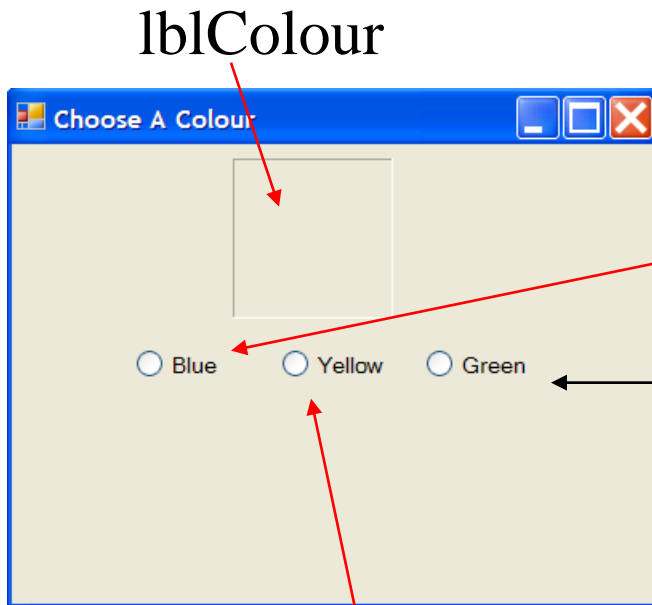


Radio Buttons and Labels



lblColour

radBlue

radGreen

radYellow

```
Public Class Form1
```

```
Private Sub radBlue_CheckedChanged(ByVal  
    lblColour.BackColor = Color.Blue
```

```
End Sub
```

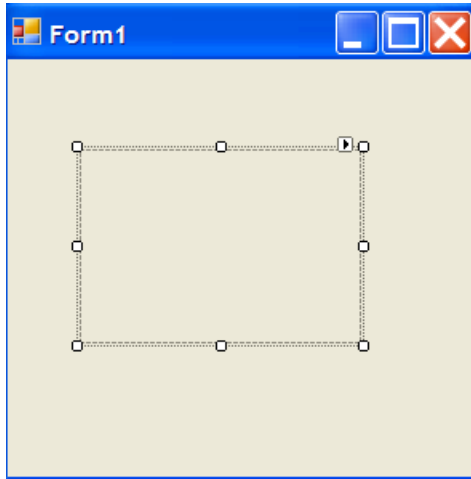
```
Private Sub radYellow_CheckedChanged(ByVal  
    lblColour.BackColor = Color.Yellow  
End Sub
```



```
Private Sub radGreen_CheckedChanged(ByVal  
    lblColour.BackColor = Color.Green  
End Sub
```

```
End Class
```

The PictureBox Tool

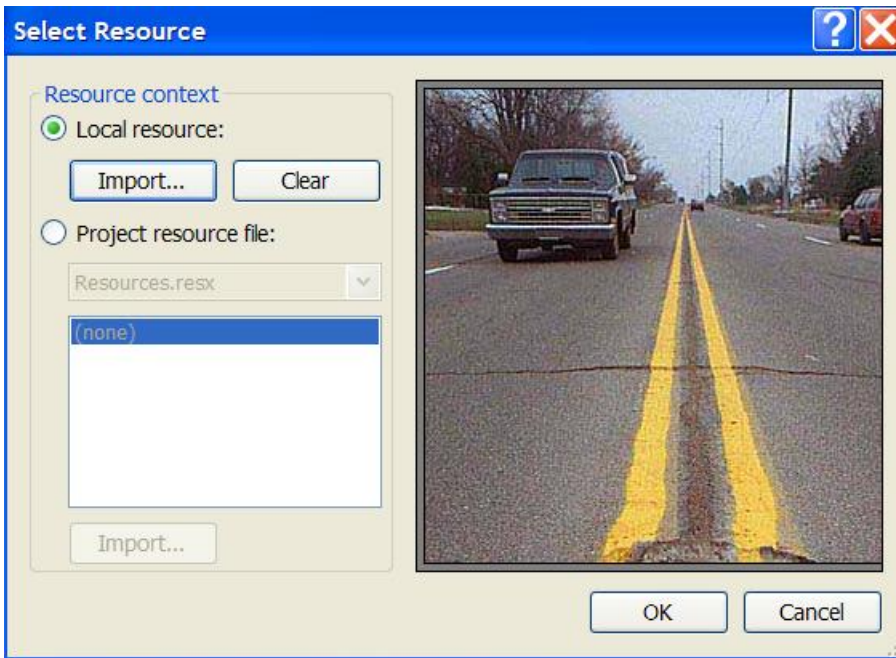
First, choose the PictureBox tool and draw an object of type PictureBox on the form. Lets call the object picOne. Notice the 'pic' prefix that we will use for naming PictureBox objects.



ContextMenuStrip	(none)
Cursor	Default
Dock	None
Enabled	True
ErrorImage	 System.Drawing.Bitmap
GenerateMember	True
Image	 System.Drawing.Bi ...
ImageLocation	

Now, with the picOne highlighted, go to the Image property and click on the ellipsis button.

Browse For Images



Making sure the Local resource button is chosen, click the Import... button and browse for the image you want to use. Then click okay.

Usually the image will not fit exactly in the PictureBox object.

You have two ways to fix this.

You can resize the PictureBox until it is the correct size for the image or you can set the 'SizeMode' property of the PictureBox objects to StretchImage.

This will adjust the image so that it fits into the PictureBox.

The Visible Property

Many objects that can be created have a visible property that can be set to true or false. PictureBox objects have this property which can be very handy in making images appear and disappear. You should remember that it is the whole PictureBox that is disappearing, not just the image within it.



```
Private Sub btnShow_Click(ByVal sender As  
    picRoadKill.Visible = True  
End Sub
```

```
Private Sub btnHide_Click(ByVal sender As  
    picRoadKill.Visible = False  
End Sub
```

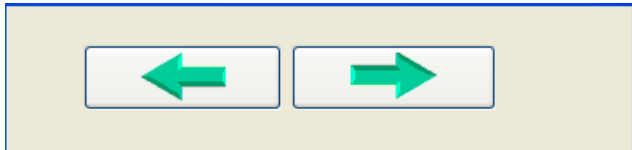
Using Images For Other Types of Objects

Other tools than just PictureBox objects can have images assigned to them. For instance, by adding images to buttons you can improve the visual appeals of your projects.


Start by creating or downloading your image. It is best to put the image file in a folder that is within your VB project folder.



When you run your project, VB looks for the directory you used when you first attached the image. If you move the project folder and the image isn't within the folder, the image will get lost.



With the button object highlighted, click on the image property and browse for your image.

FlatAppearance	
FlatStyle	Standard
Font	Microsoft Sans Serif, 7.8pt
ForeColor	ControlText
GenerateMember	True
Image	 System.Drawing.Bitmap
ImageAlign	MiddleCenter
ImageIndex	(none)

The Assignment Operator

=

This symbol, which is commonly referred to as the equals sign in math, has a different meaning in programming.

We call this symbol the assignment operator in Computer Science.

Whatever is on the left side of the assignment operator is assigned the value of whatever is on the right side.

$x = 5$ We now say x has been assigned the value of 5.

This might seem like a silly and unnecessary difference in terminology but look at this at this example.

$x = x + 1$

Mathematically, this statement is not possible. x cannot be EQUAL to $x + 1$, however...

x can be ASSIGNED the value of $x + 1$.

If x has an initial value of 5, assigning the value of $x + 1$ to x is the same as saying x now has a value of $5 + 1$.

Put another way...

if $x = x + 1$

x now has a value of 6

Practicing Assignment Statements

For each, assume x
has an initial value
of 3

	New value of x
$x = x + 2$	5
$x = x - 2$	1
$x = x * x$	9

Using val in a Statement

There will often be cases when numbers are entered in a text box. Though the entries look like numbers, it should be noted that the so called numbers are still really text.

To use them in calculations a programmer should precede the numeric text wrapped in brackets with the word val.

`txtAnswer.Text = txtNumberOne.Text + 4` wrong!!

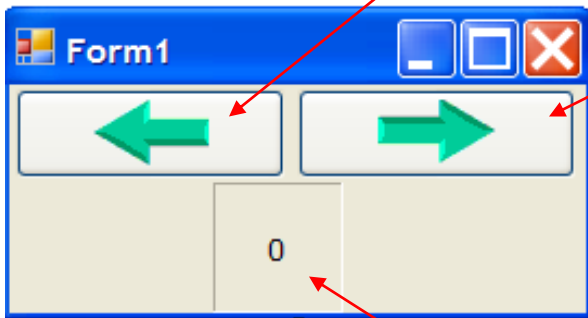
`txtAnswer.Text = val(txtNumberOne.Text) + 4` right!!

Converting Text Input for Numeric Calculations

subtracts 1 to the number

```
Private Sub btnSubtract_Click(ByVal sender As  
    lblCount.Text = Val(lblCount.Text) - 1  
End Sub
```

```
Private Sub btnAdd_Click(ByVal sender As System  
    lblCount.Text = Val(lblCount.Text) + 1  
End Sub
```



converts the text to numeric form for calculations.

looks at the text number in the label

adds 1 to the number

Assigns the new value as a text property to lblCount