Chương 4:
Lập trình ứng dụng windows Froms

GV: LÊ THỌ

FACULTY OF INFORMATION TECHNOLOGY
THU DUC COLLEGE OF TECHNOLOGY

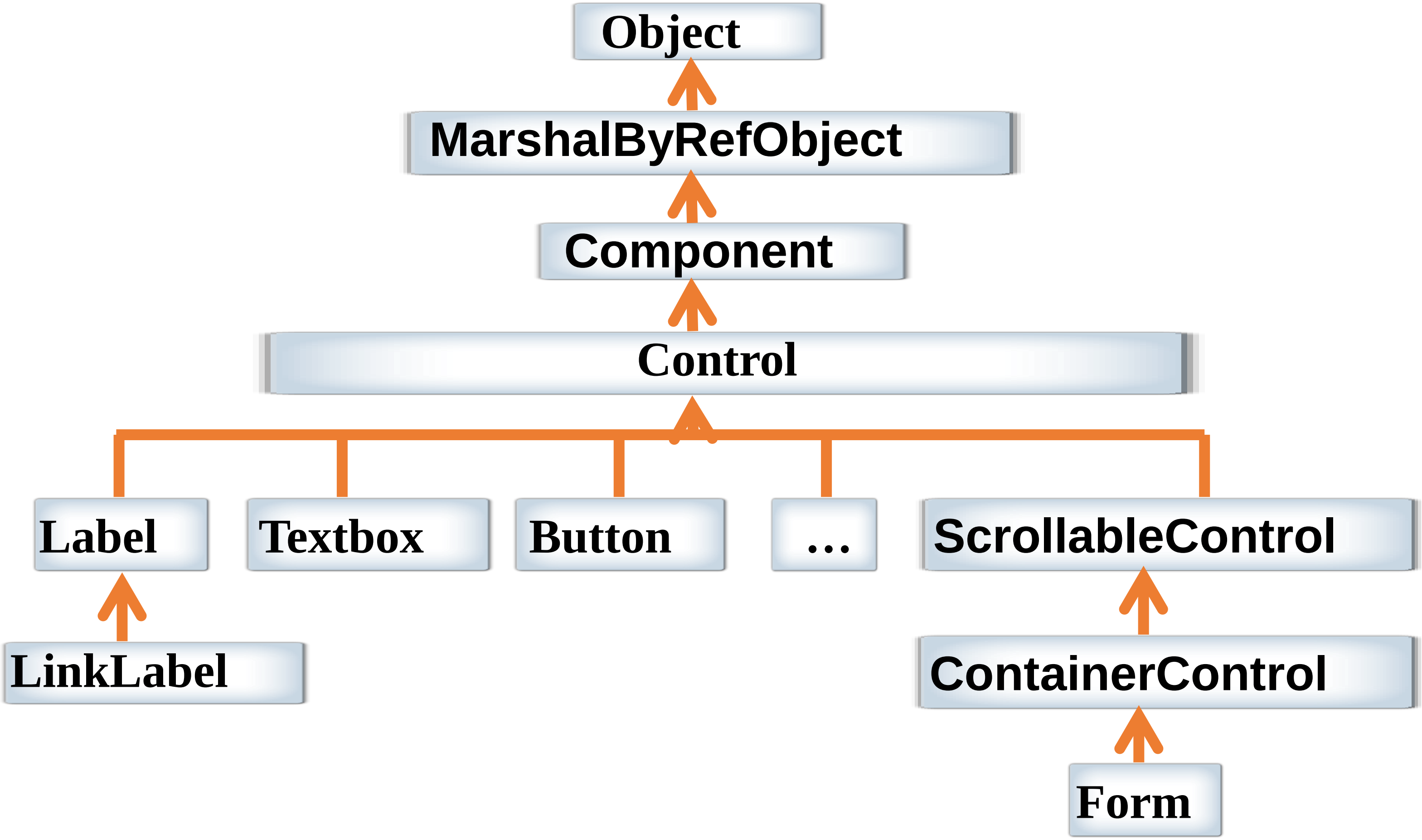
MỤC TIÊU

- Vận dụng các control để thiết kế ứng dụng
- Xử lý các sự kiện với các control

NỘI DUNG

- Các thành phần ứng dụng windows form
- Sử dụng các control để thiết kế ứng dụng
- Sử dụng các điều khiển chuẩn

Các thành phần của Windows Forms



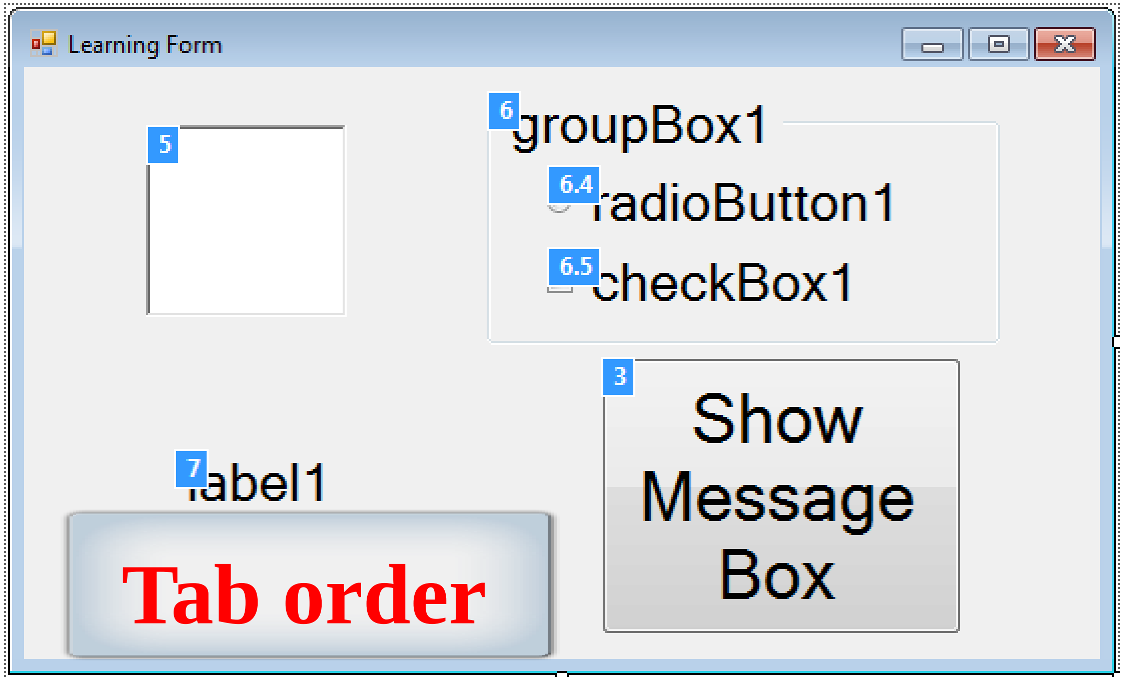
Windows Form

Create Form

Popular Properties

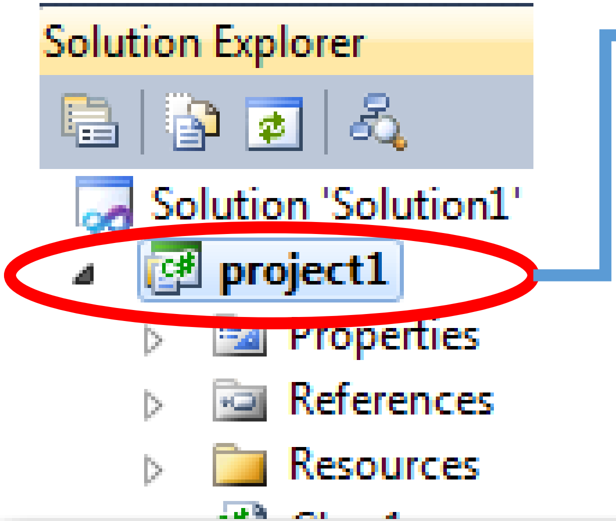
Popular Events

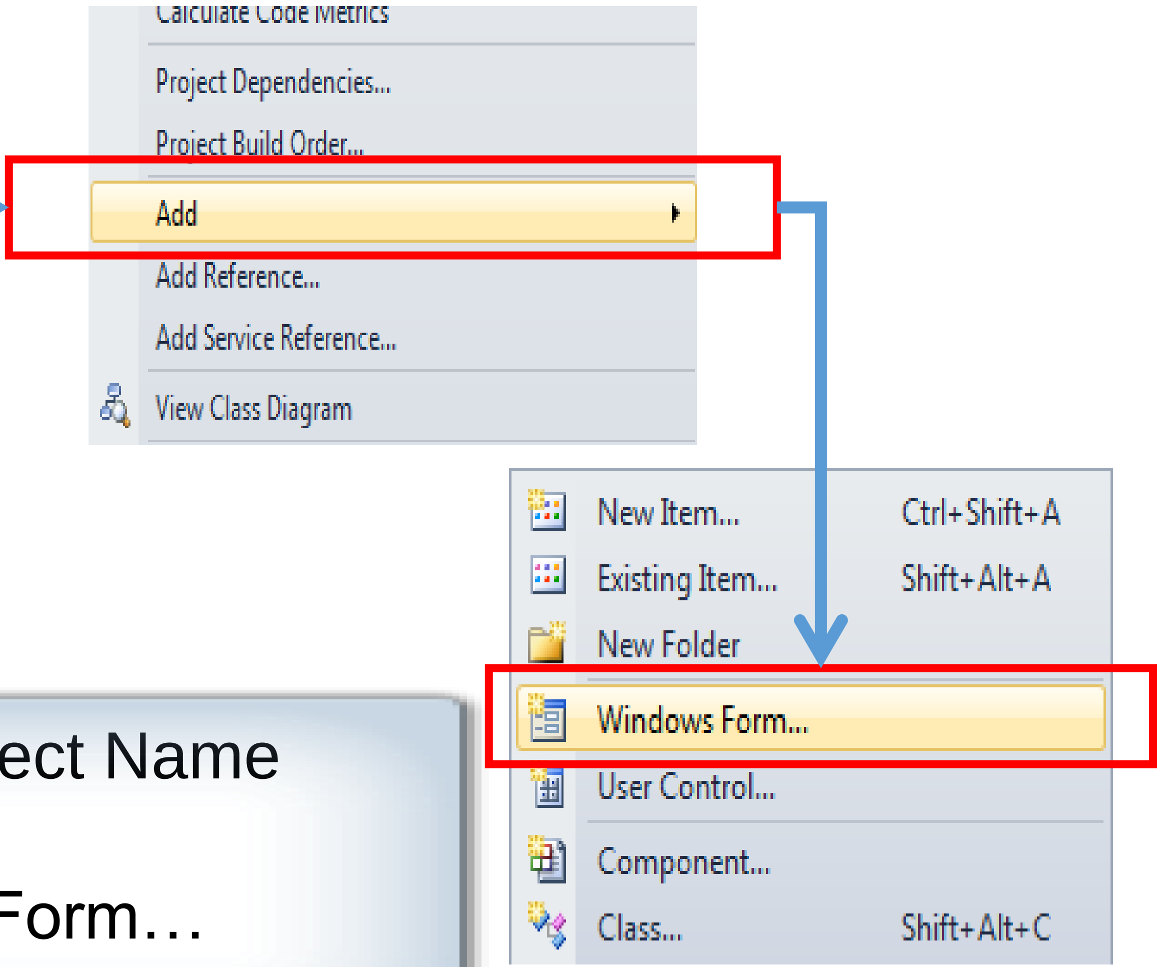
Represents a window or dialog box that makes up an application’s user interface



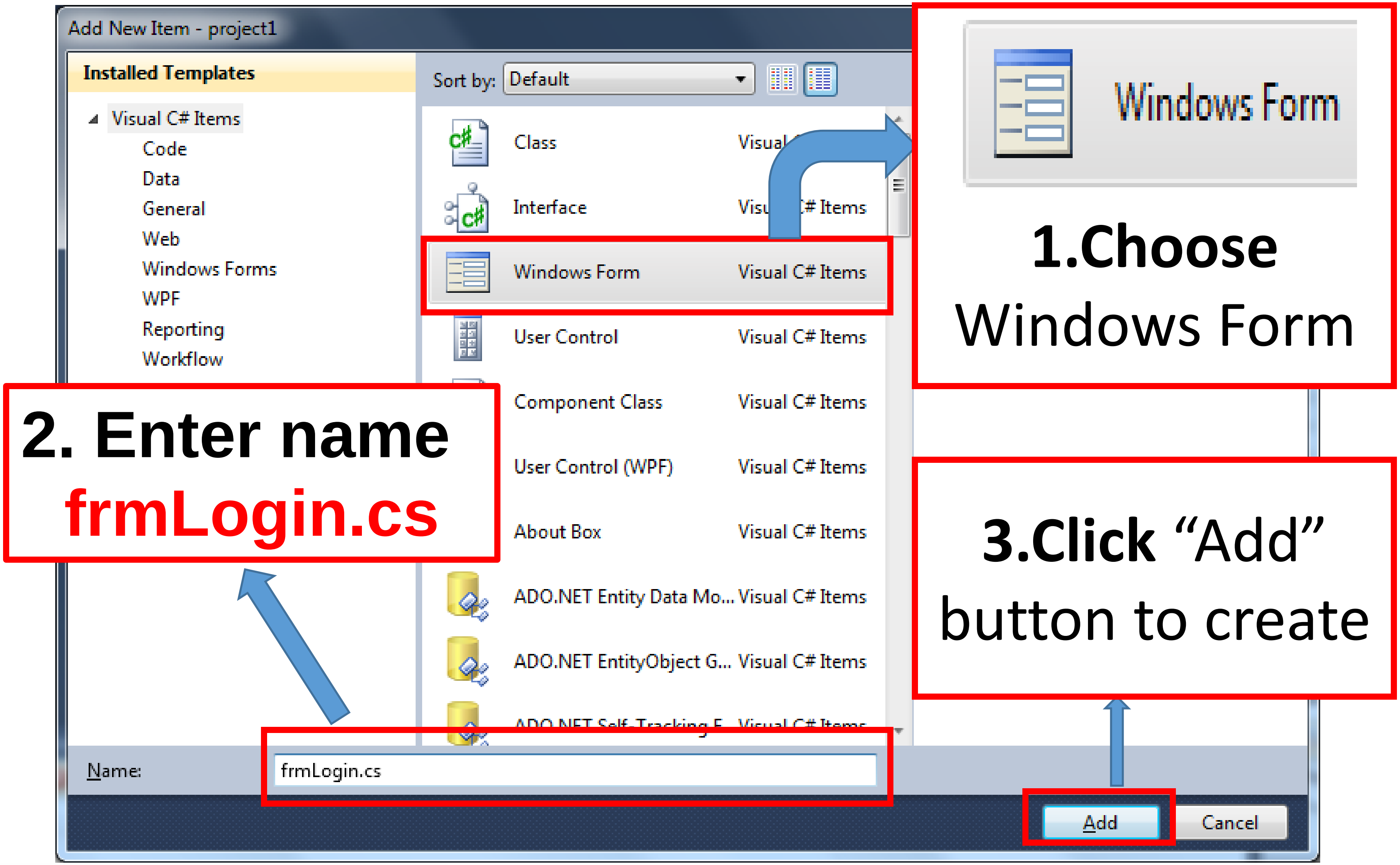
Tab order

Create **Form**





1. Right Click on Project Name
2. Choose Add Item
3. Choose Windows Form…

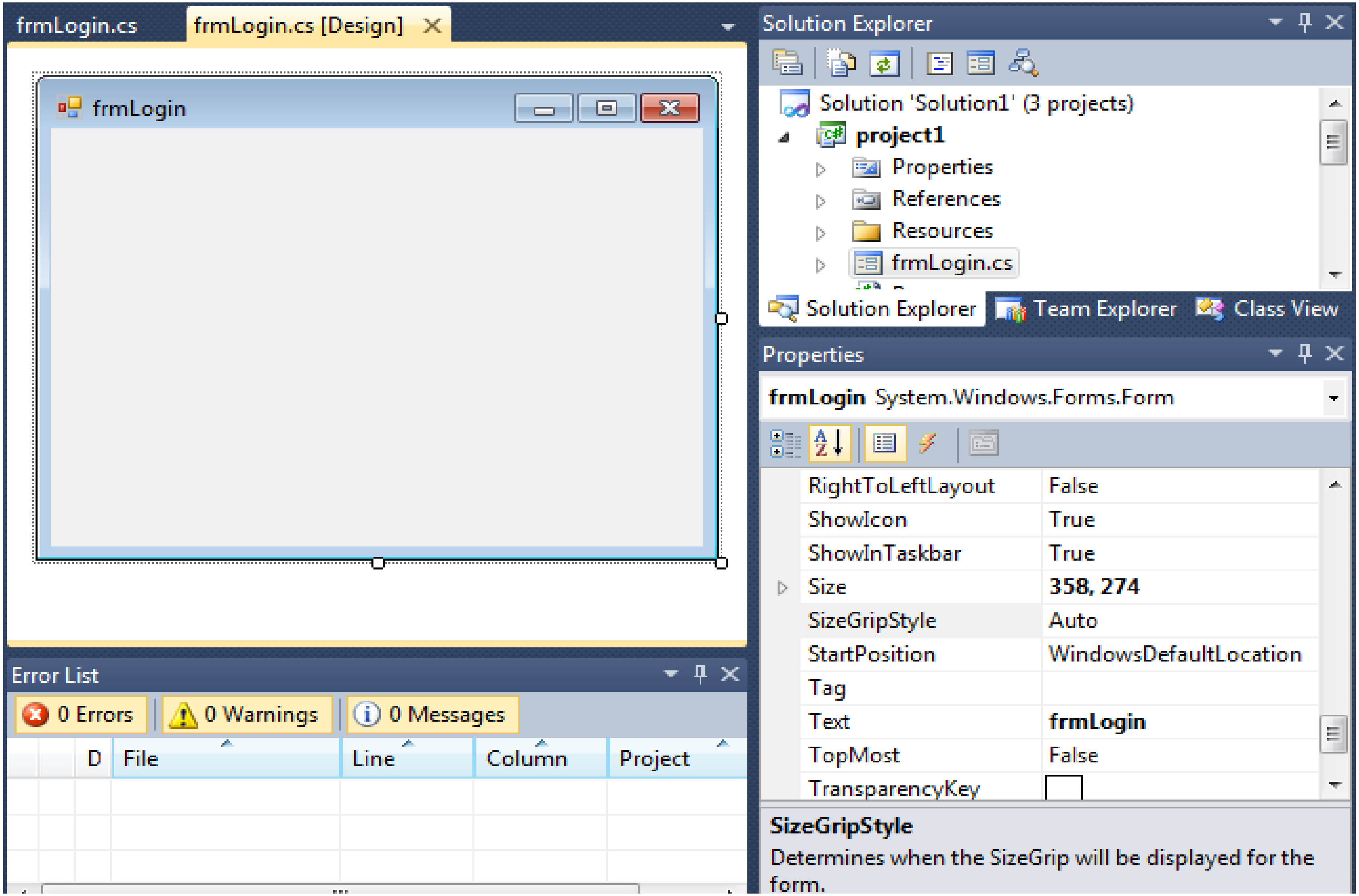


Name: frmLogin.cs

1.Choose Windows Form

2. Enter name **frmLogin.cs**

3.Click "Add" button to create



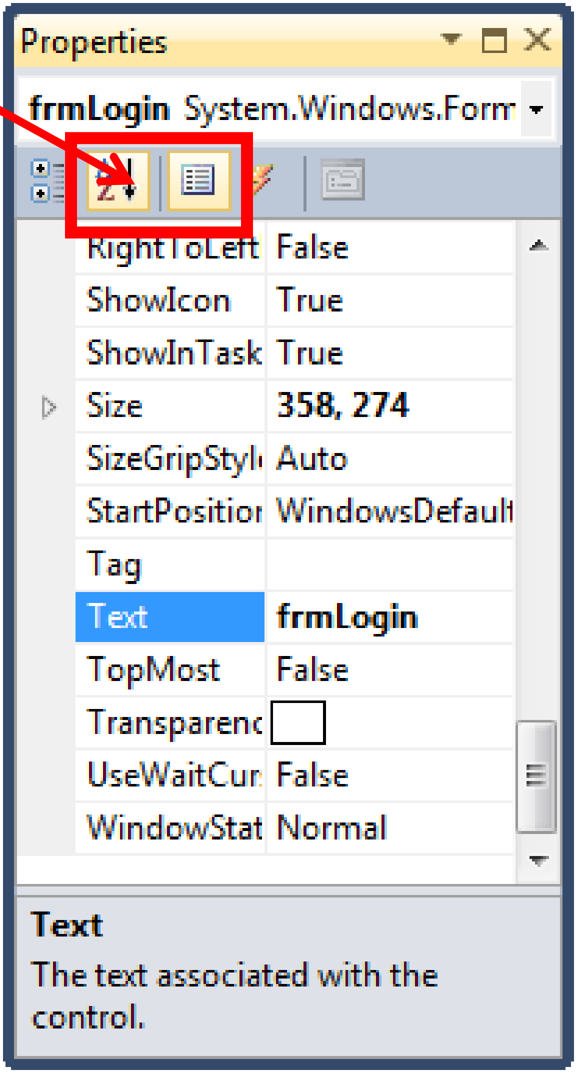
frmLogin System.Windows.Forms.Form

RightToLeftLayout	False
ShowIcon	True
ShowInTaskbar	True
Size	**358, 274**
SizeGripStyle	Auto
StartPosition	WindowsDefaultLocation
Tag	
Text	**frmLogin**
TopMost	False
TransparencyKey	□

SizeGripStyle

Determines when the SizeGrip will be displayed for the form.

Popular **Properties**



Properties

frmLogin System.Windows.Form

RightToLeft	False
ShowIcon	True
ShowInTask	True
Size	358, 274
SizeGripStyl	Auto
StartPositior	WindowsDefault
Tag	
Text	frmLogin
TopMost	False
Transparenc	
UseWaitCur	False
WindowStat	Normal

Text

The text associated with the control.

Name	Description
Text	Gets or sets the text associated with this control
Name	Gets or sets the Name
Size	Gets or sets the size of the form.
WindowState	Gets or sets the form's window state. **Normal; Minimized; Maximized**
Font	Gets or sets the font of the text displayed by the control

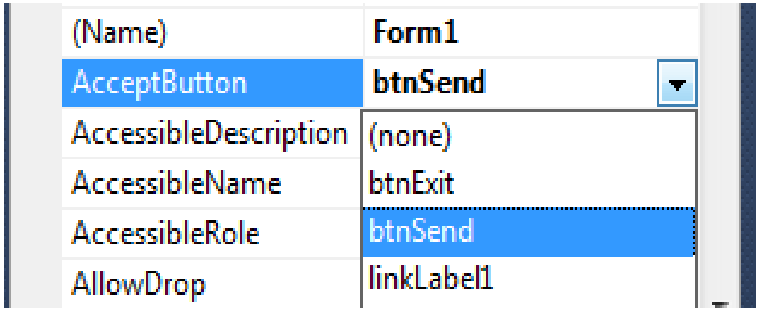
Name	Description
StartPosition	Gets or sets the starting position of the form at run time. **Manual** **CenterScreen** **WindowsDefaultLocation** **WindowsDefaultBounds** **CenterParent**
TopMost	Gets or sets a value indicating whether the form should be displayed as a topmost form.

Name	Description
FormBorderStyle	Gets or sets the border style of the form **None** **FixedSingle** **Fixed3D** **FixedDialog** **Sizable** **FixedToolWindow** **SizableToolWindow**

Name	Description
BackColor	Gets or sets the background color for the control
BackGroundImage	Gets or sets the background image displayed in the control
MainMenuStrip	Gets or sets the primary menu container for the form
ForceColor	Gets or sets the foreground color of the control

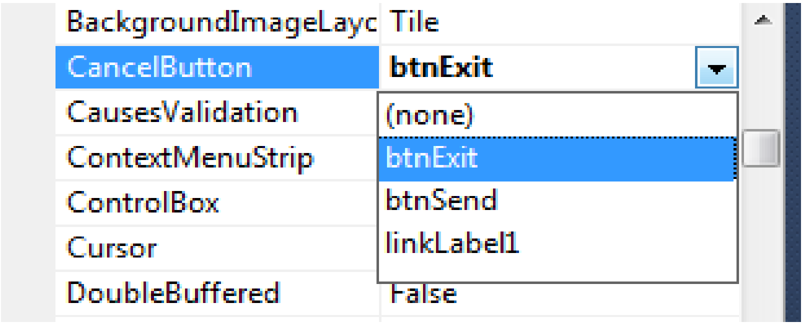
Name	Description
Cursor	Gets or sets the cursor that is displayed when the mouse pointer is over the control
Location	Gets or sets the Point that represents the upper-left corner of the Form in screen coordinates
Icon	Gets or sets the icon for the form.
Opacity	Gets or sets the opacity level of the form

Name	Description
AcceptButton	Gets or sets the button on the form that is clicked when the user presses the ENTER key.
CancelButton	Gets or sets the button control that is clicked when the user presses the ESC key



(Name) Form1
AcceptButton btnSend
AccessibleDescription
AccessibleName
AccessibleRole
AllowDrop

(none)
btnExit
btnSend
linkLabel1



BackgroundImageLayc Tile
CancelButton btnExit
CausesValidation
ContextMenuStrip
ControlBox
Cursor
DoubleBuffered False

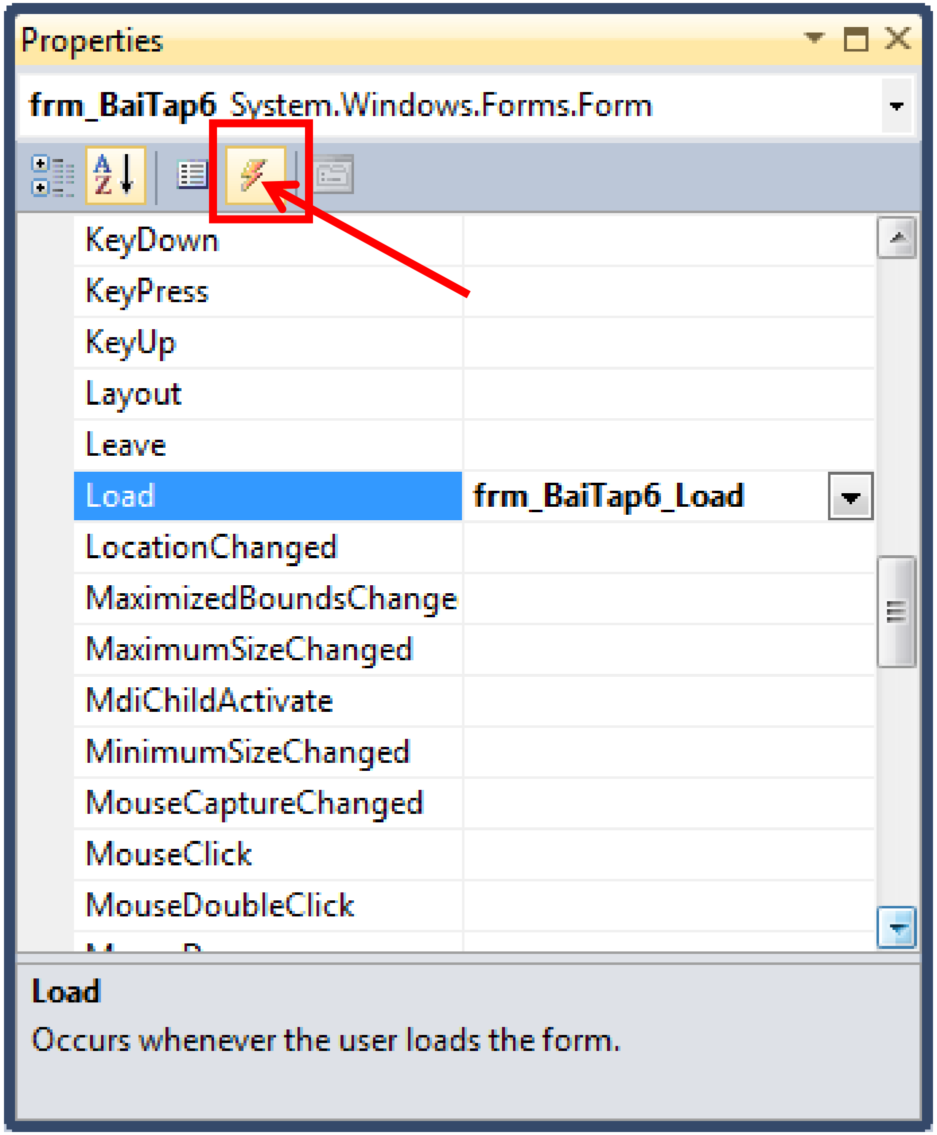
(none)
btnExit
btnSend
linkLabel1

DEMO

Windows Forms

Popular Events

For Windows Forms & Another Controls



Properties

frm_BaiTap6 System.Windows.Forms.Form

KeyDown
KeyPress
KeyUp
Layout
Leave
Load frm_BaiTap6_Load
LocationChanged
MaximizedBoundsChange
MaximumSizeChanged
MdiChildActivate
MinimumSizeChanged
MouseCaptureChanged
MouseClick
MouseDoubleClick

Load

Occurs whenever the user loads the form.

Name	Description
Click	Occurs when the control is clicked. (Inherited from Control.)
DoubleClick	Occurs when the control is double -clicked. (Inherited from Control.)
Load	Occurs before a form is displayed for the first time.
FormClosing	Occurs before the form is closed.
FormClosed	Occurs after the form is closed.

Name	Description
KeyDown	Occurs when a key is pressed while the control has focus. (Inherited from Control.)
KeyPress	Occurs when a key is pressed while the control has focus. (Inherited from Control.)
KeyUp	Occurs when a key is released while the control has focus. (Inherited from Control.)

Name	Description
MouseClick	Occurs when the control is clicked by the mouse. (Inherited from Control.)
MouseDouble Click	Occurs when the control is double clicked by the mouse. (Inherited from Control.)
MouseDown	Occurs when the mouse pointer is over the control and a mouse button is pressed.(Control)

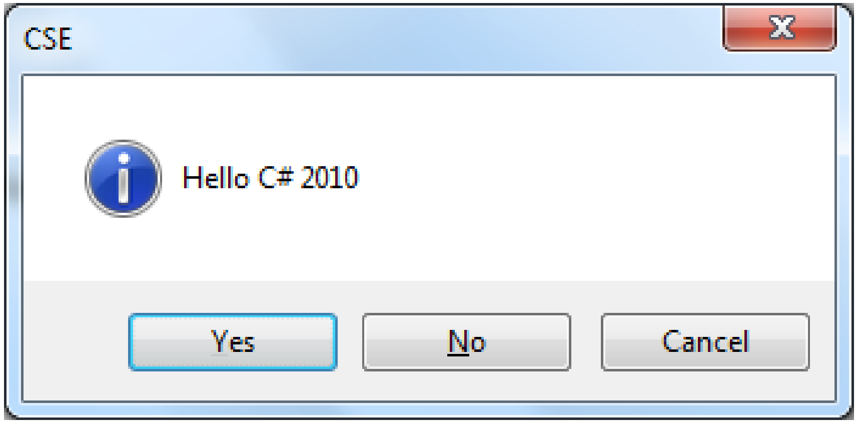
Name	Description
MouseEnter	Occurs when the mouse pointer enters the control.(Control)
MouseHover	Occurs when the mouse pointer rests on the control(Control)
MouseLeave	Occurs when the mouse pointer leaves the control. (Control)
MouseMove	Occurs when the mouse pointer is moved over the control. (Control)

Name	Description
MouseUp	Occurs when the mouse pointer is over the control and a mouse button is released.(Control)
MouseWheel	Occurs when the mouse wheel moves while the control has focus.(Control)

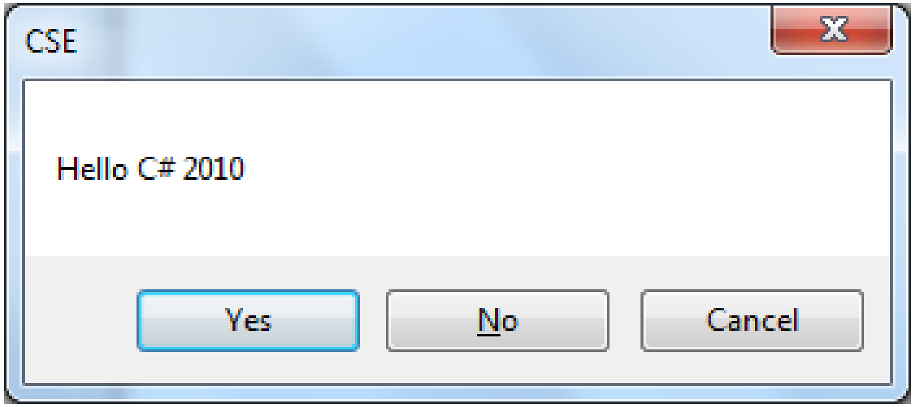
DEMO

Events

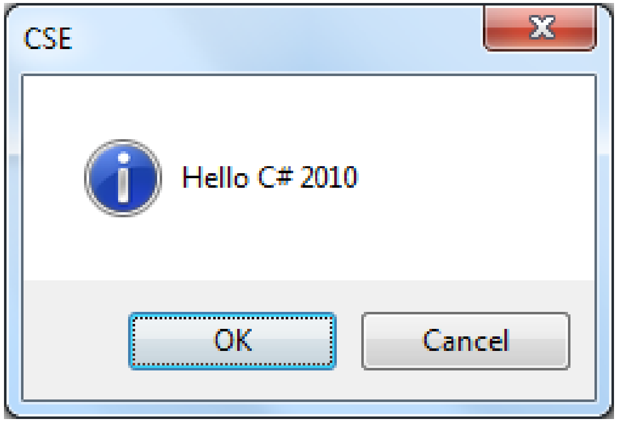
MessageBox class


CSE

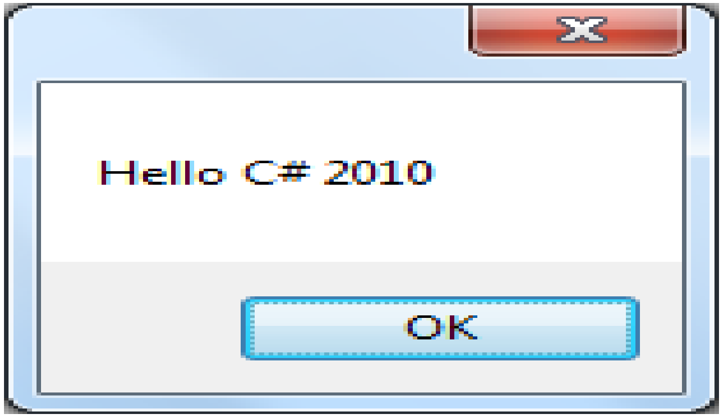
Hello C# 2010


CSE

Hello C# 2010

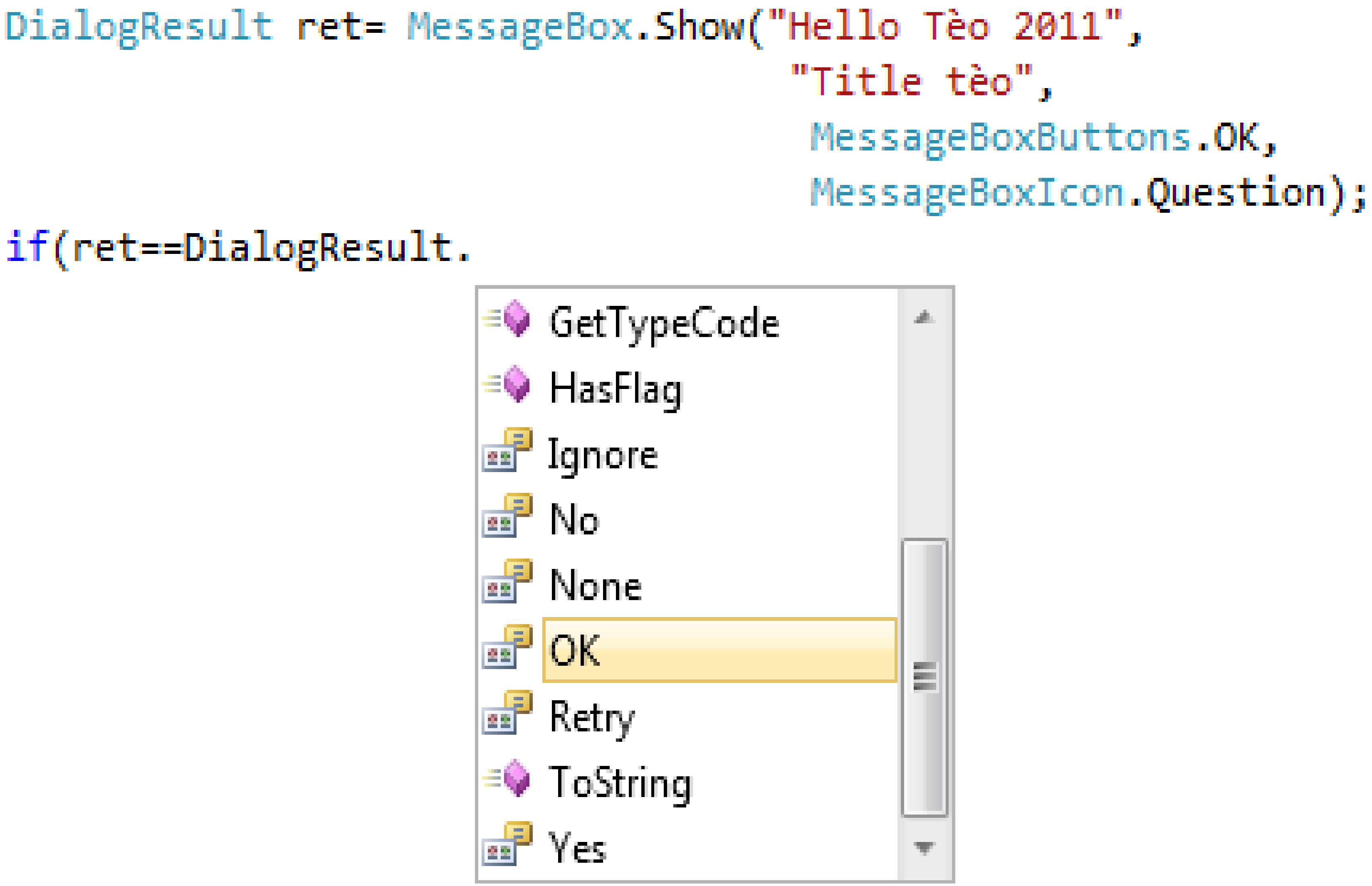

CSE

Hello C# 2010


Hello C# 2010

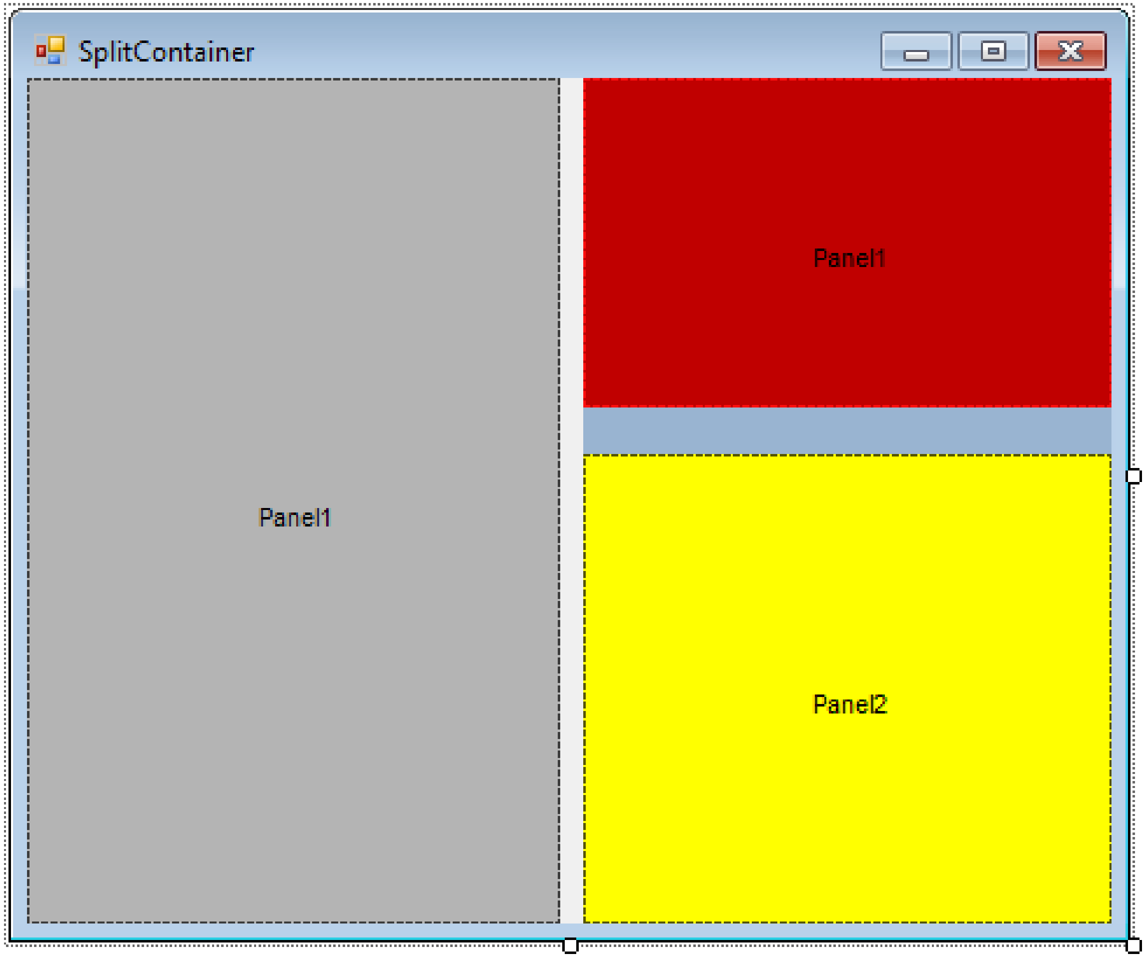
Name	Description
Show(string)	Show a message box with text.
Show(string,string)	Show a message box with text and caption.
Show(string,string, MessageBoxButtons)	Show a message box with text, caption, and buttons.
Show(string,string, MessageBoxButtons, MessageBoxIcon)	Show a message box with text, caption, buttons, and icon.

```
MessageBox.Show("Hello  2017");

MessageBox.Show("Hello 2017","Title");

MessageBox.Show("Hello  2017","Title",
            MessageBoxButtons.YesNoCancel);

MessageBox.Show("Hello  2017","Title",
                        MessageBoxButtons.OK,
                       MessageBoxIcon.Question);
```



```
DialogResult ret= MessageBox.Show("Hello Tèo 2011",
                                  "Title tèo",
                                   MessageBoxButtons.OK,
                                   MessageBoxIcon.Question);
if(ret==DialogResult.
```

Panel & SplitContainer

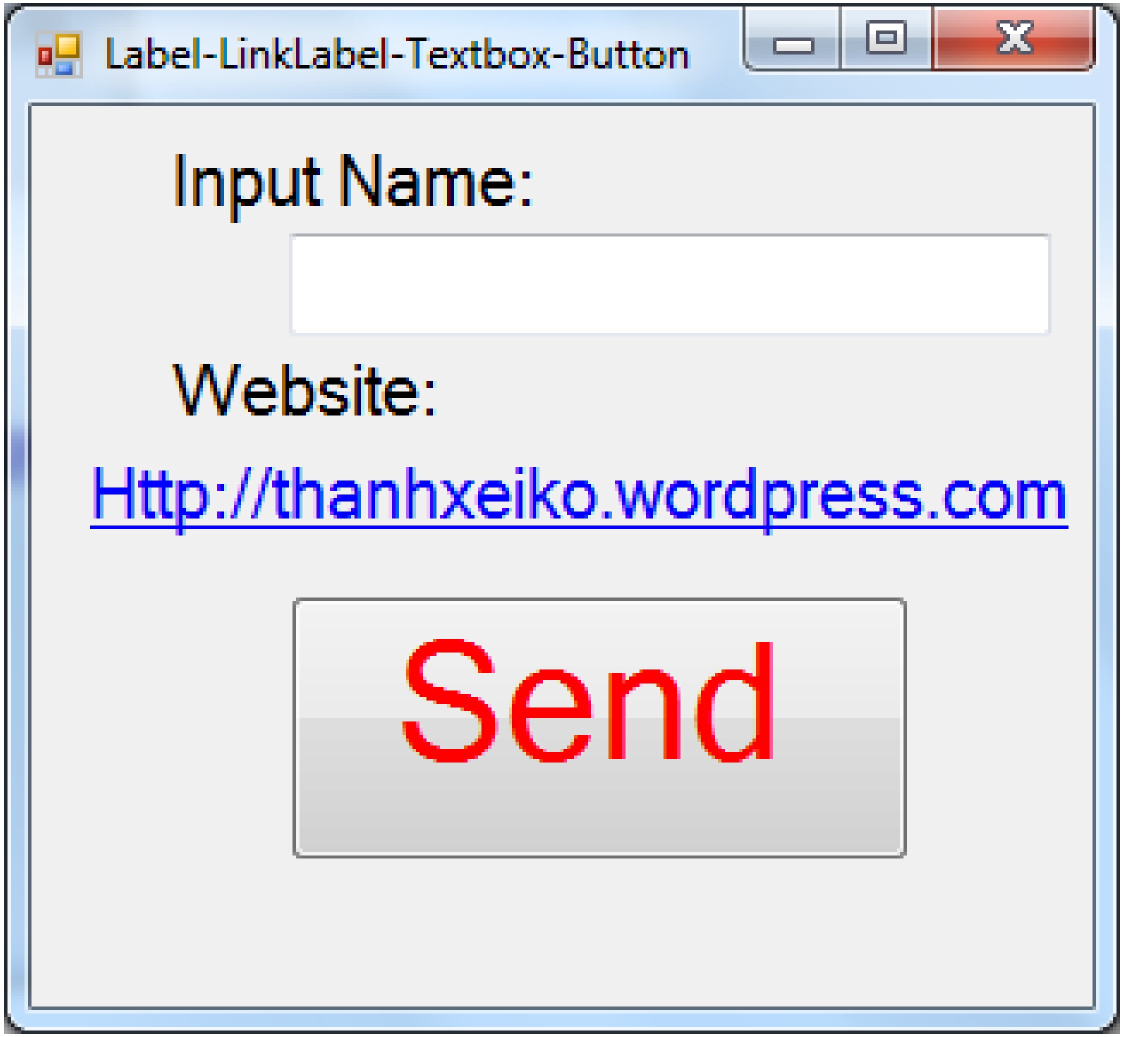


Label

LinkLabel

TextBox

Button



Label-LinkLabel-Textbox-Button

Input Name:

Website:

Http://thanhxeiko.wordpress.com

Send

LinkLabel Events

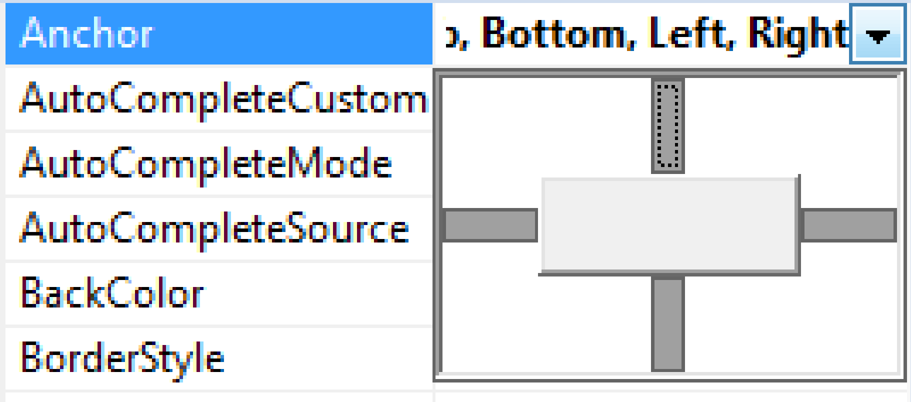
Name	Description
LinkClicked	The LinkClicked event is handled to perform tasks when the user clicks on a link in the control. It is passed an instance of the LinkLabelLinkClickedEventArgs class

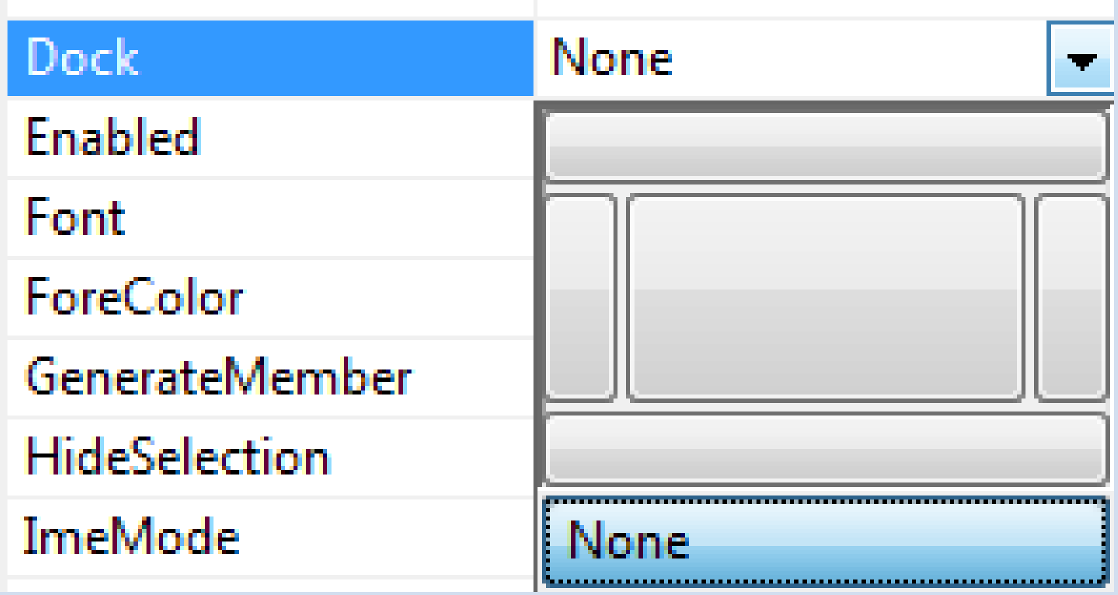
```
private void linkLabel1_LinkClicked(object sender,
                                  LinkLabelLinkClickedEventArgs e)
        { switch (e.Button)
            {
                case MouseButtons.Left:
                  System.Diagnostics.Process.Start(linkLabel1.Text);
                    break;
                case MouseButtons.Right:
                        MessageBox.Show("Right Click");
                    break;
                case MouseButtons.Middle:
                    MessageBox.Show("Middle Click", "Title",
                                  MessageBoxButtons.YesNoCancel,
                                  MessageBoxIcon.Information);
                    break;
            }
        }
```

How add LinkLabel control & event at Runtime?

```
private LinkLabel linkTeo;
private void Form1_Load(object sender, EventArgs e)
{ //Create the linklabel
 linkTeo = new LinkLabel();
  //set location for linklabel
linkTeo.Location =new  System.Drawing.Point(100, 100);
 //Set the text for linklabel
 linkTeo.Text= "http://thanhxeiko.wordpress.com";
 //Add the controls to the form
 this.Controls.Add(linkTeo);
 //Add an event at runtime
 linkTeo.LinkClicked += linkTeo_Clicked;
}
private void linkTeo_Clicked(object sender,
            LinkLabelLinkClickedEventArgs e)
{  if (sender is LinkLabel)
      MessageBox.Show(((LinkLabel)sender).Text);
  MessageBox.Show("Add event at runtime");
}
```

TextBox – Properties - Events

Name	Description
Anchor	Gets or sets the edges of the container to which a control is bound and determines how a control is resized with its parent **Top** **Left** **Right** **Bottom**  Anchor ›, Bottom, Left, Right AutoCompleteCustom AutoCompleteMode AutoCompleteSource BackColor BorderStyle

Name	Description
Dock	Gets or sets which control borders are docked to its parent control and determines how a control is resized with its parent **None** **Top** **Left** **Right** **Bottom** **Fill**  Dock None Enabled Font ForeColor GenerateMember HideSelection ImeMode None

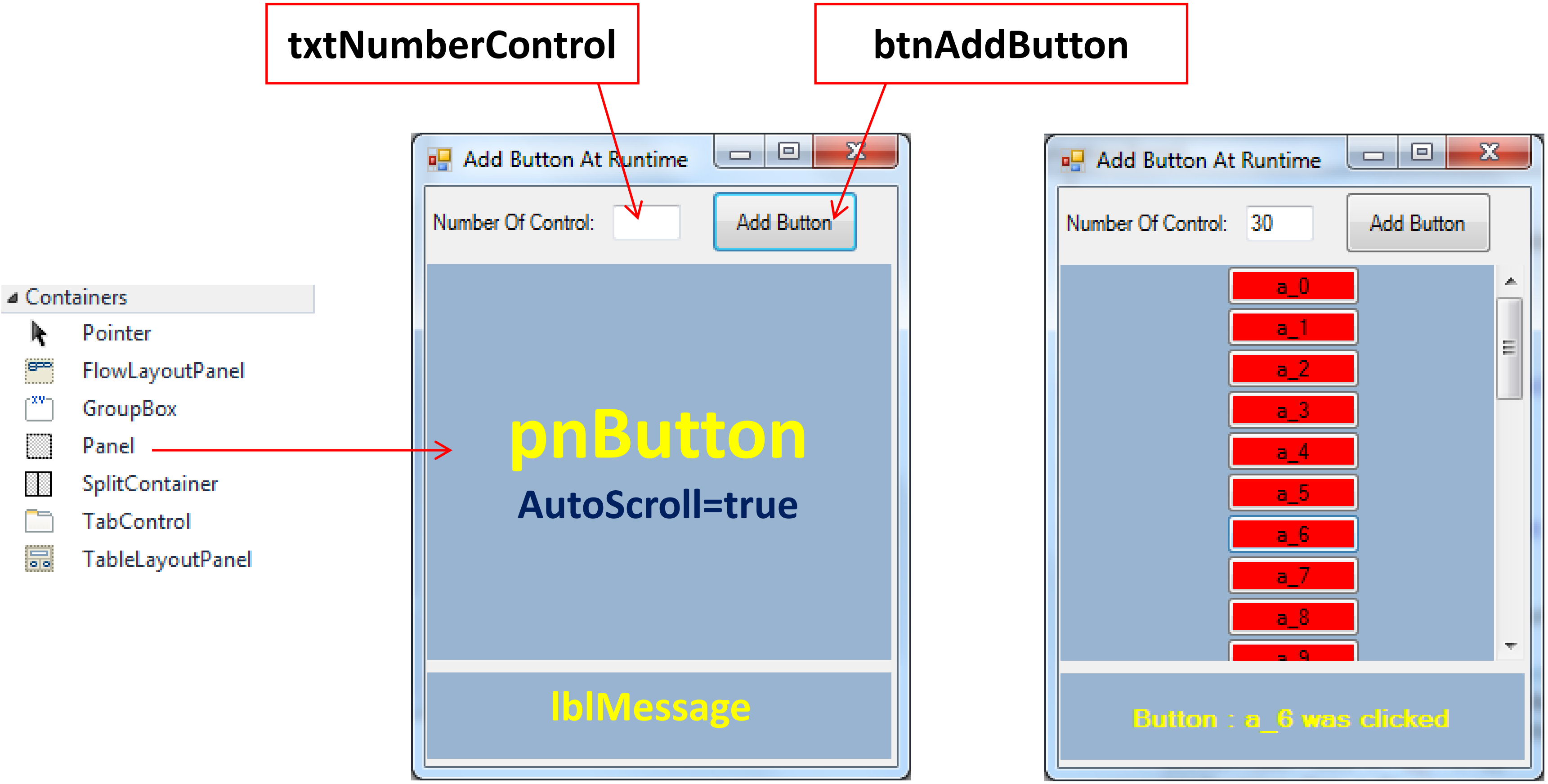
Name	Description
Multiline	Gets or sets a value indicating whether this is a
PasswordChar	Gets or sets the character used to mask characters of a password in a single-line TextBox control
ReadOnly	Gets or sets a value indicating whether text in the text box is read-only.

Name	Description
UseSystemPasswordChar	Gets or sets a value indicating whether the text in the TextBox control should appear as the default password character
Visible	Gets or sets a value indicating whether the control and all its child controls are displayed.
Enabled	Gets or sets a value indicating whether the control can respond to user interaction.

Name	Description
TextChanged	Occurs when the **Text** property value changes.

```
 private void txtName_TextChanged
(object sender, EventArgs e)
  {
     this.Text = txtName.Text;
  }
```

Add Button & Event at Runtime



txtNumberControl

btnAddButton

pnButton

AutoScroll=true

lblMessage

Number Of Control: 30

Button : a_6 was clicked

```
private void btnAddButton_Click(object sender, EventArgs e)
{
 pnButton.Controls.Clear();
 for (int i = 0; i < Int32.Parse(txtNumberControl.Text); i++)
  {Button btnRuntime = new Button();
    btnRuntime.BackColor = Color.Red;
    btnRuntime.Location = new System.Drawing.Point
             (pnButton.Width/2-btnRuntime.Width/2,
              i * btnRuntime.Height);
               btnRuntime.Text = "a_" + i;
               btnRuntime.Tag = i;
               btnRuntime.Click += btnRuntime_click;
               pnButton.Controls.Add(btnRuntime);}
}
private void btnRuntime_click(object sender, EventArgs e)
{
 Button btn = (Button)sender;
 lblMessage.Text = "Button : "+btn.Text +" was clicked";
}
```

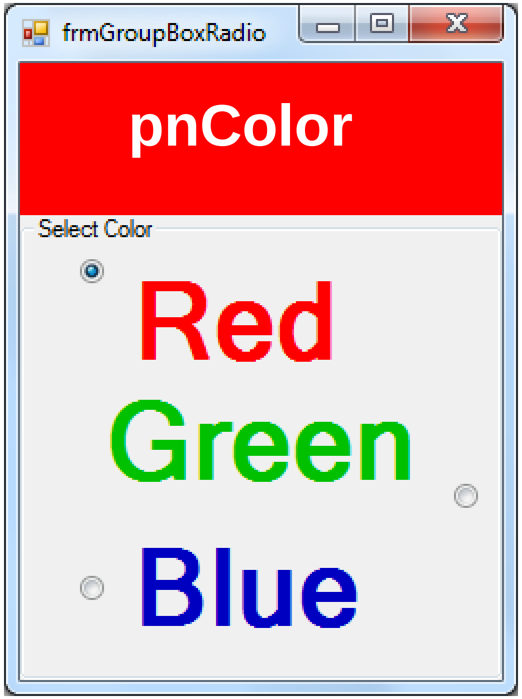
GroupBox & RadioButton

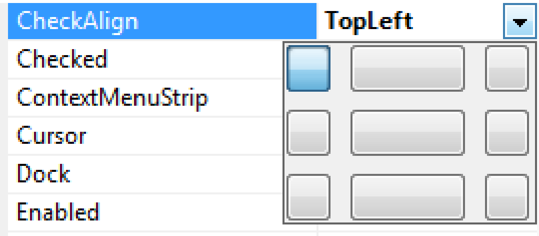
Name	Description
Checked	Gets or sets a value indicating whether the control is checked
CheckAlign	Gets or sets the location of the check box portion of the RadioButton

```
if (radRed.Checked == true)
{
 //do something for Red
}
```

```
radBlue.CheckAlign =
ContentAlignment.MiddleLeft;
```

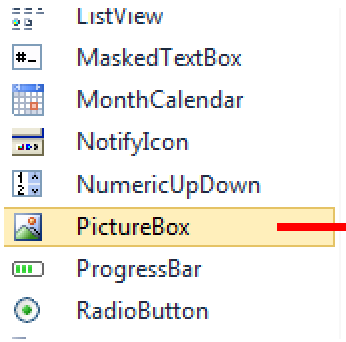
GroupBox & RadioButton

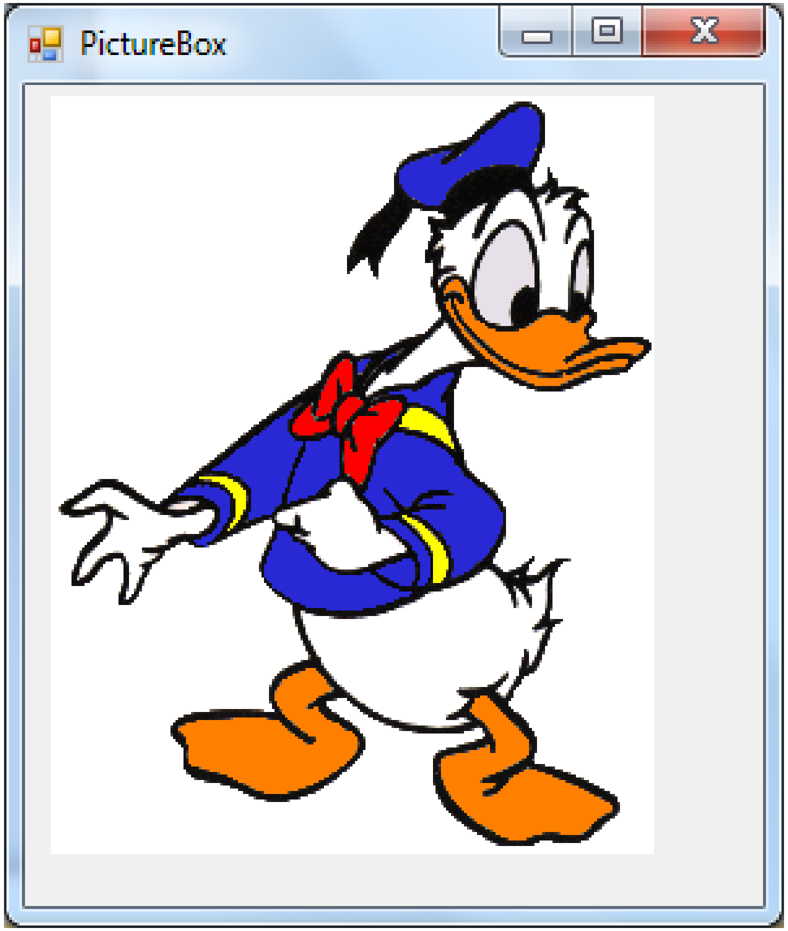



CheckAlign TopLeft
Checked
ContextMenuStrip
Cursor
Dock
Enabled

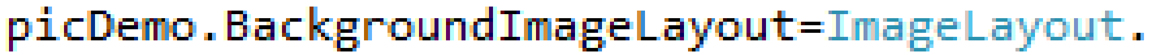
```
private void frmGroupBoxRadio_Load
(object sender, EventArgs e)
{
   radRed.CheckedChanged +=
        rad_CheckedChanged;
   radGreen.CheckedChanged +=
        rad_CheckedChanged;
   radBlue.CheckedChanged +=
        rad_CheckedChanged;
}

private void rad_CheckedChanged
(object sender, EventArgs e)
{
   RadioButton rad = (RadioButton)sender;
   pnColor.BackColor = rad.ForeColor;
}
```

PictureBox control

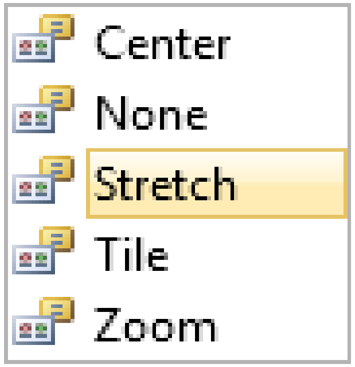




Name	Description
BackgroundImage	Gets or sets the background image displayed in the control
BackgroundImage Layout	Gets or sets the background image layout as defined in the ImageLayout enumeration

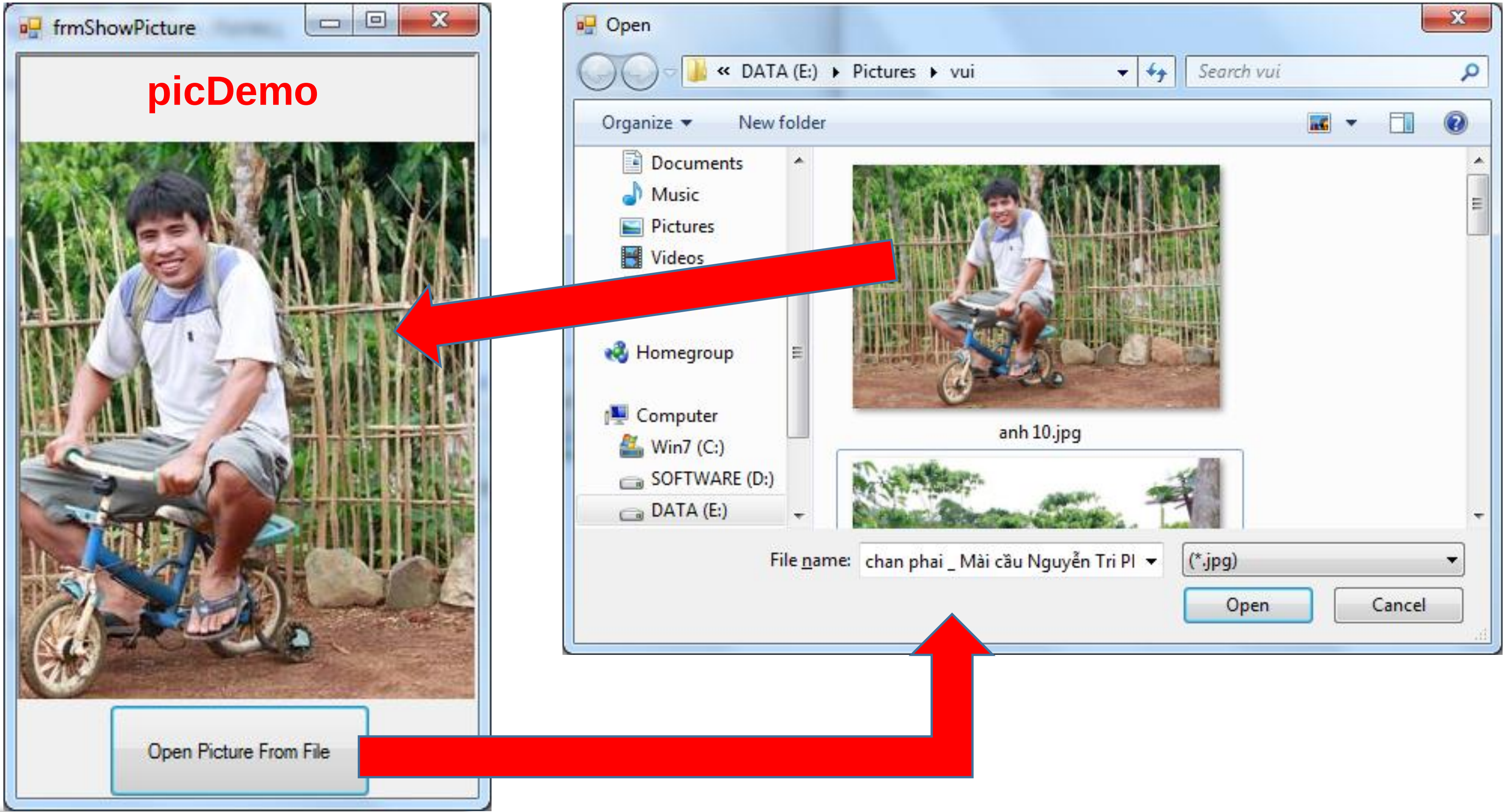


```
picDemo.BackgroundImageLayout=ImageLayout.
```



- Center
- None
- Stretch
- Tile
- Zoom

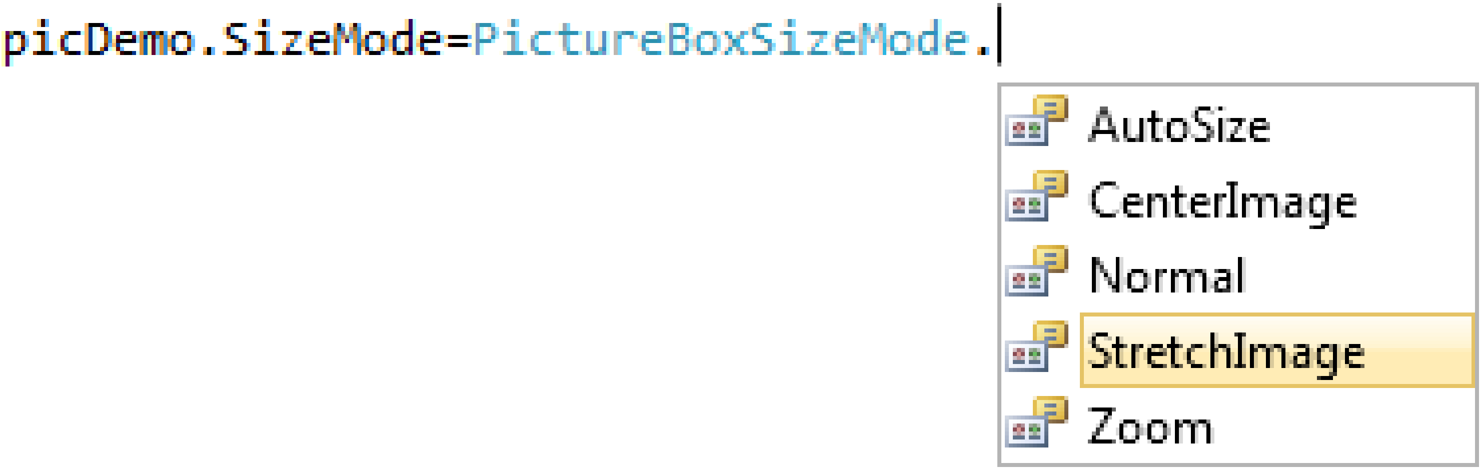
Get Picture from OpenFileDialog



File name: chan phai _ Mài cầu Nguyễn Tri Pl

(*.jpg)

Name	Description
Image	Gets or sets the image that is displayed by PictureBox
SizeMode	Indicates how the image is displayed.

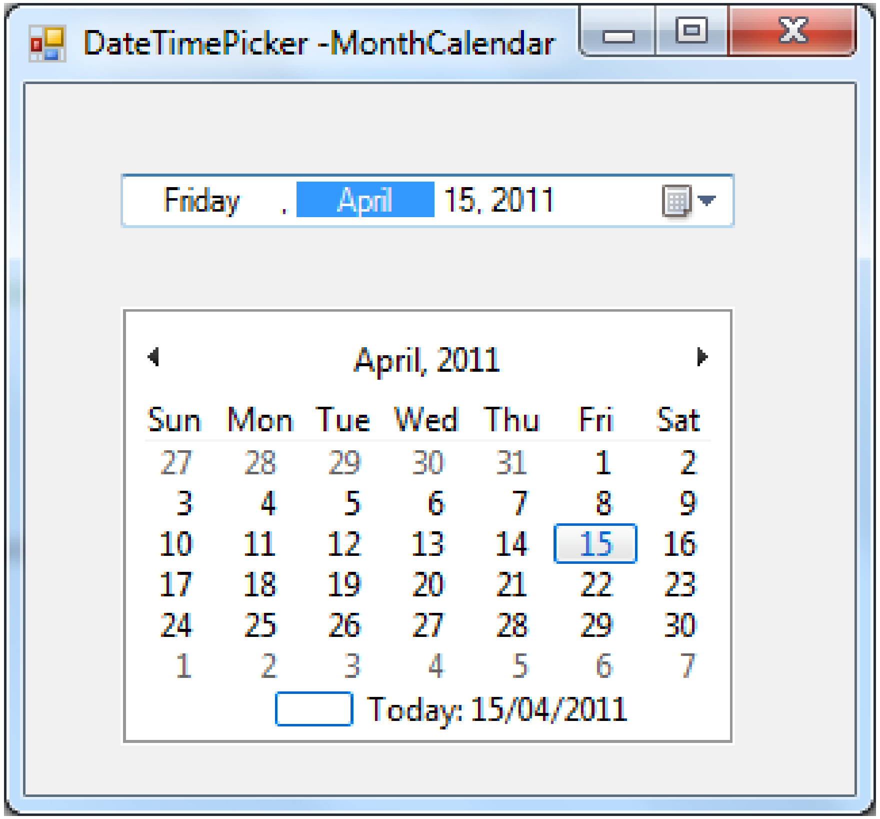


```
picDemo.SizeMode=PictureBoxSizeMode.
```

```
private void btnOpenPic_Click
(object sender, EventArgs e)
 {
 OpenFileDialog fileOpenDlg = new OpenFileDialog();
 fileOpenDlg.Filter = "(*.jpg)|*.jpg|(*.doc)|*.doc";
 if (fileOpenDlg.ShowDialog() == DialogResult.OK)
 {
  //picDemo.Image = new Bitmap(fileOpenDlg.OpenFile());
  //Or
  picDemo.Image = Image.FromFile(fileOpenDlg.FileName);
  //Or
  //picDemo.BackgroundImage = new Bitmap(fileOpenDlg.OpenFile());
 }

}
```

Bài tập áp dụng

DateTimePicker & MonthCalendar



DateTimePicker -MonthCalendar

Friday , April 15, 2011

April, 2011

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

Today: 15/04/2011

Name	Description
Format	Gets or sets the format of the date and time displayed in the control
CustomFormat	Gets or sets the custom date/time format string
Value	Gets or sets the date/time value assigned to the control.

```
dateTimePicker1.Format = DateTimePickerFormat.Custom;
dateTimePicker1.CustomFormat = "MMMM dd, yyyy - dddd
```

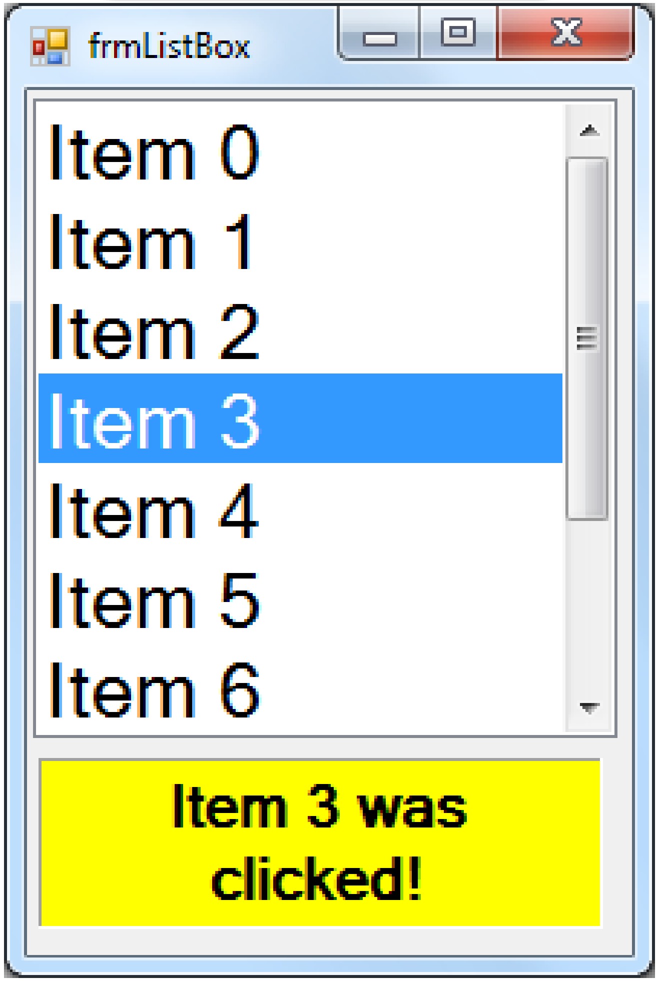
Next Slide to see list Custom Format

Format string	Description
d	The one- or two-digit day.
dd	The two-digit day. Single-digit day values are preceded by a 0.
ddd	The three-character day-of-week abbreviation.
dddd	The full day-of-week name.
h	The one- or two-digit hour in 12-hour format.
hh	The two-digit hour in 12-hour format. Single digit values are preceded by a 0.
H	The one- or two-digit hour in 24-hour format.

Format string	Description
HH	The two-digit hour in 24-hour format. Single digit values are preceded by a 0.
m	The one- or two-digit minute.
mm	The two-digit minute. Single digit values are preceded by a 0.
M	The one- or two-digit month number.
mm	The two-digit minute. Single digit values are preceded by a 0.
M	The one- or two-digit month number.
MM	The two-digit month number. Single digit values are preceded by a 0.
MMM	The three-character month abbreviation.

Format string	Description
MMMM	The full month name.
s	The one- or two-digit seconds.
ss	The two-digit seconds. Single digit values are preceded by a 0.
t	The one-letter A.M./P.M. abbreviation (A.M. is displayed as "A").
tt	The two-letter A.M./P.M. abbreviation (A.M. is displayed as "AM").
y	The one-digit year (2001 is displayed as "1").
yy	The last two digits of the year (2001 is displayed as "01").
yyyy	The full year (2001 is displayed as "2001").

ListBox



Item 3 was clicked!

```
private void
frmListBox_Load(object sender,
EventArgs e)
{ listBox1.Items.Clear();
 for (int i = 0; i < 10; i++)
listBox1.Items.Add("Item " + i);
}
private void
listBox1_SelectedIndexChanged
(object sender, EventArgs e)
{
lblMessage.Text = listBox1.Text
 +" was clicked!";
}
```

And We can use AddRange method to add data:

```
private void frmListBox_Load(object
sender, EventArgs e)
 {
string[] strArr = new string[] {
“Tèo",“Tí","Bin","Bo"};
listBox1.Items.AddRange(strArr);
}
```

```
public class CStudent
    {   private string m_strID;
        private string m_strName;
        public CStudent(string strID,
                             string strName)
        { this.m_strID = strID;
          this.m_strName = strName;
        }
        public string ID
        { get { return this.m_strID; }
          set { this.m_strID = value; }
        }
        public string Name
        { get { return this.m_strName; }
          set { this.m_strName = value; }
        }
    }
```

```
using System.Collections;
```

Also We can use DataSource to display data

```
ArrayList arr = new ArrayList();
for(int i=0;i<10;i++)
{arr.Add(new CStudent("ID_"+i,"Name
"+i));
}
listBox1.DataSource = arr;
listBox1.ValueMember = "ID";
listBox1.DisplayMember = "Name";
```

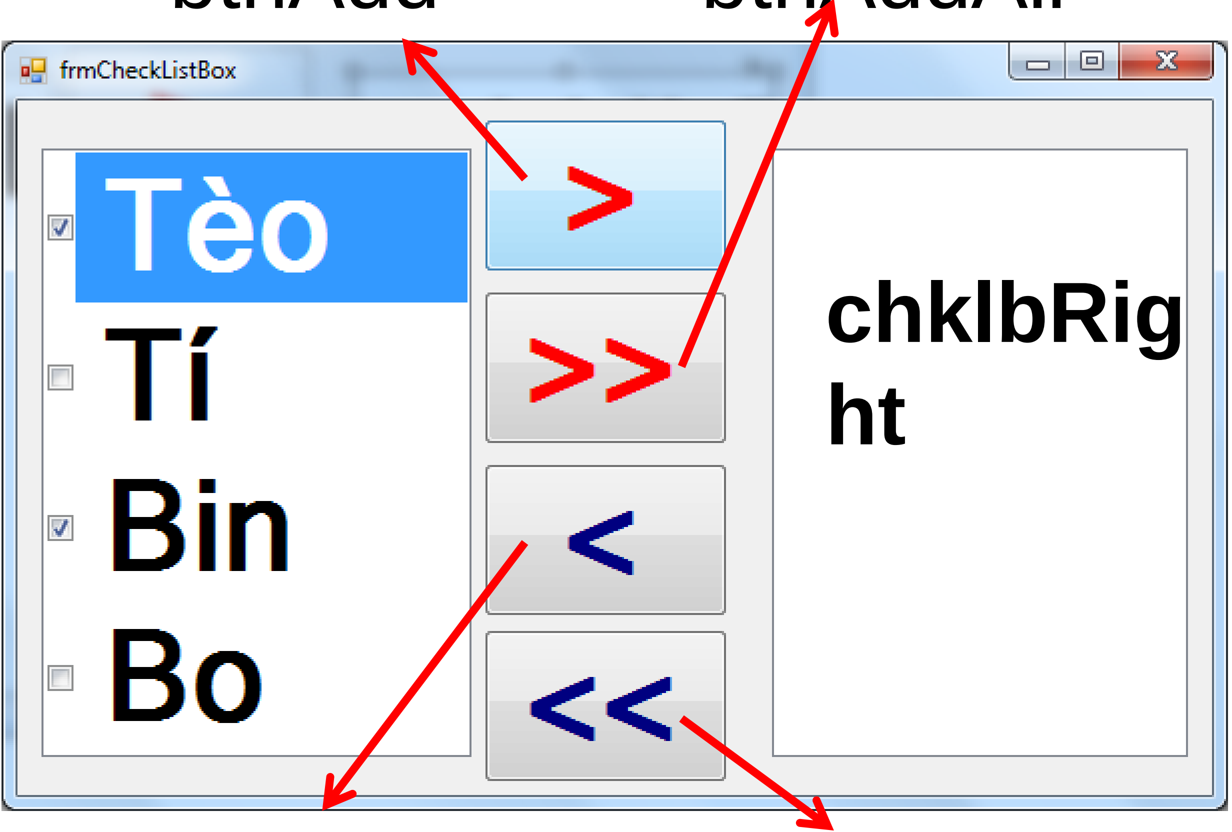
To get object from Listbox, We could use coding below:

```
if (listBox1.SelectedItem != null)
  {
   CStudent svTeo = (CStudent )
                listBox1.SelectedItem;
   lblMessage.Text = aStudent.Name
          +  " Was selected";
  }
```

CheckedListBox

btnAdd btnAddAll

chklbLeft



frmCheckListBox

☑ Tèo

□ Tí

☑ Bin

□ Bo

chklbRight

btnRemove btnRemoveAll

Use Items to add data

```
private void frmCheckListBox_Load
(object sender, EventArgs e)
  { chklbLeft.Items.Add("Tèo");
     chklbLeft.Items.Add("Tí");
     chklbLeft.Items.Add("Bin");
     chklbLeft.Items.Add("Bo");
  }
```

Or we could use **AddRange**

```
chklbLeft.Items.AddRange(new string[] {
"Tèo","Tí","Bin","Bo"});
```

How to process Items Checked???

Case 1:

```
CheckedListBox.CheckedIndexCollection
indexCollection = chklbLeft.CheckedIndices;
string strChecked = "";
foreach (int i in indexCollection)
{
strChecked += i + ";";
}
MessageBox.Show(strChecked);
```

How to process Items Checked???

Case 2:

```
CheckedListBox.CheckedItemCollection items
= chklbLeft.CheckedItems;
string strChecked = "";
foreach (string s in items)
{
strChecked += s + ";";
}
 MessageBox.Show(strChecked);
```

How to process Items Checked???

Case 3:

```
string strChecked = "";
for (int i = 0; i <
chklbLeft.Items.Count; i++){
  if (chklbLeft.GetItemChecked(i))
   {
    //Process Item checked here
   }
}
```

Go back ChecklistBox Example:

```
private void btnAdd_Click
(object sender, EventArgs e)
{
 foreach(int i in chklbLeft.CheckedIndices)
 {
  chklbRight.Items.Add(chklbLeft.Items[i]);
 }
 foreach (string s in chklbRight.Items)
 {chklbLeft.Items.Remove(s);}
}
```

```
private void btnAddAll_Click
(object sender, EventArgs e)
{               chklbRight.Items.AddRange
                (chklbLeft.Items);
     chklbLeft.Items.Clear();
}
```

```
private void btnRemove_Click
(object sender, EventArgs e)
{
foreach (string s in
     chklbRight.CheckedItems)
          chklbLeft.Items.Add(s);
foreach(string s in     chklbLeft.Items)
          chklbRight.Items.Remove(s);
}
```

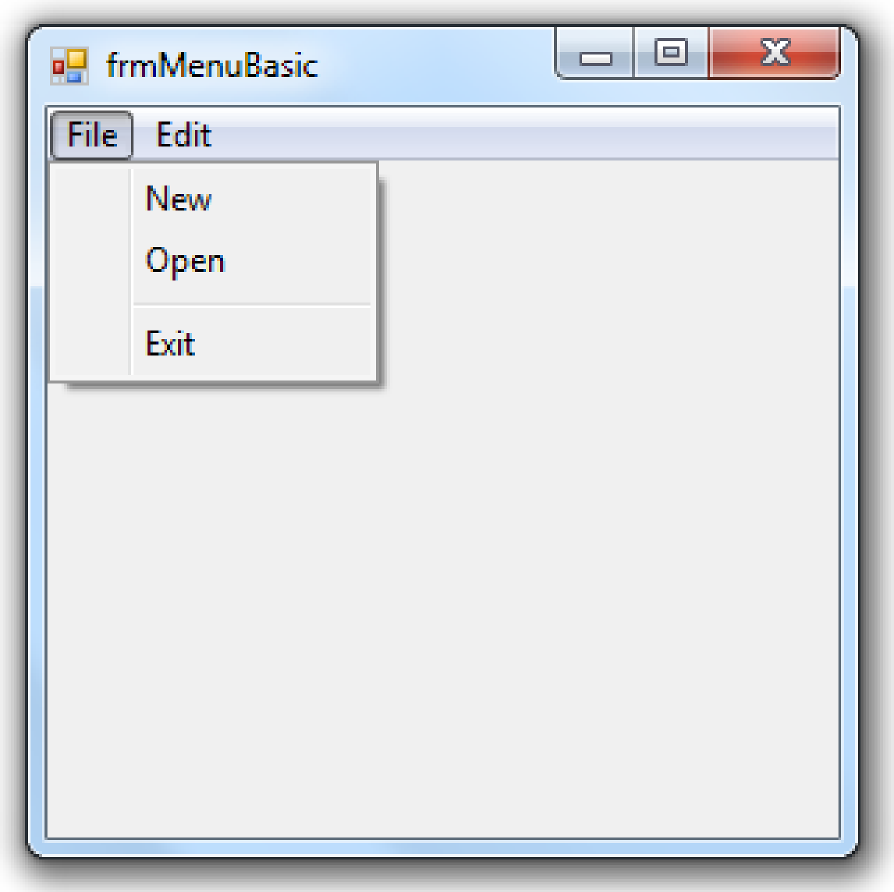
```
private void btnRemoveAll_Click
(object sender, EventArgs e)
{
 chklbLeft.Items.AddRange
          (chklbRight.Items);

 chklbRight.Items.Clear();
}
```

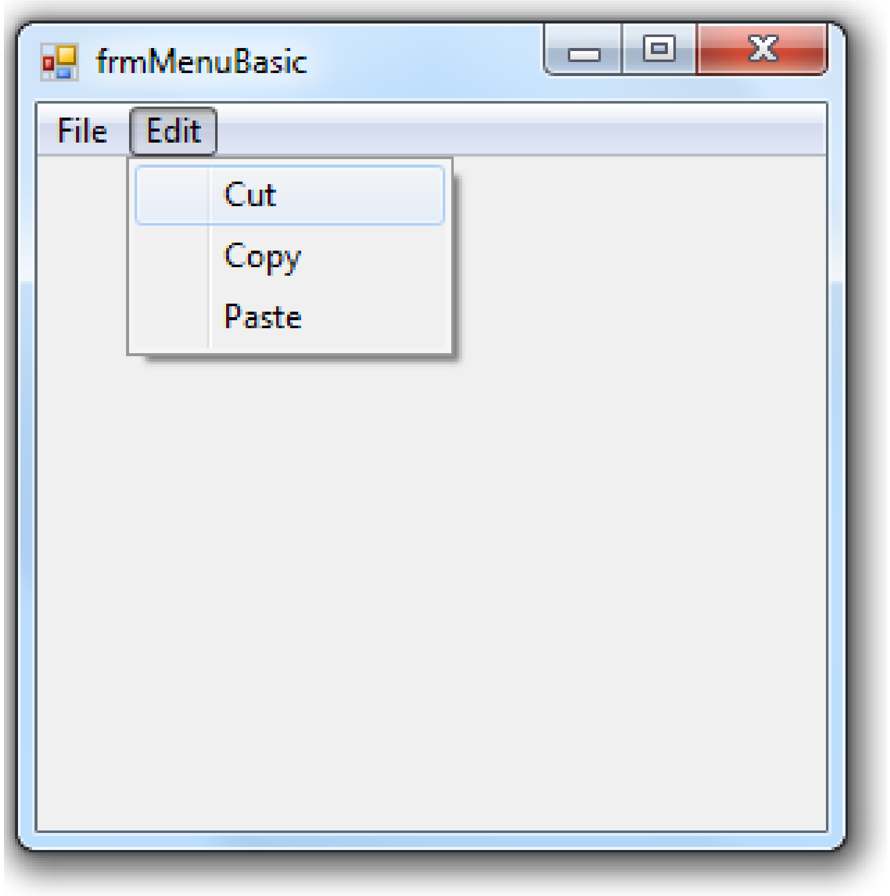
Bài tập áp dụng

Menu (2 ways to use)

MainMenu	MenuStrip
MenuItem	ToolStripMenuItem
Menu Property	MainMenuStrip Property

Coding to create Menu at Runtime





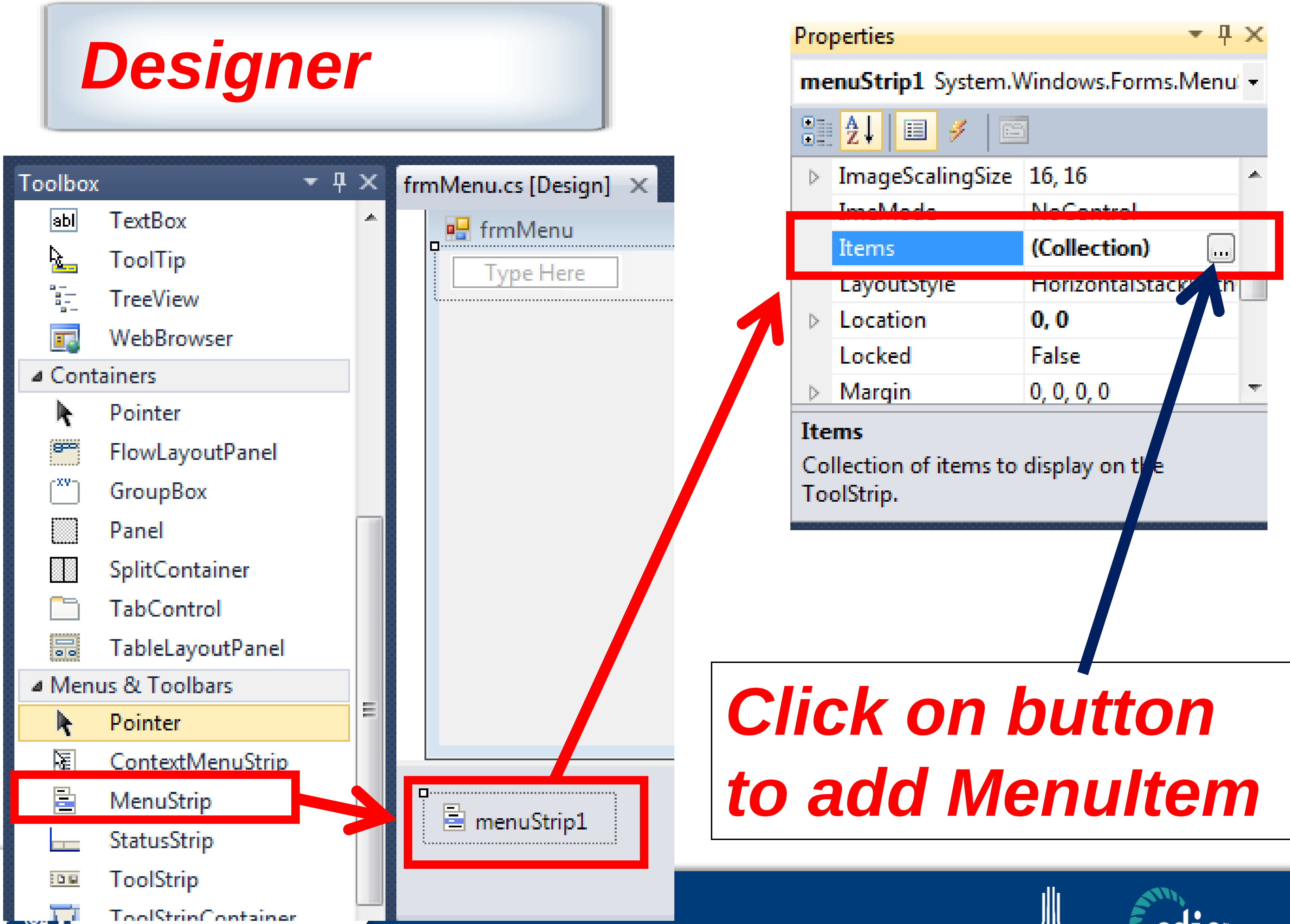
```
private MainMenu mainMenuBar;
private MenuItem menuFile, menuEdit,
menuFileNew, menuFileOpen, menuFileExit,
menuEditCut, menuEditCopy, menuEditPaste;
private void createMenu()
{
mainMenuBar = new MainMenu();
this.Menu = mainMenuBar;
menuFile=new MenuItem("File");
menuFileNew = new MenuItem("New");
menuFileOpen = new MenuItem("Open");
menuFileExit = new MenuItem("Exit");
menuFile.MenuItems.Add(menuFileNew);
menuFile.MenuItems.Add(menuFileOpen);
```

```
private void createMenu(){……
menuFile.MenuItems.Add("-");
menuFile.MenuItems.Add(menuFileExit);
mainMenuBar.MenuItems.Add(menuFile);
menuEdit = new MenuItem("Edit");
menuEditCut = new MenuItem("Cut");
menuEditCopy = new MenuItem("Copy");
menuEditPaste = new MenuItem("Paste");
menuEdit.MenuItems.AddRange(new MenuItem[] {
menuEditCut,menuEditCopy,menuEditPaste});
mainMenuBar.MenuItems.Add(menuEdit);
attachEvents();}
```

```
private void attachEvents()
 {
menuFileNew.Click    +=  process_MenuClick;
menuFileOpen.Click  +=  process_MenuClick;
menuFileExit.Click     +=  process_MenuClick;
menuEditCut.Click     +=  process_MenuClick;
menuEditCopy.Click   +=  process_MenuClick;
menuEditPaste.Click  +=  process_MenuClick;
}
```

```
private void process_MenuClick
(object sender, EventArgs e)
{
 if (sender.Equals(menuFileExit))
 {
  Application.Exit();
 }
}

private void frmMenuBasic_Load
(object sender, EventArgs e)
 {createMenu();
 }
```

Designer

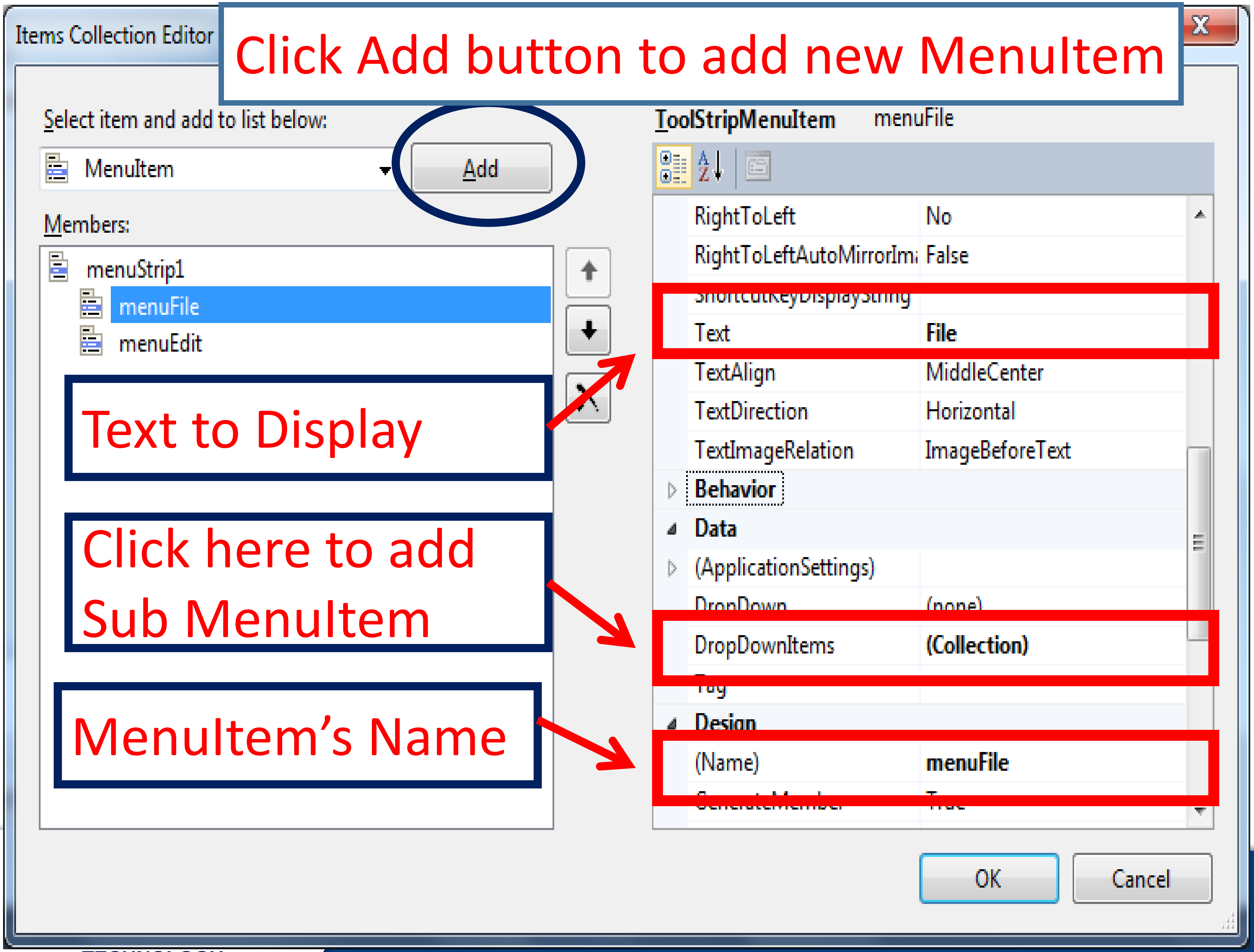


menuStrip1 System.Windows.Forms.Menu

ImageScalingSize	16, 16
ImageMode	NoControl
Items	(Collection)
LayoutStyle	HorizontalStackWith
Location	0, 0
Locked	False
Margin	0, 0, 0, 0

Items
Collection of items to display on the ToolStrip.

Click on button to add MenuItem



Click Add button to add new MenuItem

Items Collection Editor

Select item and add to list below:

MenuItem

Members:

- menuStrip1
  - menuFile
  - menuEdit

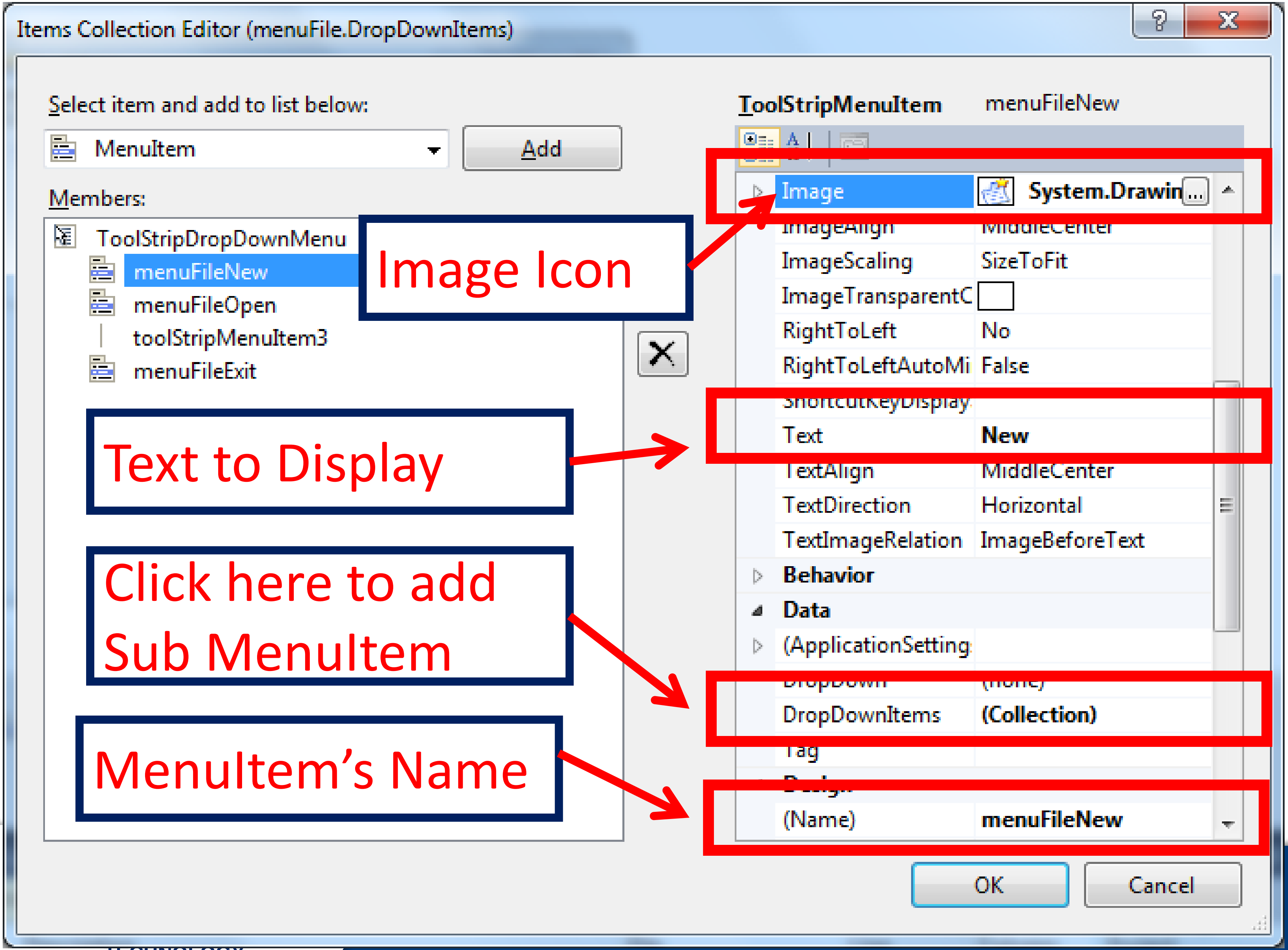
Text to Display

Click here to add Sub MenuItem

MenuItem's Name

ToolStripMenuItem menuFile

RightToLeft	No
RightToLeftAutoMirrorIm	False
ShortcutKeyDisplayString	
Text	File
TextAlign	MiddleCenter
TextDirection	Horizontal
TextImageRelation	ImageBeforeText
Behavior	
Data	
(ApplicationSettings)	
DropDown	(none)
DropDownItems	(Collection)
Tag	
Design	
(Name)	menuFile
GenerateMember	True



Items Collection Editor (menuFile.DropDownItems)

Select item and add to list below:

MenuItem | Add

Members:

- ToolStripDropDownMenu
  - menuFileNew
  - menuFileOpen
  - toolStripMenuItem3
  - menuFileExit

ToolStripMenuItem menuFileNew

Image	System.Drawin...
ImageAlign	MiddleCenter
ImageScaling	SizeToFit
ImageTransparentC	
RightToLeft	No
RightToLeftAutoMi	False
ShortcutKeyDisplay	
Text	New
TextAlign	MiddleCenter
TextDirection	Horizontal
TextImageRelation	ImageBeforeText
Behavior	
Data	
(ApplicationSetting	
DropDown	(none)
DropDownItems	(Collection)
Tag	
Design	
(Name)	menuFileNew

OK | Cancel

Image Icon

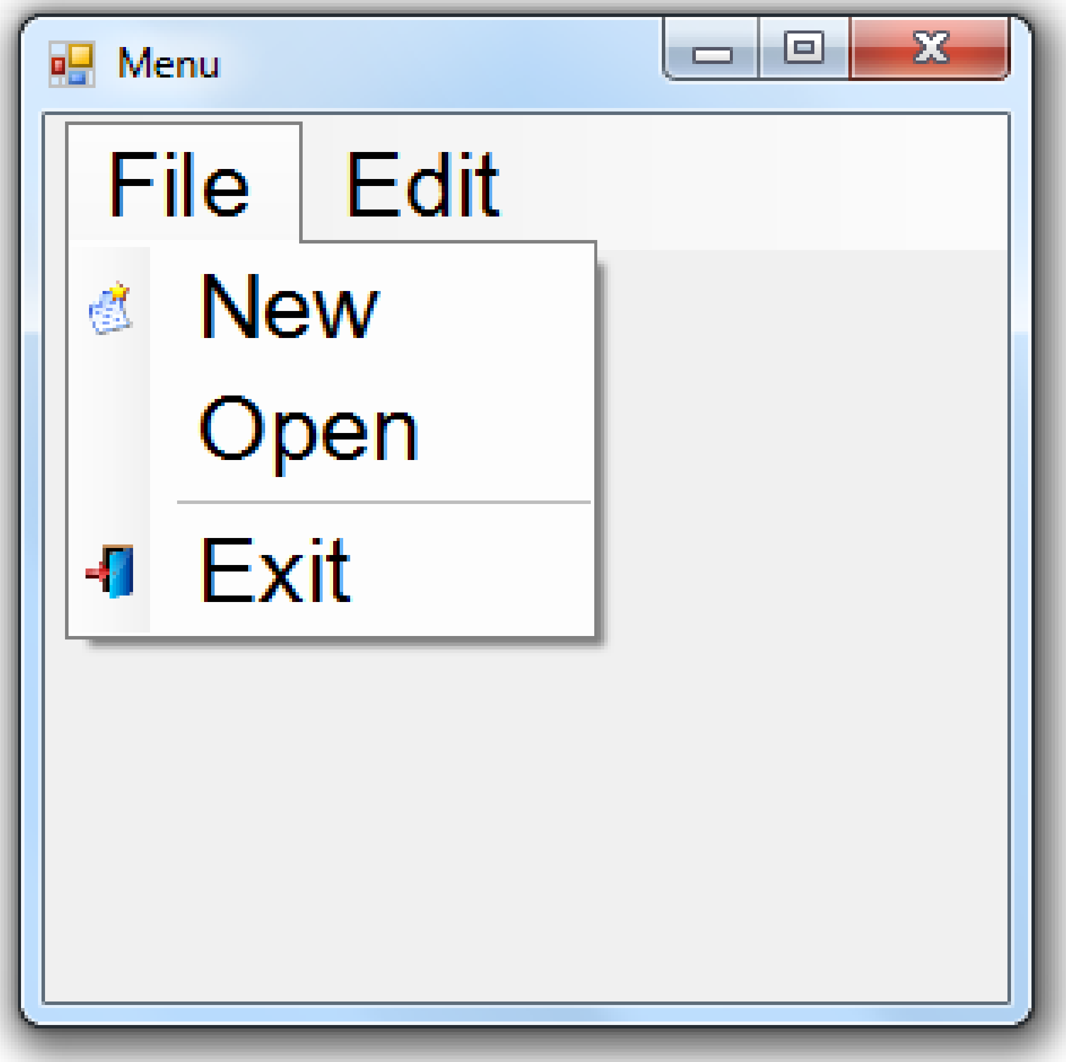
Text to Display

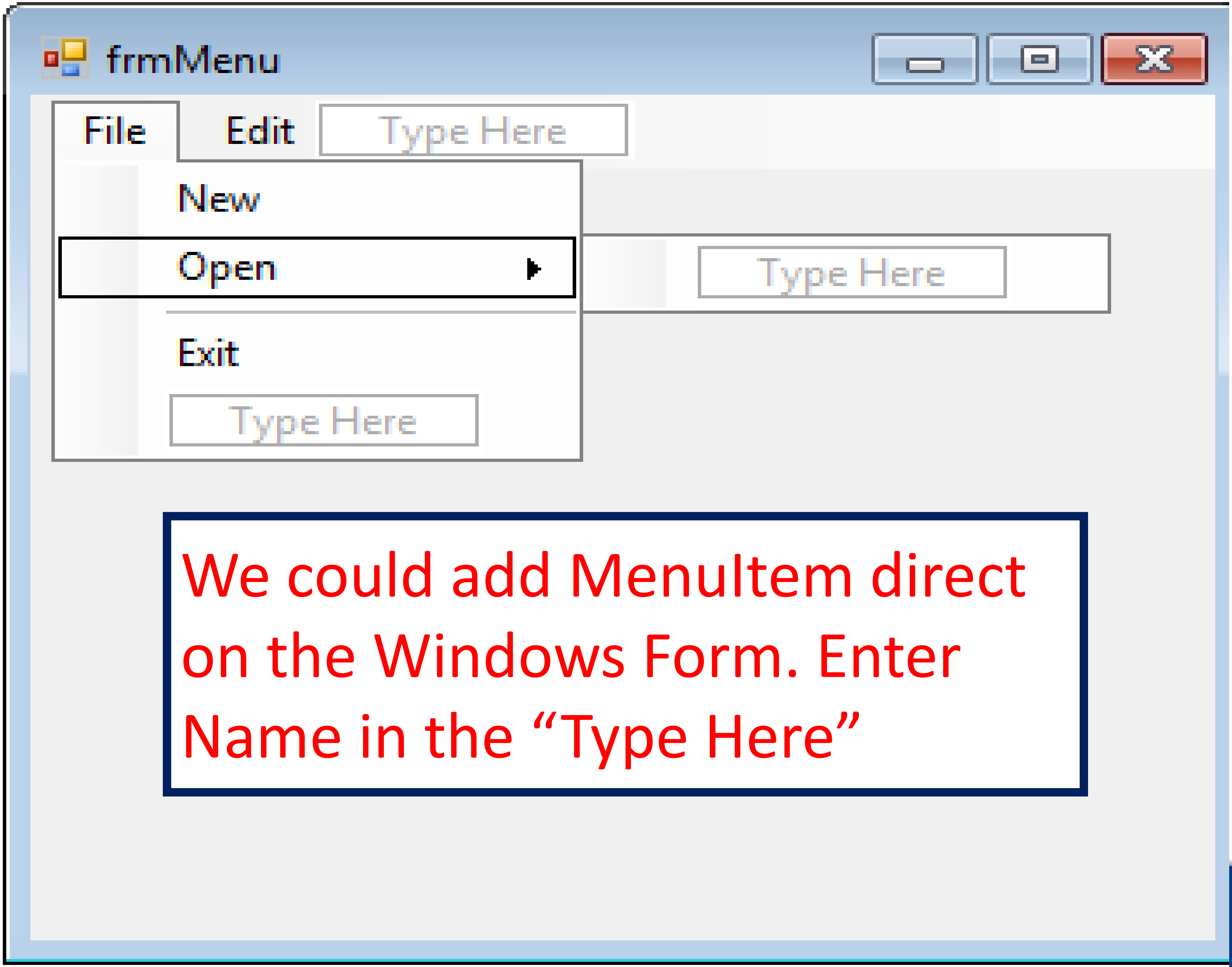
Click here to add Sub MenuItem

MenuItem's Name

TECHNOLOGY

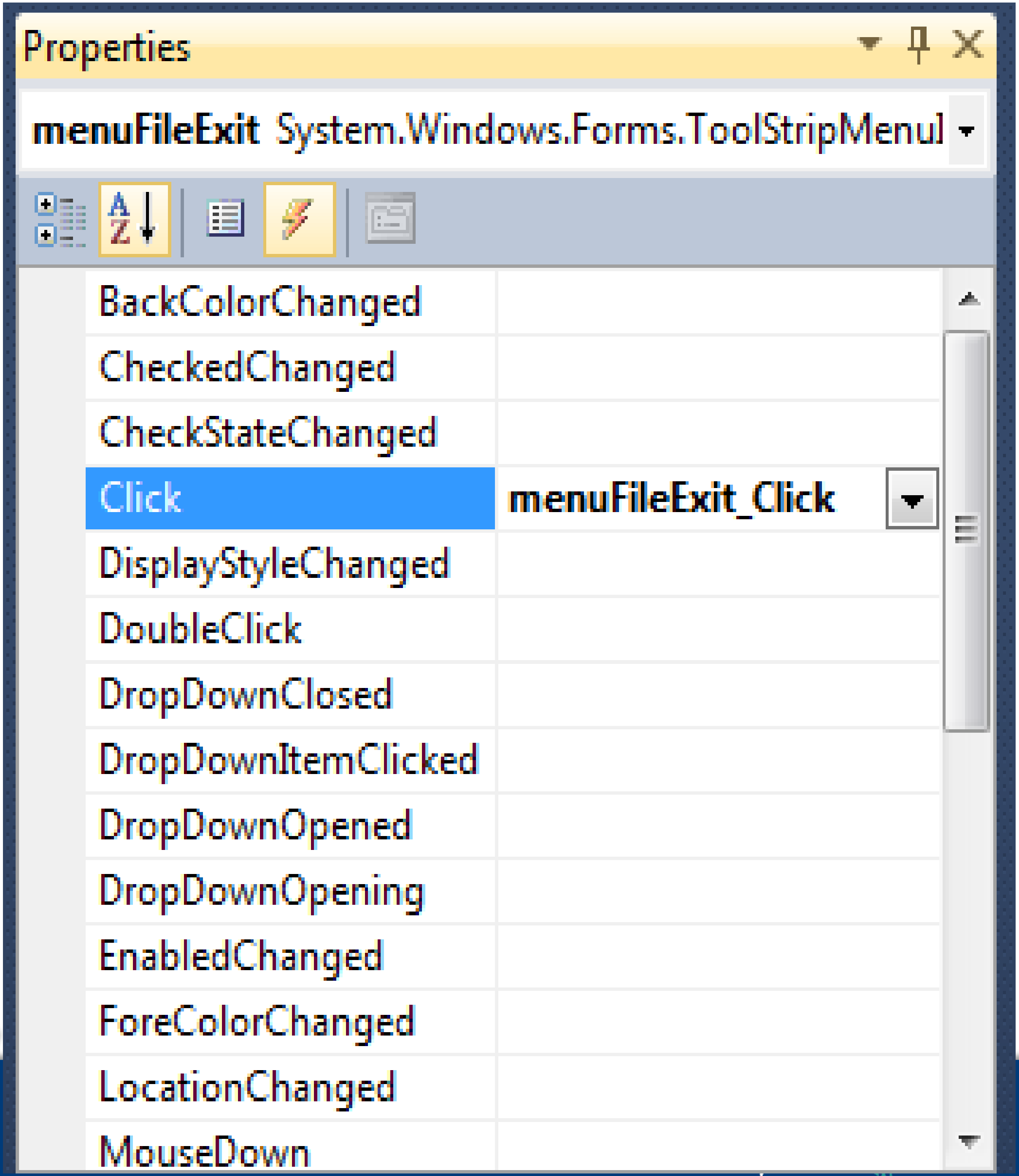
MenuStrip





We could add MenuItem direct on the Windows Form. Enter Name in the “Type Here”

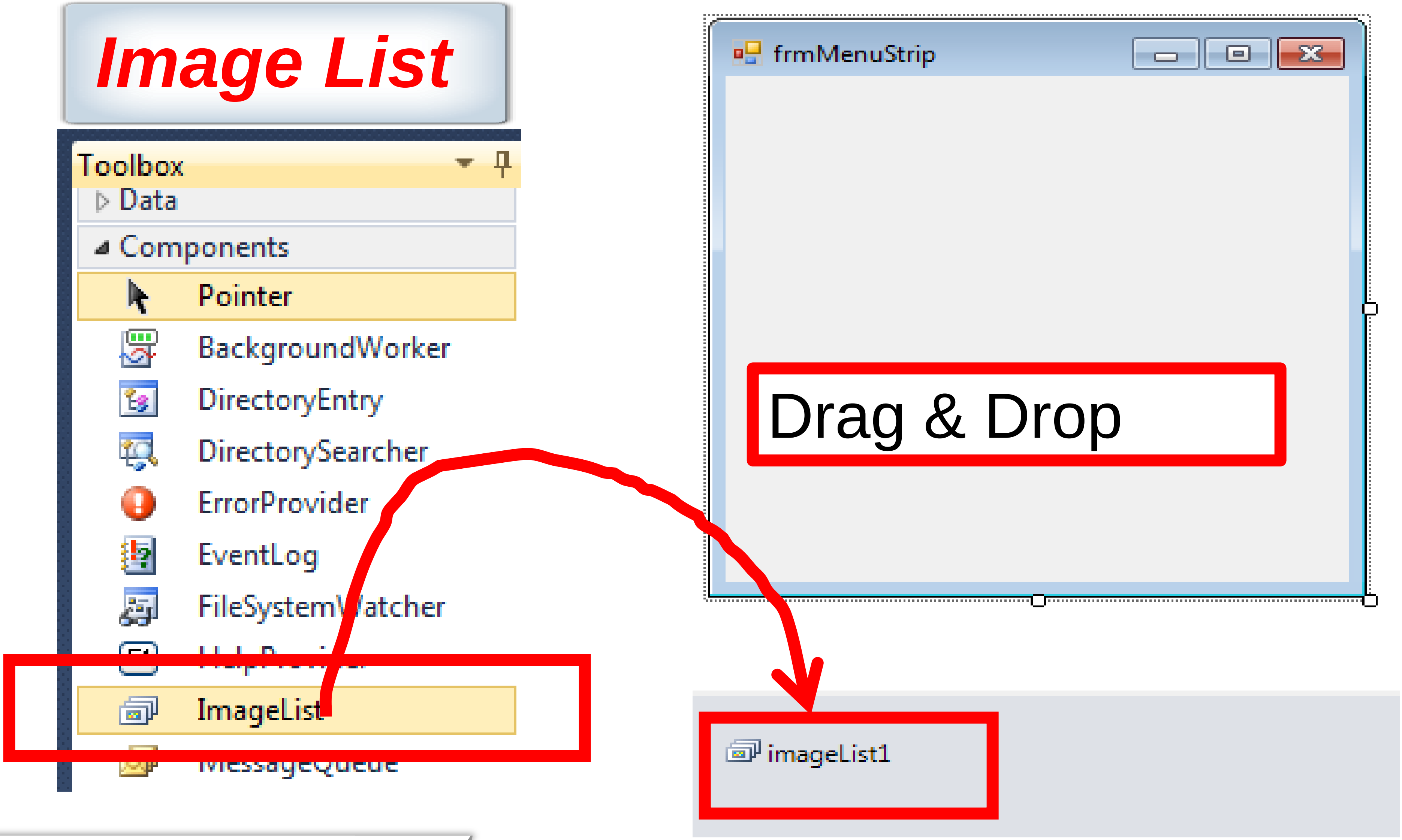
Add Event for each MenuItem



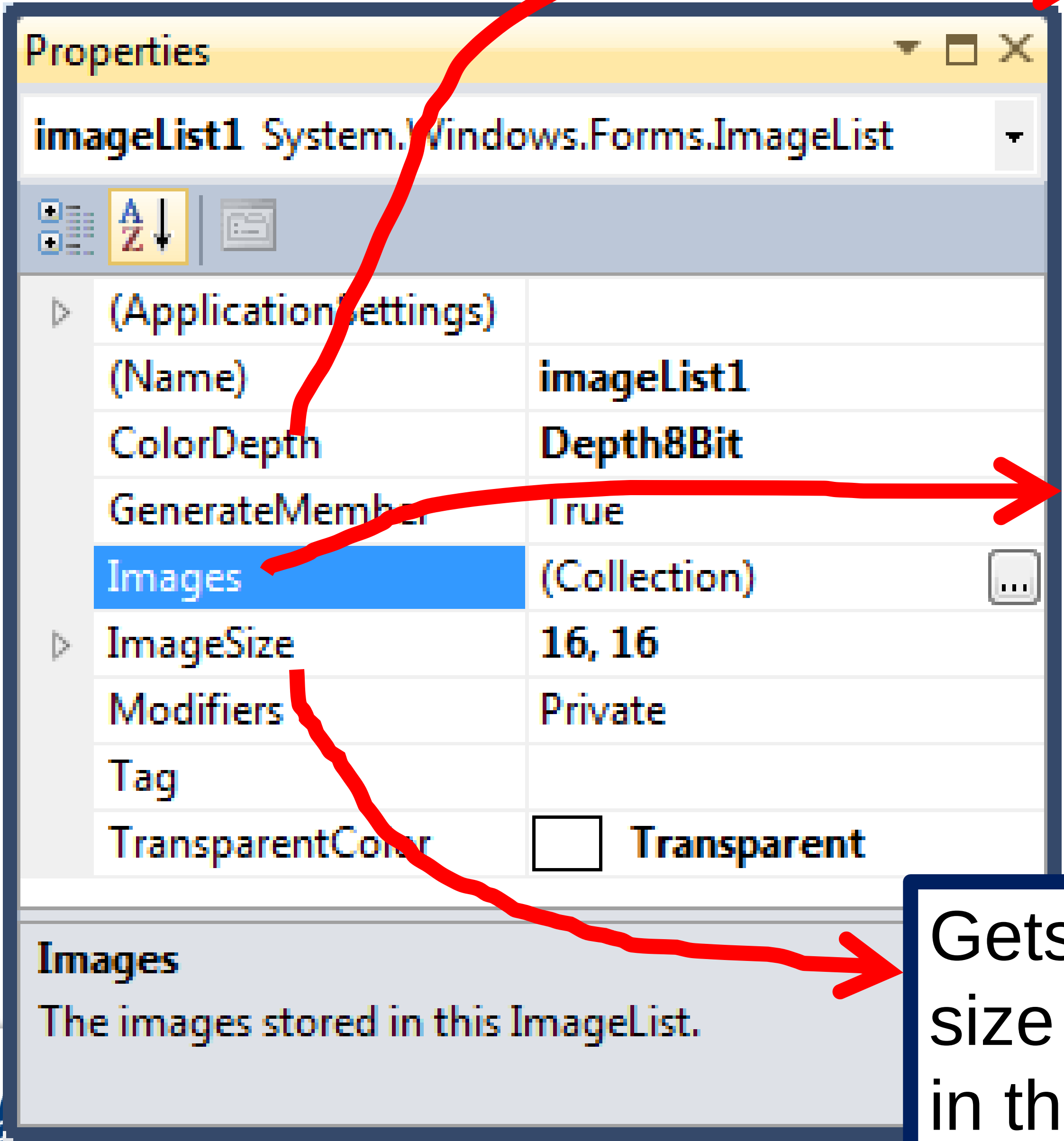
menuFileExit System.Windows.Forms.ToolStripMenuI

BackColorChanged	
CheckedChanged	
CheckStateChanged	
Click	menuFileExit_Click
DisplayStyleChanged	
DoubleClick	
DropDownClosed	
DropDownItemClicked	
DropDownOpened	
DropDownOpening	
EnabledChanged	
ForeColorChanged	
LocationChanged	
MouseDown	

Image List



Drag & Drop



imageList1 System.Windows.Forms.ImageList

(ApplicationSettings)	
(Name)	imageList1
ColorDepth	Depth8Bit
GenerateMember	True
Images	(Collection)
ImageSize	16, 16
Modifiers	Private
Tag	
TransparentColor	□ Transparent

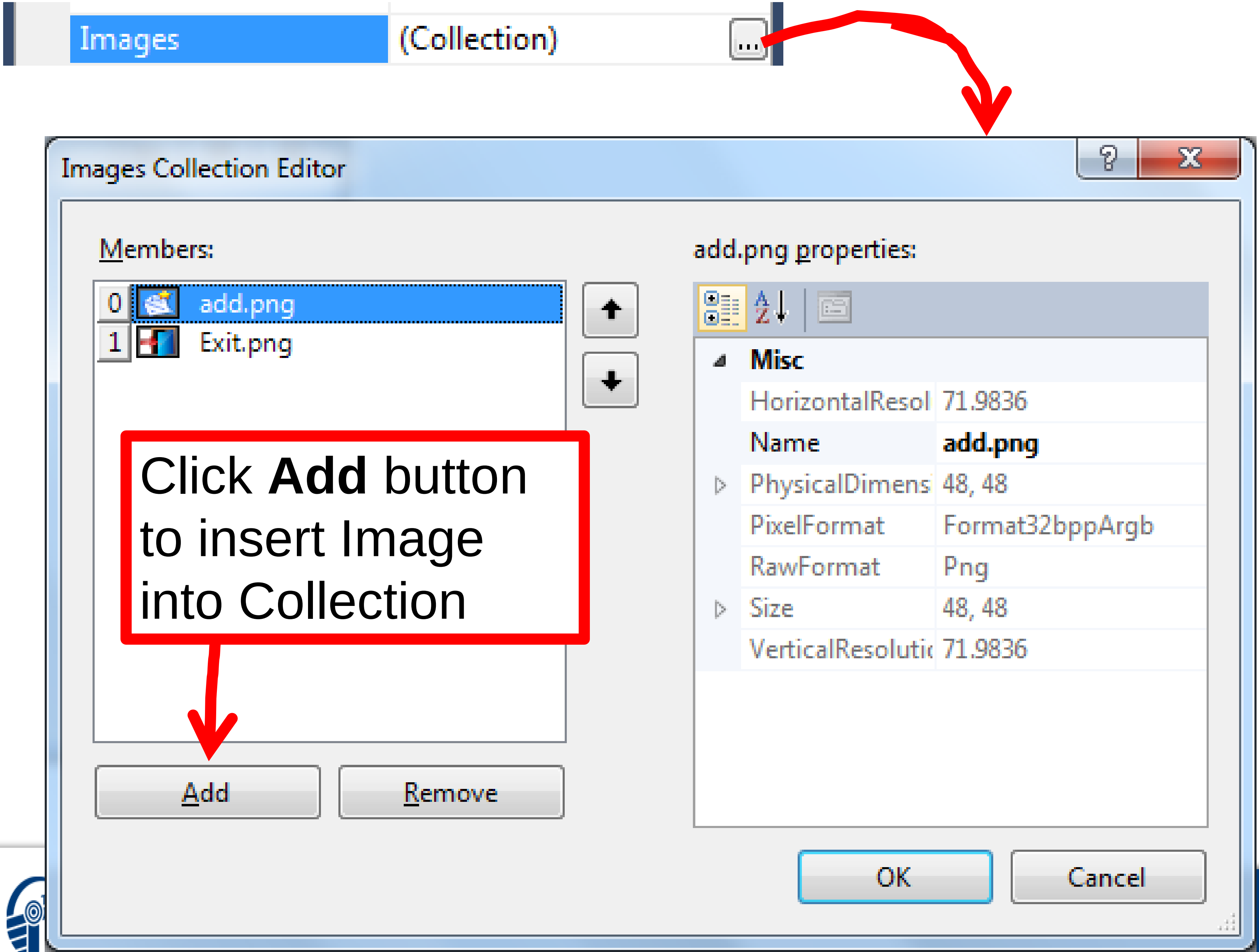
Images

The images stored in this ImageList.

Gets the color depth of the image list

Gets the ImageList.
ImageCollection for this image list.
See Next Slide

Gets or sets the size of the images in the image list



Images (Collection)

Images Collection Editor

Members:

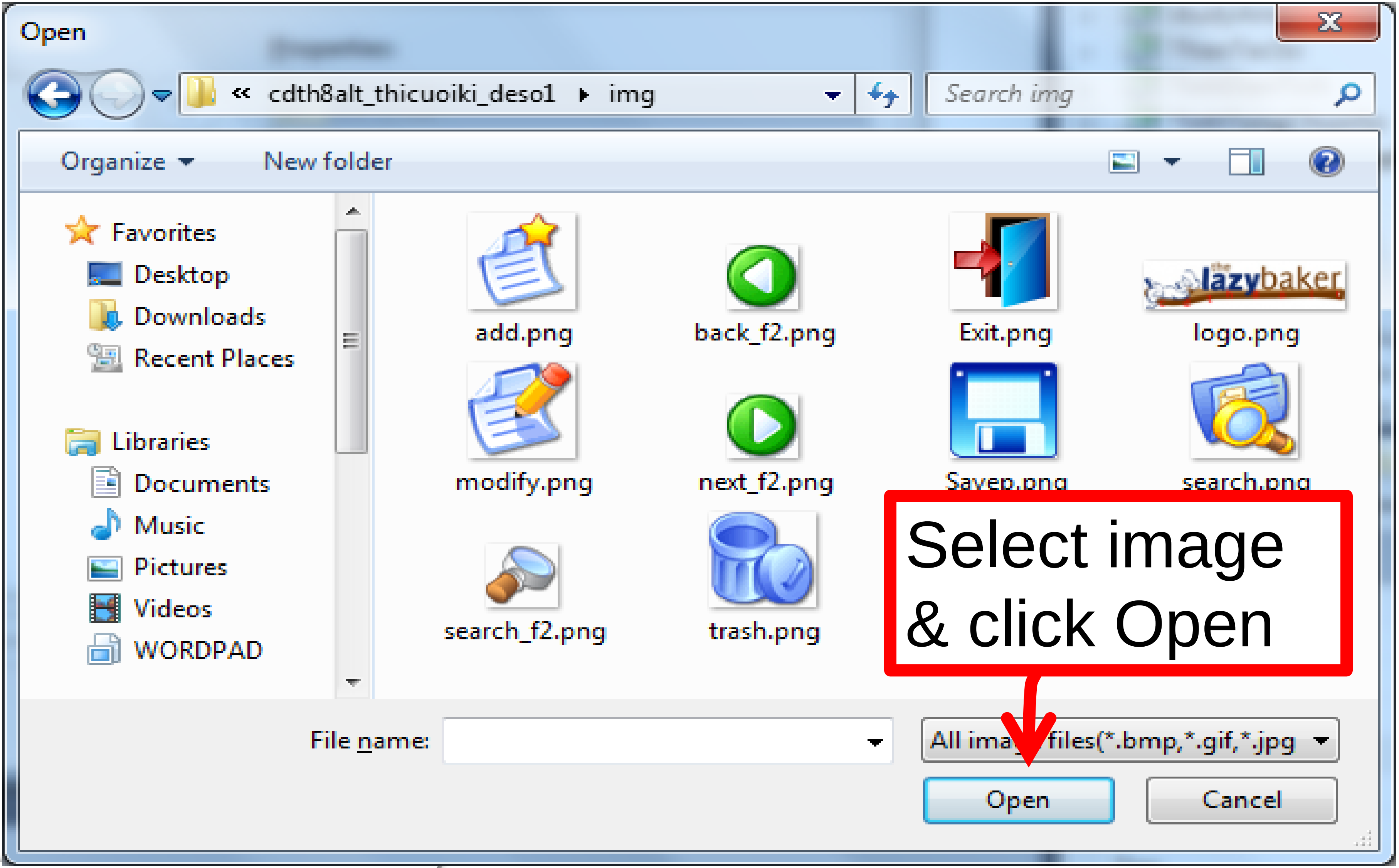
0 add.png
1 Exit.png

add.png properties:

Misc

HorizontalResol	71.9836
Name	add.png
PhysicalDimens	48, 48
PixelFormat	Format32bppArgb
RawFormat	Png
Size	48, 48
VerticalResoluti	71.9836

Click **Add** button to insert Image into Collection

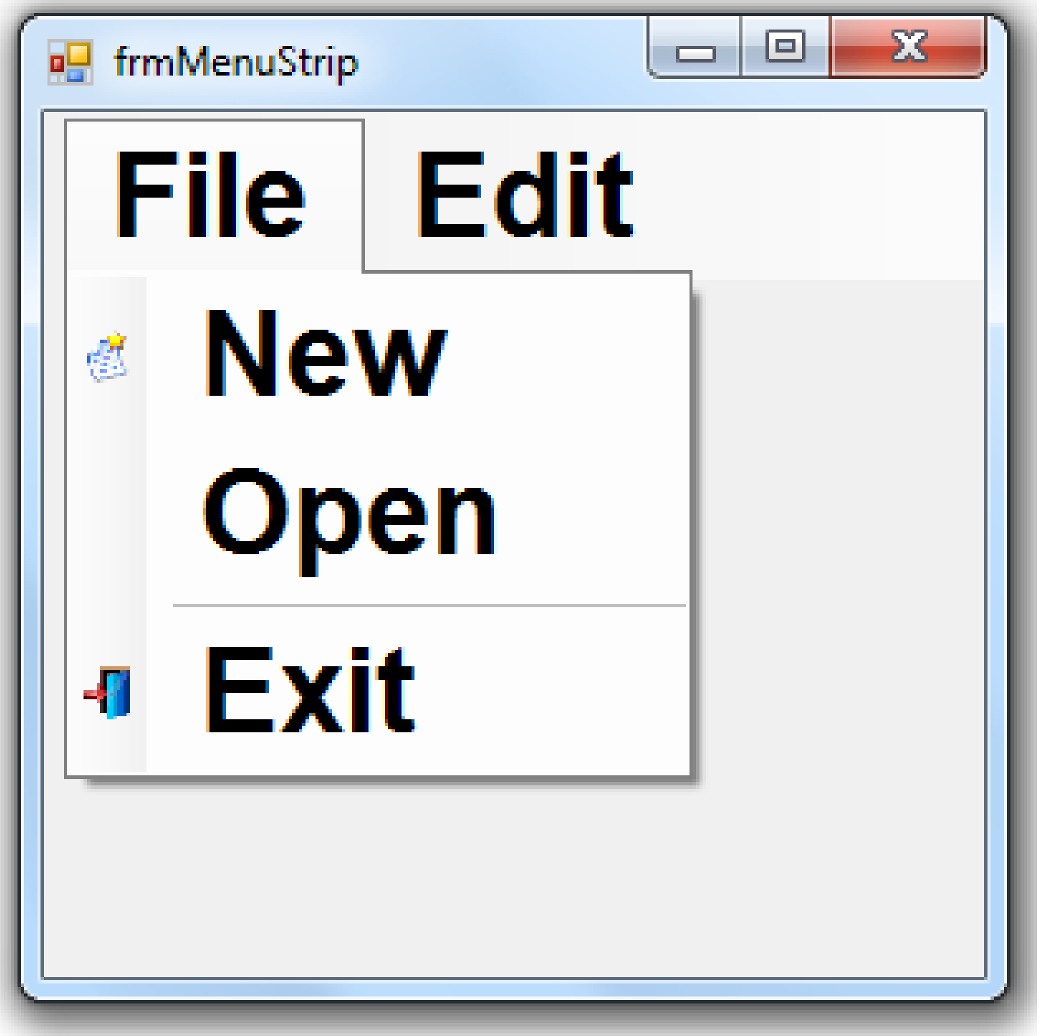


Select image & click Open

MenuStrip

At

Runtime



```
private MenuStrip menuBar;
private ToolStripMenuItem
menuFile, menuEdit, menuFileNew,
menuFileOpen, menuFileExit,
menuEditCut, menuEditCopy,
menuEditPaste;
```

```
private void createMenu()
{
menuBar = new MenuStrip();
menuBar.Font = new Font("arial", 36,
FontStyle.Bold, GraphicsUnit.Pixel);
this.MainMenuStrip = menuBar;
menuFile = new
ToolStripMenuItem("File");
menuFileNew = new
ToolStripMenuItem("New");
menuFileNew.Image =
imageList1.Images[0];
```

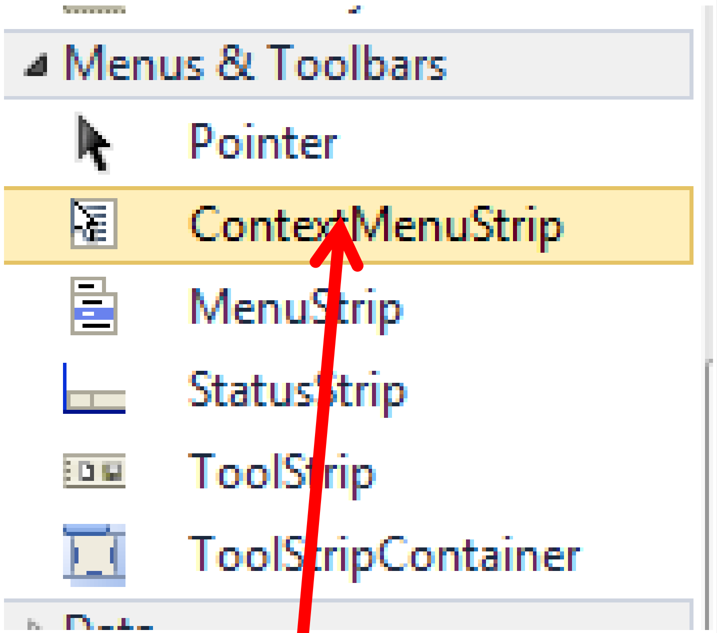
```
menuFileOpen = new
ToolStripMenuItem("Open");
ToolStripSeparator sp = new
ToolStripSeparator();
menuFileExit = new
ToolStripMenuItem("Exit");
menuFileExit.Image =
imageList1.Images[1];
menuFile.DropDownItems.Add(
menuFileNew);
menuFile.DropDownItems.Add(
menuFileOpen);
```

```
menuFile.DropDownItems.Add(sp);
menuFile.DropDownItems.Add(
menuFileExit);
menuEdit = new
ToolStripMenuItem("Edit");
menuEditCut = new
ToolStripMenuItem("Cut");
menuEditCopy = new
ToolStripMenuItem("Copy");
menuEditPaste = new
ToolStripMenuItem("Paste");
```

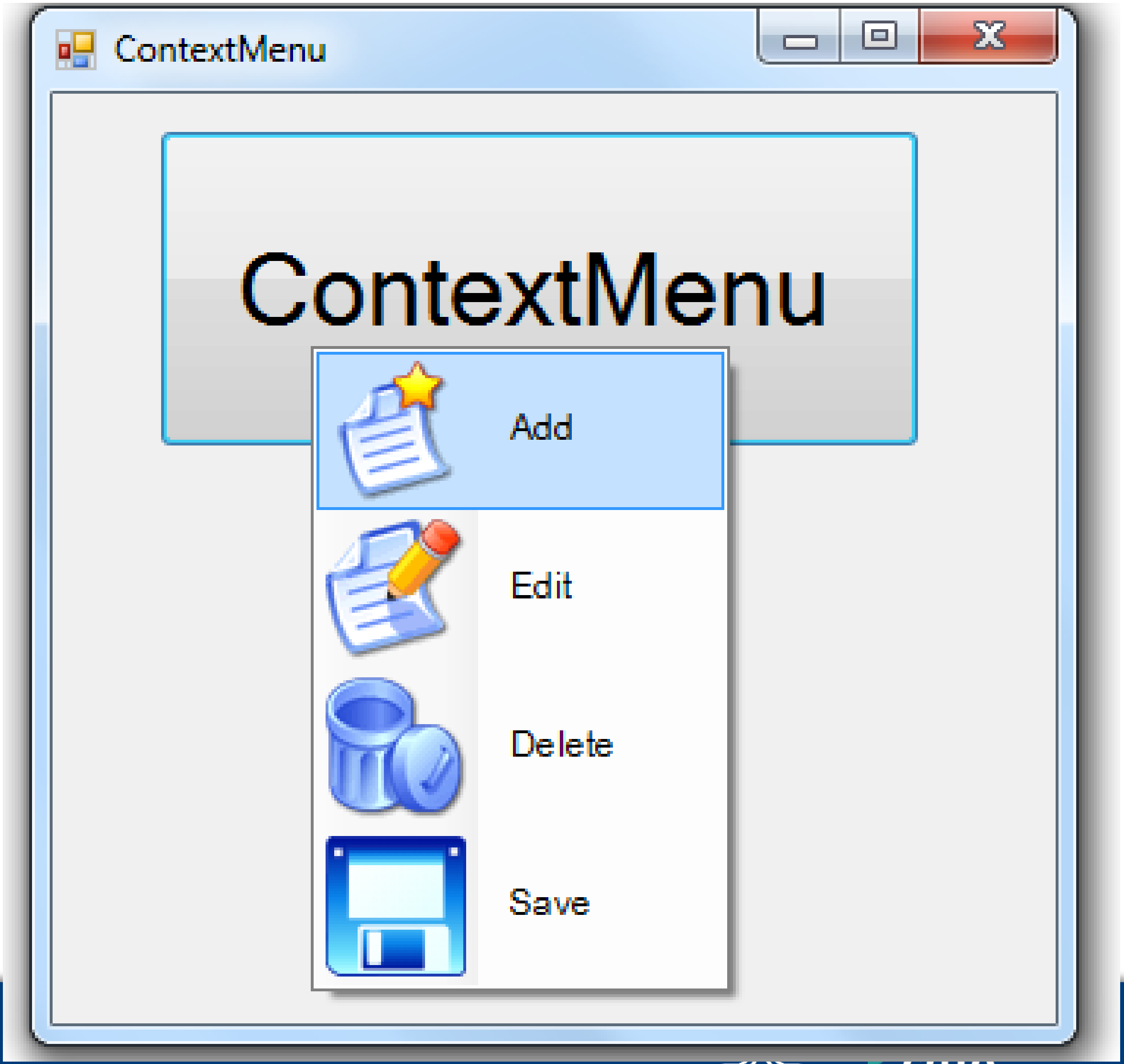
```
menuEdit.DropDownItems.AddRange(new
ToolStripItem[] {
menuEditCut,menuEditCopy,
menuEditPaste});
menuBar.Items.AddRange(new
ToolStripItem[] {
menuFile,menuEdit});
this.Controls.Add(menuBar);
attachEvents();
}
private void attachEvents()
{menuFileExit.Click += processClick; }
```

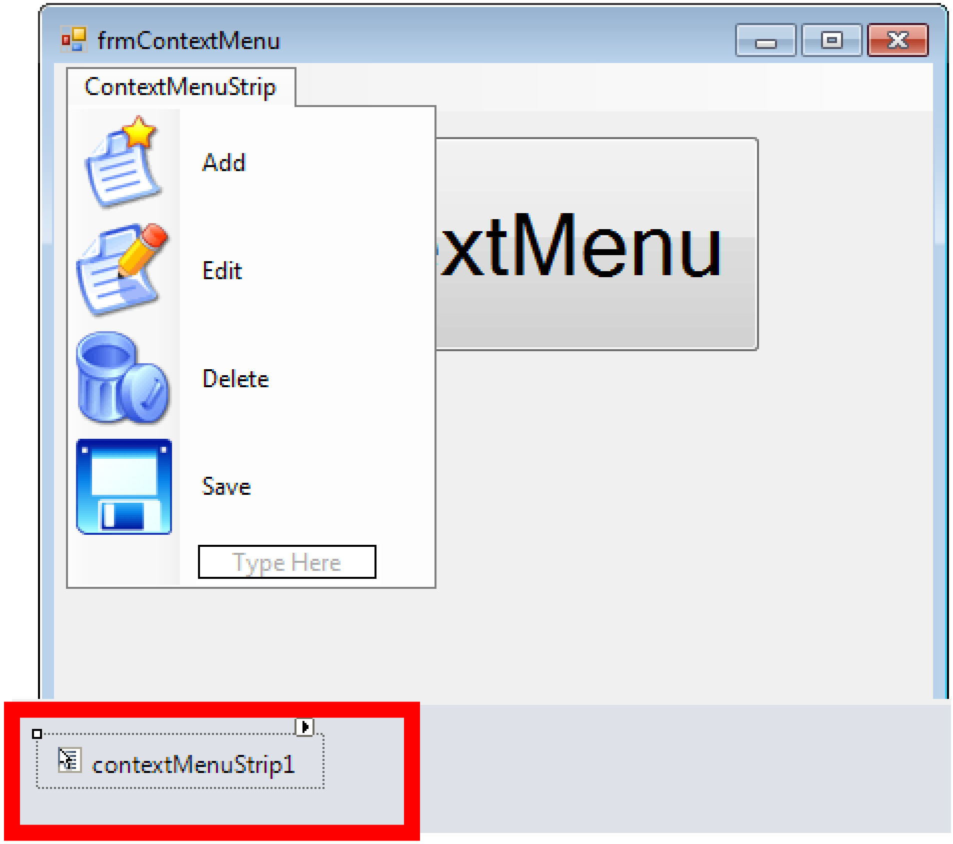
```
private void processClick
(object o, EventArgs e)
{
 if (o.Equals(menuFileExit))
      Application.Exit();
}
private void frmMenuStrip_Load
(object sender, EventArgs e)
{
 createMenu();
}
```

ContextMenuStrip

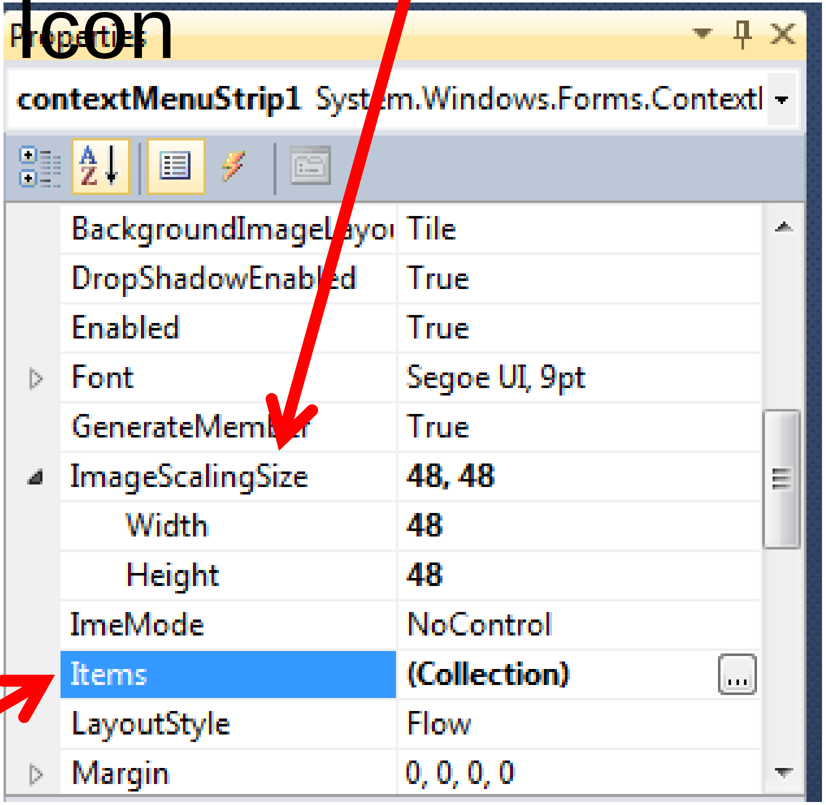


Drag & Drop into the FORM





ImageScalingSize to set Width, Height Icon

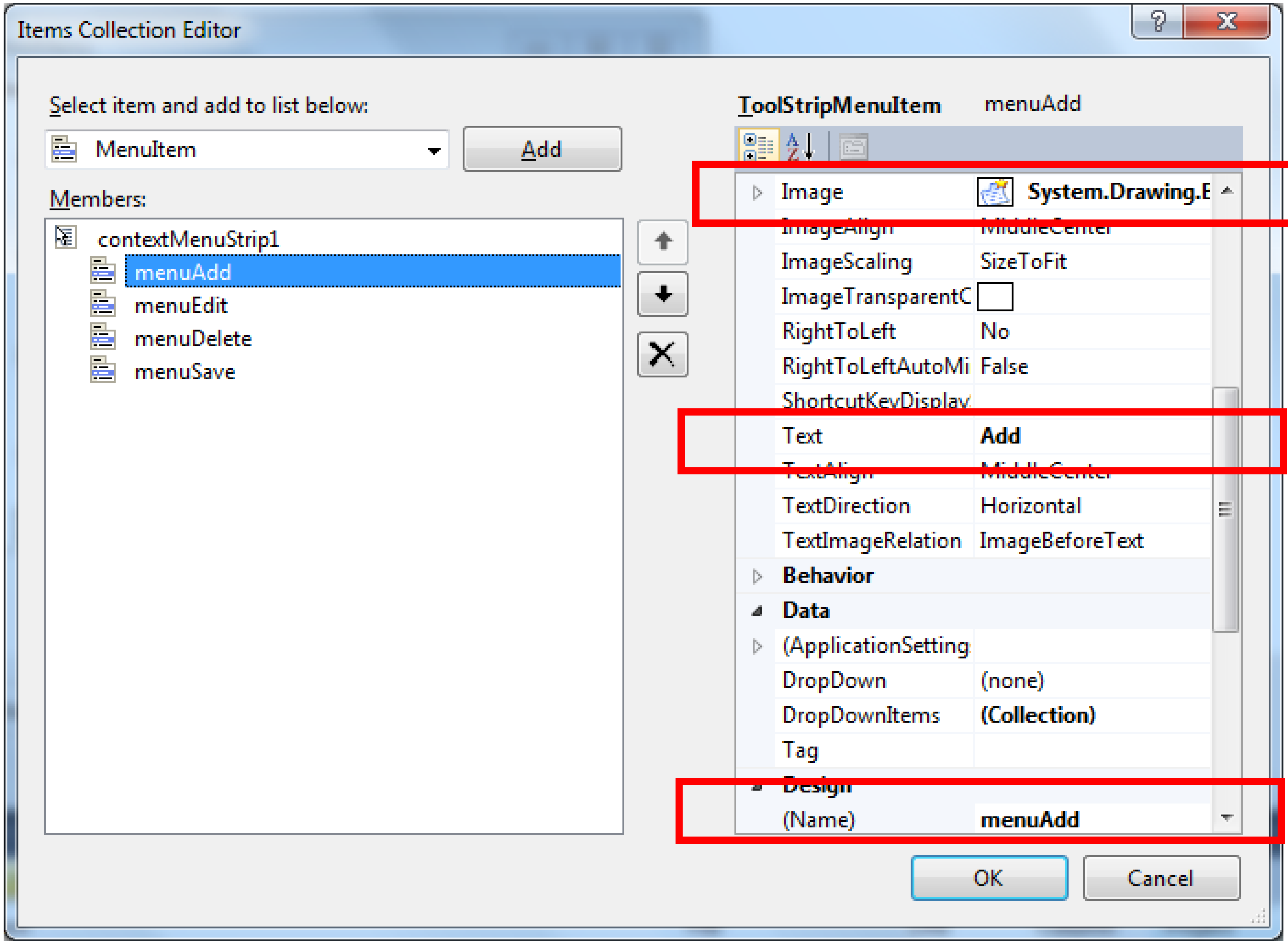


contextMenuStrip1 System.Windows.Forms.ContextI

BackgroundImageLayou	Tile
DropShadowEnabled	True
Enabled	True
Font	Segoe UI, 9pt
GenerateMember	True
ImageScalingSize	48, 48
Width	48
Height	48
ImeMode	NoControl
Items	(Collection)
LayoutStyle	Flow
Margin	0, 0, 0, 0

Click Items (Collection) to add MenuItem

As the same MenuStrip



Items Collection Editor

Select item and add to list below:

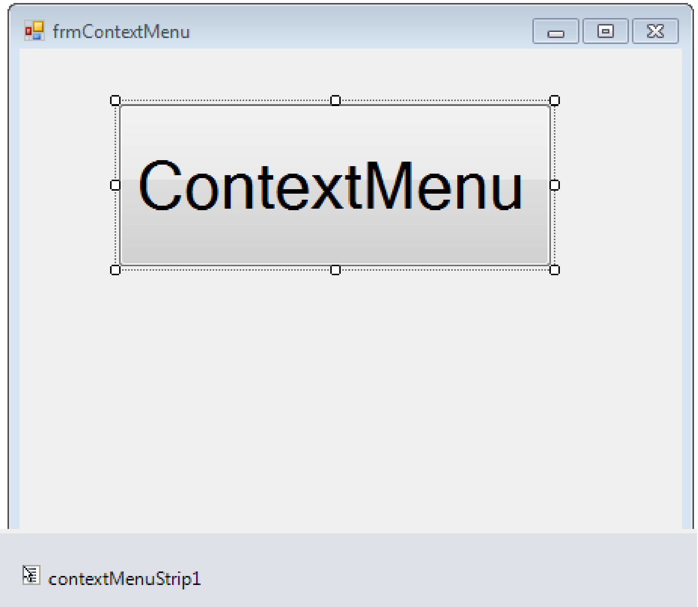
MenuItem

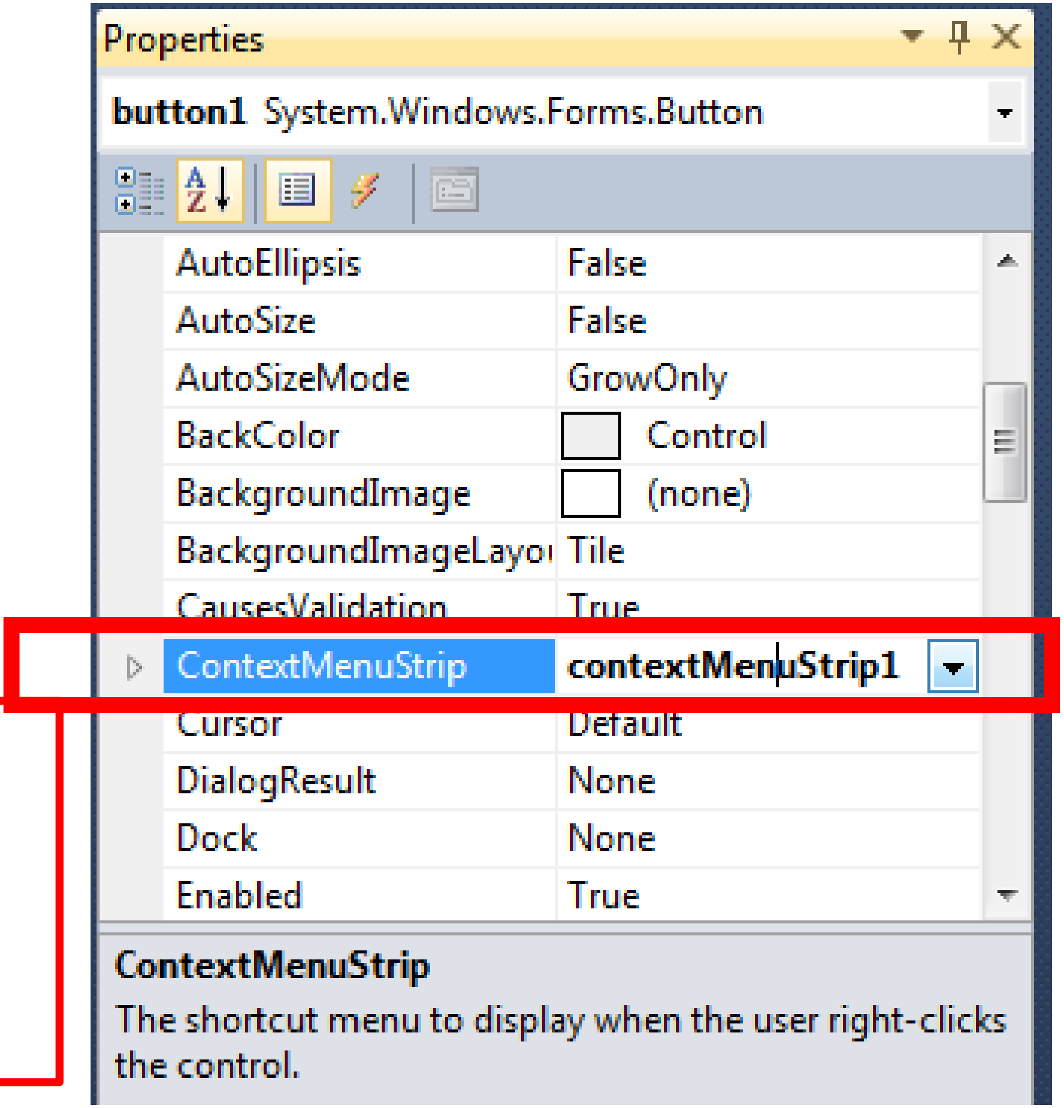
Members:

- contextMenuStrip1
  - menuAdd
  - menuEdit
  - menuDelete
  - menuSave

ToolStripMenuItem menuAdd

Image	System.Drawing.E
ImageAlign	MiddleCenter
ImageScaling	SizeToFit
ImageTransparentC	
RightToLeft	No
RightToLeftAutoMi	False
ShortcutKeyDisplay	
Text	Add
TextAlign	MiddleCenter
TextDirection	Horizontal
TextImageRelation	ImageBeforeText
Behavior	
Data	
(ApplicationSetting	
DropDown	(none)
DropDownItems	(Collection)
Tag	
Design	
(Name)	menuAdd





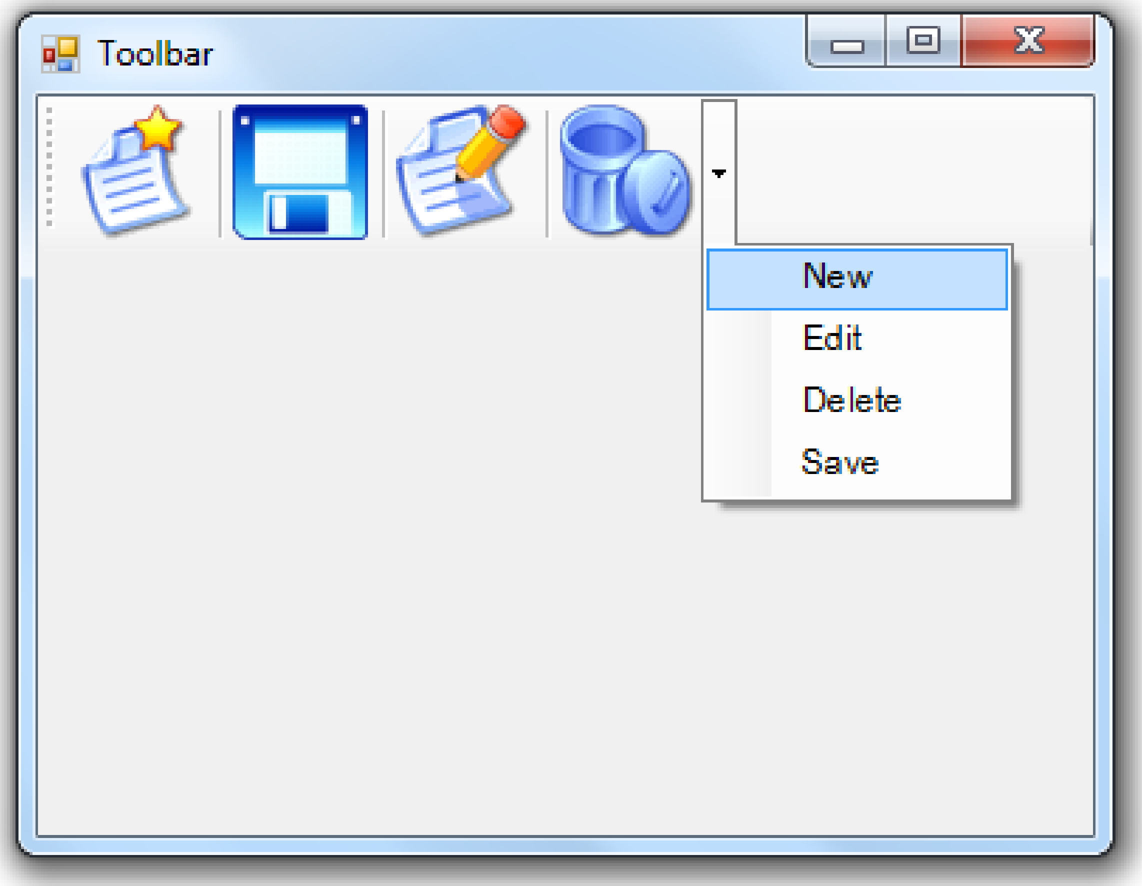
button1 System.Windows.Forms.Button

AutoEllipsis	False
AutoSize	False
AutoSizeMode	GrowOnly
BackColor	Control
BackgroundImage	(none)
BackgroundImageLayou	Tile
CausesValidation	True
ContextMenuStrip	contextMenuStrip1
Cursor	Default
DialogResult	None
Dock	None
Enabled	True

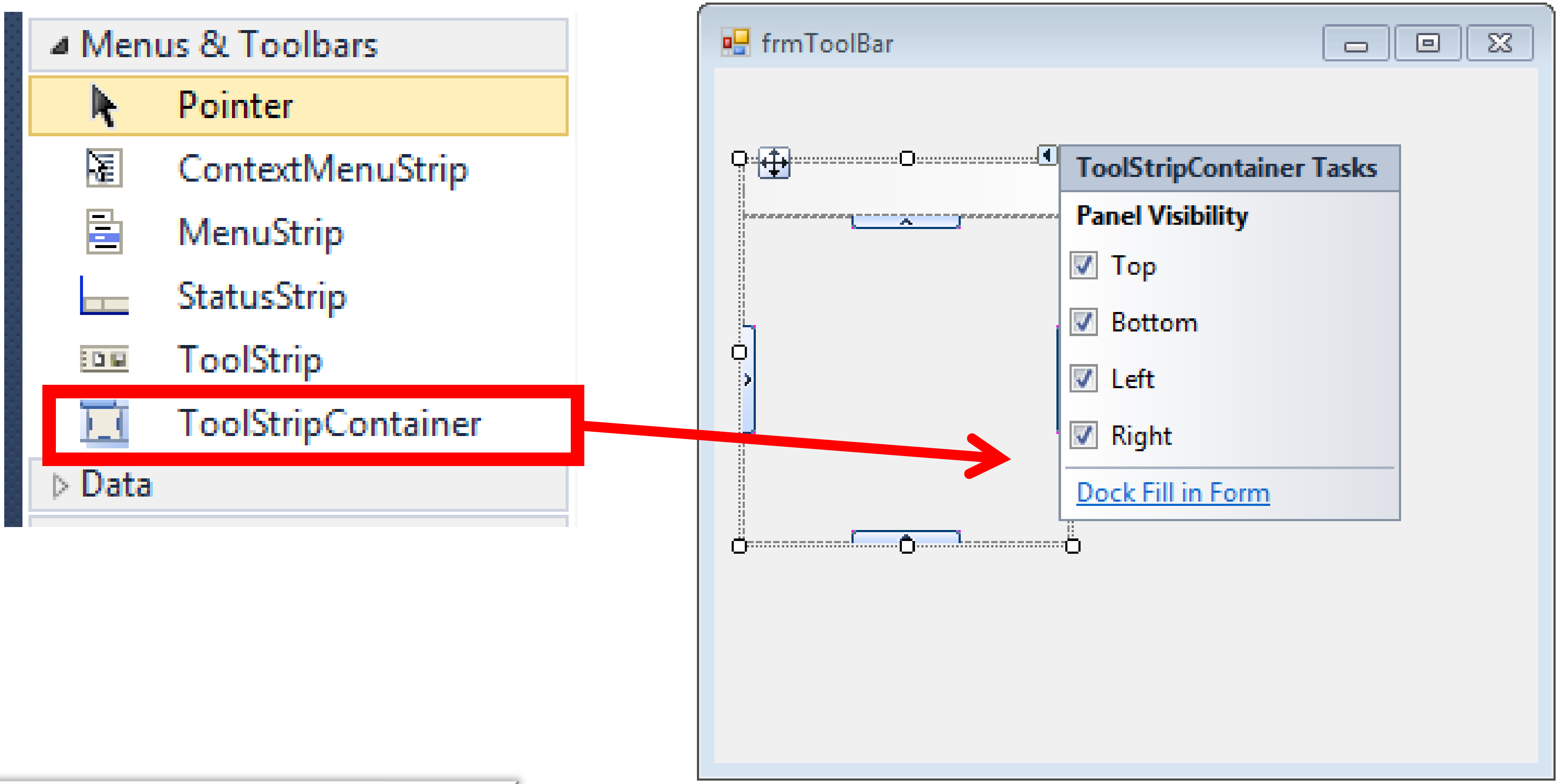
ContextMenuStrip
The shortcut menu to display when the user right-clicks the control.

1. Choose button
2. Set ContextMenuStrip
3. Attach event as the same MenuStrip

ToolStripContainer & ToolStrip



ToolStripContainer



ToolStripContainer Tasks

Panel Visibility

☑ Top

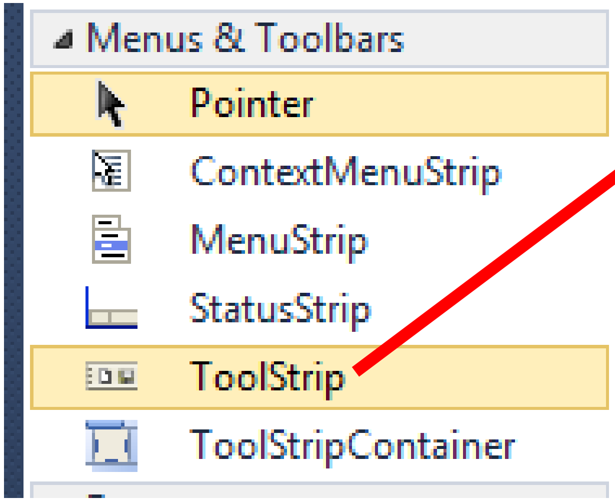
☑ Bottom

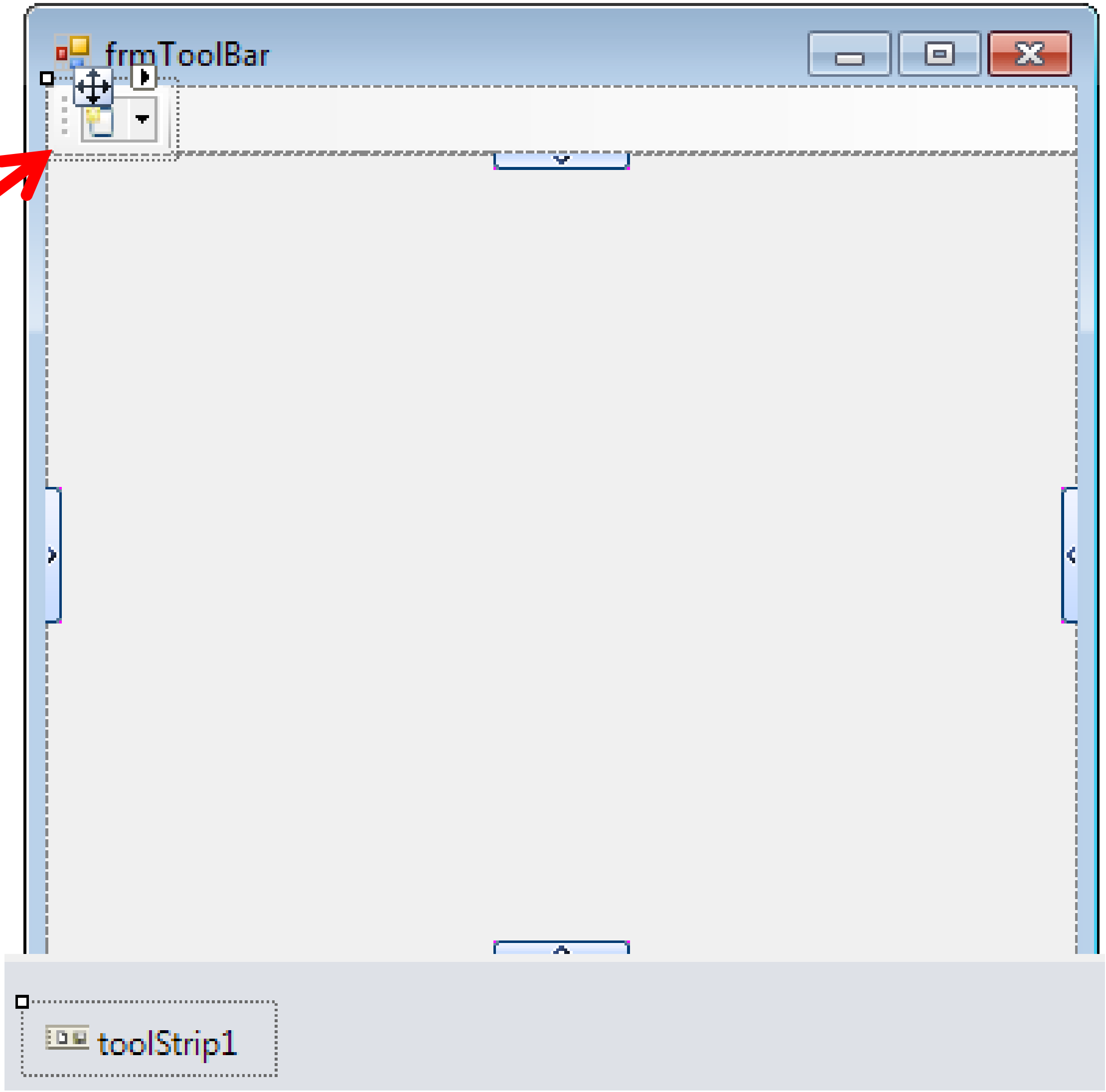
☑ Left

☑ Right

Dock Fill in Form

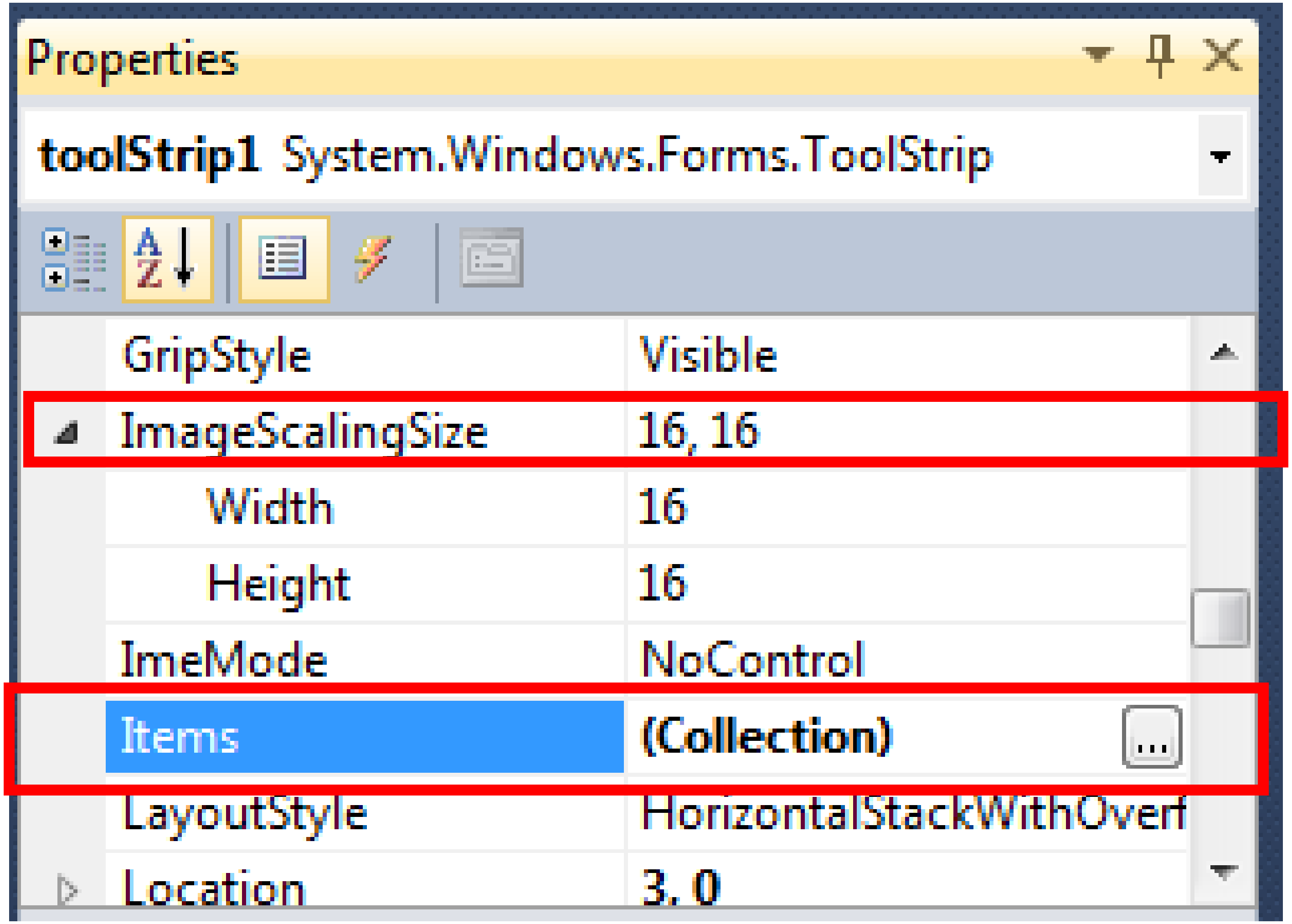
ToolStripContainer Provides panels on each side of the form and a central panel that can hold one or more controls





Drag & Drop **ToolStrip** into the Form

As the same **MenuStrip**

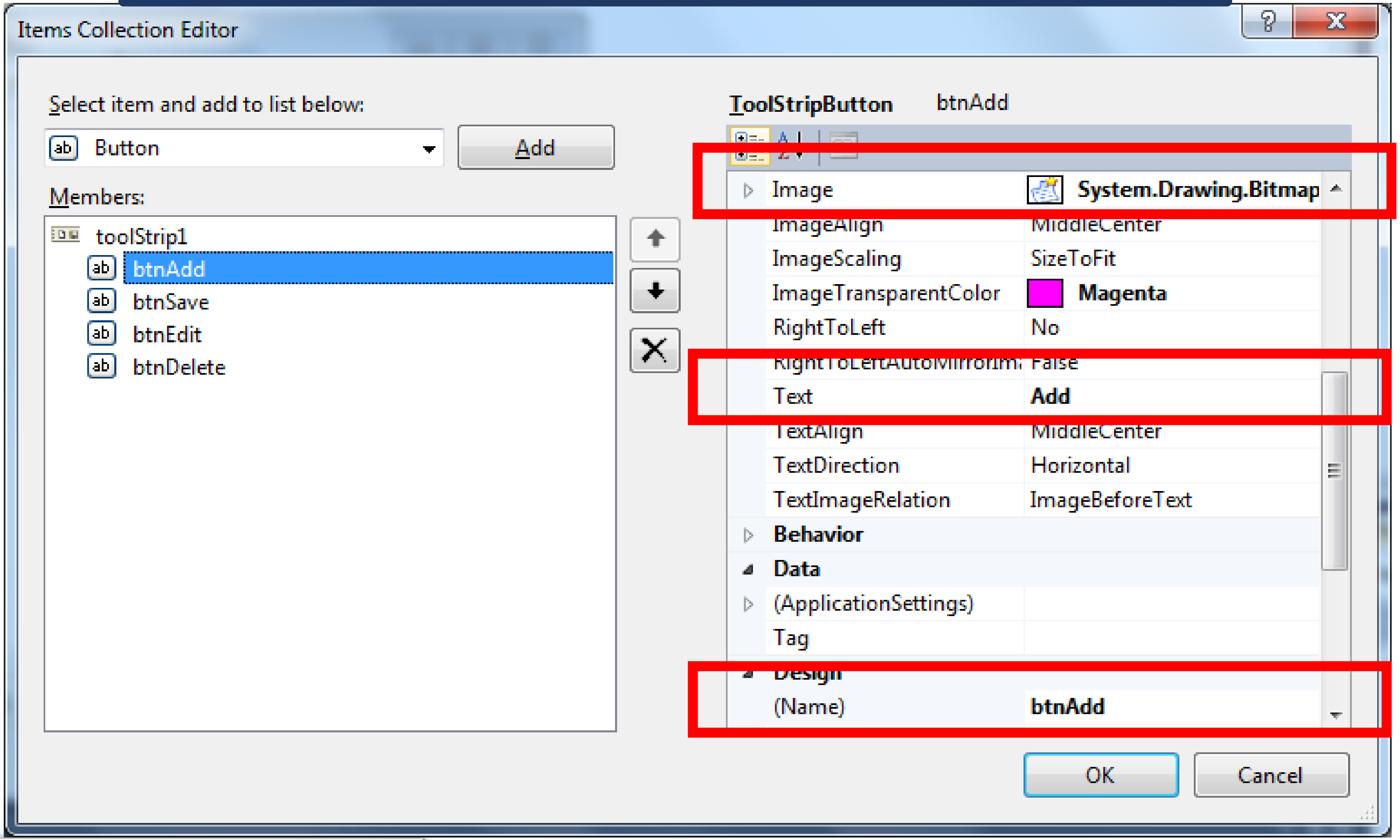


Properties

toolStrip1 System.Windows.Forms.ToolStrip

GripStyle	Visible
ImageScalingSize	16, 16
Width	16
Height	16
ImeMode	NoControl
Items	(Collection)
LayoutStyle	HorizontalStackWithOverf
Location	3, 0

Add Items Collection for ToolBar



Items Collection Editor

Select item and add to list below:

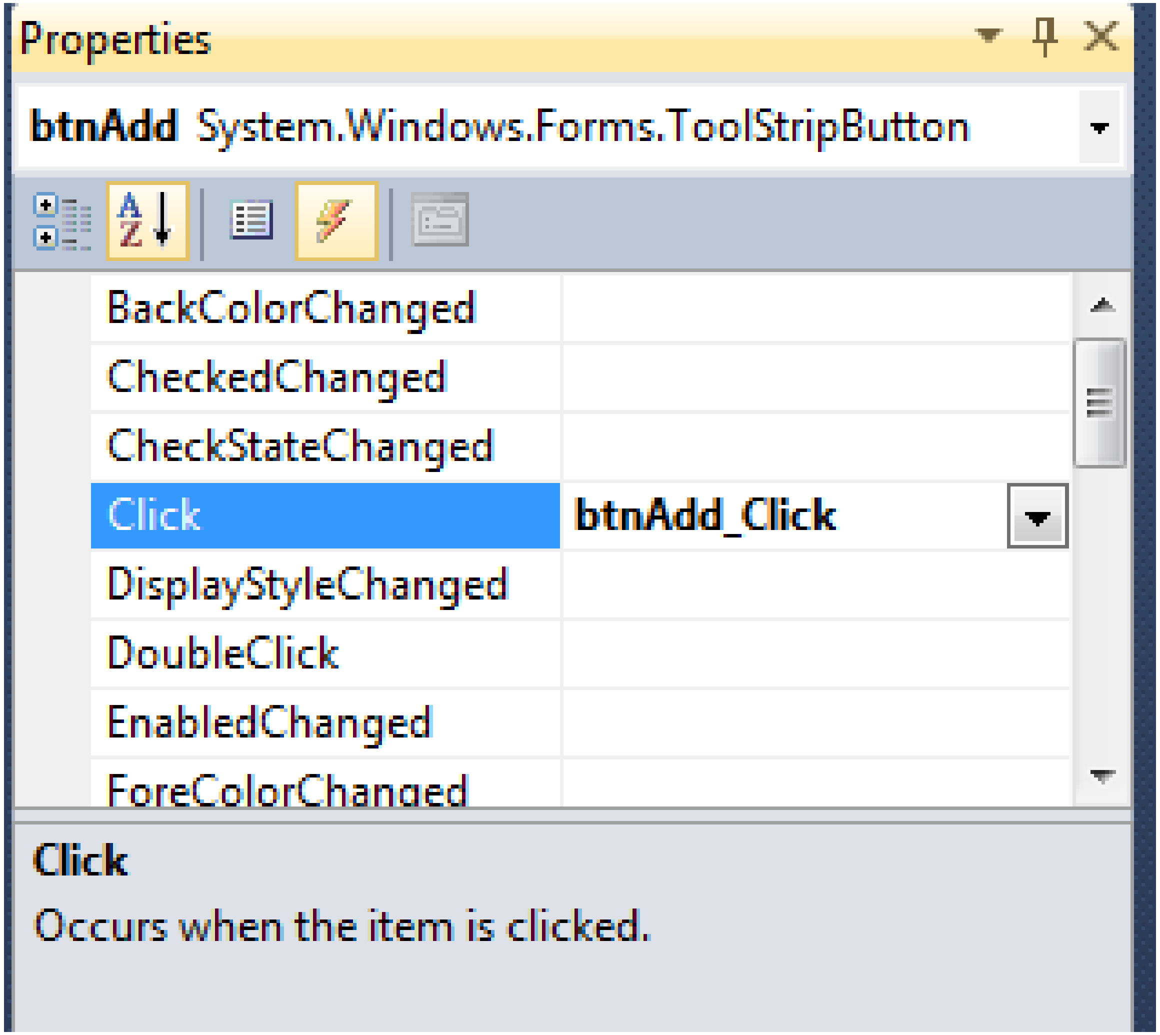
Members:

- toolStrip1
  - btnAdd
  - btnSave
  - btnEdit
  - btnDelete

ToolStripButton btnAdd

Image	System.Drawing.Bitmap
ImageAlign	MiddleCenter
ImageScaling	SizeToFit
ImageTransparentColor	🟪 Magenta
RightToLeft	No
RightToLeftAutoMirrorIm:	False
Text	Add
TextAlign	MiddleCenter
TextDirection	Horizontal
TextImageRelation	ImageBeforeText
Behavior	
Data	
(ApplicationSettings)	
Tag	
Design	
(Name)	btnAdd

Attach Events for each Item



Properties

btnAdd System.Windows.Forms.ToolStripButton

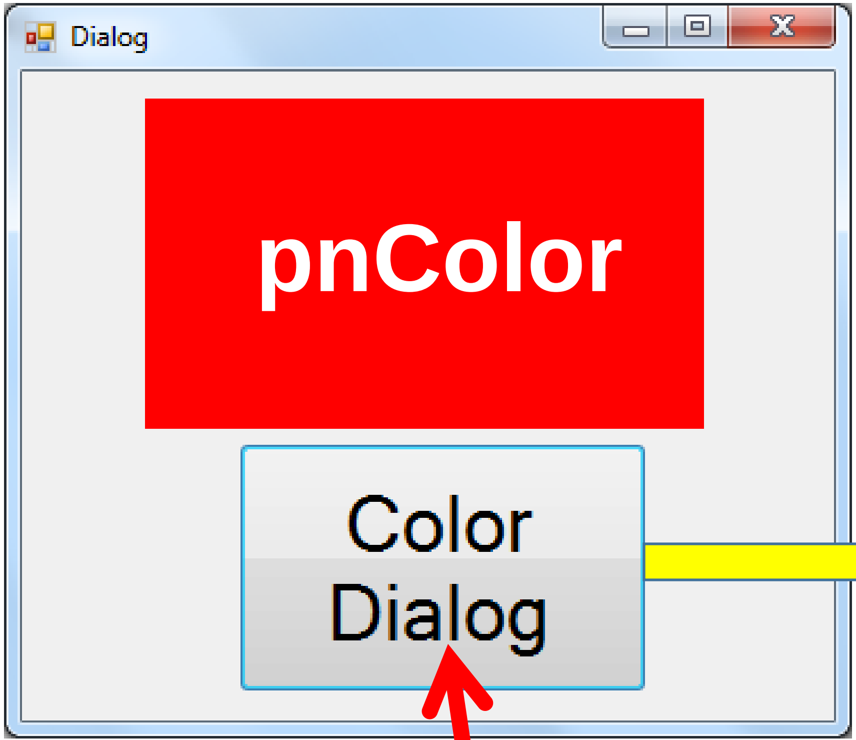
BackColorChanged	
CheckedChanged	
CheckStateChanged	
Click	btnAdd_Click
DisplayStyleChanged	
DoubleClick	
EnabledChanged	
ForeColorChanged	

Click

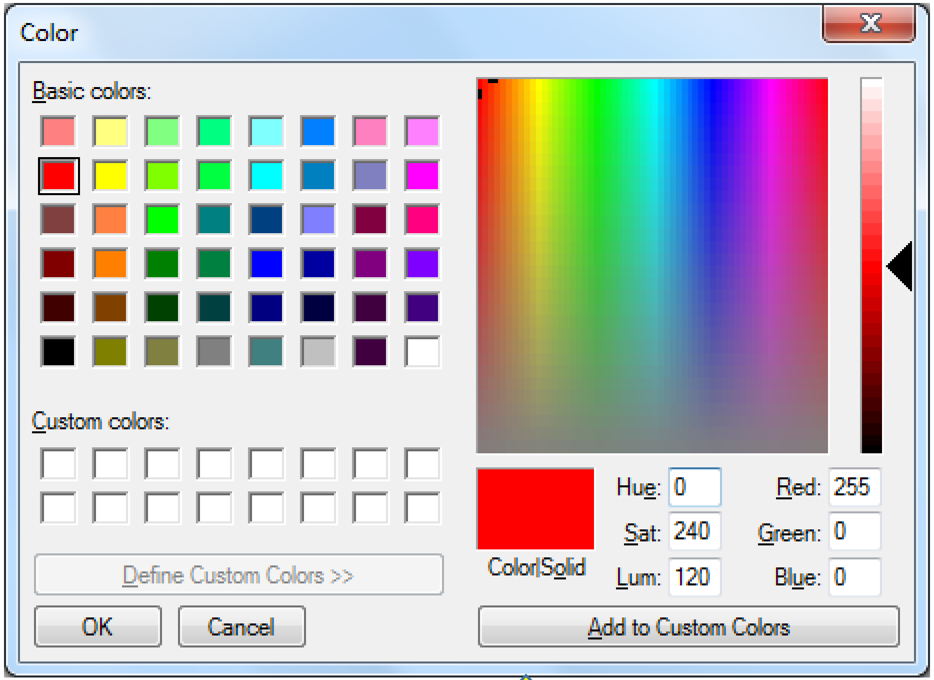
Occurs when the item is clicked.

Bài tập áp dụng

ColorDialog



btnColor

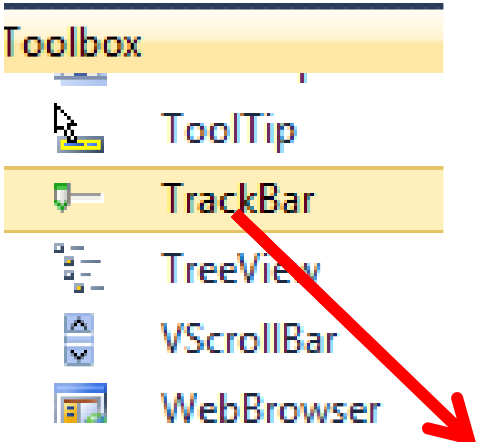


Hue: 0
Sat: 240
Lum: 120
Red: 255
Green: 0
Blue: 0

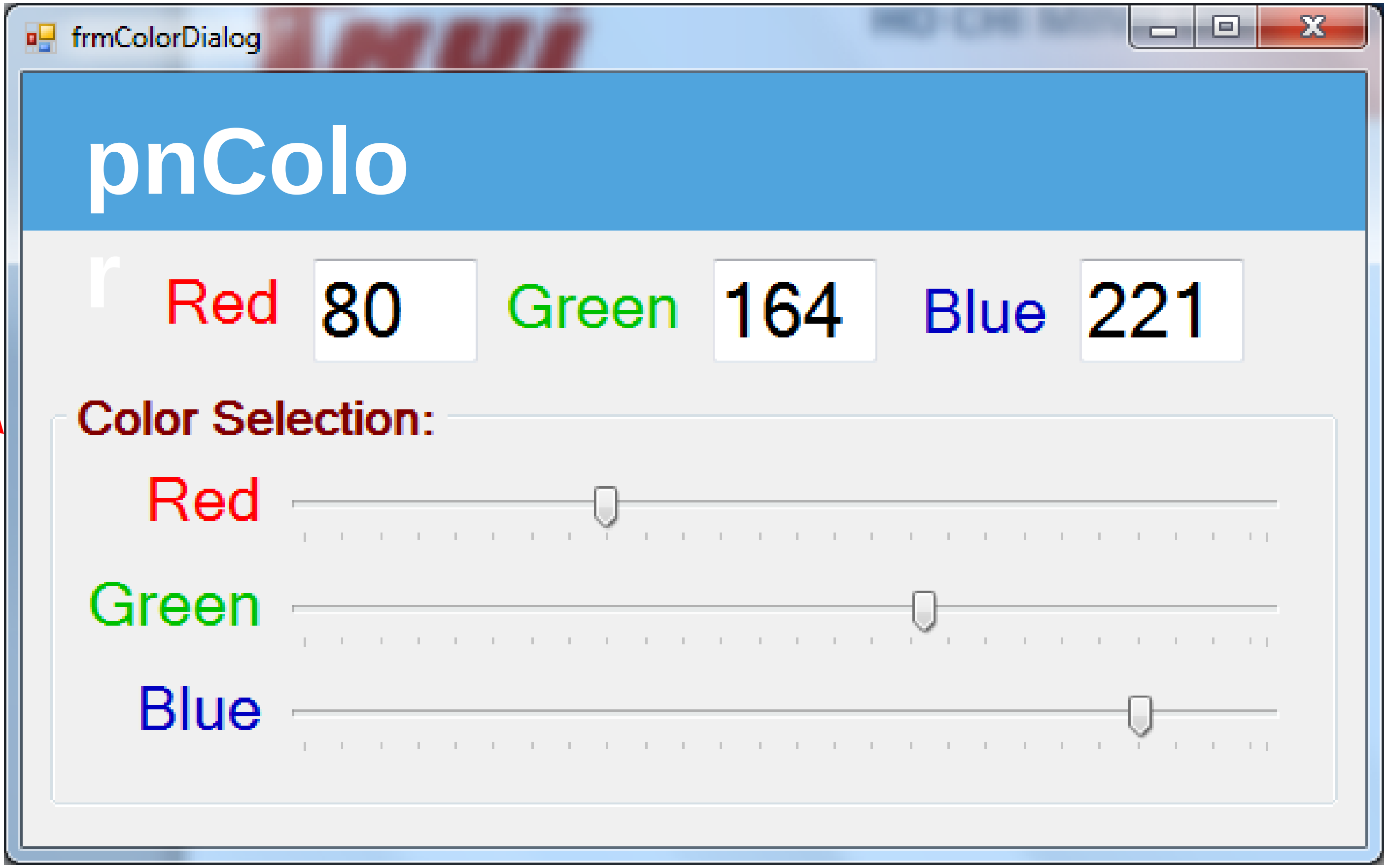
How to use ColorDialog?

```
private void btnColor_Click
(object sender, EventArgs e)
{
ColorDialog cldlg= new ColorDialog();
cldlg.Color = pnColor.BackColor;

if(cldlg.ShowDialog()==DialogResult.OK)
 pnColor.BackColor = cldlg.Color;
}
```

If you want to set color by the **RED**, **GREEN**, **BLUE**. Please see figure below:



trackBarRed
txtRed
trackBarGreen
txtGreen
trackBarBlue
txtBlue



Red 80 Green 164 Blue 221

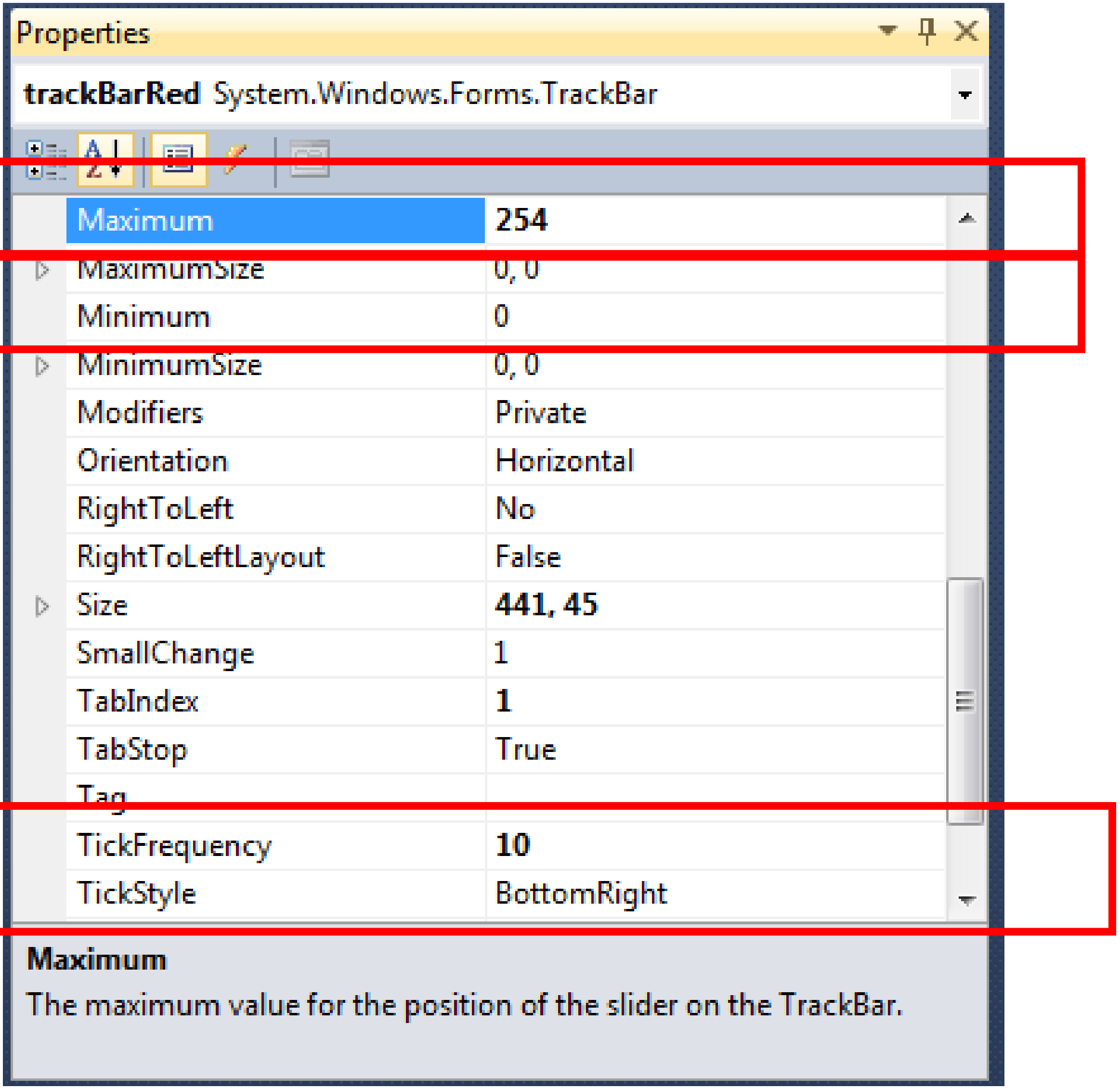
Color Selection:

Red

Green

Blue

For each trackbar, please set properties as the same figure



Properties

trackBarRed System.Windows.Forms.TrackBar

Maximum	**254**
MaximumSize	0, 0
Minimum	0
MinimumSize	0, 0
Modifiers	Private
Orientation	Horizontal
RightToLeft	No
RightToLeftLayout	False
Size	**441, 45**
SmallChange	1
TabIndex	**1**
TabStop	True
Tag	
TickFrequency	**10**
TickStyle	BottomRight

Maximum
The maximum value for the position of the slider on the TrackBar.

```
private void setColor(){
int nRed = trackBarRed.Value;
int nGreen = trackBarGreen.Value;
int nBlue = trackBarBlue.Value;
   txtRed.Text = nRed+"";
   txtBlue.Text = nBlue + "";
   txtGreen.Text = nGreen+"";
pnColor.BackColor = Color.FromArgb(nRed,
nGreen,nBlue);
}
```

Create Event or trackbar control

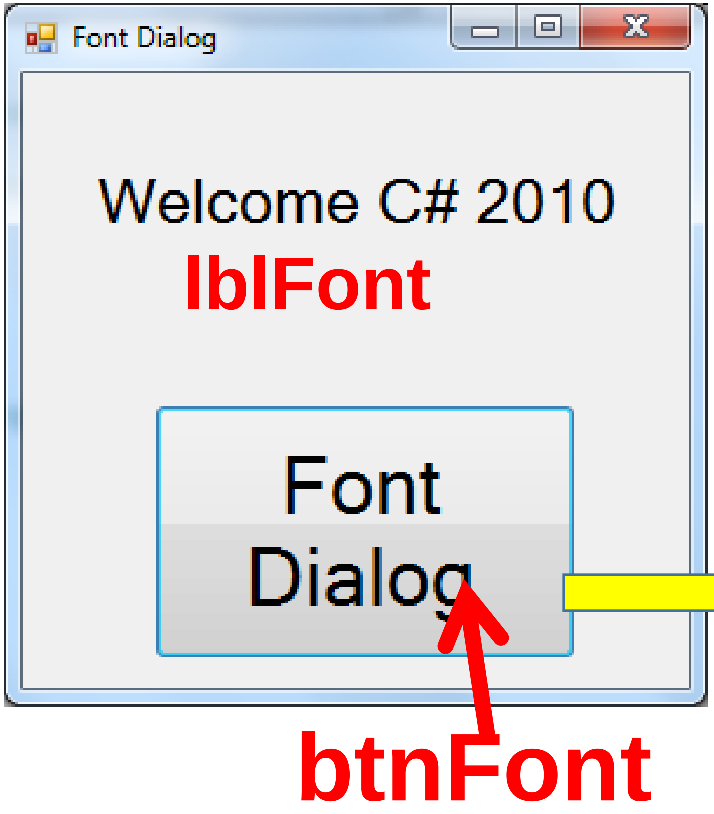
```
private void processTrackBar
(object sender, EventArgs e)
    {
      setColor();
    }
```

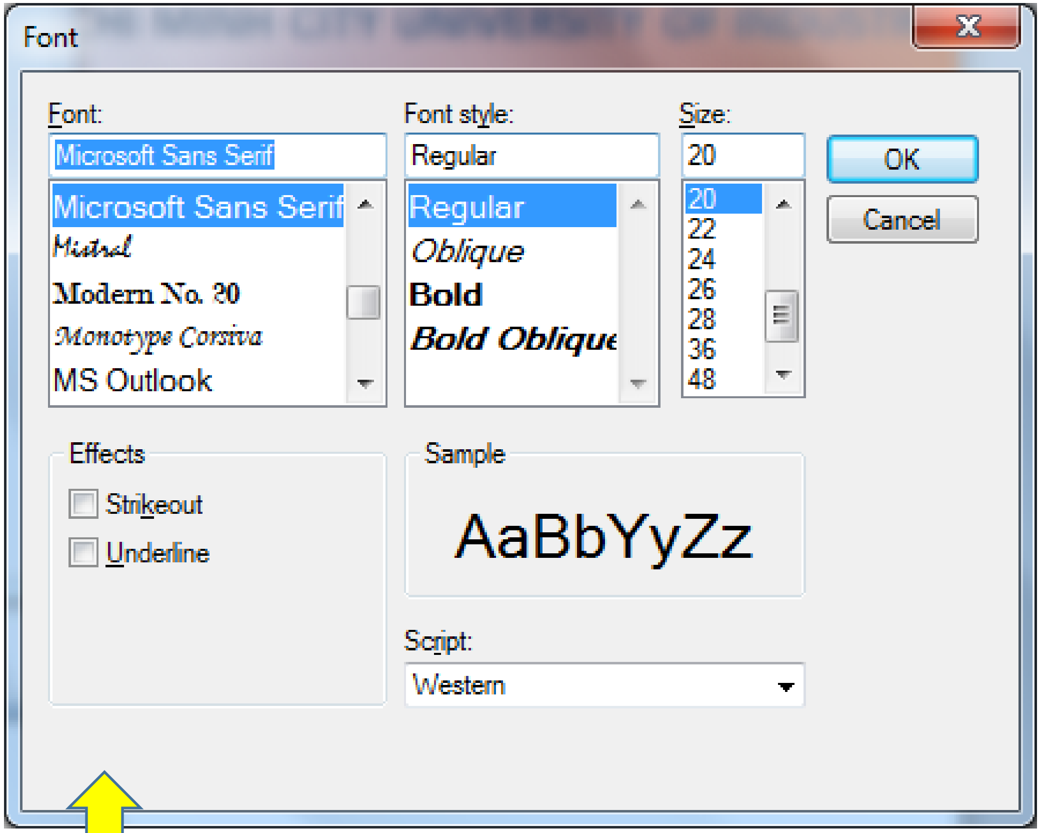
```
private void frmColorDialog_Load
(object sender, EventArgs e)
{ trackBarBlue.Scroll  += processTrackBar;
  trackBarGreen.Scroll += processTrackBar;
  trackBarRed.Scroll += processTrackBar;
  setColor();
}
```

Name	Description
Maximum	The Maximum property sets the value of the track bar when the slider is all the way to the right
Minimum	Gets or sets the lower limit of the range this TrackBar is working with.
Value	Gets or sets a numeric value that represents the current position of the scroll box on the track bar

Name	Description
LargeChange	Sets how many positions to move if the bar is clicked on either side of the slider
SmallChange	sets how many positions to move if the keyboard arrows are used to move the slider
TickFrequency	Establishes how many positions are between each tick-mark

FontDialog





Font

Font: Microsoft Sans Serif

- Microsoft Sans Serif
- Mistral
- Modern No. 20
- Monotype Corsiva
- MS Outlook

Font style: Regular

- Regular
- Oblique
- Bold
- Bold Oblique

Size: 20

- 20
- 22
- 24
- 26
- 28
- 36
- 48

Effects

- □ Strikeout
- □ Underline

Sample

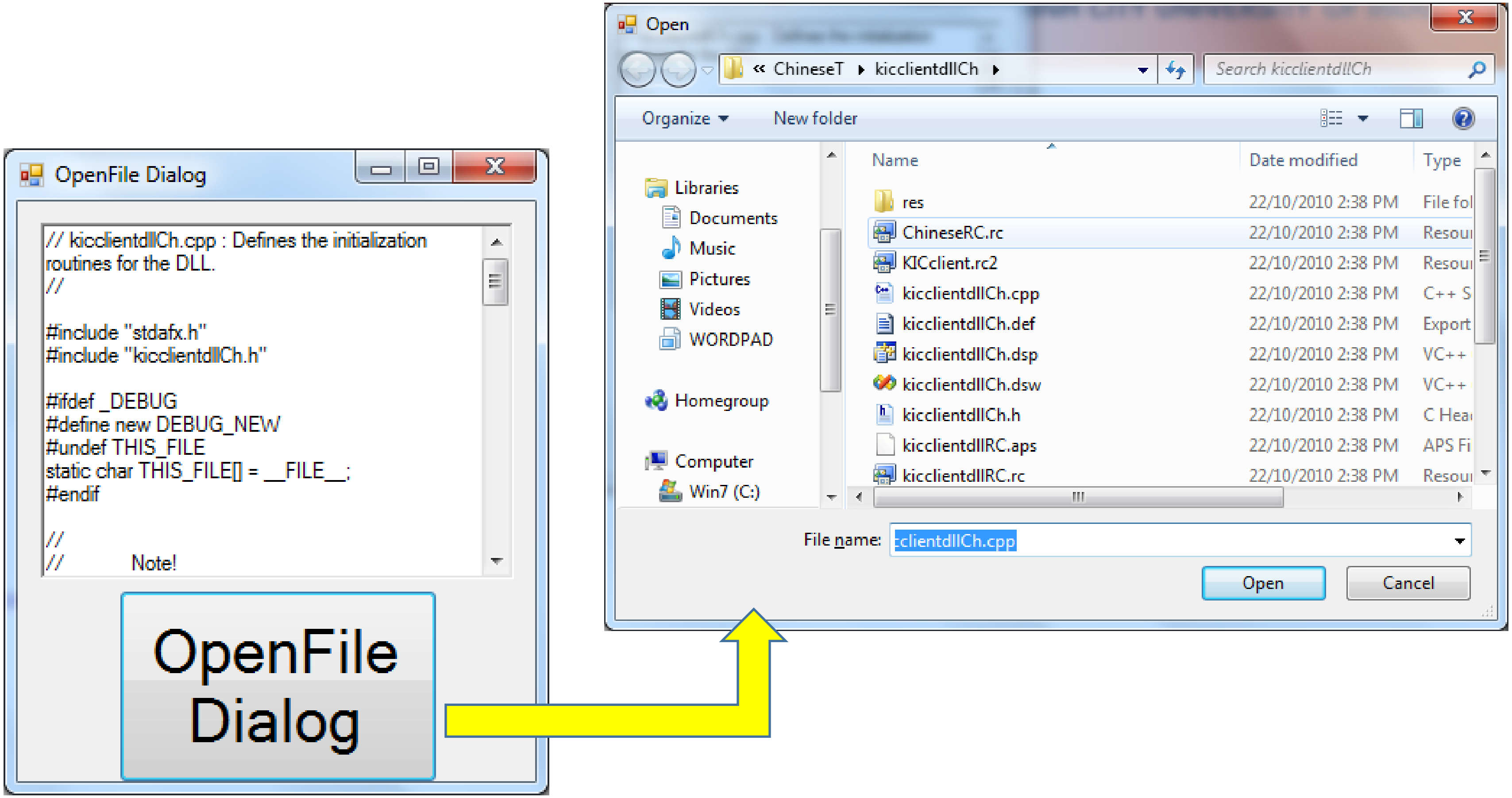
AaBbYyZz

Script: Western

How to use FontDialog?

```
private void btnFont_Click
(object sender, EventArgs e)
{
 FontDialog ftDialog = new
               FontDialog();
ftDialog.Font = lblFont.Font;
if (ftDialog.ShowDialog() ==
     DialogResult.OK)
 {
  lblFont.Font = ftDialog.Font;
 }
}
```

RichTextBox & StreamFile

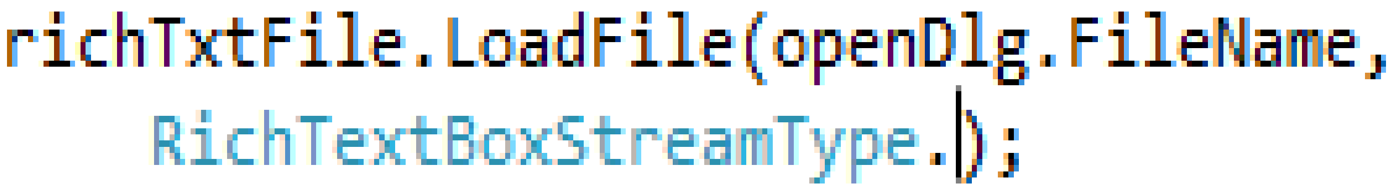


```
// kicclientdllCh.cpp : Defines the initialization
routines for the DLL.
//

#include "stdafx.h"
#include "kicclientdllCh.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

//
//          Note!
```

Name	Date modified	Type
res	22/10/2010 2:38 PM	File fol
ChineseRC.rc	22/10/2010 2:38 PM	Resour
KICclient.rc2	22/10/2010 2:38 PM	Resour
kicclientdllCh.cpp	22/10/2010 2:38 PM	C++ S
kicclientdllCh.def	22/10/2010 2:38 PM	Export
kicclientdllCh.dsp	22/10/2010 2:38 PM	VC++
kicclientdllCh.dsw	22/10/2010 2:38 PM	VC++
kicclientdllCh.h	22/10/2010 2:38 PM	C Hea
kicclientdllRC.aps	22/10/2010 2:38 PM	APS Fi
kicclientdllRC.rc	22/10/2010 2:38 PM	Resour

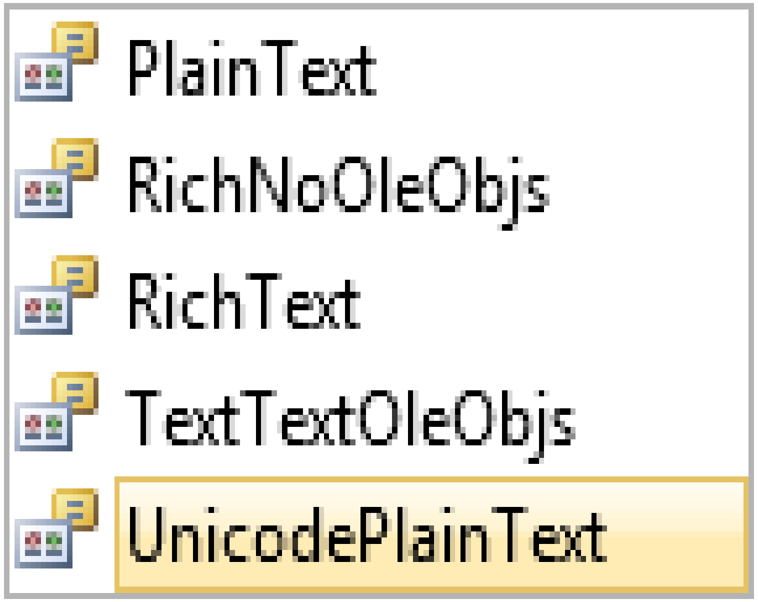
File name: cclientdllCh.cpp

```
OpenFileDialog openDlg = new
OpenFileDialog();
openDlg.Filter = "(*.txt)|*.txt|(All)|*.*";
if (openDlg.ShowDialog() ==
DialogResult.OK)
{
rtFile.LoadFile(openDlg.FileName,
RichTextBoxStreamType.PlainText);
}
```



```
richTxtFile.LoadFile(openDlg.FileName,
    RichTextBoxStreamType.);
```

You could choose StreamType



- PlainText
- RichNoOleObjs
- RichText
- TextTextOleObjs
- UnicodePlainText

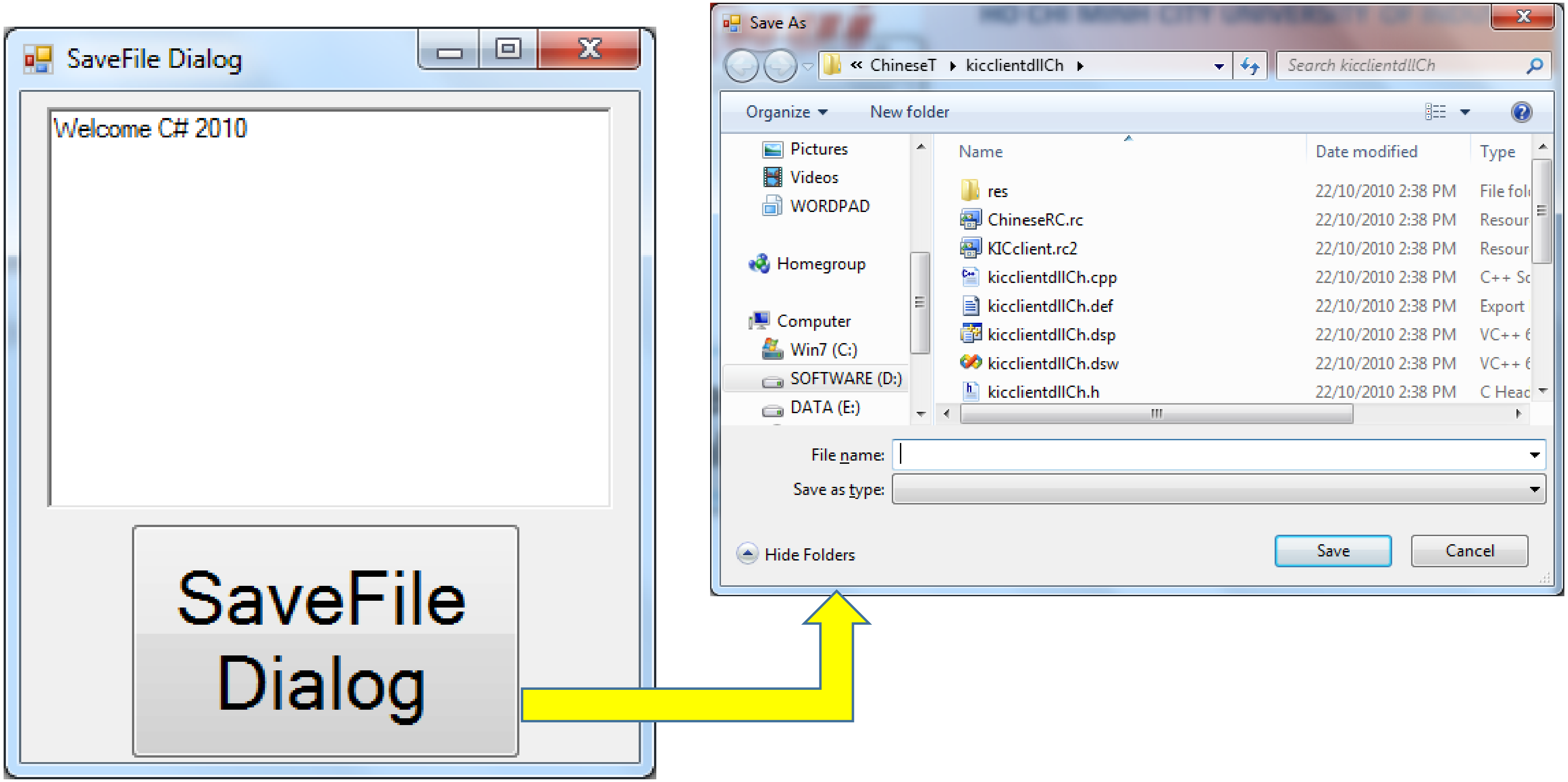
But We could use StreamFile to read Data.

You should Know this

Add this command at the top File

```
using System.IO;

if (openDlg.ShowDialog() ==
DialogResult.OK)
 {                //Open to read
Stream stream = openDlg.OpenFile();
StreamReader reader = new
StreamReader(stream);
richTxtFile.Text = reader.ReadToEnd();
reader.Close();
}
```

RichTextBox & StreamFile



Name	Date modified	Type
res	22/10/2010 2:38 PM	File fol
ChineseRC.rc	22/10/2010 2:38 PM	Resour
KICclient.rc2	22/10/2010 2:38 PM	Resour
kicclientdllCh.cpp	22/10/2010 2:38 PM	C++ Sc
kicclientdllCh.def	22/10/2010 2:38 PM	Export
kicclientdllCh.dsp	22/10/2010 2:38 PM	VC++ 6
kicclientdllCh.dsw	22/10/2010 2:38 PM	VC++ 6
kicclientdllCh.h	22/10/2010 2:38 PM	C Head

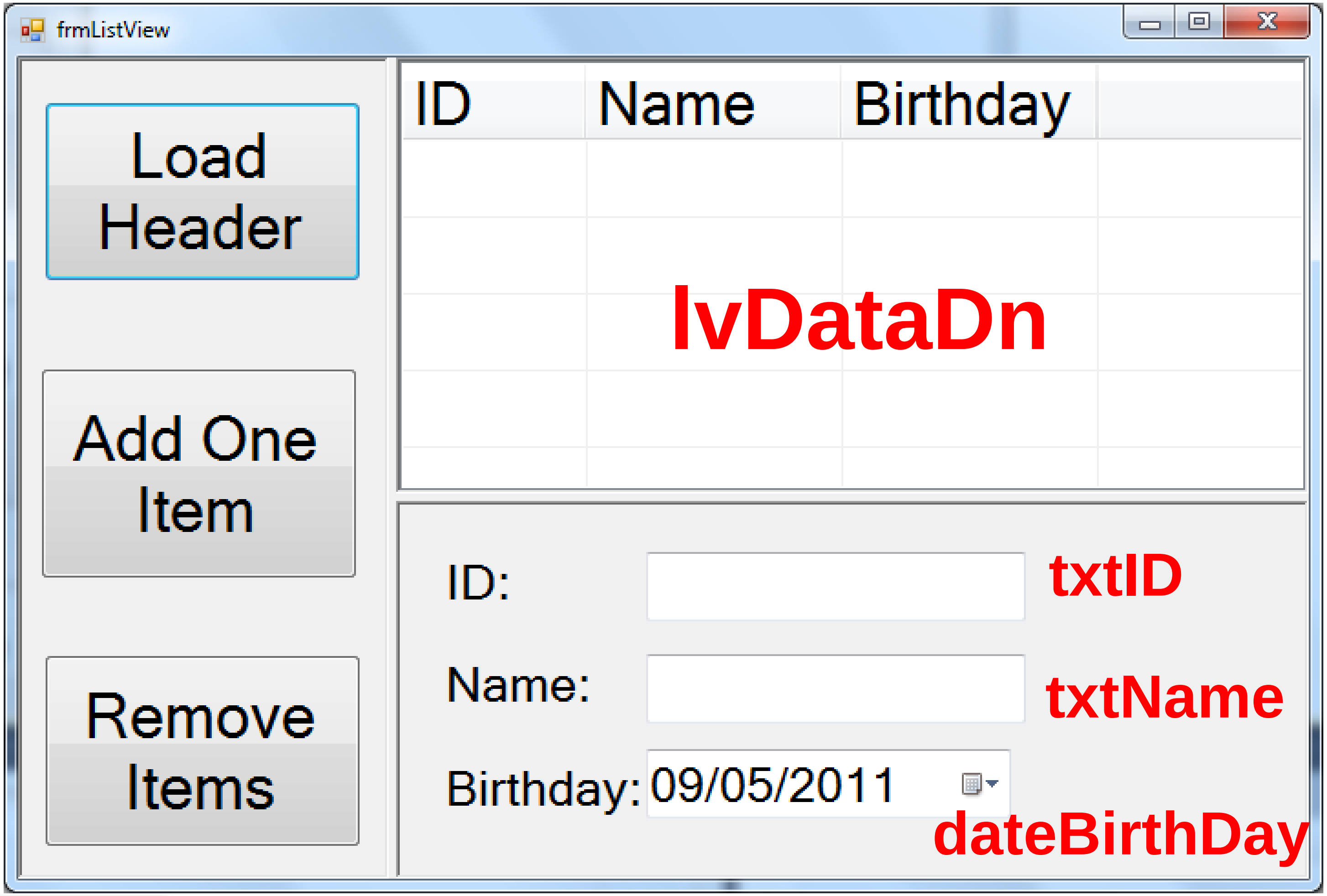
```
SaveFileDialog saveDlg = new
SaveFileDialog();
saveDlg.Filter = "(*.txt)|*.txt|(All)|*.*";
if (saveDlg.ShowDialog() ==
DialogResult.OK)
{
rtFile.SaveFile(saveDlg.FileName,RichTextBo
xStreamType.PlainText);
}
```

Use StreamFile to Write Data.

```
if (saveDlg.ShowDialog() ==
DialogResult.OK)
{
Stream stream = saveDlg.OpenFile();//Open
to Write
StreamWriter writer = new
StreamWriter(stream);
writer.WriteLine(rtFile.Text);
writer.Close();
}
```

Bài tập áp dụng

ListView



btnHeader

btnAdd

btnRemove

Birthday: 09/05/2011

```
 private void btnLoadHeader_Click
(object sender, EventArgs e)
 {
     ColumnHeader hdID = new
              ColumnHeader();
     hdID.Text = "ID";
     ColumnHeader hdName = new
              ColumnHeader();
    hdName.Text = "Name";
     ColumnHeader hdBirthday = new
              ColumnHeader();
     hdBirthday.Text = "Birthday";
```

```
lvDataDn.GridLines = true;
lvDataDn.FullRowSelect = true;
lvDataDn.MultiSelect = true;
lvDataDn.HideSelection = false;

lvDataDn.View = View.Details;

hdID.Width = 100;
hdName.Width =140; hdBirthday.Width = 140;
lvDataDn.Columns.Clear();
lvDataDn.Columns.AddRange(new
ColumnHeader[] {
hdID,hdName,hdBirthday});
}
```

At Design Time

????

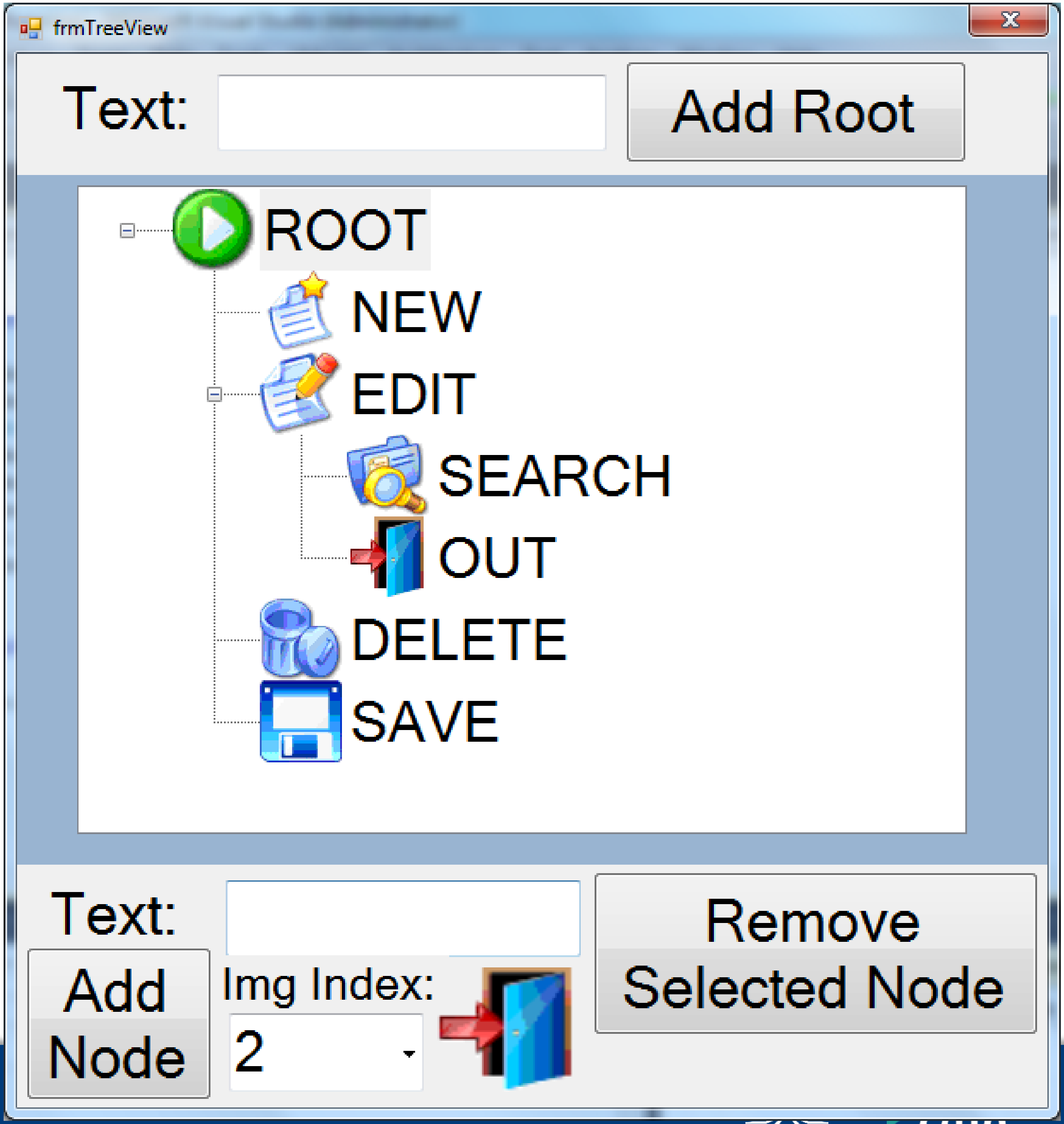
```
 private void btnAdd_Click
(object sender, EventArgs e)
{
ListViewItem itemID = new
     ListViewItem();
itemID.Text = txtID.Text;
ListViewItem.ListViewSubItem itemName  =
new
ListViewItem.ListViewSubItem();
itemName.Text=txtName.Text;
itemID.SubItems.Add(itemName);
ListViewItem.ListViewSubItem itemBirthday =
new ListViewItem.ListViewSubItem();
            itemBirthday.Text =
dateBirthDay.Value.Day + "/" +
dateBirthDay.Value.Month + "/" +
dateBirthDay.Value.Year;

itemID.SubItems.Add(itemBirthday);
            lvDataDn.Items.Add(itemID);
}
```

```
 private void btnRemove_Click
(object sender, EventArgs e)
{ListView.
SelectedListViewItemCollection
list = lvDataDn.SelectedItems;
foreach (ListViewItem item in list)
     lvDataDn.Items.Remove(item);
}
```

We could use Indices

```
ListView.SelectedIndexCollection
indices =
lvDataDn.SelectedIndices;
foreach (int index in indices)
{
    //Process Index here
}
```

Tree View



- ROOT
  - NEW
  - EDIT
    - SEARCH
    - OUT
  - DELETE
  - SAVE

Img Index: 2

Popular Properties & Method

ImageList

ImageIndex

SelectedImageIndex

TreeNode

SelectedNode

Nodes

Add

Remove

Expand

Collaps
e

```
private void frmTreeView_Load
(object sender, EventArgs e)
{cboImageIndex.Items.Clear();
 for (int i = 0; i <
    imgList.Images.Count;i++ )
   {
    cboImageIndex.Items.Add(i);
   }
  tvSample.ImageList = imgList;
}
```

```
private void btnAddRoot_Click
(object sender, EventArgs e)
{tvSample.Nodes.Clear();
TreeNode rootNode = new
TreeNode(txtRoot.Text);
rootNode.ImageIndex =
Int32.Parse(cboImageIndex.Text);
rootNode.SelectedImageIndex =
rootNode.ImageIndex;
tvSample.Nodes.Add(rootNode);
txtRoot.Text = "";
}
```

```
private void
cboImageIndex_SelectedIndexChanged
(object sender, EventArgs e)
 {

picShow.Image=imgList.Images[Int32
.Parse(cboImageIndex.Text)];
}
```

```
private void
btnRemove_Click(object sender,
EventArgs e)
 {
  TreeNode tNode =
         tvSample.SelectedNode;

  tvSample.Nodes.Remove(tNode);
}
```

Bài tập áp dụng