

UNIT 6

CHAPTER 1: (Overview VB.NET)

Lesson 36: (Other VB.NET Application)



Forms now have an Opacity property. Is it a necessary feature? Probably not, but it could be used for some useful purpose. Probably. Somewhere. At least while you're working with it, you'll look busy.

On the Form1 you've been working with so far, find its Opacity property and change it to 50 percent. Now run the project and you'll notice that the form is now quite see-through.

The form still works fine (not that this current form is doing anything). You can click and drag the form,; you can press the buttons.

It works fine, but it is translucent at the moment. Close the project, and you can add some code to take advantage of this new, if not quite critical, feature.

Go to the Toolbox and add a timer to your form. Unlike VB6, the timer doesn't actually appear on the form, but it appears in a small window below the form. Click on the timer in this window and then set the timer's Enabled property to True.

The interval of 100 is fine.

Now click on the Form1.vb tab to get to the code. In the Sub New routine, add the following line of code right after the TODO comment: Form1 .Opacity = 0

This merely sets the opacity to 0, which means that the form will be completely invisible when the form loads.

Next, use the Class Name and Method Name drop-down boxes at the top of the code window to choose the Tick event of the Timer 1 control. This adds an event procedure for the Timer1.Tick event. Inside that event procedure, enter the following code:

```
Form1 .Opacity = form1 .Opacity + 0.01 If form1.Opacity >= 1 Then
timer1.Enabled = False
beep()
End If
```

This code merely increments the opacity by .01 each time the Tick event fires. When it finally reaches 1 (100 percent), setting its Enabled property to false shuts off the timer, and the beep is called just to make a sound so that you know when it is finished.

If you are at all confused by the code, Listing 2 shows how the whole listing will look (with the Windows Form Designer generated code hidden).



Listing Your First VB.NET Application, with All the Code Discussed So Far

```
Imports System.ComponentModel
Imports System.Drawing
Imports System.Windows
Public Class Form1 »
    Inherits System.Windows.Forms.Form
    Public Sub New() MyBase.New
        Form1 = Me
        'This call is required by the Win Form Designer.
        InitializeComponent
        TODO: Add any initialization after the InitializeComponent() call
        Form1.Opacity = 0
    End Sub
    Form overrides dispose to clean up the component list.
    Overrides Public Sub Dispose() '-
        MyBase.Dispose
        components.Dispose
    End Sub
```

```
' Windows Form Designer generated code Public Sub Timer1_Tick(ByVal sender As Object, _  
ByVal e As System.EventArgs) Handles Timer1.Tick form1.Opacity  
= form1.Opacity + 0.01  
If form1.Opacity >= 1 Then  
    timer1.Enabled = False  
    beep()  
End If  
End Sub  
End Class
```

Run the project. You should see your form fade in slowly as the opacity is increased. You will hear a beep when the form is fully opaque.

Summary



This part of the course pack sought to get you into the new VB.NET IDE quickly. You have created your first VB.NET application, and you have seen some of the new features presented in the IDE. It is now easier to have controls maintain their positions even when the form is resized. Help is more readily available. The Server Explorer makes it easier to locate and use resources throughout your enterprise. New project types open up a new world for VB.NET developers.

References:



Sams teach yourself visual basic net in 21 days

- Duncan Mackenzie and Kent Sharkey

Using visual basic .net

- Brian Siler and Spotts