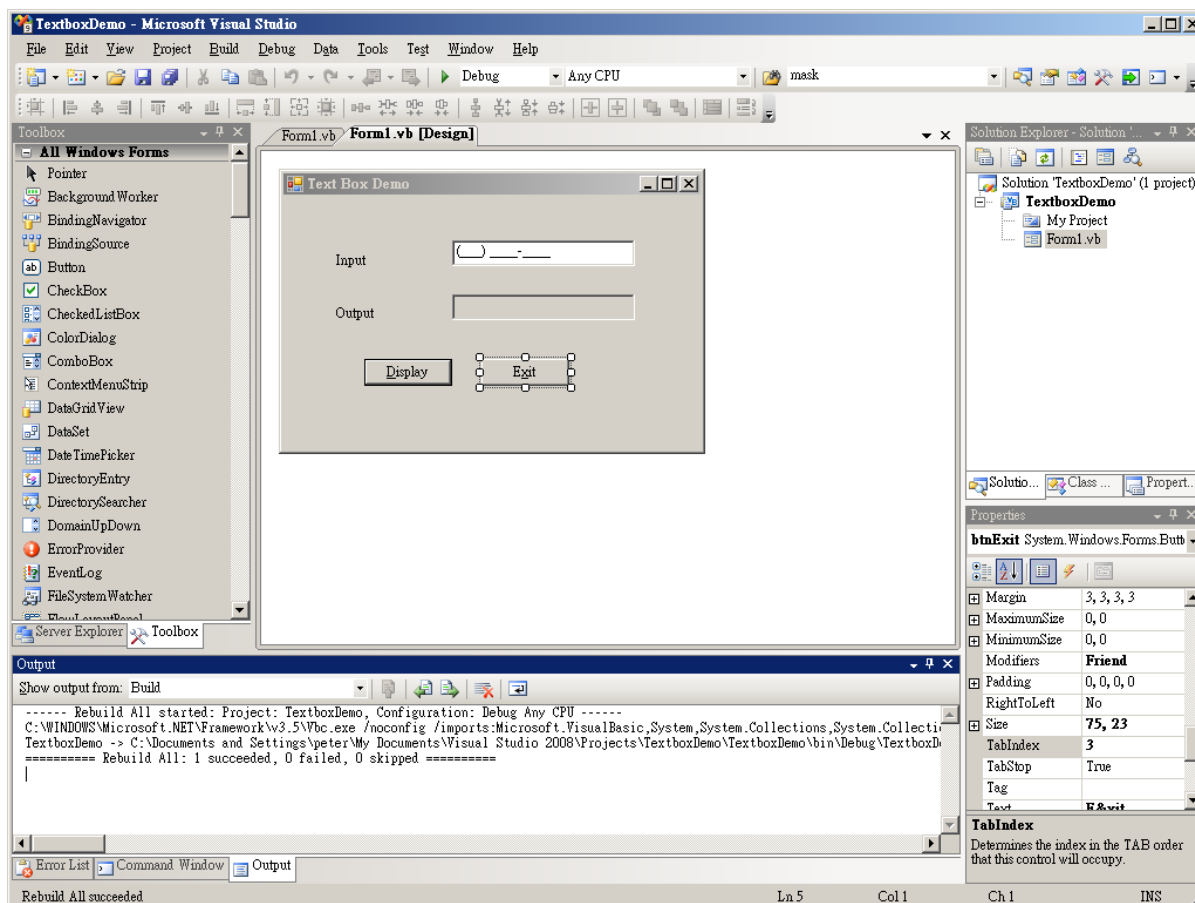


1. Text Box Demo

1. Open the Microsoft Visual Studio and start a new Visual Basic Project named as **TextboxDemo**. From the Toolbox, drag 2 **Button** controls, 2 **Label** controls, a **Masked TextBox** control, a **TextBox** control onto the form and customize the properties.

Object	Name	Property	Property Value
Form	frmMain	Text	Text Box Demo
		AcceptButton	btnDisplay
		CancelButton	btnExit
Button	btnDisplay	Text	&Display
	btnExit	Text	E&xit
Label	Label1	Text	Input
	Label2	Text	Output
Masked TextBox	txtInput	Text	(Blank)
		Mask	(999) 9999-9999
TextBox	txtOutput	Text	(Blank)
		ReadOnly	True



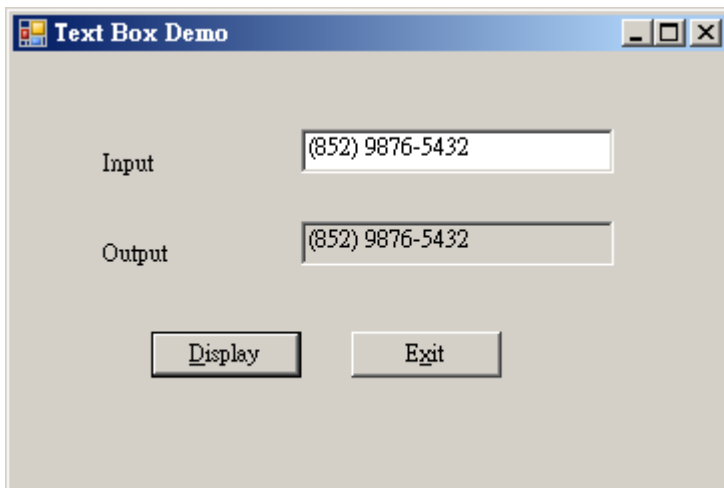
2. In the **Click** event procedure of the Display button (**btnDisplay**), add the following code.

```
' Display the text  
txtOutput.Text = txtInput.Text
```

3. In the **Click** event procedure of the Exit button (**btnExit**), add the following code.

```
' Exit the program  
End
```

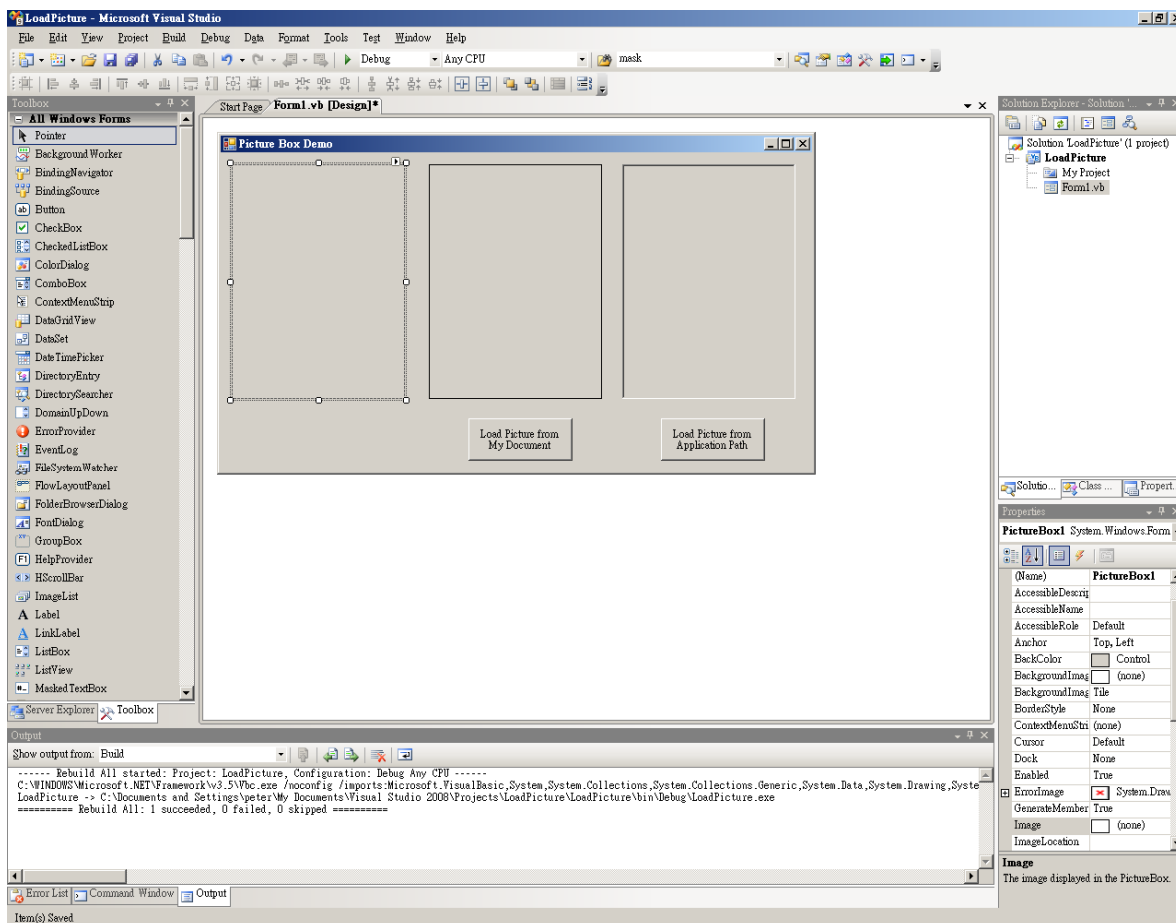
4. Build and execute your program.
- A. Input “852987654321” in the textbox, and then press **[Enter]**, what can you occurs?
 - B. What will happen if you press **[ESC]** button?
 - C. Re-execute the program, and then Input “852987654321” in the textbox, and then press **[Alt] + [D]**, what can you occurs?
 - D. What will happen this time if you press **[Alt] + [X]** button?



2. Loading Picture

1. Open the Microsoft Visual Studio and start a new Visual Basic Project named as **LoadPicture**. From the Toolbox, drag 2 **Button** controls and 3 **PictureBox** controls onto the form and customize the properties.

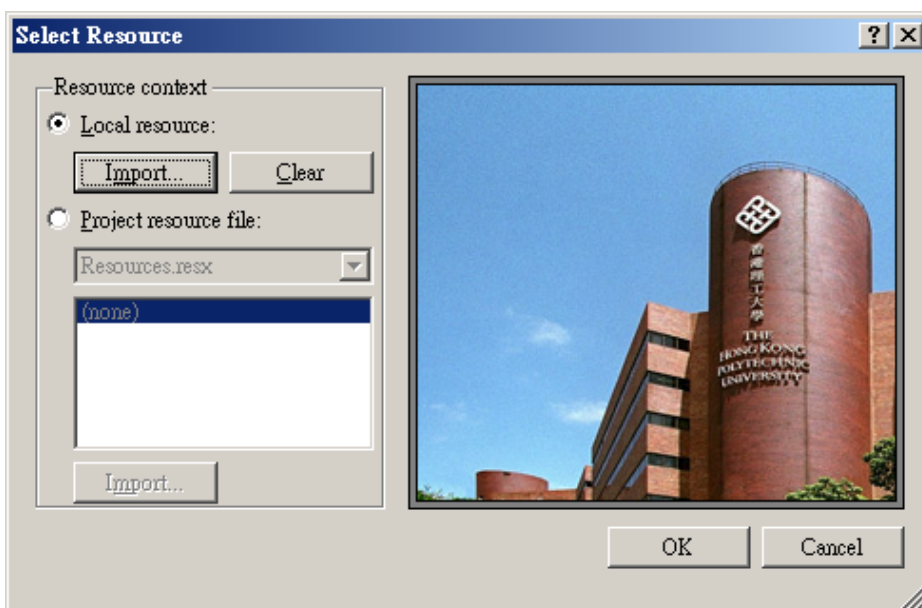
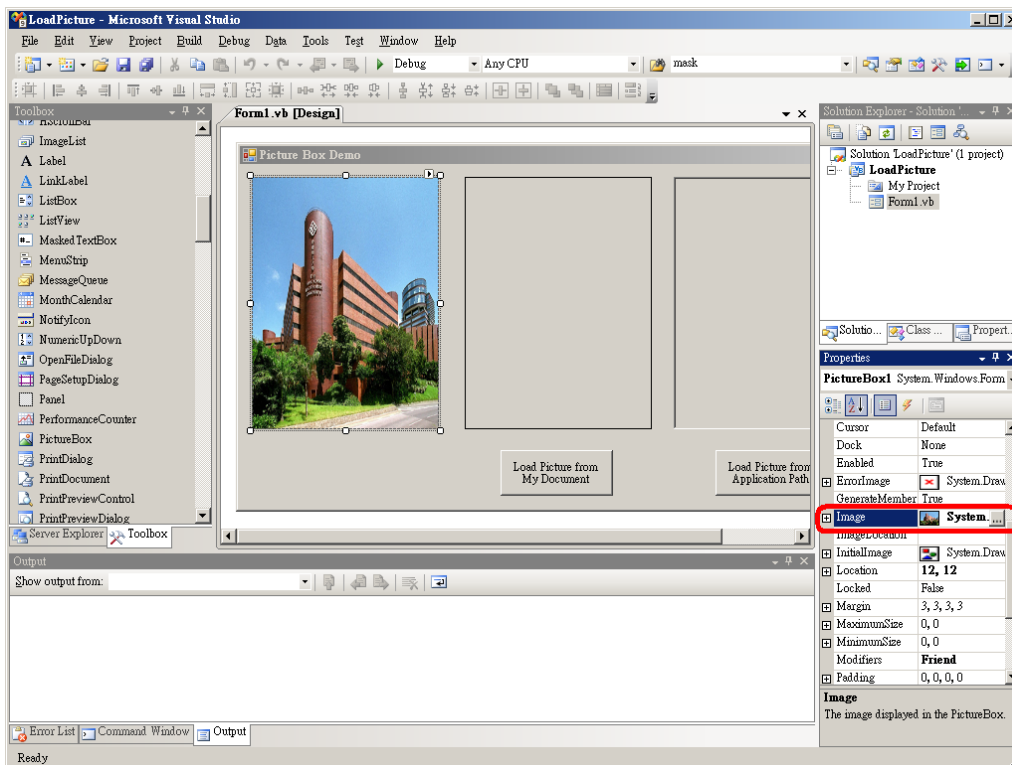
Object	Name	Property	Property Value
Form	frmMain	Text	Picture Box Demo
Button	btnMyDoc	Text	Load Picture from My Document
	btnAppPath	Text	Load Picture from Application Path
PictureBox	PictureBox1	BorderStyle	None
		SizeMode	StretchImage
	PictureBox2	BorderStyle	FixedSingle
		SizeMode	CenterImage
	PictureBox3	BorderStyle	Fixed3D
		SizeMode	Zoom



- Use the Internet Image Search Engine (such as Google, Yahoo etc) to find three JPEG images, download them into your hard disk and rename them as follow (You can also download the image file from <http://www.peter-lo.com/Teaching/I135-1-A/Source3.zip>):

Picture Name	Store Location
Pic01.jpg	C:\Temp\
Pic02.jpg	The “My Document” in your computer
Pic03.jpg	The “Debug” folder for your Visual Studio current project (My Documents\Visual Studio 2008\Projects\LoadPicture\LoadPicture \bin\Debug)

- In the **Image** property for the **PictureBox1** control, use the resource browser to import the image from **C:\Temp\Pic01.jpg**.



4. In the **Click** event procedure of the Load Picture from My Document button (**btnMyDoc**), add the following code.

```
' Load the image from My Document  
PictureBox2.Image = _  
Image.FromFile(System.Environment.GetFolderPath _  
(System.Environment.SpecialFolder.Personal) & "\Pic02.jpg")
```

5. In the **Click** event procedure of the Load Picture from Application Path button (**btnAppPath**), add the following code.

```
' Load the image from project startup path  
PictureBox3.Image = _  
Image.FromFile(Application.StartupPath & "\Pic03.jpg")
```

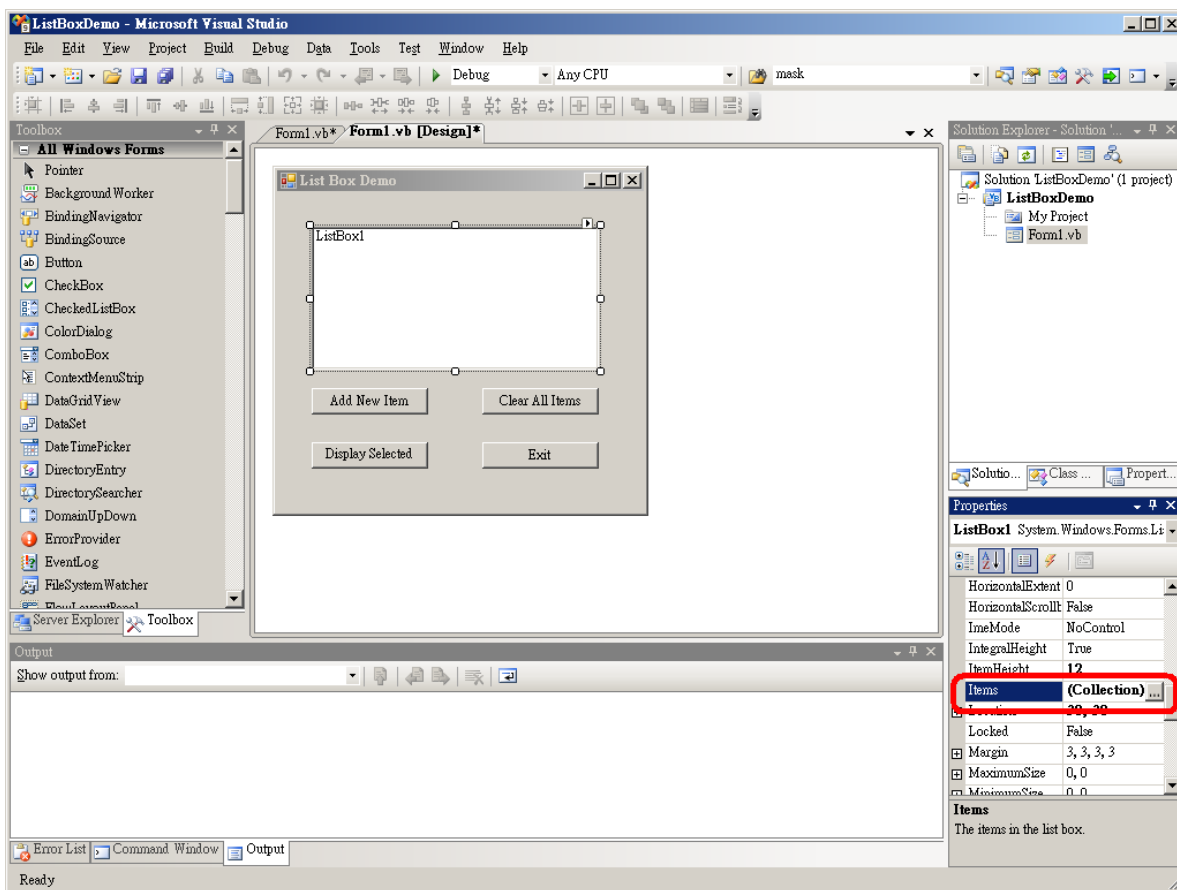
6. Build and execute your program. Can you find the difference for border style and size mode?



3. List Box

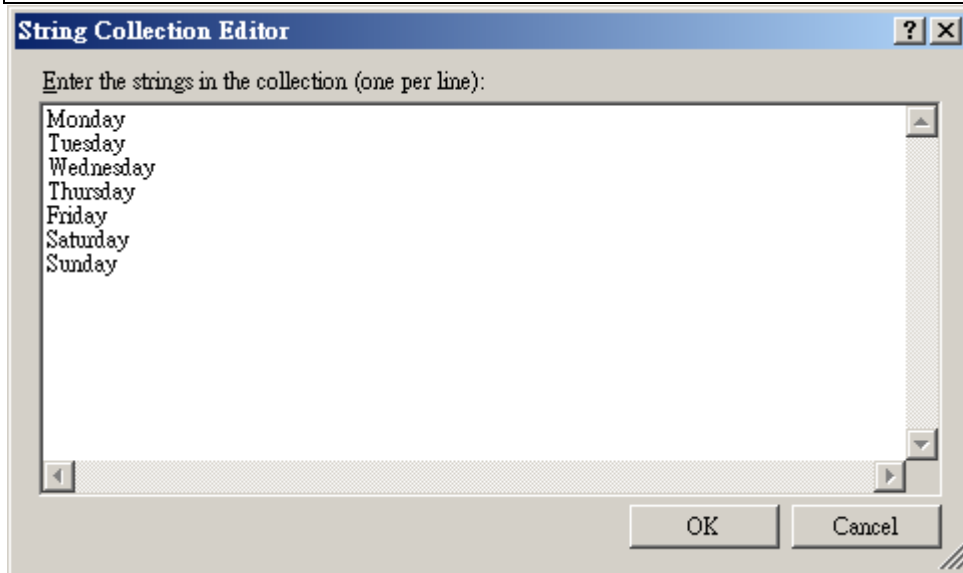
1. Open the Microsoft Visual Studio and start a new Visual Basic Project named as **ListBoxDemo**. From the Toolbox, drag 4 **Button** controls and a **List Box** control onto the form and customize the properties.

Object	Name	Property	Property Value
Form	frmMain	Text	List Box Demo
Button	btnAdd	Text	Add New Item
	btnClear	Text	Clear All Item
	btnDisplay	Text	Display Selected Item
	btnExit	Text	Exit
Listbox	Listbox1	N/A	N/A



- Click the “**Collection [...]**” button in the **Items** properties for the **ListBox1** control to active the String Collection Editor, and input the following string to it

```
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
```



- In the **Click** event procedure of the Add New Item button (**btnAdd**), add the following code:

```
' Add a new item to the Listbox
ListBox1.Items.Add("Hello World")
```

- In the **Click** event procedure of the Clear All Item button (**btnClear**), add the following code:

```
' Remove all Item from the Listbox
ListBox1.Items.Clear()
```

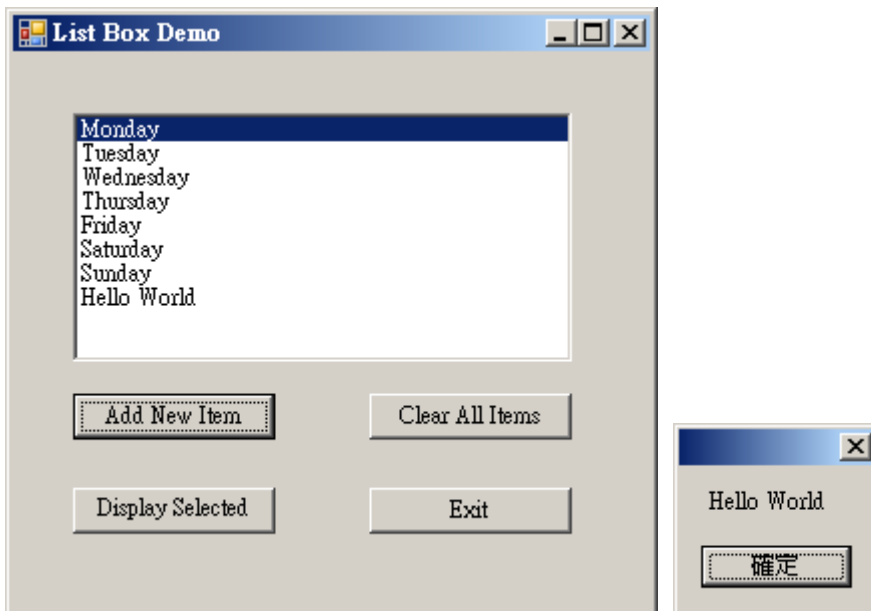
- In the **Click** event procedure of the Display Selected Item button (**btndisplay**), add the following code:

```
' Display the selected item
MessageBox.Show(Me.ListBox1.SelectedItem)
```

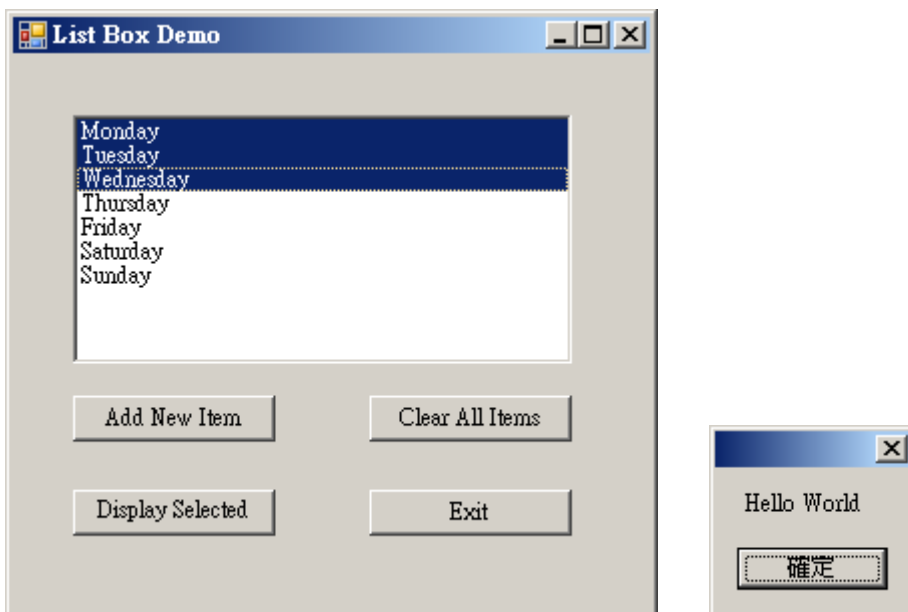
- In the **Click** event procedure of the Exit button (**btnExit**), add the following code:

```
' Exit the program
End
```

- 7. Build and execute your program.
 - A. Press the **[Add New Item]** button to add the new item to the list box.
 - B. Select one item in the list box, and then press the **[Display Selected Item]** button. Can you see the dialog box showing your selected value?
 - C. Press **[Clear All Item]** button to clear the items, how many items in your list box now?
 - D. Press the **[Exit]** button to quit the program



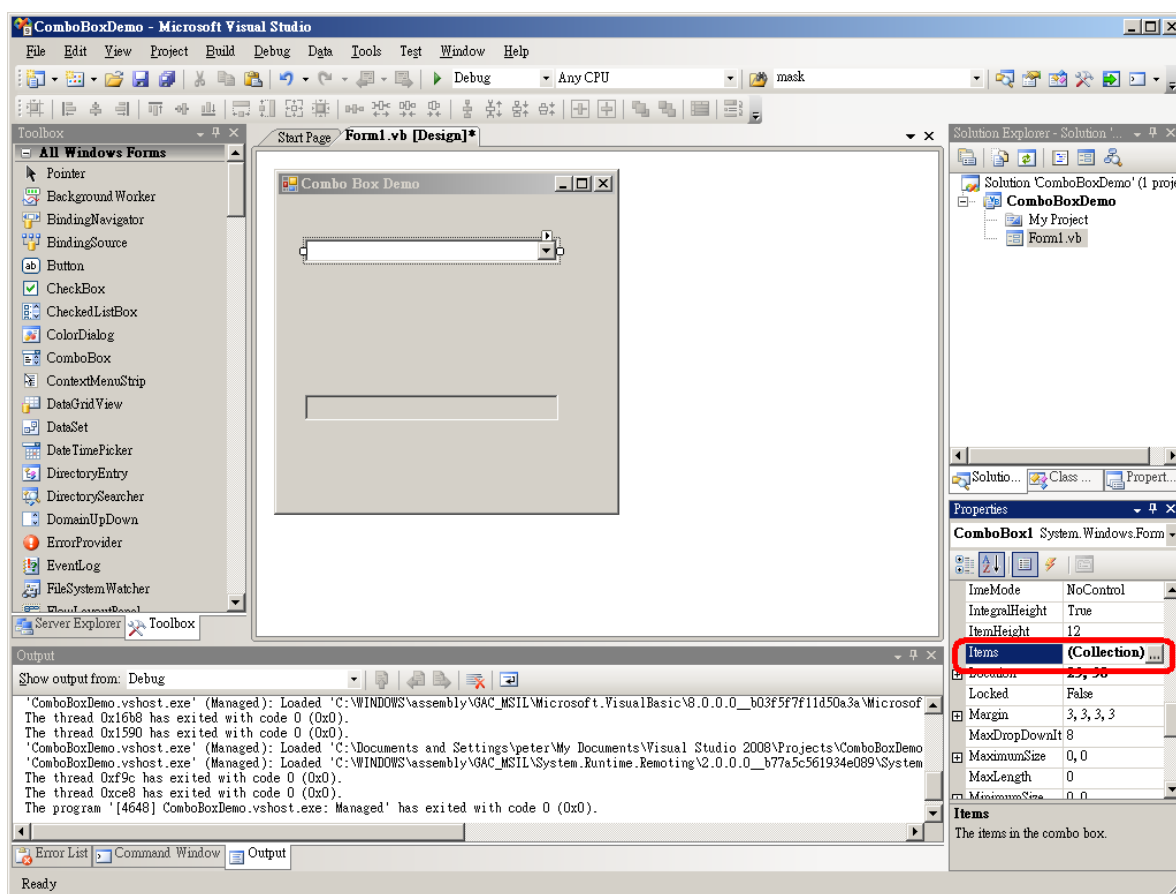
- 8. Change the **SelectionMode** property for **ListBox1** control from **One** to **MultiSimple**. Then rebuild and execute your program. Select several items in your list box, and then press the **[Display Selected Item]** button. Can you see the dialog box showing your selected value? What do you observe?



4. Combo Box

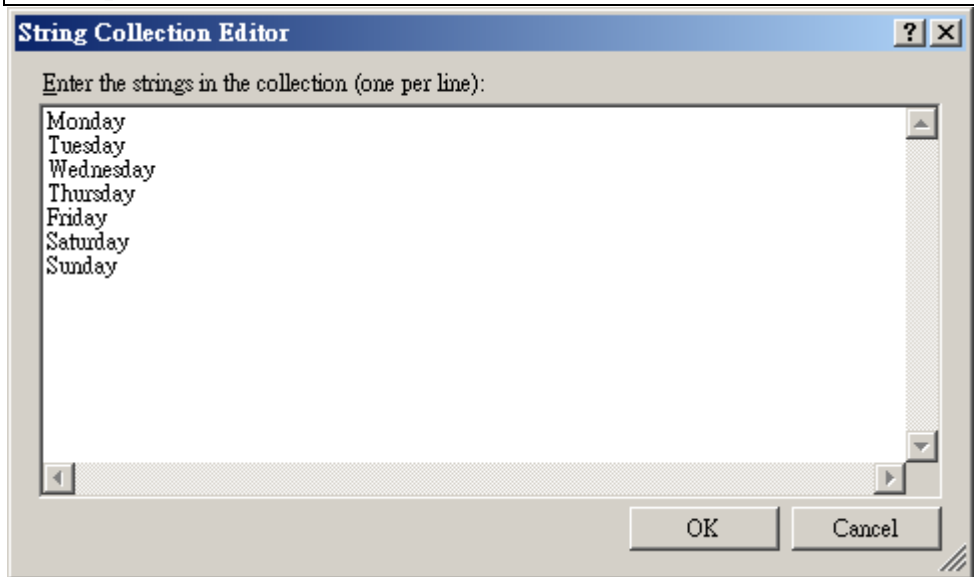
1. Open the Microsoft Visual Studio and start a new Visual Basic Project named as **ComboBoxDemo**. From the Toolbox, drag one **Button** control, one **Textbox** control and a **Combo Box** control onto the form and customize the properties.

Object	Name	Property	Property Value
Form	frmMain	Text	Combo Box Demo
TextBox	txtOutput	ReadOnly	True
ComboBox	ComboBox1	N/A	N/A



2. Click the [...] button in the Items properties for the Combo Box control to activate the **String Collection Editor**, and put the following string in to the Collection of the Combo Box control.

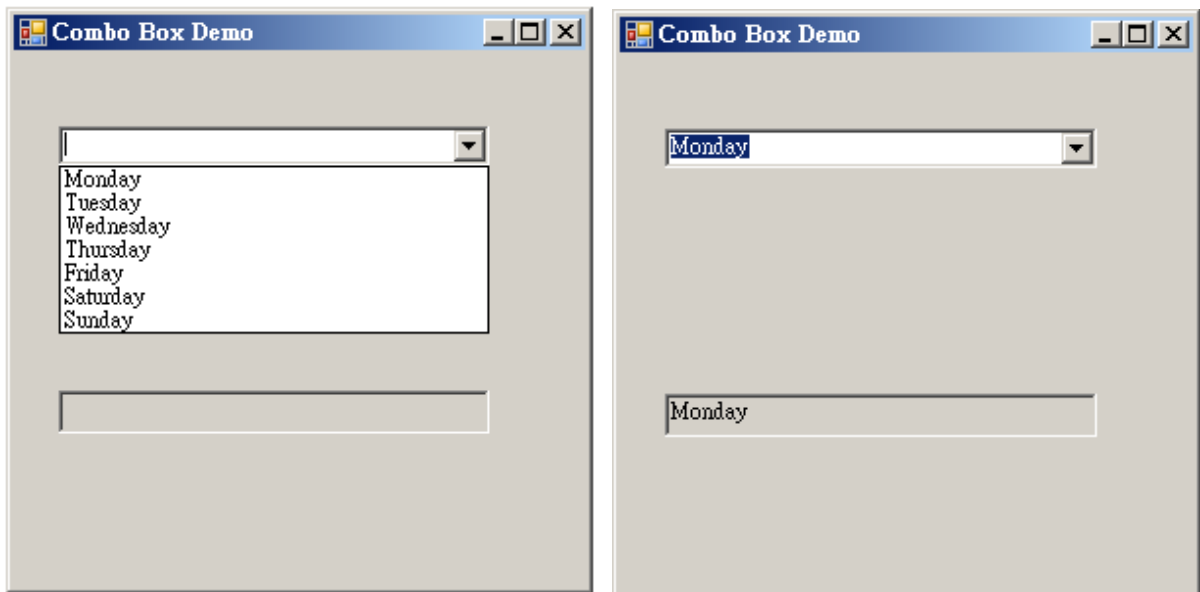
```
Monday  
Tuesday  
Wednesday  
Thursday  
Friday  
Saturday  
Sunday
```



3. In the **Selected Index Changed** event procedure of the Combo Box control (**ComboBox1**), add the following code:

```
' Change the output text when user change the combobox value  
txtOutput.Text = ComboBox1.Text
```

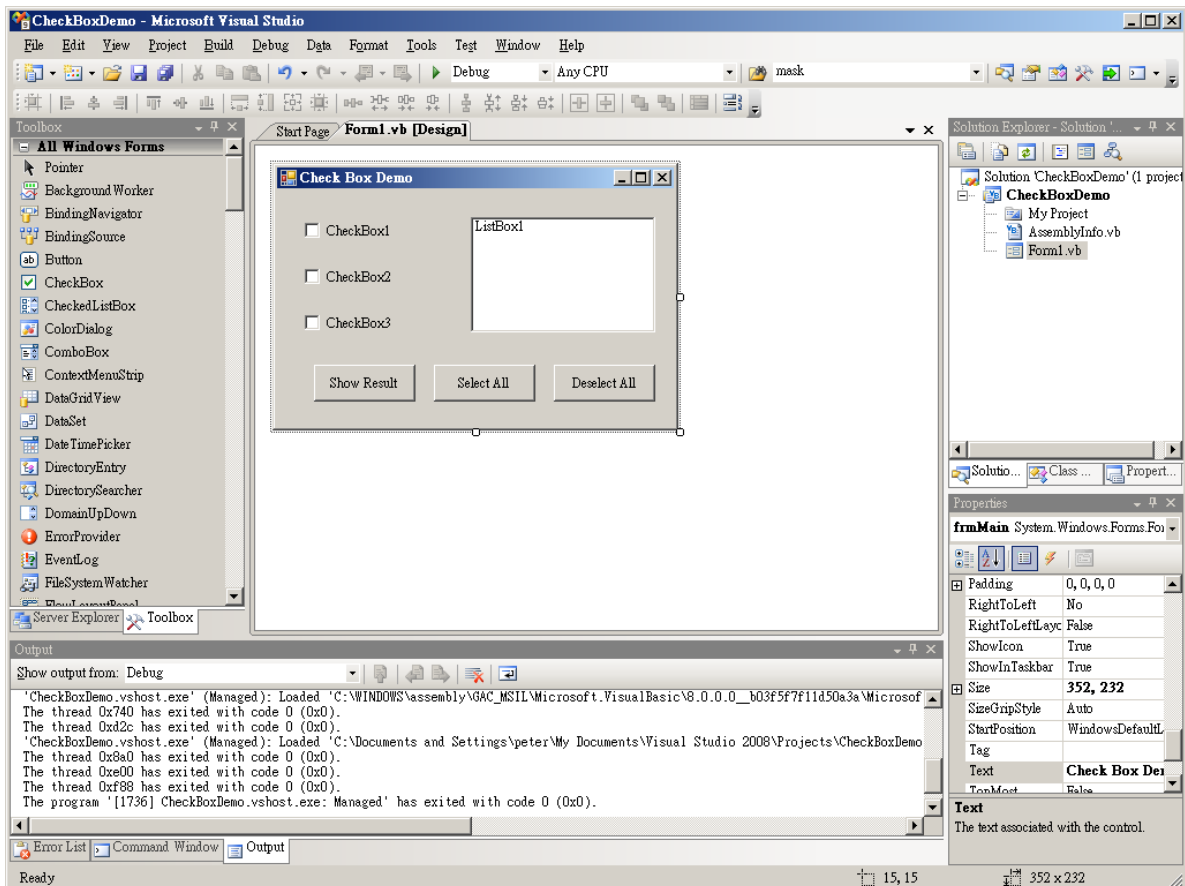
4. Save the project and build the solution, and then execute it.



5. Checkbox

1. Open the Microsoft Visual Studio and start a new Visual Basic Project named as **CheckBoxDemo**. From the Toolbox, drag 3 **Button** controls, one **List Box** control and 3 **Check Box** controls onto the form and customize the properties.

Object	Name	Property	Property Value
Form	frmMain	Text	Checked Box Demo
Button	btnCheck	Text	Show Result
	btnSelect	Text	Select All
	btnDeselect	Text	Deselect All
List Box	ListBox1	N/A	N/A
Check Box	Checkbox1	Text	Check Box 1
		ThreeState	False
	Checkbox2	Text	Check Box 2
		ThreeState	True
	Checkbox3	Text	Check Box 3
		ThreeState	True



2. In the **Click** event procedure of the Show Result button control (**btnShow**), add the following code:

```
' Clear the List Box
ListBox1.Items.Clear()

' Show the result
ListBox1.Items.Add("Check Box 1 = " & CheckBox1.CheckState)
ListBox1.Items.Add("Check Box 2 = " & CheckBox2.CheckState)
ListBox1.Items.Add("Check Box 3 = " & CheckBox3.CheckState)
```

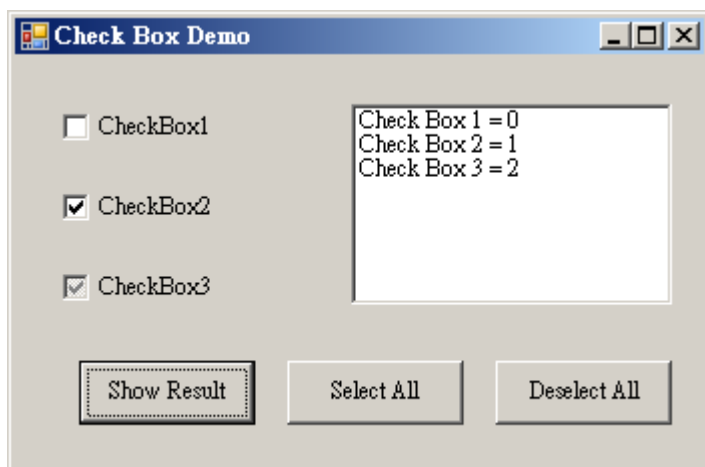
3. In the **Click** event procedure of the Select All button control (**btnSelect**), add the following code:

```
' Select all checkbox
CheckBox1.Checked() = True
CheckBox2.Checked() = True
CheckBox3.Checked() = True
```

4. In the **Click** event procedure of the Deselect All button control (**btnDeselect**), add the following code:

```
' Deselect all checkbox
CheckBox1.Checked() = False
CheckBox2.Checked() = False
CheckBox3.Checked() = False
```

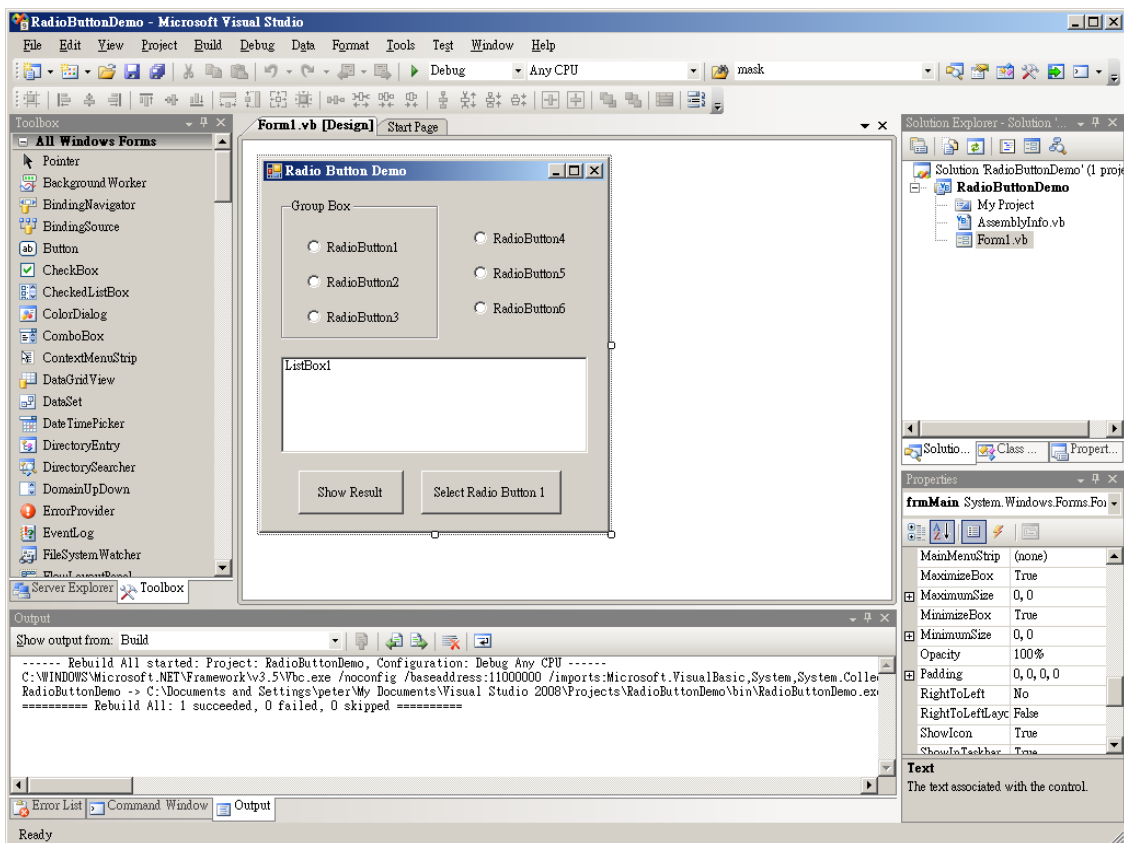
5. Save the project and build the solution, and then execute it. Can you observe the



6. Radio Button

1. Open the Microsoft Visual Studio and start a new Visual Basic Project named as **RadioButtonDemo**. From the Toolbox, drag 2 **Button** controls, one **List Box** control, one **Group Box** controls onto the form. Then drag 3 **Radio Button** controls into the Group box and 3 **Radio Button** controls outside the Group box. Finally customize the properties of the controls as follow:

Object	Name	Property	Property Value
Form	frmMain	Text	Radio Button Demo
Button	btnShow	Text	Show Result
	btnSelect1	Text	Select All
List Box	ListBox1	N/A	N/A
Group Box	GroupBox1	Text	Group Box
Radio Button	RadioButton1	Text	Radio Button 1
	RadioButton2	Text	Radio Button 2
	RadioButton3	Text	Radio Button 3
	RadioButton4	Text	Radio Button 4
	RadioButton5	Text	Radio Button 5
	RadioButton6	Text	Radio Button 6



2. In the **Click** event procedure of the Show Result control (**btnShow**), add the following code.

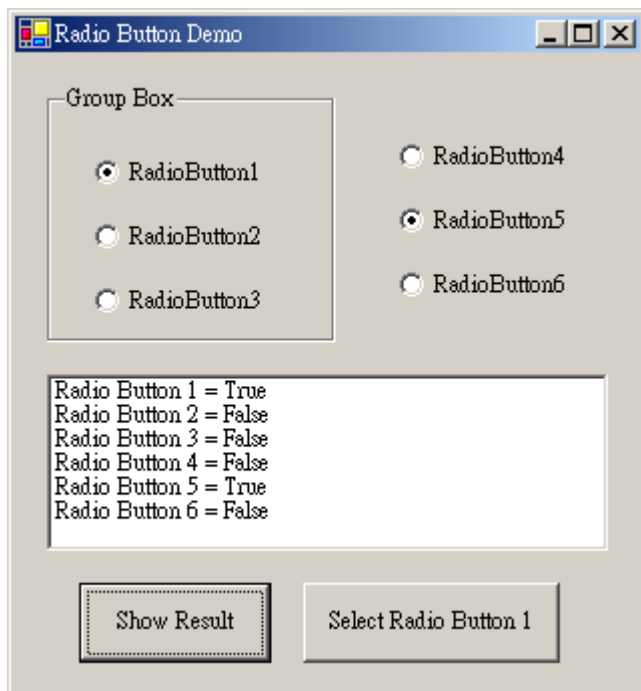
```
' Clear the List Box
ListBox1.Items.Clear()

' Show the result
ListBox1.Items.Add("Radio Button 1 = " & RadioButton1.Checked)
ListBox1.Items.Add("Radio Button 2 = " & RadioButton2.Checked)
ListBox1.Items.Add("Radio Button 3 = " & RadioButton3.Checked)
ListBox1.Items.Add("Radio Button 4 = " & RadioButton4.Checked)
ListBox1.Items.Add("Radio Button 5 = " & RadioButton5.Checked)
ListBox1.Items.Add("Radio Button 6 = " & RadioButton6.Checked)
```

3. In the **Click** event procedure of the Select Radio Button 1 control (**btnSelect1**), add the following code.

```
' Select Radio Button 1
RadioButton1.Select()
```

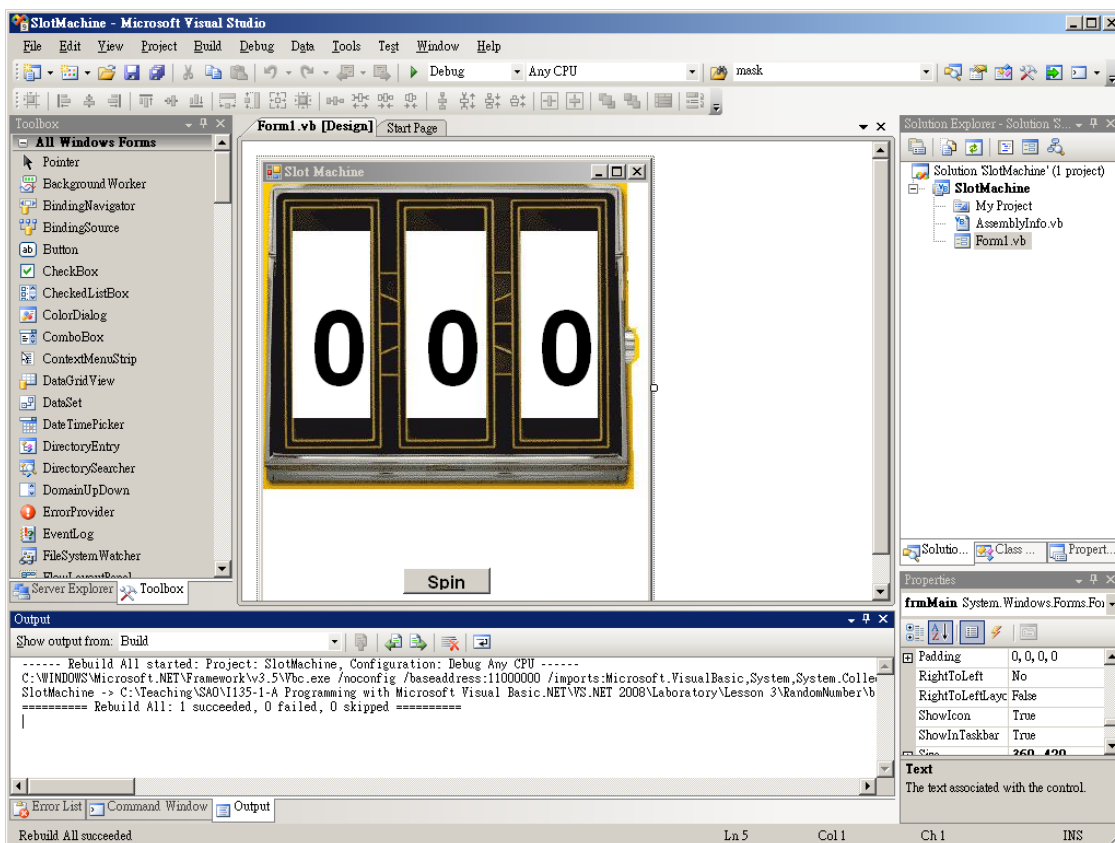
4. Save the project and build the solution, and then execute it. What is the different between the radio button in the Group Box and outside the Group Box?



7. Generate a Random Number

1. Open the Microsoft Visual Studio and start a new Visual Basic Project named as **SlotMachine** (The image file for the Slot Machine can be downloaded from the following URL:<http://www.peter-lo.com/Teaching/I135-1-A/Source3.zip>).

Object	Name	Property	Property Value
Form	frmMain	Text	Slot Machine
		BackgroundImage	SlotMachine.jpg
		Size Width	360
		Size Height	420
		AcceptButton	btnSpin
		StartPosition	CenterScreen
Button	btnSpin	Text	Spin
		Font Size	12
		Font Name	Arial
		Font Style	Bold
Label	Label1	Text	0
		Font Size	72
		Font Name	Arial
		Font Style	Bold
		Back Color	Transparent
	Label2	Text	0
		Font Size	72
		Font Name	Arial
		Font Style	Bold
		Back Color	Transparent
	Label3	Text	0
		Font Size	72
		Font Name	Arial
		Font Style	Bold
		Back Color	Transparent



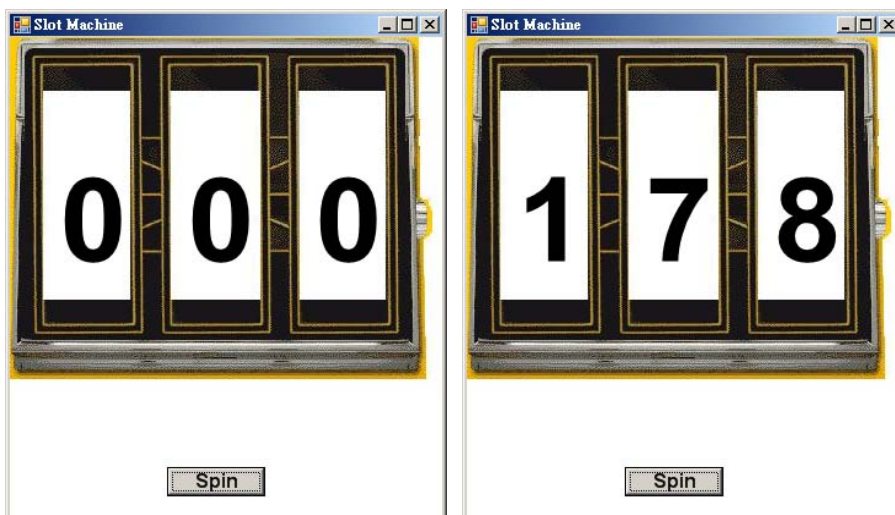
2. In the **Click** event procedure of the button control (**btnSpin**), add the following code.

```

' Declare a random generator
Dim RandomGenerator As New Random

' Generate a random number from 0 to 9
Label1.Text = RandomGenerator.Next(0, 9)
Label2.Text = RandomGenerator.Next(0, 9)
Label3.Text = RandomGenerator.Next(0, 9)
    
```

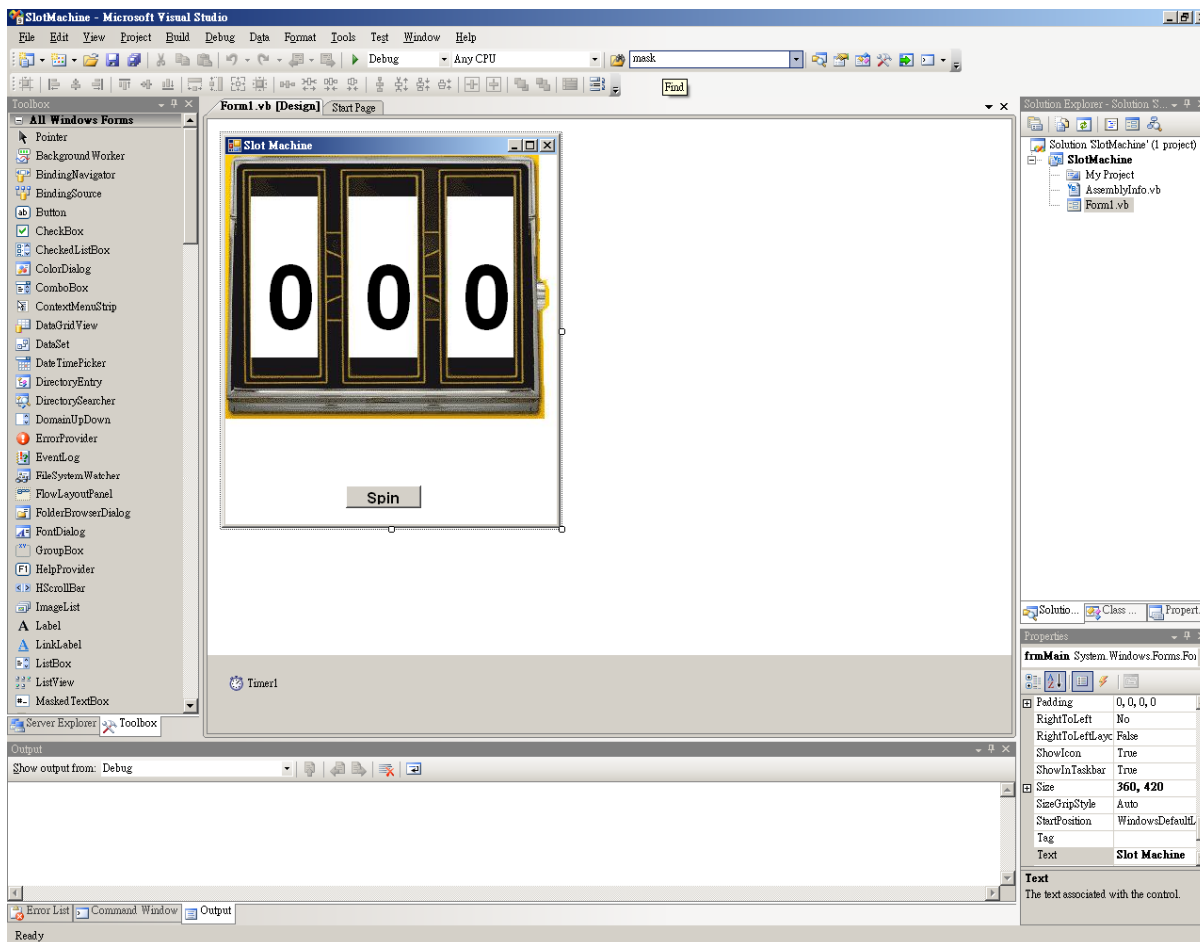
3. Save the project and build the solution. Then execute it and press the [**Spin**] button.



8. Timer Control

1. Open the Microsoft Visual Studio and open the previous Visual Basic Project **SlotMachine**. From the Toolbox, drag a **Timer** controls onto the form and customize the properties.

Object	Name	Property	Property Value
Timer	Timer1	Interval	100



2. Remove the **Click** event for the button control (**btnSpin**). You can double click the button control (btnSpin), and then delete/remark all program code inside the event.
3. Declare a public variable for controlling spin stop.

```
Public StopFlag As Integer
```

4. In the **Tick** event procedure of the Timer control (**Timer1**), add the following code.

```
' Declare a random generator
Dim RandomGenerator As New Random

' Slow down the display when user release the button
If StopFlag < 0 Then
    Timer1.Interval() += 10
End If

If StopFlag <> 0 Then
    ' Generate a random number from 0 to 9
    Label1.Text = RandomGenerator.Next(0, 9)
    Label2.Text = RandomGenerator.Next(0, 9)
    Label3.Text = RandomGenerator.Next(0, 9)

    ' Increase the counter
    StopFlag += 1
Else
    ' Stop the timer
    Timer1.Enabled = False

    If Label1.Text = Label2.Text And _
        Label1.Text = Label3.Text Then
        ' Gain 20 credit for all 3 numbers equal
        MsgBox("You Win 20 credits")
    ElseIf Label1.Text = Label2.Text Or _
        Label1.Text = Label3.Text Or _
        Label2.Text = Label3.Text Then
        ' Gain 5 credit for any 2 numbers equal
        MsgBox("You Win 5 credits")
    End If
End If
```

5. In the **MouseDown** event procedure of the button control (**btnSpin**), add the following code.

```
' Set the Stop Flag to False (-ve as true)
StopFlag = 10

' Define the time interval at 50/1000 s
Timer1.Interval = 50

' Start the timer
Timer1.Enabled = True
```

6. In the **MouseUp** event procedure of the button control (**btnSpin**), add the following code.

```
' Set the Stop Flag to True (-ve as true)  
StopFlag = -5
```

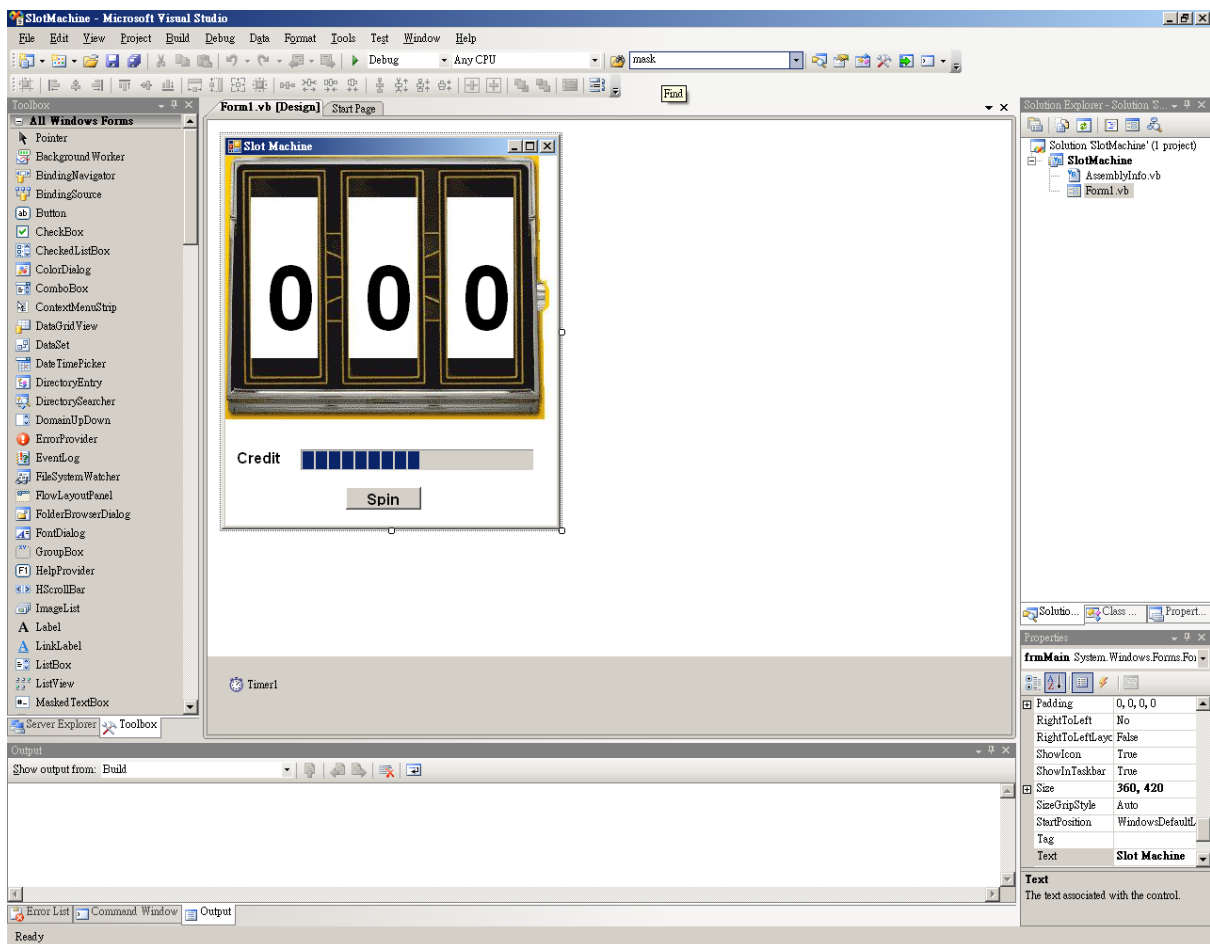
7. Build and run your program. Press the [**Spin**] button to randomize the number, and then release the button to obtain the result.



9. Progress Bar Handling

1. Open the Microsoft Visual Studio and open the previous Visual Basic Project **SlotMachine**. From the Toolbox, drag a **Progress Bar** controls onto the form and customize the properties.

Object	Name	Property	Property Value
Label	Label4	Text	Credit
		Font Size	12
		Font Name	Arial
		Font Style	Bold
		Back Color	Transparent
Progress Bar	ProgressBar1	Value	50
		Maximum	100
		Minimum	0



2. In the **Tick** event procedure of the Timer control (**Timer1**), modify the code as follow:

```
' Declare a random generator
Dim RandomGenerator As New Random

' Slow down the display when user release the button
If StopFlag < 0 Then
    Timer1.Interval() += 10
End If

If StopFlag <> 0 Then
    ' Generate a random number from 0 to 9
    Label1.Text = RandomGenerator.Next(0, 9)
    Label2.Text = RandomGenerator.Next(0, 9)
    Label3.Text = RandomGenerator.Next(0, 9)

    ' Increase the counter
    StopFlag += 1
Else
    ' Stop the timer
    Timer1.Enabled = False

    If Label1.Text = Label2.Text And _
        Label1.Text = Label3.Text Then
        ' Increase 20 credit and show in Progress Bar
        ProgressBar1.Value += 20

        ' Gain 20 credit for all 3 numbers equal
        MsgBox("You Win 20 credits")
    ElseIf Label1.Text = Label2.Text Or _
        Label1.Text = Label3.Text Or _
        Label2.Text = Label3.Text Then
        ' Increase 5 credit and show in Progress Bar
        ProgressBar1.Value += 5

        ' Gain 5 credit for any 2 numbers equal
        MsgBox("You Win 5 credits")
    End If
End If
```

3. In the **MouseDown** event procedure of the button control (**btnSpin**), modify the code as follow:

```
' Reduce 5 credit for each spin
If ProgressBar1.Value > 0 Then
    ' Reduce 5 credit for each spin
    ProgressBar1.Value -= 5

    ' Set the Stop Flag to False (-ve as true)
    StopFlag = 10

    ' Define the time interval at 50/1000 s
    Timer1.Interval = 50

    ' Start the timer
    Timer1.Enabled = True
End If
```

4. Save the project and build the solution, and then execute it. Can you win the game?

