Student Learning Objectives:

* Working with list box controls
* Working with the timer control
* Working with the random class

**There are 4 print screens/code each worth 25%**

List Box Control

**Project #1**

Reading from a text file and displaying on a List box

**Design**

Graphical user interface, application

Description automatically generated

**Code**

A picture containing text

Description automatically generated

**Copy and paste the quotes in a notepad**

Always keep your eyes open. Keep watching. Because whatever you see can inspire you

What you get by achieving your goals is not as important as what you become by achieving your goals

If the plan does not work, change the plan, but never the goal

**Project #2**

Removing duplicates in a list box. Add a second button control (see blue arrow)

Graphical user interface, application

Description automatically generated

Code  
Text, letter

Description automatically generated

**Project #3**

**Design**

Combo Box

**Graphical user interface, text, application, email

Description automatically generated**

**Code**

The following program uses a Simple combo box to obtain a person’s title for the first line of the address of a letter. **Note: the second batch of code in loaded from the Load event**

**Text, timeline

Description automatically generated**

Timer Control

**Project #4**

The timer control, which is not visible on the form during run time, raises an event whenever a specified amount of time has passed. (The timer control is found only in the All Windows Forms group of the Toolbox. When you double-click on the timer control in the Toolbox, it appears in the component tray, at the bottom of the Form Designer.) The length of time, measured in milliseconds, is set with the Interval property to be any integer from 1 to 2,147,483,647 (about 596 hours). The event that is raised each time Timer1.Interval milliseconds elapses is called Timer1.Tick. To begin timing, a timer must first be turned on by setting its Enabled property to True. A timer is turned off by setting its Enabled property to False. The standard prefix for the name of a timer control is tmr.

**Design**

Graphical user interface, application

Description automatically generated

**Code**

Timeline

Description automatically generated

Add or modify code so that if the text box = 10 than message box a message “Blast off” and print screen below

**#1 print screen the output with code below here**

**Graphical user interface

Description automatically generated**

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

TextBox3.Text = "0"

Timer1.Enabled = True

End Sub

Private Sub Button3\_Click(sender As Object, e As EventArgs) Handles Button3.Click

Timer1.Enabled = False

End Sub

Private Sub Timer1\_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick

TextBox3.Text = (CDbl(TextBox3.Text) + 0.1).ToString("N1")

End Sub

Random Class

**Project #5**

**Design**

A picture containing graphical user interface, application

Description automatically generated

**Code**

Text

Description automatically generated

Create a little game where you can play the computer. Be sure to modify the design and code where the player can input either **Rock, Paper or Scissor**

**#2 print screen the output with code below here**

**Graphical user interface, application

Description automatically generated**

Private Sub Button4\_Click(sender As Object, e As EventArgs) Handles Button4.Click

Dim Choices() As String = {"Rock", "Paper", "Scissors"}

Dim randomNumbers As New Random

Dim n As Integer

n = randomNumbers.Next(0, 3)

TextBox4.Text = Choices(n)

End Sub

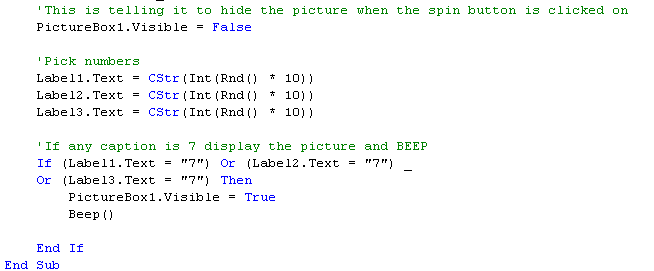
**Project #6**

**Design**

**Text

Description automatically generated**

**Code**



Graphical user interface

Description automatically generated

Graphical user interface, chart

Description automatically generated

Private Sub Button5\_Click(sender As Object, e As EventArgs) Handles Button5.Click

PictureBox1.Visible = False

Label1.Text = CStr(Int(Rnd() \* 10))

Label2.Text = CStr(Int(Rnd() \* 10))

Label3.Text = CStr(Int(Rnd() \* 10))

If (Label1.Text = "7") Or (Label2.Text = "7") Or (Label3.Text = "7") Then

PictureBox1.Visible = True

Beep()

End If

End Sub

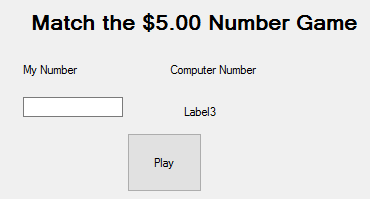
**Project #7**

Objectives:

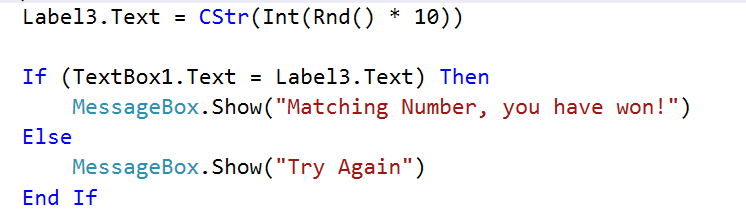
* Creating a small game using the Random Class
* Adding an event called Key Press to the Text Box controls

Creating a matching game app

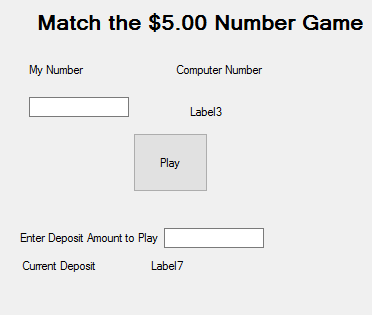
1. Create a new project and design the interface as shown below



1. Double click on the Play button control and type the following:



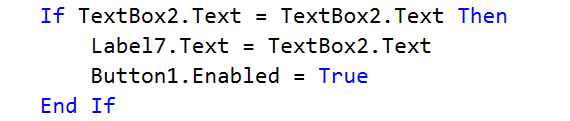
1. Now, we will add additional features to add or append winnings and deduct when losing. Add the following controls in bracket



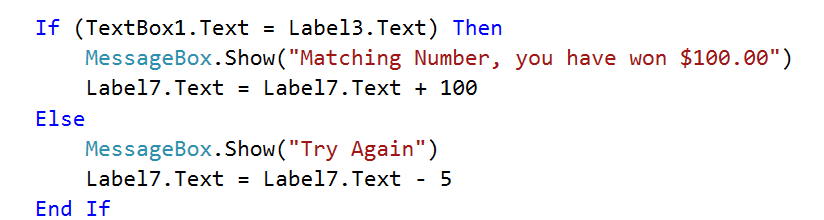
1. On the Form Load disable the Play button



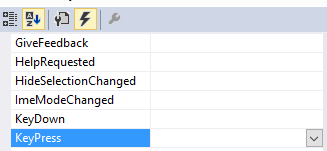
1. Double click on the text box and type in the following



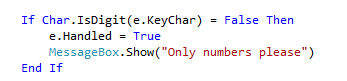
1. Double click on the Play button and add the following (See arrows)



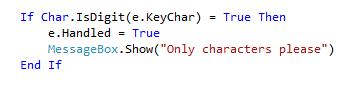
1. If a user accidently enters a character into the Text Box, the application will throw an error or NOT work correctly
2. To tackle that problem let’s add event for the two text box controls
   1. Go to the properties for Text Box1
   2. Click on the Lightning Bolt Icon  on top
   3. Double click on the empty space beside KeyPress



* 1. Enter the following code:



1. Write or copy the code above into the 2nd text box control
2. Try replacing the True Boolean value to True



**#3 print screen the output for project #7 below here**

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**#4 copy and paste the code for project #7 below here**

Private Sub Button6\_Click(sender As Object, e As EventArgs) Handles Button6.Click

Label7.Text = CStr(Int(Rnd() \* 10))

If (TextBox5.Text = Label7.Text) Then

MessageBox.Show("MAtching Number, You WON!")

Label10.Text = Label10.Text + 100

Else

MessageBox.Show("Try Again")

Label10.Text = Label10.Text - 5

End If

End Sub

Private Sub TextBox6\_TextChanged(sender As Object, e As EventArgs) Handles TextBox6.TextChanged

Button6.Enabled = False

If TextBox6.Text = TextBox6.Text Then

Label10.Text = TextBox6.Text

Button6.Enabled = True

End If

End Sub

Private Sub TextBox5\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox5.KeyPress

If Char.IsDigit(e.KeyChar) = False Then

e.Handled = True

MessageBox.Show("Only numbers please")

End If

End Sub

End Class

**Submit this document to module 10 class exercise**