
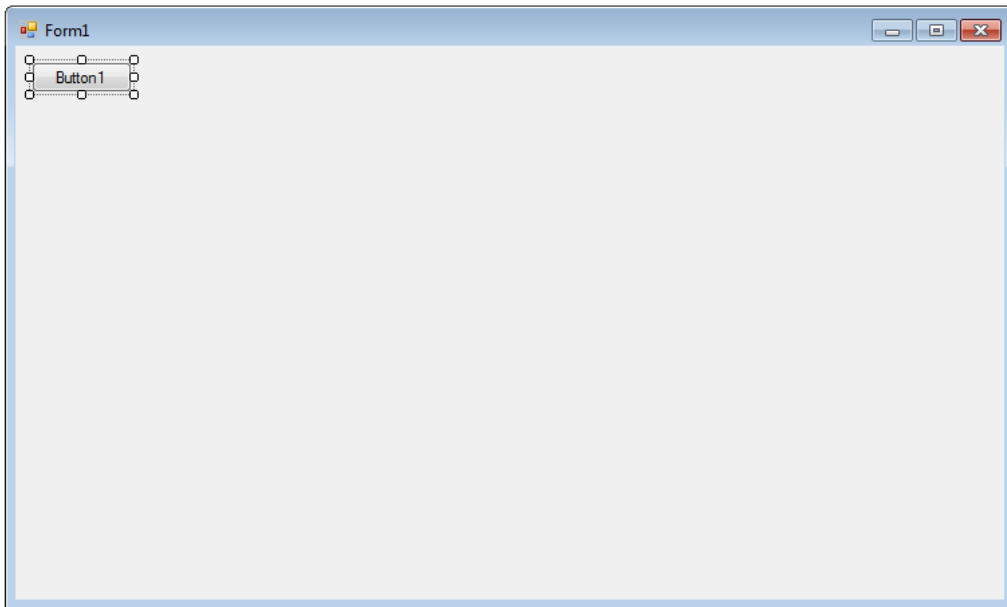



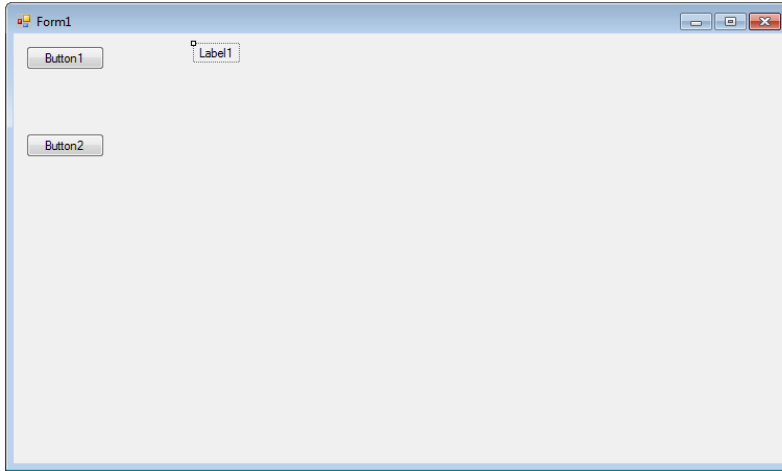
Designing the Lucky Seven Game

In this lesson, you will start building the Lucky Seven game by creating a new project and then using controls in the toolbox to build the user interface.

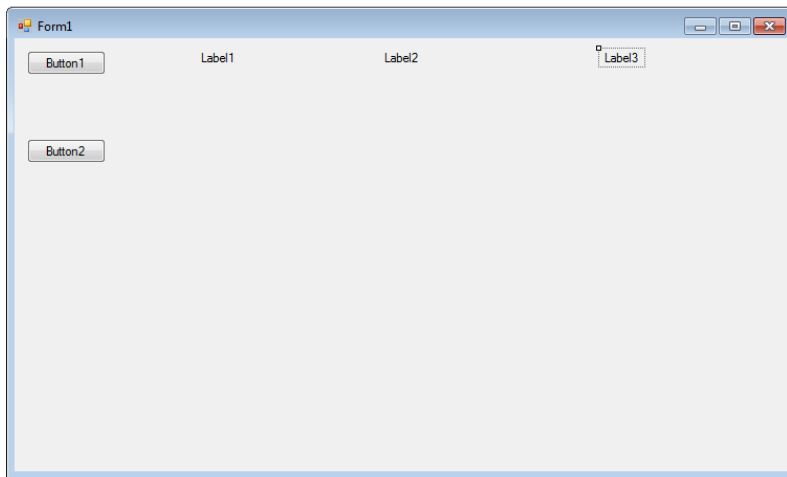
1. Start Visual Basic. On the **File** menu, click **New Project**. If Visual Basic opens to the New Project window, name the project lucky 7 game.
2. Click the **Form1** window to select it. Enlarge the form by dragging on the handles to make room for the objects in the program. As you resize the form, scroll bars will appear as shown.
3. Resize the Project window so you can display the Toolbox and the Properties window.
4. Click the **Button** control  in the toolbox. Move the mouse pointer over the form. The mouse pointer changes to crosshairs when it rests on the form. The crosshairs are designed to help you draw the rectangular shape of a button. Hold down the left mouse button and drag. The button object takes shape and snaps to the grid formed by the intersection of dots on the form. Create your first command button close to the upper left corner of the form as shown. By default, the name of the button is **Button1**.



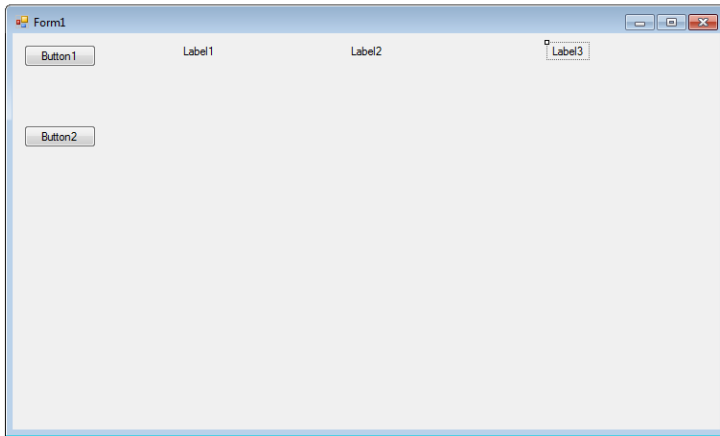
5. Add a second button below the first button on the form. Make the second button the same size as the first. You may move or resize the button after you place it by dragging on the handles.
6. Click the **Label** control  in the toolbox, and then place the mouse pointer over the form. Drag to create a small, rectangular box like the one shown below.




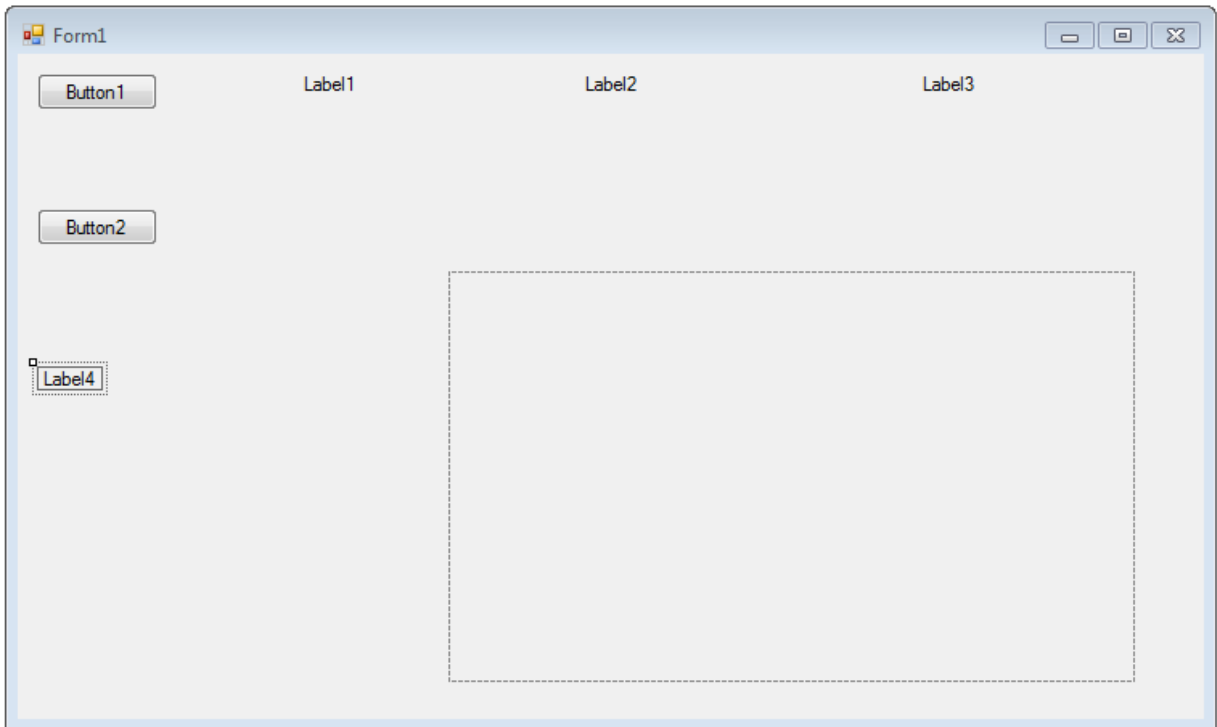
7. By default, the label you created will be called **Label1**, the first label in the program. Create two more labels beside the first one, as shown. Make all three labels the same size.



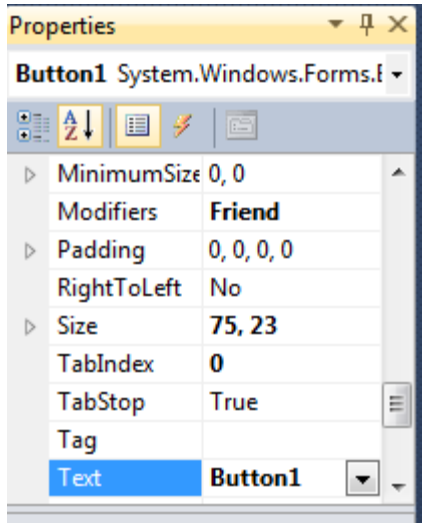
8. Create a larger **Label4** directly below the two command buttons, as shown.



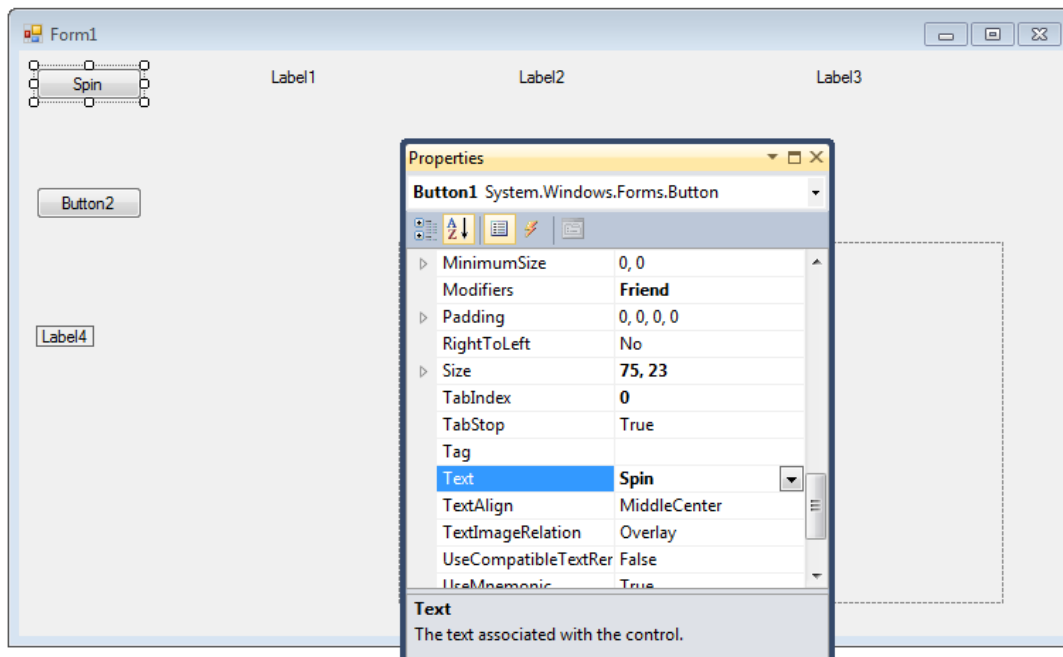
- Next, you will add an image box to the form to display the jackpot the user wins when a lucky seven is drawn. Before you begin, click [here](#) to download the image to your computer. Save the image to your **My Documents/My Pictures folder**.
- Select the **Picture** control  in the toolbox. Create a large rectangular box directly beneath the Label1, Label2 and Label3 controls, as shown.



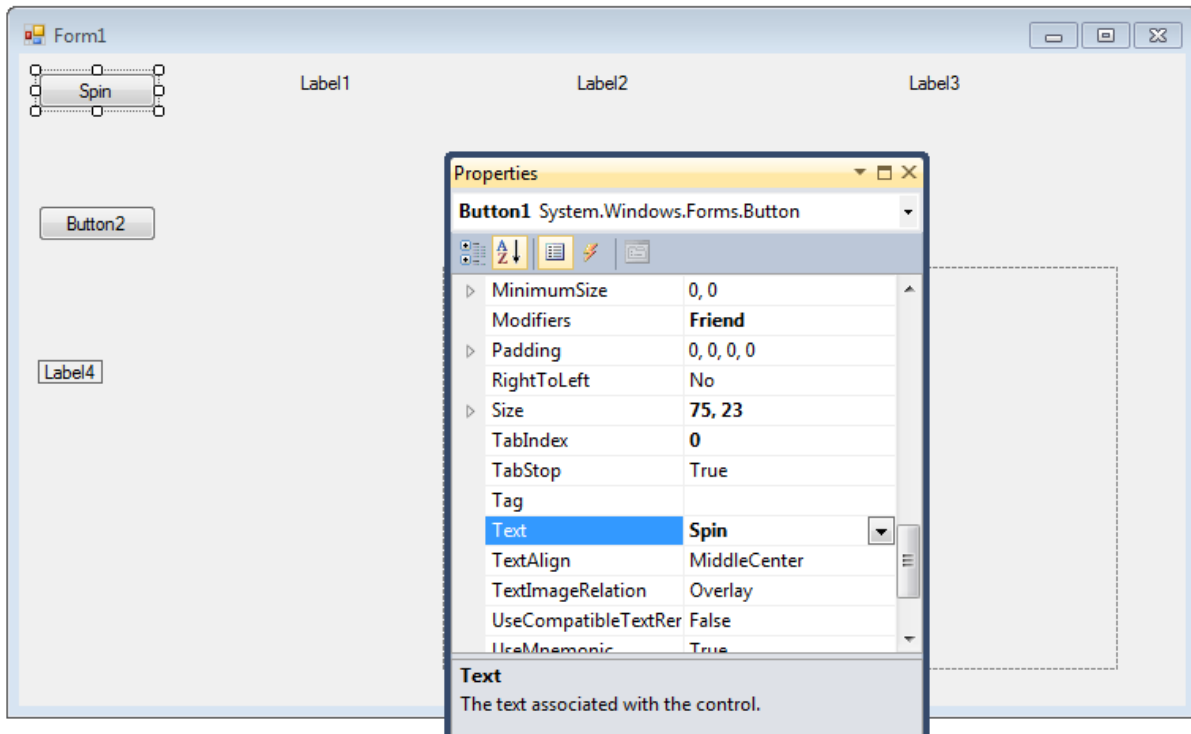
- Now we are ready to set the properties for these controls. Open the **Properties** window, if it is not already open.
- Select the first button, called Button1. Once selected, it will be surrounded by selection handles.
- In the **Properties** window, double-click the **Text** property in the left-hand column, as shown.



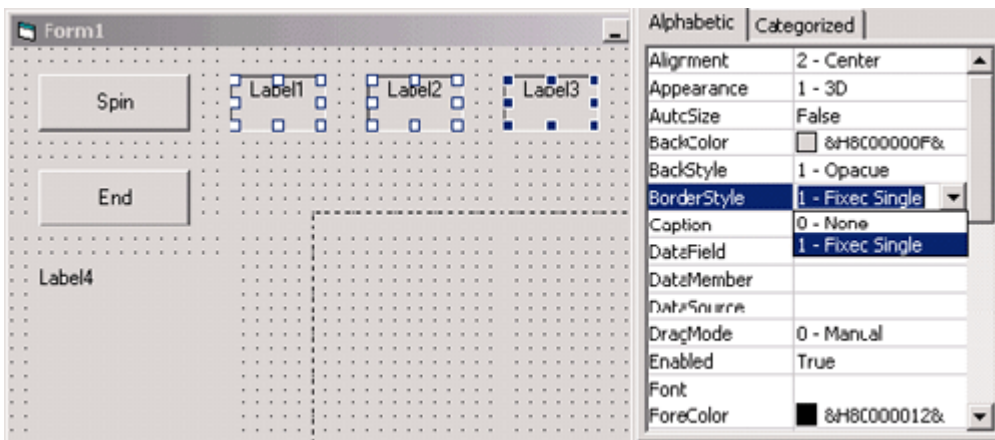
14. Notice that the current name, Spin, is also selected in the right-hand column. You will rename the button1 button. Type **Spin** and then press **Enter**. Notice that the text property changes to Spin in the Properties window and on your form, as shown.



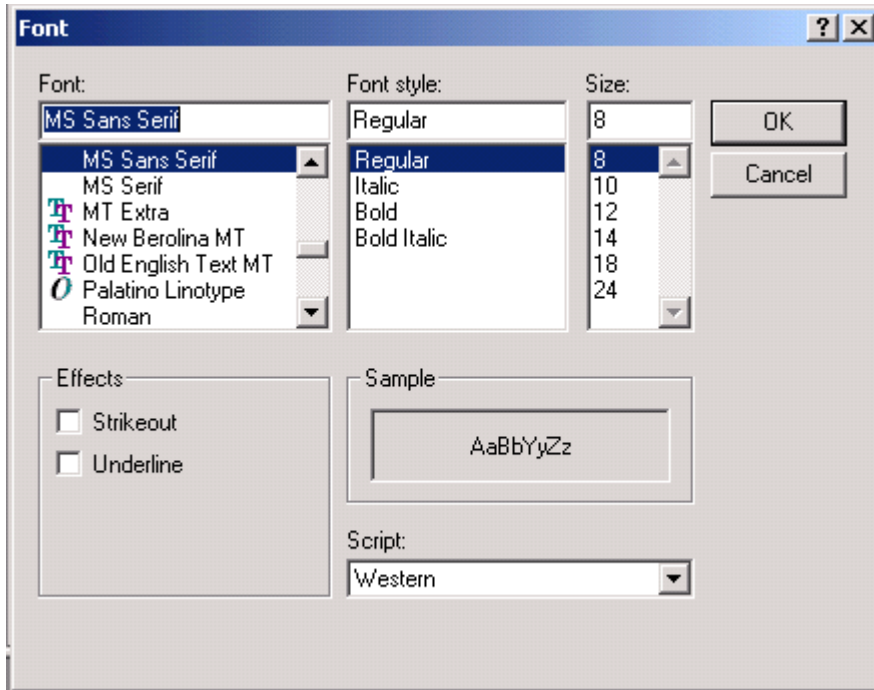
15. Select **Button2** and change the text to **End**, as shown.
16. Next, you will set the properties for Label1, Label2 and Label3. Click **Label1** to select it. Hold down the **Shift** key and click **Label2** and **Label3**. Selection handles will appear around each of the three labels. Release the Shift key.
17. In the Properties window, double-click the **ImageAlign** property. A drop-down list box appears in the right-hand column. Click the drop-down arrow to display a list of alignment options, as shown.



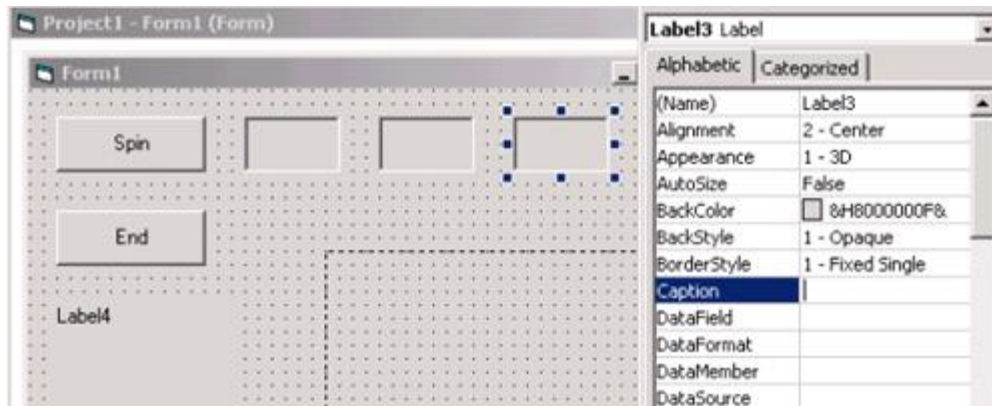
18. Click the **MiddleCenter** option from this list.
19. In the Properties window, double-click the **BorderStyle** property. Click the drop-down list arrow and select **1-Fixed Single**, as shown.



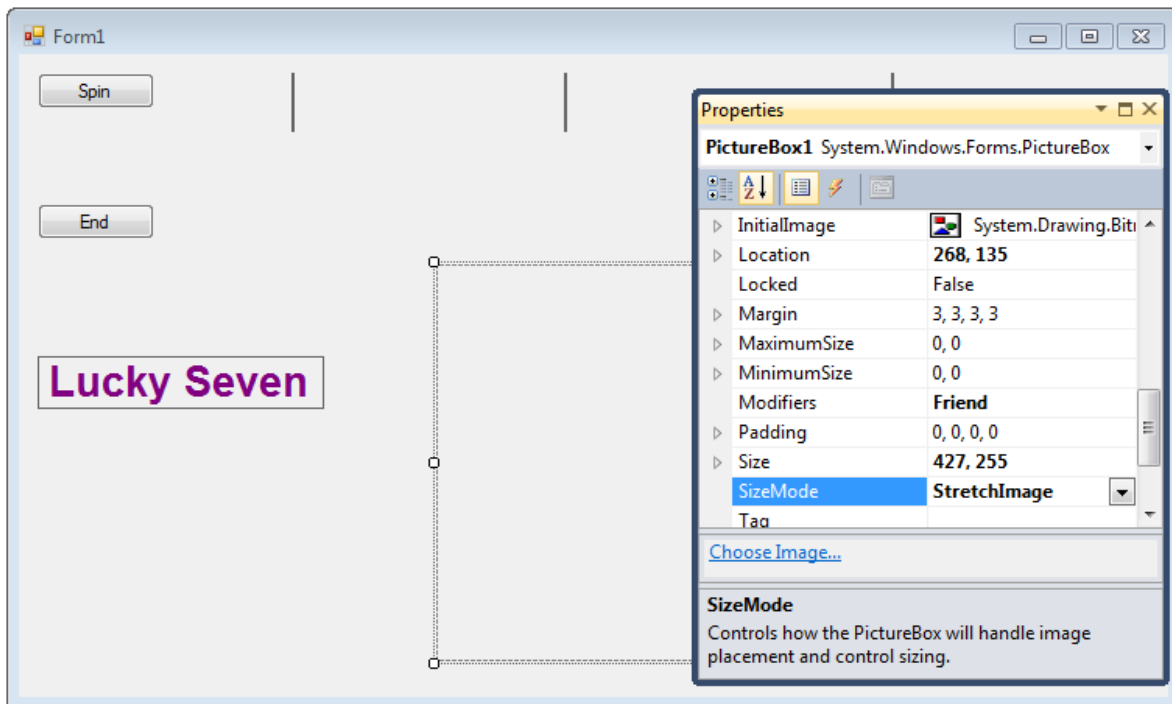
20. Double-click the **Font** property in the Property window. The **Font** dialog box appears, as shown.



21. Change the font to **Times New Roman**, the font style to **Bold** and the point size to **24**, then click **OK**.
22. Click the form to remove the selection handles from the three labels.
23. Select **Label1**. Double-click the **Caption (Text)** property in the Property window. Press the **Delete** key on the keyboard to delete the text (caption). Delete the text (captions) for Label2 and Label3, as shown.



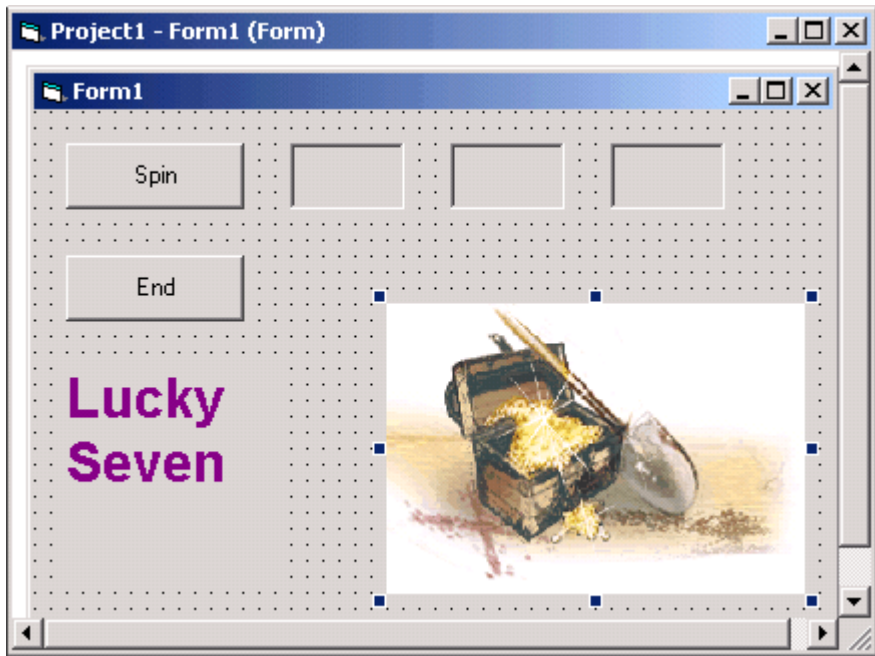
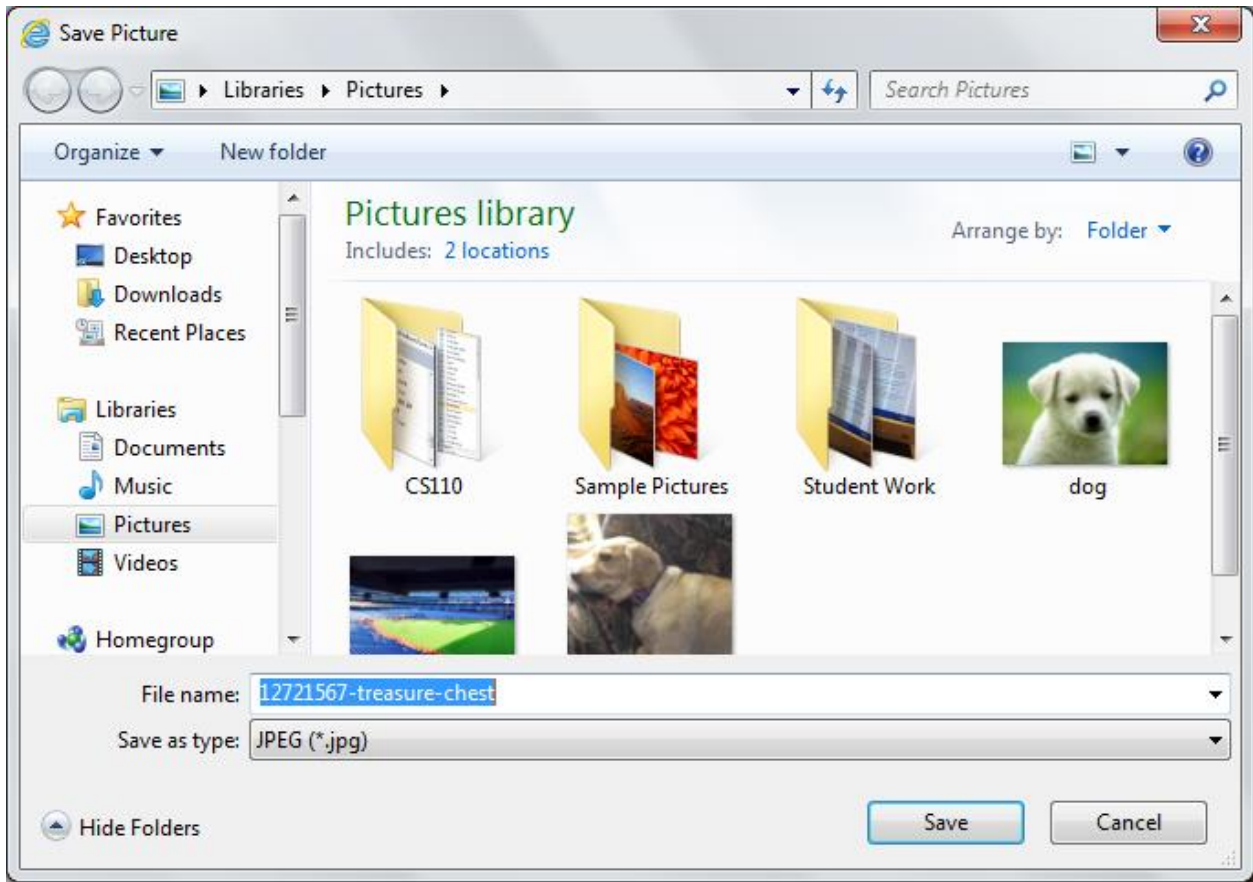
24. Now you will change the **Text**, **Font** and **Fore Color** properties on the last label, Label4. Click Label4 to select it.
25. Double-click the **Text** property and type **Lucky Seven** in the right-hand column.
26. Double-click the **Font** property and use the Font dialog box to change the font to **Arial**, the style to **Bold** and the point size to **20**. Click **OK**.
27. Double-click the **ForeColor** property in the Property window. A list box appears with a **System** tab and a **Custom** tab. Select the **Custom** tab, then click the box containing the dark purple color. All the properties that were changed in steps 24 to 27 are shown below.

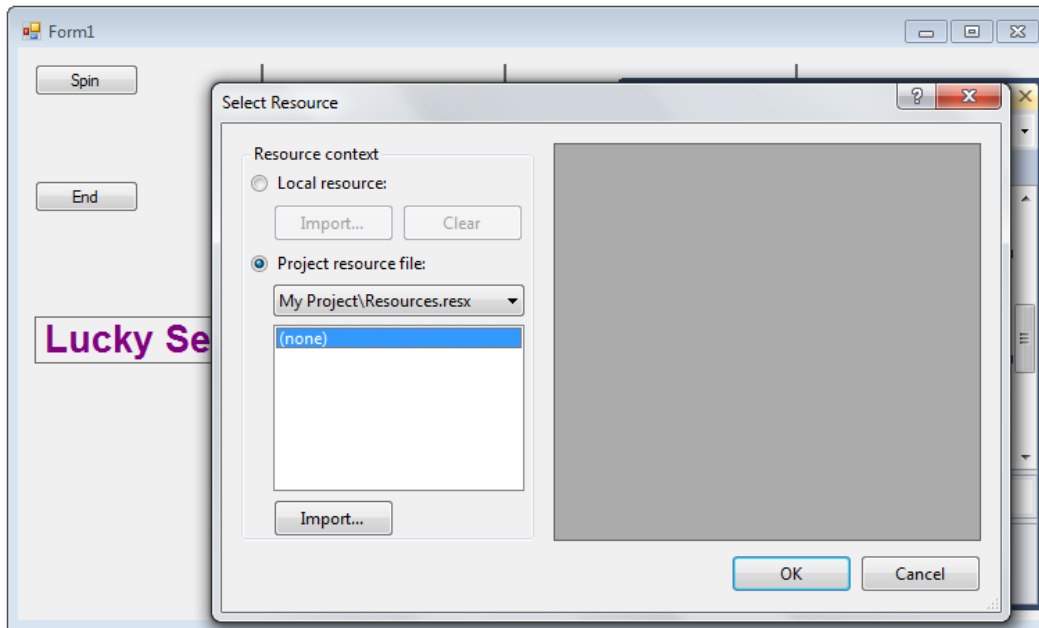


28. Next, you will set the **Size Mode** property to size the graphic that you will add to the form.
29. Select the image box. Double-click the **SizeMode** property, click the drop-down list arrow and select **StretchImage**, as shown. See above

30. Double-click the **Picture** property. The **Load Picture dialog box** appears, as shown.

31. Navigate to **My Documents/My Pictures**, as shown, and then select the image called **treasurechest_r06_c1** and click **Open**. The image is loaded into the image box on the form, as shown.





32. Now you will change the **Visible** property so that the treasure chest will be invisible when the program starts. Double-click the **Visible** property.
33. Click the drop-down list arrow and select **False**.
34. Now you are ready to write the code. Close the Properties window.
35. Double-click the **End** command button on the form. This will open the **Code** window, as shown.

36. Inside the code window, there are program statements that indicate the beginning and end of this particular Visual Basic subroutine, also known as an event procedure. A subroutine, or event procedure, is a block of code associated with a particular object in the interface. The body of the procedure always fits between these lines. The cursor is flashing on the line between:


```
Private Sub Command2_Click( )
End Sub
```

```
Private Sub Command2_Click( )
End Sub
```

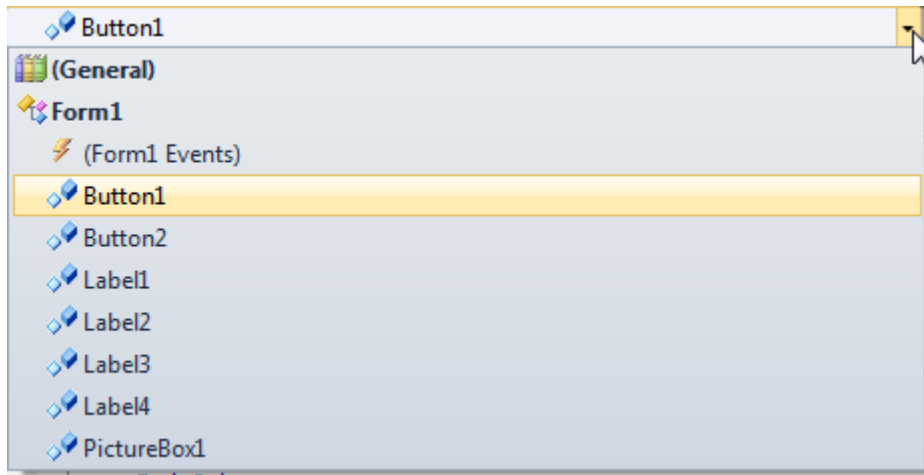
37. Type **Me.Close**, then press the **Down** arrow key on the keyboard, as shown.

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
    Me.Close()

End Sub
End Class
```

38. Move the cursor to the beginning of the line with the **Me.Close** statement. Press the spacebar **four** times to move the End statement four spaces to the right, as shown.

39. Now you will write the code for the **Button1** Spin button. Click the Object drop-down arrow and choose Command1 from the drop-down list, as shown.



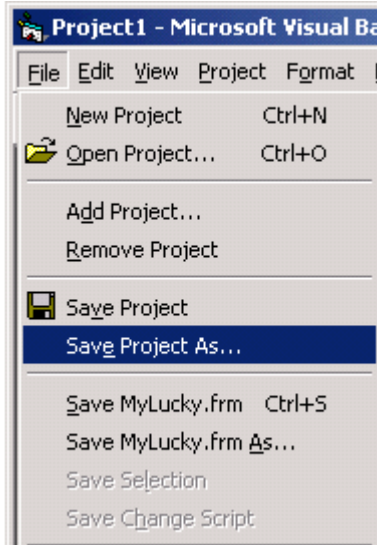
40. Type the lines of code **exactly as shown**.

```
PictureBox1.Visible = False 'hide coins
Label1.Text = Int (Rnd * 10) 'pick numbers
Label2.Text = Int (Rnd * 10)
Label3.Text = Int (Rnd * 10)
'if any text is 7 display treasure chest and beep
If (Label1.Text = 7) Or (Label2.Text = 7) _
Or (Label3.Text = 7) Then
PictureBox1.Visible = True
Beep
End If
```

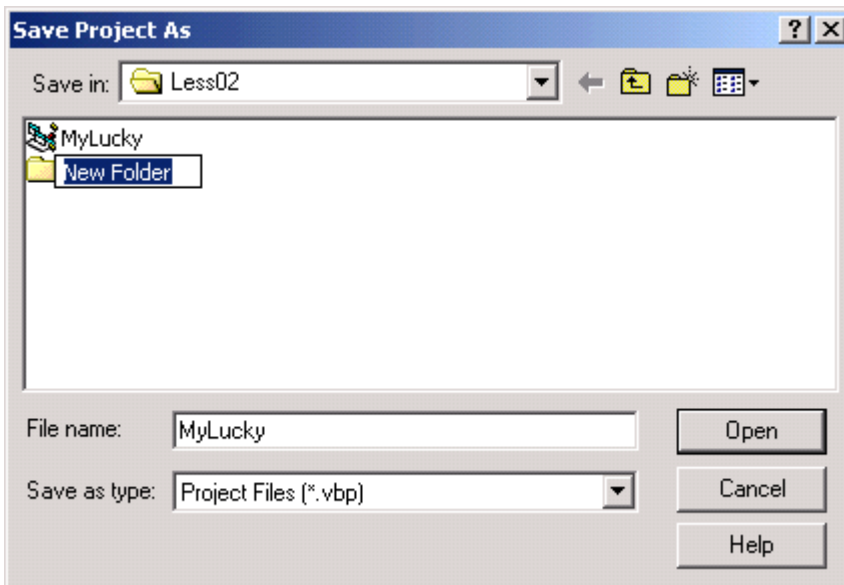
41. When you have finished, the code window should look exactly as shown.

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    PictureBox1.Visible = False 'hide coins
    Label1.Text = Int(Rnd() * 10) 'pick numbers
    Label2.Text = Int(Rnd() * 10)
    Label3.Text = Int(Rnd() * 10)
    'if any caption is 7 display treasure chest and beep
    If (Label1.Text = 7) Or (Label2.Text = 7) _
    Or (Label3.Text = 7) Then
        PictureBox1.Visible = True
        Beep()
    End If
End Sub
End Class
```

42. Students find this first attempt at entering code to be the most difficult part of the lesson. It must be typed **exactly** as shown or it will not work. Pay close attention to your typing. I promise you that if you enter it **exactly as shown**, it will work!
43. The last steps involve **Saving** our project and then running it. Click **File->Save Project as** just as in the image shown below.



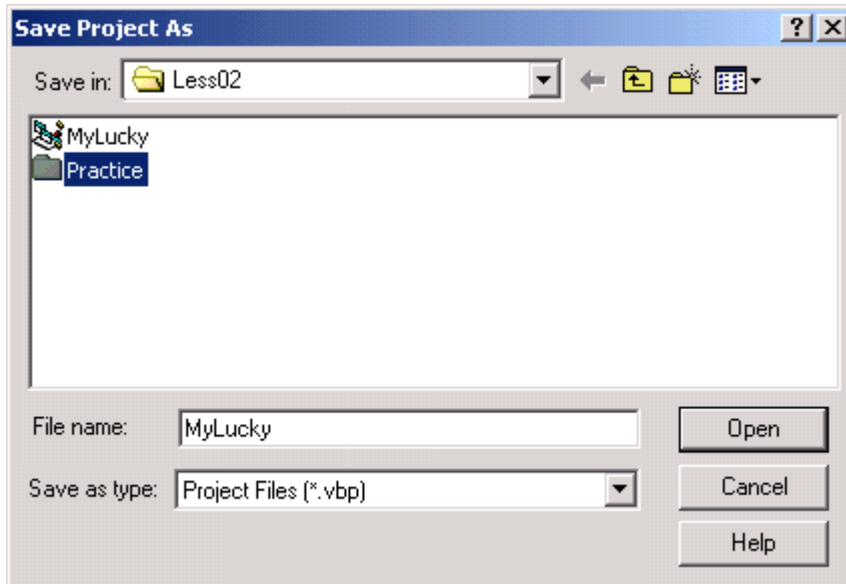
44. Click the **Create New Folder** button  on the toolbar and a new folder appears, as shown.



45. In the **New Folder** box, type **Practice** and press **Enter**. A new folder named Practice is created as shown.

Note: For the rest of the course, you will save all your projects in the Practice folder.

46. In the File name box, type **MyLucky** and press **Enter**. Your Lucky Seven form will be saved as **MyLucky . frm**.



47. The **Save Project As** dialog box appears again. The **Practice** folder should already be in the **Save in** box. In the **File name** box, type **MyLucky** and press **Enter**. This will save the project as **MyLucky .vbp**.
48. Congratulations! You are now ready to run your first Visual Basic program. Click **Run** on the menu bar and select **Start**. The game interface that you created appears just as you designed it. Click the **Spin** button. Three random numbers will appear in the boxes. If number 7 shows in one of the boxes, the image will also appear and your computer will beep. If the number 7 does not show in one of the boxes, the image will not appear and you will not hear a beep. Continue to click the spin button to play the game. When you are finished playing, click the **End** button to end the game.

