected
FirstNode	Gets the first child tree node in the tree node collection Nodes
FullPath	Gets the path from the root tree node to the current tree node
ImageIndex	Gets/sets the image list index value of the image displayed when the tree node is in the unselected state
LastNode	Gets the last child tree node in the collection Nodes
NextNode	Gets the next sibling tree node

Nodes	Gets the collection of TreeNode objects assigned to the current tree node
PrevNode	Gets the previous sibling tree node
SelectedImageIndex	Gets or sets the image list index value of the image that is displayed when a tree node is selected
Text	Gets/sets the text of the TreeNode

Methods of TreeNode

Collapse	Collapses the TreeNode
Expand	Expands the tree node
ExpandAll	Expands all the child tree nodes
GetNodeCount	Returns the number of child tree nodes

Example: Displays the directory file structure on a computer.



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.IO;

namespace WindowsApplication15
{
   public partial class Form1 : Form
   {
      public Form1()
      {  InitializeComponent();  }

      private void Form1_Load(object sender, EventArgs e)
      {
         directory.Nodes.Add ("C:\\");
         PopulateTreeView ("C:\\", directory.Nodes[0]);
      }
```

```
      public void PopulateTreeView(string directoryValue,
                                   TreeNode parentNode)
      {  string[] directoryArray =
         Directory.GetDirectories(directoryValue);
         try
         {
            if (directoryArray.Length != 0)
            {
               foreach (string d in directoryArray)
               {  TreeNode myNode = new TreeNode(d);
                  parentNode.Nodes.Add(myNode);
                  PopulateTreeView(d, myNode);
               }
            }
         }
         catch (UnauthorizedAccessException e)
         {
            parentNode.Nodes.Add("Access denied." + e.Message);
         }
      }
   }
}
```

```
namespace WindowsApplication15
{
   partial class Form1
   {
      private System.ComponentModel.IContainer components = null;

      protected override void Dispose(bool disposing) {  …  }

      #region Windows Form Designer generated code

      private void InitializeComponent()
      {
         this.directory = new System.Windows.Forms.TreeView();
         this.SuspendLayout();

         // directory
         this.directory.Location = new System.Drawing.Point(12, 25);
         this.directory.Name = "directory";
         this.directory.Size = new System.Drawing.Size(268, 217);
         this.directory.TabIndex = 0;
```

```
         // Form1
         this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
         this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
         this.ClientSize = new System.Drawing.Size(292, 266);
         this.Controls.Add(this.directory);
         this.Name = "Form1";
         this.Text = "TreeView Demo";
         this.Load += new System.EventHandler(this.Form1_Load);
         this.ResumeLayout(false);
      }

      #endregion

      private System.Windows.Forms.TreeView directory;
   }
}
```

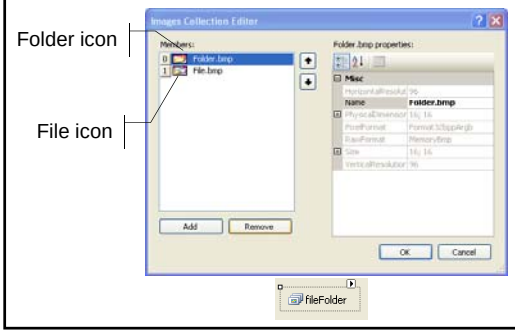
Control **ListView**

Displays a list from which the user can select one or more items (class ListViewItem) and can display icons alongside the list items in a variety of ways.

ImageList component is used to display images used as icons.

Defining icons for list items

1. ToolBox ⇒ ImageList ⇒ property Images ⇒ Image Collection Editor ⇒ Add ⇒ choose image ⇒ OK



2. ListView ⇒
   property SmallImageList ⇒ object ImageList
   property LargeImageList ⇒ object ImageList
3. Icons for list items ListViewItem –
   set the item's ImageIndex property to the appropriate array index

ListView properties

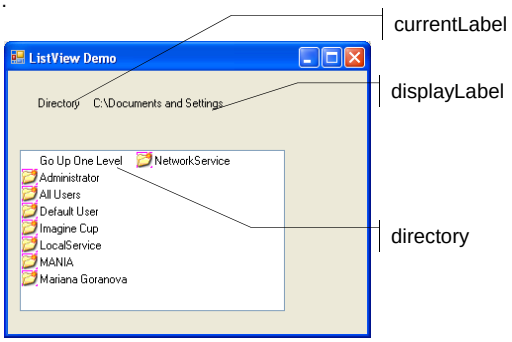
Activation	Determines how the user activates an item: OneClick (<L>), TwoClick (<L²>, item changes color when selected) and Standard (<L²>)
CheckBoxes	Indicates (true/false) whether items appear with checkboxes
LargeImageList	Indicates the ImageList used when displaying large icons
Items	Returns the collection of ListViewItems in the control
MultiSelect	Determines (true/false) whether multiple selection is allowed
SelectedItems	Lists the collection of currently selected items

SmallImageList	Indicates the ImageList used when displaying small icons
View	Determines appearance of ListViewItems: LargeIcon, SmallIcon, List and Details

ListView events

ItemActivate	Generated when an item in the ListView is activated; does not specify which item is activated

Example: Displays files and folders in a ListView, along with small icons representing each file or folder.




```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.IO;

namespace WindowsApplication16
{
   public partial class Form1 : Form
   {
      private string currentDirectory = Directory.GetCurrentDirectory();

      public Form1()
      {
         InitializeComponent();
      }
```

```
   private void directory_Click(object sender, EventArgs e)
   {  // If a directory is selected
      if (directory.SelectedItems.Count != 0)
      {  // If the first element is selected (always is Go Up One Level)
         if (directory.Items[0].Selected)
         {  DirectoryInfo directoryObject = new DirectoryInfo(currentDirectory);
            // If it is not a tree root
            if (directoryObject.Parent != null)
               LoadFilesInDirectory(directoryObject.Parent.FullName);
         }
         else // Traverse the tree
         {  string chosen = directory.SelectedItems[0].Text;
            if (Directory.Exists(currentDirectory + "\\" + chosen))       // Choose
            {  if (currentDirectory == "C:\\")
                  LoadFilesInDirectory(currentDirectory + chosen);    // directory
               else                                                   // or
                  LoadFilesInDirectory(currentDirectory+"\\"+chosen);// file
            }
         }
         displayLabel.Text = currentDirectory;
      }
   }
```

```
public void LoadFilesInDirectory(string currentDirectoryValue)
{  try
   {  directory.Items.Clear();
      directory.Items.Add("Go Up One Level");
      currentDirectory = currentDirectoryValue;
      DirectoryInfo newCurrentDirectory =
         new DirectoryInfo(currentDirectory);
      DirectoryInfo[] directoryArray =
         newCurrentDirectory.GetDirectories();
      FileInfo[] fileArray = newCurrentDirectory.GetFiles();
      foreach (DirectoryInfo d in directoryArray)
      {  ListViewItem newDirectoryItem = directory.Items.Add(d.Name);
         newDirectoryItem.ImageIndex=0;            // icon for a directory
      }
      foreach (FileInfo file in fileArray)
      {  ListViewItem newFileItem = directory.Items.Add(file.Name);
         newFileItem.ImageIndex = 1;               // icon for a file
      }
   }
   catch (UnauthorizedAccessException)
   {  MessageBox.Show("Attension: You don't have a prevelege",
         "Attention", 0, MessageBoxIcon.Warning);        }
}
```

```
private void Form1_Load(object sender, EventArgs e)
{
   Image folderImage = Image.FromFile (currentDirectory+"\\Folder.bmp");
   Image fileImage = Image.FromFile (currentDirectory+"\\File.bmp");
   fileFolder.Images.Add (folderImage);
   fileFolder.Images.Add (fileImage);
   LoadFilesInDirectory (currentDirectory);
   displayLabel.Text = currentDirectory;
}
}
}
```

```
namespace WindowsApplication16
{
  partial class Form1
  {
    private System.ComponentModel.IContainer components = null;

    protected override void Dispose(bool disposing)    {  …  }

    #region Windows Form Designer generated code

    private void InitializeComponent()
    {
      …
      this.currentLabel = new System.Windows.Forms.Label();
      this.displayLabel = new System.Windows.Forms.Label();
      this.directory = new System.Windows.Forms.ListView();
      this.fileFolder =
        new System.Windows.Forms.ImageList(this.components);
```

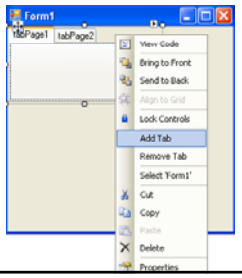
```
      // directory
      this.directory.Location = new System.Drawing.Point(12, 81);
      this.directory.Name = "directory";
      this.directory.SmallImageList = this.fileFolder;
      this.directory.View = System.Windows.Forms.View.List;
      this.directory.Click += new System.EventHandler(this.directory_Click);
      // fileFolder
      this.fileFolder.ImageStream =
            ((System.Windows.Forms.ImageListStreamer)
            (resources.GetObject("fileFolder.ImageStream")));
      this.fileFolder.TransparentColor = System.Drawing.Color.Transparent;
      this.fileFolder.Images.SetKeyName(0, "Folder.bmp");
      this.fileFolder.Images.SetKeyName(1, "File.bmp");
      // Form1
      this.Controls.Add(this.directory);
      this.Controls.Add(this.displayLabel);
      this.Controls.Add(this.currentLabel);
      this.Text = "ListView Demo";
      this.Click += new System.EventHandler(this.directory_Click);
      this.Load += new System.EventHandler(this.Form1_Load);
  }
  #endregion
```

```
    private System.Windows.Forms.Label currentLabel;
    private System.Windows.Forms.Label displayLabel;
    private System.Windows.Forms.ListView directory;
    private System.Windows.Forms.ImageList fileFolder;
  }
}
```

Control TabControl

Creates tabbed windows – fit a large number of controls. Contains TabPage objects that contain controls.

<R> TabControl ⇒ AddTab



Properties

ImageList	Specifies images to be displayed on a tab
ItemSize	Specifies tab size
MultiLine	Indicates (true/false) whether multiple rows of tabs can be displayed
SelectedIndex	Indicates index of TabPage that is currently selected
SelectedTab	Indicates the TabPage that is currently selected
TabCount	Returns the number of tabs
TabPages	Gets the collection of TabPage within the TabControl

Events

SelectedIndexChange	Generated when SelectedIndex changes

Example: Uses TabControl to display various options relating to the text on a label


```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication17
{
   public partial class Form1 : Form
   {
      public Form1()
      {  InitializeComponent();  }
      private void black_CheckedChanged(object sender, EventArgs e)
      {  displayLabel.ForeColor = Color.Black;  }

      private void red_CheckedChanged(object sender, EventArgs e)
      {  displayLabel.ForeColor = Color.Red;  }

      private void green_CheckedChanged(object sender, EventArgs e)
      {  displayLabel.ForeColor = Color.Green;  }
```

```
      private void size12_CheckedChanged(object sender, EventArgs e)
      {  displayLabel.Font = new Font(displayLabel.Font.Name, 12);  }

      private void size16_CheckedChanged(object sender, EventArgs e)
      {  displayLabel.Font = new Font(displayLabel.Font.Name, 16);  }

      private void size20_CheckedChanged(object sender, EventArgs e)
      {  displayLabel.Font = new Font(displayLabel.Font.Name, 20);  }

      private void hello_CheckedChanged(object sender, EventArgs e)
      {  displayLabel.Text = "Hello!";  }

      private void goodbye_CheckedChanged(object sender, EventArgs e)
      {  displayLabel.Text = "Goodbye!";  }
   }
}
```

```
namespace WindowsApplication17
{
   partial class Form1
   {
      private System.ComponentModel.IContainer components = null;

      protected override void Dispose(bool disposing)  {  …  }

      #region Windows Form Designer generated code

      private void InitializeComponent()
      {
         this.options = new System.Windows.Forms.TabControl();
         this.color = new System.Windows.Forms.TabPage();
         this.green = new System.Windows.Forms.RadioButton();
         this.red = new System.Windows.Forms.RadioButton();
         this.black = new System.Windows.Forms.RadioButton();
         this.size = new System.Windows.Forms.TabPage();
         this.message = new System.Windows.Forms.TabPage();
         this.about = new System.Windows.Forms.TabPage();
         this.displayLabel = new System.Windows.Forms.Label();
```

```
         this.size20 = new System.Windows.Forms.RadioButton();
         this.size16 = new System.Windows.Forms.RadioButton();
         this.size12 = new System.Windows.Forms.RadioButton();
         this.goodbye = new System.Windows.Forms.RadioButton();
         this.hello = new System.Windows.Forms.RadioButton();
         this.messageLabel = new System.Windows.Forms.Label();
         …
         // options
         this.options.Controls.Add(this.color);
         this.options.Controls.Add(this.size);
         this.options.Controls.Add(this.message);
         this.options.Controls.Add(this.about);
         this.options.Name = "options";
         this.options.SelectedIndex = 0;

         // color
         this.color.Controls.Add(this.green);
         this.color.Controls.Add(this.red);
         this.color.Controls.Add(this.black);
         this.color.Name = "color";
         this.color.Text = "Color";
```

```
        // green
        this.green.Name = "green";
        this.green.Text = "Green";
        this.green.CheckedChanged +=
                new System.EventHandler(this.green_CheckedChanged);
        // red
        this.red.Name = "red";
        this.red.Text = "Red";
        this.red.CheckedChanged +=
                new System.EventHandler(this.red_CheckedChanged);

        // black
        this.black.Checked = true;
        this.black.Name = "black";
        this.black.Text = "Black";
        this.black.CheckedChanged +=
                new System.EventHandler(this.black_CheckedChanged);
```

```
        // size
        this.size.Controls.Add(this.size20);
        this.size.Controls.Add(this.size16);
        this.size.Controls.Add(this.size12);
        this.size.Name = "size";
        this.size.Text = "Size";

        // message
        this.message.Controls.Add(this.goodbye);
        this.message.Controls.Add(this.hello);
        this.message.Name = "message";
        this.message.Text = "Message";

        // about
        this.about.Controls.Add(this.messageLabel);
        this.about.Name = "about";
        this.about.Text = "About";
```

```
        // displayLabel
        this.displayLabel.Dock = System.Windows.Forms.DockStyle.Bottom;
        this.displayLabel.Font = new System.Drawing.Font
         ("Microsoft Sans Serif", 12F, System.Drawing.FontStyle.Regular,
         System.Drawing.GraphicsUnit.Point, ((byte)(204)));
        this.displayLabel.Name = "displayLabel";
        this.displayLabel.Size = new System.Drawing.Size(195, 91);
        this.displayLabel.Text = "TabControl Demo";
        this.displayLabel.TextAlign =
                System.Drawing.ContentAlignment.MiddleCenter;

        // size20
        this.size20.Name = "size20";
        this.size20.Size = new System.Drawing.Size(63, 17);
        this.size20.Text = "20 point";
        this.size20.CheckedChanged +=
                new System.EventHandler(this.size20_CheckedChanged);
        // size16
        this.size16.Name = "size16";
        this.size16.Text = "16 point";
        this.size16.CheckedChanged +=
                new System.EventHandler(this.size16_CheckedChanged);
```

```
        // size12
        this.size12.Checked = true;
        this.size12.Name = "size12";
        this.size12.Text = "12 point";
        this.size12.CheckedChanged +=
                new System.EventHandler(this.size12_CheckedChanged);

        // goodbye
        this.goodbye.Name = "goodbye";
        this.goodbye.Text = "Goodbye!";
        this.goodbye.CheckedChanged +=
         new System.EventHandler(this.goodbye_CheckedChanged);
        // hello
        this.hello.Name = "hello";
        this.hello.Text = "Hello!";
        this.hello.CheckedChanged +=
                new System.EventHandler(this.hello_CheckedChanged);
        // messageLabel
        this.messageLabel.Name = "messageLabel";
        this.messageLabel.Size = new System.Drawing.Size(177, 37);
        this.messageLabel.Text =
           "Tabs are used to organize controls and conserve screen space.";
```

```
        // Form1
        this.Controls.Add(this.displayLabel);
        this.Controls.Add(this.options);
        this.Name = "Form1";
        this.Text = "TabControl Demo";
    }
    #endregion
    private System.Windows.Forms.TabControl options;
    private System.Windows.Forms.TabPage color;
    private System.Windows.Forms.TabPage size;
    private System.Windows.Forms.TabPage message;
    private System.Windows.Forms.TabPage about;
    private System.Windows.Forms.Label displayLabel;
    private System.Windows.Forms.RadioButton green;
    private System.Windows.Forms.RadioButton red;
    private System.Windows.Forms.RadioButton black;
    private System.Windows.Forms.RadioButton size20;
    private System.Windows.Forms.RadioButton size16;
    private System.Windows.Forms.RadioButton size12;
    private System.Windows.Forms.RadioButton goodbye;
    private System.Windows.Forms.RadioButton hello;
    private System.Windows.Forms.Label messageLabel;
  }
}
```

Validating User Input

A control CausesValidation property indicates whether the control raises validation events. If CausesValidation is set to true (default) for a control, when that control receives the focus, the previous control loosing the focus will be validated.

Validation Events

Validating	Occurs when focus leaves a control. If control's data is not well-formatted, we can set the Cancel property of the CancelEventArgs parameter to prevent the focus from changing.
Validated	Fires after the Validating event but before the control loses focus. We cannot cancel this event and it is not useful for checking the user's input.

Control ErrorProvider

Displays error information.

Methods

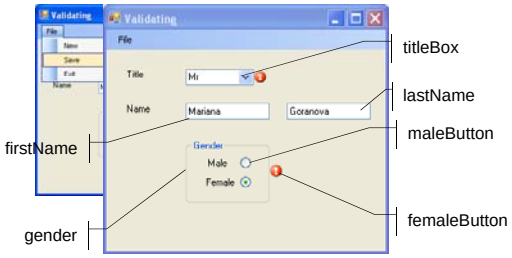
SetError — Specifies the *control* that the error icon should appear next to, and the *value* of the error message string.

```
public void SetError (Control control, string value);
```

Validating data:

1. Validate the contents of a single control
   - Validating event
   - CausesValidation = true for all controls on the form to allow the Validation event to be raised
2. Validate the contents of multiple controls or an entire form – form-level validation:
   - Create a method that validates all the data on the form
   - Call this method when the data input is complete (Save button)
3. Indicate which values are in error and display error information – use an ErrorProvider control
   - Call the SetError method to display an error icon and record the error message as a ToolTip

Example: Form with a simple cross-check between the contents of the Title list box and the radio buttons in the Gender group box. When the Save menu is selected the method is called that validates all the data on the form.




```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication18
{  public partial class Form1 : Form
   {  public Form1()
      {  InitializeComponent();
         Reset();
      }
      private void Reset()
      {  titleBox.Text = "Mr";
         firstName.Text = "";
         lastName.Text = "";
         maleButton.Checked = true;
         errorProvider.SetError (gender, "");
         errorProvider.SetError (titleBox, "");
      }
```

```
      private bool checkTitleAndGender()
      {  if (titleBox.Text == "Mr")
         {  if (!maleButton.Checked)
            {  errorProvider.SetError(gender,
                  "If the title is Mr the gender must be male");
               errorProvider.SetError(titleBox,
                  "If the gender is female the title must be Mrs, Miss, or Ms");
               return false;
            }
         }
         else if (titleBox.Text == "Mrs" || titleBox.Text == "Miss" ||
                  titleBox.Text == "Ms")
         {  if (!femaleButton.Checked)
            {  errorProvider.SetError(gender,
                  "If the title is Mrs, Miss, or Ms the gender must be female");
               errorProvider.SetError(titleBox,
                  "If the gender is male the title must be Mr");
               return false;
            }
         }
         errorProvider.SetError (gender, "");
         errorProvider.SetError (titleBox, "");
         return true;
      }
```

```
      private void newToolStripMenuItem_Click(object sender, EventArgs e)
      {
         Reset();
      }

      private void exitToolStripMenuItem_Click(object sender, EventArgs e)
      {
         this.Close();
      }

      private void saveToolStripMenuItem_Click(object sender, EventArgs e)
      {
         // If the check is not successful
         if (!checkTitleAndGender())
            e.Cancel = true;
      }
   }
}
```

```
namespace WindowsApplication18
{
   partial class Form1
   {  private System.ComponentModel.IContainer components = null;

      protected override void Dispose(bool disposing)  {  …  }

      #region Windows Form Designer generated code
      private void InitializeComponent()
      {  …
         // menuStrip1
         this.menuStrip1.Items.AddRange(new
           System.Windows.Forms.ToolStripItem[] {this.fileToolStripMenuItem});
         this.menuStrip1.Name = "menuStrip1";
         this.menuStrip1.Text = "menuStrip1";
         // fileToolStripMenuItem
         this.fileToolStripMenuItem.DropDownItems.AddRange(
             new  System.Windows.Forms.ToolStripItem[] {
                 this.newToolStripMenuItem, this.saveToolStripMenuItem,
                 this.exitToolStripMenuItem});
         this.fileToolStripMenuItem.Name = "fileToolStripMenuItem";
         this.fileToolStripMenuItem.Text = "File";
```

```
         // newToolStripMenuItem
         this.newToolStripMenuItem.Name = "newToolStripMenuItem";
         this.newToolStripMenuItem.Text = "New";
         this.newToolStripMenuItem.Click +=
                 new System.EventHandler(this.newToolStripMenuItem_Click);

         // saveToolStripMenuItem
         this.saveToolStripMenuItem.Name = "saveToolStripMenuItem";
         this.saveToolStripMenuItem.Text = "Save";
         this.saveToolStripMenuItem.Click +=
                 new System.EventHandler(this.saveToolStripMenuItem_Click);

         // exitToolStripMenuItem
         this.exitToolStripMenuItem.Name = "exitToolStripMenuItem";
         this.exitToolStripMenuItem.Text = "Exit";
         this.exitToolStripMenuItem.Click +=
                  new System.EventHandler(this.exitToolStripMenuItem_Click);

         // titleLabel
         this.titleLabel.Name = "titleLabel";
         this.titleLabel.Text = "Title";
```

```
         // titleBox
         this.titleBox.Items.AddRange(new object[] {"Mr","Mrs","Miss","Ms"});
         this.titleBox.Name = "titleBox";

         // nameLabel
         this.nameLabel.Name = "nameLabel";
         this.nameLabel.Text = "Name";

         // firstName
         this.firstName.Name = "firstName";
         // lastName
         this.lastName.Name = "lastName";
         // gender
         this.gender.Controls.Add(this.femaleButton);
         this.gender.Controls.Add(this.maleButton);
         this.gender.Name = "gender";
         this.gender.Text = "Gender";
         // maleButton
         this.maleButton.Name = "maleButton";
         this.maleButton.RightToLeft =
                   System.Windows.Forms.RightToLeft.Yes;
         this.maleButton.Text = "Male   ";
```

```
         // femaleButton
         this.femaleButton.Name = "femaleButton";
         this.femaleButton.RightToLeft =
                   System.Windows.Forms.RightToLeft.Yes;
         this.femaleButton.Text = "Female";
         // errorProvider
         this.errorProvider.ContainerControl = this;
         // Form1
         this.Controls.Add(this.gender);
         this.Controls.Add(this.lastName);
         this.Controls.Add(this.firstName);
         this.Controls.Add(this.nameLabel);
         this.Controls.Add(this.titleBox);
         this.Controls.Add(this.titleLabel);
         this.Controls.Add(this.menuStrip1);
         this.MainMenuStrip = this.menuStrip1;
         this.Name = "Form1";
         this.Text = "Validating";
      }

      #endregion
```

```
      private System.Windows.Forms.MenuStrip menuStrip1;
      private System.Windows.Forms.ToolStripMenuItem
          fileToolStripMenuItem;
      private System.Windows.Forms.ToolStripMenuItem
          newToolStripMenuItem;
      private System.Windows.Forms.ToolStripMenuItem
          saveToolStripMenuItem;
      private System.Windows.Forms.ToolStripMenuItem
          exitToolStripMenuItem;
      private System.Windows.Forms.Label titleLabel;
      private System.Windows.Forms.ComboBox titleBox;
      private System.Windows.Forms.Label nameLabel;
      private System.Windows.Forms.TextBox firstName;
      private System.Windows.Forms.TextBox lastName;
      private System.Windows.Forms.GroupBox gender;
      private System.Windows.Forms.RadioButton femaleButton;
      private System.Windows.Forms.RadioButton maleButton;
      private System.Windows.Forms.ErrorProvider errorProvider;
   }
}
```

If we validate the contents of a single control (input a title), we use the Validating event:

```
private void titleValidating(object sender,
        System.ComponentModel.CancelEventArgs e)
{
   // If the check is not successful
   if (!checkTitleAndGender())
      e.Cancel = true;
}
```

Multiple-Document Interface (MDI) Applications

Different styles for the user interface

1. Single-document interface (SDI)
2. Multiple-document interface (MDI)
3. Explorer-style interface

Creating MDI application

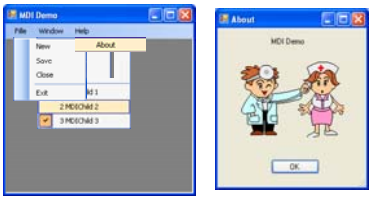
1. Creating a parent form
   - IsMdiContainer = true;
   - LayoutMdi () method – arranges child forms in an MDI parent form
2. Creating a child-form
3. Calling a child from a parent form

Determining the active MDI child

Property

ActiveMdiChild returns the current active child form.

Example: MDI application – when the user opens a New window then the menu Save appears. When the user wants to save the file he/she uses a SaveFileDialog control.



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication19
{
   public partial class Form1 : Form
   {
      private int childCount;

      public Form1()
      {
         InitializeComponent();
         childCount = 0;
      }
```

```
   private void newItemm_Click(object sender, EventArgs e)
   {
      MDIChild childForm = new MDIChild();          // Create a child form
      childForm.MdiParent = this;  // Set the parent form of the child window
      childCount++;
      childForm.Text = childForm.Text + " " + childCount;
      childForm.Show();                             // Display the child form
   }

   private void closeItem_Click(object sender, EventArgs e)
   {
      // Determining the active child form
      Form childForm = this.ActiveMdiChild;
      if (childForm != null)
         childForm.Close();
   }

   private void exitItem_Click(object sender, EventArgs e)
   {
      this.Close();
   }
```

```
      private void cascadeItem_Click(object sender, EventArgs e)
      {
         this.LayoutMdi(MdiLayout.Cascade);
      }

      private void horizontalItem_Click(object sender, EventArgs e)
      {
         this.LayoutMdi(MdiLayout.TileHorizontal);
      }

      private void verticalItem_Click(object sender, EventArgs e)
      {
         this.LayoutMdi(MdiLayout.TileVertical);
      }

      private void aboutItem_Click(object sender, EventArgs e)
      {
         About aboutDialog = new About();
         aboutDialog.ShowDialog();
      }
   }
}
```

```
namespace WindowsApplication19
{
   partial class Form1
   {
      private System.ComponentModel.IContainer components = null;

      protected override void Dispose(bool disposing) {  …  }

      #region Windows Form Designer generated code

      private void InitializeComponent()
      {
         …                                  Create an MDI window list on a MenuStrip
         // mdiMenu
         this.mdiMenu.Items.AddRange
            (new System.Windows.Forms.ToolStripItem[]
            {this.fileItem, this.windowItem, this.helpItem});
         this.mdiMenu.MdiWindowListItem = this.windowItem;
         this.mdiMenu.Name = "mdiMenu";
         this.mdiMenu.Text = "menuStrip1";
```

```
         // fileItem
         this.fileItem.DropDownItems.AddRange
            (new System.Windows.Forms.ToolStripItem[]
            {this.newItemm, this.closeItem, this.exitItem});
         this.fileItem.MergeAction =
                  System.Windows.Forms.MergeAction.MatchOnly;
         this.fileItem.MergeIndex = 0;
         this.fileItem.Name = "fileItem";
         this.fileItem.Text = "File";          MergeAction = MergeAction.MatchOnly
                                               Child menu will be nested into parent
         // newItemm                           menu
         this.newItemm.MergeAction =
                  System.Windows.Forms.MergeAction.Insert;
         this.newItemm.MergeIndex = 0;
         this.newItemm.Name = "newItemm";
         this.newItemm.Text = "New";
         this.newItemm.Click +=
                  new System.EventHandler(this.newItemm_Click);

                                               MergeAction = MergeAction.Insert
                                               Child menu is inserted into parent menu
```

```
         // closeItem
         this.closeItem.MergeAction =
                  System.Windows.Forms.MergeAction.Insert;
         this.closeItem.MergeIndex = 2;
         this.closeItem.Name = "closeItem";
         this.closeItem.Text = "Close";
         this.closeItem.Click += new System.EventHandler(this.closeItem_Click);

         // exitItem
         this.exitItem.MergeAction =
                  System.Windows.Forms.MergeAction.Insert;
         this.exitItem.MergeIndex = 4;
         this.exitItem.Name = "exitItem";
         this.exitItem.Text = "Exit";
         this.exitItem.Click += new System.EventHandler(this.exitItem_Click);

         // windowItem
         this.windowItem.DropDownItems.AddRange
            (new System.Windows.Forms.ToolStripItem[]
            {this.cascadeItem, this.horizontalItem, this.verticalItem});
         this.windowItem.Name = "windowItem";
         this.windowItem.Text = "Window";
```

```
         // cascadeItem
         this.cascadeItem.Name = "cascadeItem";
         this.cascadeItem.Text = "Cascade";
         this.cascadeItem.Click +=
                  new System.EventHandler(this.cascadeItem_Click);
         // horizontalItem
         this.horizontalItem.Name = "horizontalItem";
         this.horizontalItem.Text = "Horizontal";
         this.horizontalItem.Click +=
                  new System.EventHandler(this.horizontalItem_Click);
         // verticalItem
         this.verticalItem.Name = "verticalItem";
         this.verticalItem.Text = "Vertical";
         this.verticalItem.Click +=
                  new System.EventHandler(this.verticalItem_Click);

         // helpItem
         this.helpItem.DropDownItems.AddRange
            (new System.Windows.Forms.ToolStripItem[] {this.aboutItem});
         this.helpItem.Name = "helpItem";
         this.helpItem.Text = "Help";
```

```
         // aboutItem
         this.aboutItem.Name = "aboutItem";
         this.aboutItem.Text = "About";
         this.aboutItem.Click +=
                  new System.EventHandler(this.aboutItem_Click);

         // Form1
         this.Controls.Add(this.mdiMenu);
         this.IsMdiContainer = true;
         this.MainMenuStrip = this.mdiMenu;
         this.Name = "Form1";
         this.Text = "MDI Demo";
      }

      #endregion
```

```
      private System.Windows.Forms.MenuStrip mdiMenu;
      private System.Windows.Forms.ToolStripMenuItem fileItem;
      private System.Windows.Forms.ToolStripMenuItem newItemm;
      private System.Windows.Forms.ToolStripMenuItem closeItem;
      private System.Windows.Forms.ToolStripMenuItem exitItem;
      private System.Windows.Forms.ToolStripMenuItem windowItem;
      private System.Windows.Forms.ToolStripMenuItem cascadeItem;
      private System.Windows.Forms.ToolStripMenuItem horizontalItem;
      private System.Windows.Forms.ToolStripMenuItem verticalItem;
      private System.Windows.Forms.ToolStripMenuItem helpItem;
      private System.Windows.Forms.ToolStripMenuItem aboutItem;
   }
}
```

```
// Project ⇒ Add Windows Form

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.IO;

namespace WindowsApplication19
{
   public partial class MDIChild : Form
   {
      public MDIChild()
      {
         InitializeComponent();
      }
```

```
      private void saveItem_Click(object sender, EventArgs e)
      {
         DialogResult buttonClicked = saveFileDialog.ShowDialog();
         if (buttonClicked.Equals (DialogResult.OK))
         {
            Stream saveStream = saveFileDialog.OpenFile();
            StreamWriter saveWriter = new StreamWriter (saveStream);
            foreach (string line in editData.Lines)
               saveWriter.WriteLine (line);
            saveWriter.Close();
         }
      }
   }
}
```

```
namespace WindowsApplication19
{
   partial class MDIChild
   {
      private System.ComponentModel.IContainer components = null;

      protected override void Dispose(bool disposing)  {  …  }

      #region Windows Form Designer generated code

      private void InitializeComponent()
      {
         …
         // childMenu
         this.childMenu.Items.AddRange
            (new System.Windows.Forms.ToolStripItem[] {this.fileItem});
         this.childMenu.Name = "childMenu";
         this.childMenu.Text = "menuStrip1";
         this.childMenu.Visible = false;
```

```
         // fileItem
         this.fileItem.DropDownItems.AddRange
            (new System.Windows.Forms.ToolStripItem[]
            {this.saveItem, this.separatorItem});
         this.fileItem.MergeAction =
                  System.Windows.Forms.MergeAction.MatchOnly;
         this.fileItem.MergeIndex = 0;
         this.fileItem.Name = "fileItem";
         this.fileItem.Text = "File";

         // saveItem
         this.saveItem.MergeAction =
                  System.Windows.Forms.MergeAction.Insert;
         this.saveItem.MergeIndex = 1;
         this.saveItem.Name = "saveItem";
         this.saveItem.Text = "Save";
         this.saveItem.Click += new System.EventHandler(this.saveItem_Click);
         // separatorItem
         this.separatorItem.MergeAction =
                  System.Windows.Forms.MergeAction.Insert;
         this.separatorItem.MergeIndex = 3;
         this.separatorItem.Name = "separatorItem";
```

```
         // editData
         this.editData.Dock = System.Windows.Forms.DockStyle.Fill;
         this.editData.Multiline = true;
         this.editData.Name = "editData";

         // MDIChild
         this.Controls.Add(this.editData);
         this.Controls.Add(this.childMenu);
         this.MainMenuStrip = this.childMenu;
         this.Name = "MDIChild";
         this.Text = "MDIChild";
      }
      #endregion

      private System.Windows.Forms.MenuStrip childMenu;
      private System.Windows.Forms.ToolStripMenuItem fileItem;
      private System.Windows.Forms.ToolStripMenuItem saveItem;
      private System.Windows.Forms.ToolStripSeparator separatorItem;
      private System.Windows.Forms.TextBox editData;
      private System.Windows.Forms.SaveFileDialog saveFileDialog;
   }
}
```

```
// Project ⇒ Add Windows Form
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication19
{  public partial class About : Form
   {  public About()
      {
         InitializeComponent();
      }

      private void ok_Click(object sender, EventArgs e)
      {
         this.Close();
      }
   }
}
```

```
namespace WindowsApplication19
{
   partial class About
   {
      private System.ComponentModel.IContainer components = null;

      protected override void Dispose(bool disposing)  {  …  }

      #region Windows Form Designer generated code

      private void InitializeComponent()
      {
         // label1
         this.label1.Image =
                  global::WindowsApplication19.Properties.Resources.doctor;
         this.label1.Name = "label1";

         // label2
         this.label2.Name = "label2";
         this.label2.Text = "MDI Demo";
```

```
         // ok
         this.ok.Name = "ok";
         this.ok.Text = "OK";
         this.ok.Click += new System.EventHandler(this.ok_Click);

         // About
         this.Controls.Add(this.ok);
         this.Controls.Add(this.label2);
         this.Controls.Add(this.label1);
         this.Name = "About";
         this.Text = "About";
      }

      #endregion

      private System.Windows.Forms.Label label1;
      private System.Windows.Forms.Label label2;
      private System.Windows.Forms.Button ok;
   }
}
```

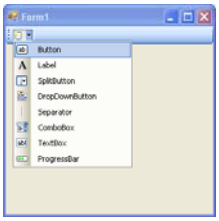
Toolbar and Status Bar

Control ToolStrip (ToolBar)

Creates advanced toolbar functionality that has consistent and professional appearance and behavior.

ToolStrip is a ToolStripItem container for items:

ToolStripLabel
ToolStripButton
ToolStripSeparator
ToolStripControlHost
ToolStripDropDownItem
ToolStripComboBox
ToolStripTextBox
ToolStripProgressBar
ToolStripDropDownButton
ToolStripSplitButton



Events for ToolStrip

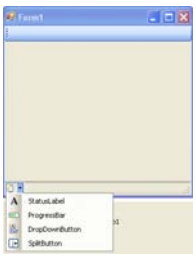
ItemClicked Occurs when the ToolStripItem is clicked (default).

Control StatusStrip (StatusBar)

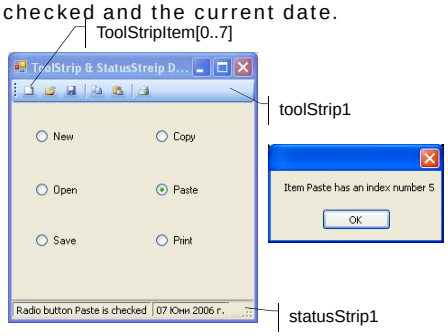
Displays information about an object being viewed on a Form, the object's components, or contextual information that relates to that object's operation within the application.

StatusStrip is a ToolStripItem container for items:

ToolStripLabel
ToolStripSplitButton
ToolStripDropDownButton
ToolStripStatusLabel
ToolStripProgressBar
ToolStripControlHost



Example: Application with toolbar and status bar. When the user clicks a tool strip item the Checked property of the corresponding radio button is set to true. The status bar labels display which radio button is checked and the current date.



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication20
{
   public partial class Form1 : Form
   {
      private string labelText;

      public Form1()
      {
         InitializeComponent();
      }
```

```
      private void toolStrip1_ItemClicked(object sender,
                                          ToolStripItemClickedEventArgs e)
      {
         switch(e.ClickedItem.Text)
         {
            case "New":
               newRadioButton.Checked = true;
               labelText = "Radio button " + newRadioButton.Text + " is checked";
               break;
            case "Open":
               openRadioButton.Checked = true;
               labelText = "Radio button "+ openRadioButton.Text + " is checked";
               break;
            case "Save":
               saveRadioButton.Checked = true;
               labelText = "Radio button "+ saveRadioButton.Text + " is checked";
               break;
            case "Copy":
               copyRadioButton.Checked = true;
               labelText = "Radio button " + copyRadioButton.Text +" is checked";
               break;
```

```
            case "Paste":
               pasteRadioButton.Checked = true;
               labelText = "Radio button " + pasteRadioButton.Text+" is checked";
               break;
            case "Print":
               printRadioButton.Checked = true;
               labelText = "Radio button " + printRadioButton.Text + " is checked";
               break;
         }
         label1.Text = labelText;
         MessageBox.Show("Item " + e.ClickedItem.ToString()
           + " has an index number " + toolStrip1.Items.IndexOf(e.ClickedItem));
      }
```

```
      private void Form1_Load(object sender, EventArgs e)
      {
         label1.ToolTipText = "Current checked radio button";
         label2.ToolTipText = "Current date";
         label1.Text = "Radio button " + newRadioButton.Text + " is checked";
         label2.Text = System.DateTime.Now.ToLongDateString();
         newRadioButton.Checked = true;
      }
   }
}
```

```
namespace WindowsApplication20
{
   partial class Form1
   {
      private System.ComponentModel.IContainer components = null;
      protected override void Dispose(bool disposing)   {  …  }

      #region Windows Form Designer generated code

      private void InitializeComponent()
      {  …
         // toolStrip1
         this.toolStrip1.Items.AddRange
            (new System.Windows.Forms.ToolStripItem[] {this.toolStripButton1,
             this.toolStripButton2, this.toolStripButton3, this.toolStripSeparator1,
             this.toolStripButton4, this.toolStripButton5, this.toolStripSeparator2,
             this.toolStripButton6});
         this.toolStrip1.Name = "toolStrip1";
         this.toolStrip1.Text = "New";
         this.toolStrip1.ItemClicked +=
            new System.Windows.Forms.ToolStripItemClickedEventHandler
            (this.toolStrip1_ItemClicked);
```

```
         // toolStripButton1
         this.toolStripButton1.Image = ((System.Drawing.Image)
                                        (resources.GetObject("toolStripButton1.Image")));
         this.toolStripButton1.Name = "toolStripButton1";
         this.toolStripButton1.Text = "New";

         // toolStripButton2
         this.toolStripButton2.Image = ((System.Drawing.Image)
                                       (resources.GetObject("toolStripButton2.Image")));
         this.toolStripButton2.Name = "toolStripButton2";
         this.toolStripButton2.Text = "Open";

         // toolStripButton3
         this.toolStripButton3.Image = ((System.Drawing.Image)
                                        (resources.GetObject("toolStripButton3.Image")));
         this.toolStripButton3.Name = "toolStripButton3";
         this.toolStripButton3.Text = "Save";

         // toolStripSeparator1
         this.toolStripSeparator1.Name = "toolStripSeparator1";
```

```
        // toolStripButton4
        this.toolStripButton4.Image = ((System.Drawing.Image)
                                (resources.GetObject("toolStripButton4.Image")));
        this.toolStripButton4.Name = "toolStripButton4";
        this.toolStripButton4.Text = "Copy";

        // toolStripButton5
        this.toolStripButton5.Image = ((System.Drawing.Image)
                                (resources.GetObject("toolStripButton5.Image")));
        this.toolStripButton5.Name = "toolStripButton5";
        this.toolStripButton5.Text = "Paste";

        // toolStripSeparator2
        this.toolStripSeparator2.Name = "toolStripSeparator2";

        // toolStripButton6
        this.toolStripButton6.Image = ((System.Drawing.Image)
                                (resources.GetObject("toolStripButton6.Image")));
        this.toolStripButton6.Name = "toolStripButton6";
        this.toolStripButton6.Text = "Print";
```

```
        // newRadioButton
        this.newRadioButton.Name = "newRadioButton";
        this.newRadioButton.Text = "New";

        // openRadioButton
        this.openRadioButton.Name = "openRadioButton";
        this.openRadioButton.Text = "Open";

        // saveRadioButton
        this.saveRadioButton.Name = "saveRadioButton";
        this.saveRadioButton.Text = "Save";

        // copyRadioButton
        this.copyRadioButton.Name = "copyRadioButton";
        this.copyRadioButton.Text = "Copy";

        // pasteRadioButton
        this.pasteRadioButton.Name = "pasteRadioButton";
        this.pasteRadioButton.Text = "Paste";
         // printRadioButton
        this.printRadioButton.Name = "printRadioButton";
        this.printRadioButton.Text = "Print";
```

```
        // statusStrip1
        this.statusStrip1.Items.AddRange
          (new System.Windows.Forms.ToolStripItem[] {this.label1, this.label2});
        this.statusStrip1.Text = "statusStrip1";

        // label1
        this.label1.BorderSides =
         ((System.Windows.Forms.ToolStripStatusLabelBorderSides)
         ((((System.Windows.Forms.ToolStripStatusLabelBorderSides.Left |
         System.Windows.Forms.ToolStripStatusLabelBorderSides.Top) |
         System.Windows.Forms.ToolStripStatusLabelBorderSides.Right) |
         System.Windows.Forms.ToolStripStatusLabelBorderSides.Bottom)));
        this.label1.BorderStyle =
                System.Windows.Forms.Border3DStyle.SunkenInner;
        this.label1.Name = "label1";
        this.label1.Text = "toolStripStatusLabel1";
```

```
        // label2
        this.label2.BorderSides =
         ((System.Windows.Forms.ToolStripStatusLabelBorderSides)
         ((((System.Windows.Forms.ToolStripStatusLabelBorderSides.Left |
         System.Windows.Forms.ToolStripStatusLabelBorderSides.Top) |
         System.Windows.Forms.ToolStripStatusLabelBorderSides.Right) |
         System.Windows.Forms.ToolStripStatusLabelBorderSides.Bottom)));
        this.label2.BorderStyle =
                System.Windows.Forms.Border3DStyle.SunkenInner;
        this.label2.Name = "label2";
        this.label2.Text = "toolStripStatusLabel1";
```

```
        // Form1
        this.Controls.Add(this.statusStrip1);
        this.Controls.Add(this.printRadioButton);
        this.Controls.Add(this.pasteRadioButton);
        this.Controls.Add(this.copyRadioButton);
        this.Controls.Add(this.saveRadioButton);
        this.Controls.Add(this.openRadioButton);
        this.Controls.Add(this.newRadioButton);
        this.Controls.Add(this.toolStrip1);
        this.Name = "Form1";
        this.Text = "ToolStrip & StatusStreip Demo";
        this.Load += new System.EventHandler(this.Form1_Load);
    }

    #endregion
```

```
        private System.Windows.Forms.ToolStrip toolStrip1;
        private System.Windows.Forms.ToolStripButton toolStripButton1;
        private System.Windows.Forms.ToolStripButton toolStripButton2;
        private System.Windows.Forms.ToolStripButton toolStripButton3;
        private System.Windows.Forms.ToolStripSeparator toolStripSeparator1;
        private System.Windows.Forms.ToolStripButton toolStripButton4;
        private System.Windows.Forms.ToolStripButton toolStripButton5;
        private System.Windows.Forms.ToolStripSeparator toolStripSeparator2;
        private System.Windows.Forms.ToolStripButton toolStripButton6;
        private System.Windows.Forms.RadioButton newRadioButton;
        private System.Windows.Forms.RadioButton openRadioButton;
        private System.Windows.Forms.RadioButton saveRadioButton;
        private System.Windows.Forms.RadioButton copyRadioButton;
        private System.Windows.Forms.RadioButton pasteRadioButton;
        private System.Windows.Forms.RadioButton printRadioButton;
        private System.Windows.Forms.StatusStrip statusStrip1;
        private System.Windows.Forms.ToolStripStatusLabel label1;
        private System.Windows.Forms.ToolStripStatusLabel label2;
    }
}
```