Lab 3 The High-Low Game

LAB GOALS

To develop a simple windows-based game named "High-Low" using VB .Net.

You will use: Buttons, Textboxes, Labels, Dim, integer, arithmetic operations, conditionals [if-then-else], random numbers [Randomize(), and Rnd() functions], type conversion functions [CInt (), CStr], Compiler directives such as Option Strict, Option Explicit, input validation and error checking [IsNumeric() function].



appropriate names, so that it is easier to

identify them when you are writing code behind each button. (e.g., ButtonSubmit, ButtonNewGame, and ButtonQuit)

Step 2: Double click the **Quit** button and write the code that allows your program to properly exit.

Close()

Now add the following two lines at the beginning of the program (before the line "Public Class Form1"):

Option Explicit On Option Strict On

Also add the following variable declarations to your program (immediately after the line "Public Class Form1"):

Dim UserGuess As Integer Dim RandomNumber As Integer Dim NumberOfGuesses As Integer

- Run: Compile and Run your program. If your program does not compile, fix the syntax errors and compile the program again. Once you are able to successfully run the program, check to see if the Quit button works properly.
- Step 3: Save your project and make sure it is saved on you Desktop. (if you have any doubt about this ask me in the lab)

Step 4: Download the following four images from our OnCource resources and save them on your desktop.

	📑 HighLow.jpg	
	Success.jpg	
	🛋 thumbs-down.jpg	
	嘴 thumbs-up.jpg	

Step 5: Load the picturebox with the "HighLow.jpg" image in to the PictureBox.

Select Resource	
Resource context <u>L</u> ocal resource: <u>Import</u> <u>C</u> lear	
Project resource file: My Project\Resources.resx	HIGH
I <u>m</u> port	
	OK Cancel

Step 6: Double click the NewGame button and write the code to initialize your variables. Then use VB's Random number generator to generate a random number between 0 and 100:

```
NumberOfGuesses = 0
                                                        ' Starting a new game
                                                        ' Give it a default value
          UserGuess = 50
          LabelFeedback.Text = "Starting a new game, enter your guess above.."
          ' Generate a random number
                                                                  📕 HighLow
                                                                                                             Randomize()
                                                                           High-Low Game by "Your Name"
          RandomNumber = CInt(Rnd() * 100)
       Compile and Run your code again to see if it runs properly. You
Run:
       should notice that when the NewGame button is pressed, the
       feedback "Starting a new game....." should appear in the label.
                                                                        Enter Your Guess: (1-100)
                                                                                                Submit
                                                                                                             Game
Step 7: Double click the Submit button and type the following code:
            UserGuess = CInt(TextBoxGuess.Text)
                                                    'Take the user's guess and convert it to an integer
```

New

Quit

```
If UserGuess < RandomNumber Then
    LabelFeedback.Text = "Higher...
ElseIf UserGuess > RandomNumber Then
   LabelFeedback.Text = "Lower.."
Else
    LabelFeedback.Text = "You got it !"
End If
```

- Run your code again to see if it runs properly. Type a number in the textbox, and then click the Submit Button. Follow the Run: directions of the game to see if you can guess the number randomly generated by the program.
- Step 8: Once you are confident that the program is running. Try improving the game:
- **Question?** How would you improve your game? Can you tell the user how many tries it took them to guess the number? Can you make the program a little more robust? Can you give the user a visual feedback when they guess a wrong number?
- Replace the code in step 7 with the code below. That will make your program more robust (see Answer: ISNumeric(TextBoxGuess.Text)), it keeps track of the number of guesses (see NumberOfGuesses = NumberOfGuesses + 1) and it gives you more visual feedback when the user does the wrong thing!

```
If (IsNumeric(TextBoxGuess.Text)) Then
    UserGuess = CInt(TextBoxGuess.Text)
                                            'Take the user's guess and convert it to an integer
   NumberOfGuesses = NumberOfGuesses + 1 'Increment the NumberOfGuesses that user has made
    If UserGuess < RandomNumber Then</pre>
        LabelFeedback.Text = "Higher.."
    ElseIf UserGuess > RandomNumber Then
        LabelFeedback.Text = "Lower.."
    Else
        LabelFeedback.Text = "You got it in " & CStr(NumberOfGuesses) & " tries."
    End If
Else
    TextBoxGuess.Text = "" 'clear out the Guess textbox
    TextBoxGuess.BackColor = Color.Red
   MessageBox.Show( "Enter a number between 1 and 100", _
                      "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
    TextBoxGuess.BackColor = Color.White
End If
```

Step 9: Making the game a little more dynamic, by providing more visual feedback to the user. For example, when the user guesses a number that is low than what the program has picked, we would like to display the following image. To indicated that their next guess should be a little higher. To do this we need to have the ability to load a picture into our PictureBox during the execution of the program (while the program is running)

Loading Images



- In order to load an image in a PictureBox, during the execution of the program, you need the following:
 - 1) Place the pictures that we need in the bin/debug folder of your project.

G V V HighLow	▶ HighLow ▶ bin ▶ Debug				✓ Search Debug	
Organize 👻 🔚 Preview	w Share with 🔻 Print New folder				:= - 🗌 🔞	
🔶 Favorites	Name	Date modified	Туре	Size		
Nesktop	📋 HighLow.exe	12/3/2015 11:00 AM	CONFIG File	1 KB		
🗼 Downloads	E HighLow	12/3/2015 10:28 AM	JPEG image	42 KB		
🖳 Recent Places 😑	HighLow.vshost	12/3/2015 12:17 PM	Application	23 KB		
	HighLow.vshost.exe	12/3/2015 11:00 AM	CONFIG File	1 KB		
🥃 Libraries	Success	12/3/2015 12:29 PM	JPEG image	60 KB		
Documents	🔊 thumbs-down	12/3/2015 10:21 AM	JPEG image	79 KB		
J Music	📄 thumbs-up	12/3/2015 10:22 AM	JPEG image	5 KB		
Pictures	Windo Item type: JPEG image	12/3/2015 11:00 AM	CONFIG File	1 KB		
🚼 Videos	Windo Rating: Unrated	12/3/2015 11:00 AM	Application	23 KB		
	Windo Size: 4.03 KB	12/3/2015 11:00 AM	CONFIG File	1 KB		
👰 Computer	WindowsApplication1.vshost.exe.manifest	6/6/2012 2:06 AM	MANIFEST File	1 KB		
🕌 BOOTCAMP (C:) 🛫						
4 items selected Date taken: Specify date taken Ra		ting: ជំជំជំជំ Size: 183 KB		A	Authors: Add an author	
	Tags: Add a tag Dimensi	ions: (multiple values)	Title: Add a title	1		

2) Add the following Sub Procedure to your code. This sub procedure will load the image into the PictureBox.

```
Private Sub DisplayImage(ByVal TheImageFile As String)
   Dim MyImage As Bitmap
   PictureBox1.SizeMode = PictureBoxSizeMode.StretchImage
   MyImage = New Bitmap(TheImageFile)
   PictureBox1.Image = CType(MyImage, Image)
End Sub
```

3) Now, in order to load different images into the picture box, the programmer must call the above sub procedure. For example at the beginning of the game, we load the "new game" image.

Later when the user guesses a number and we would like to provide visual feedback as to **<u>higher</u>** or **<u>lower</u>** we load different images to provide this feedback.

Add the highlighted lines below to your ButtonSubmit_Click module and run the program to see the result.

```
Private Sub ButtonSubmit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
                         Handles ButtonSubmit.Click
    If (IsNumeric(TextBoxGuess.Text)) Then
        UserGuess = CInt(TextBoxGuess.Text)
                                            'Take the user's guess and convert it to an integer
        NumberOfGuesses = NumberOfGuesses + 1 'Increment the NumberOfGuesses that user has made
        If UserGuess < RandomNumber Then
           LabelFeedback.Text = "Higher.."
        ElseIf UserGuess > RandomNumber Then
           DisplayImage("thumbs-down.jpg") ' Load the Thumbs Down image
LabelFeedback.Text = "Lower.."
        Else
           LabelFeedback.Text = "You got it in " & CStr(NumberOfGuesses) & " tries."
        End If
    Else
        TextBoxGuess.Text = "" 'clear out the Guess textbox
        TextBoxGuess.BackColor = Color.Red
        MessageBox.Show("Enter a number between 1 and 100", "Error",
                         MessageBoxButtons.OK, MessageBoxIcon.Error)
        TextBoxGuess.BackColor = Color.White
    End If
```

```
End Sub
```