جامعة نجران
NAJRAN UNIVERSITY

328CSS-3 Human Computer Interaction

Visual Basic, Lab Set: 2

Course Instructor: Maha Alwuthaynani

جامعة نجران
NAJRAN UNIVERSITY

Lab Activity 3

Introduction:

- **To List box,**
- **Combo box,**
- **Date time picker**

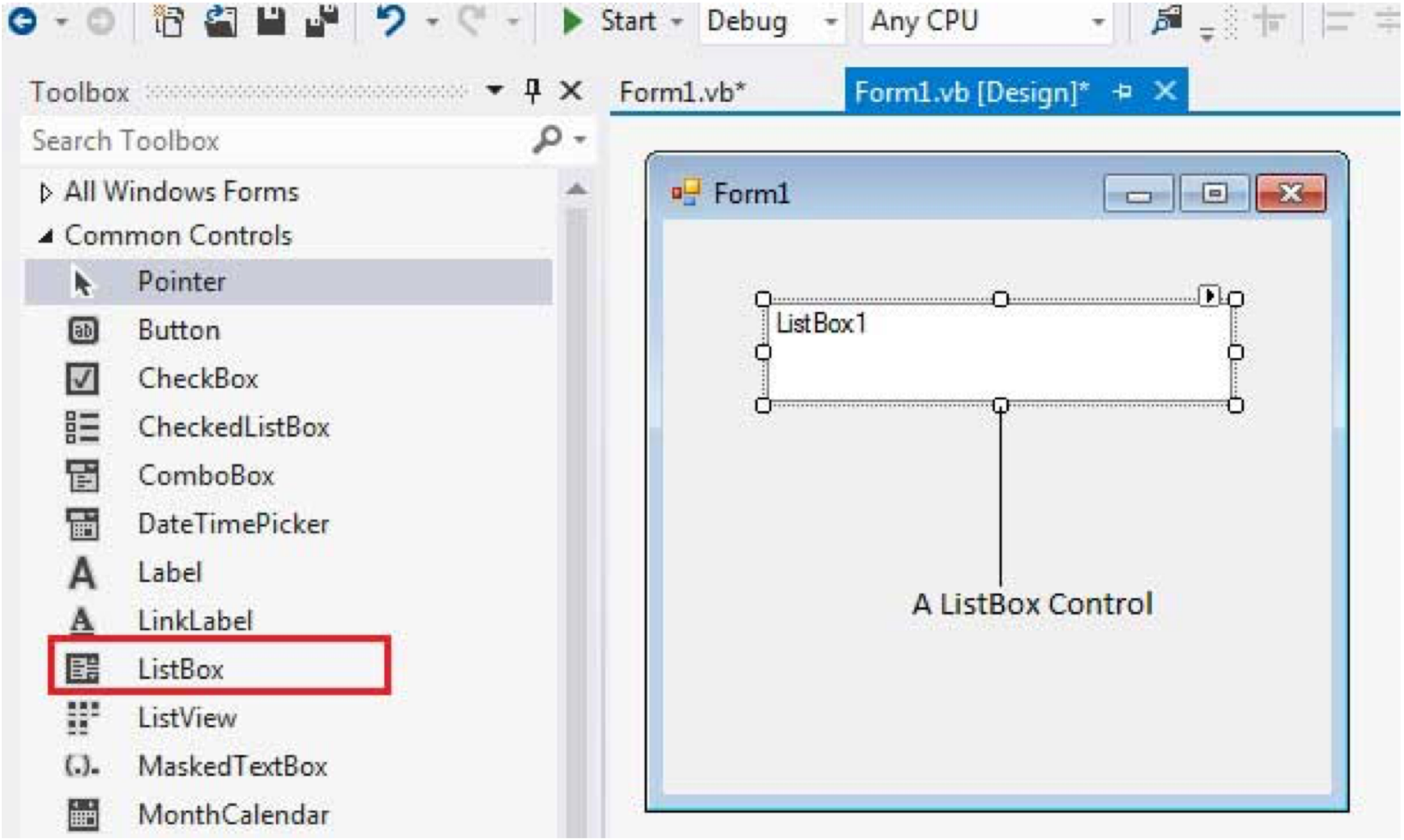
جامعة نجران
NAJRAN UNIVERSITY

ListBox

- The ListBox represents a Windows control to display a list of items to a user. A user can select an item from the list. It allows the programmer to add items at design time by using the properties window or at the runtime.
- Let's create a list box by dragging a ListBox control from the Toolbox and dropping it on the form.

جامعة نجران
NAJRAN UNIVERSITY

ListBox View



جامعة نجران
NAJRAN UNIVERSITY

ListBox Properties

Property	Description
BorderStyle	Gets or sets the type of border drawn around the list box.
ColumnWidth	Gets of sets the width of columns in a multicolumn list box.
ItemHeight	Gets or sets the height of an item in the list box.
Items	Gets the items of the list box.
SelectedItem	Gets or sets the currently selected item in the list box.
Text	Gets or searches for the text of the currently selected item in the list box.
TopIndex	Gets or sets the index of the first visible item of a list box.

جامعة نجران
NAJRAN UNIVERSITY

Example Code

Code Part 1:

```
Public Class Form1

Private Sub Form1_Load(sender As Object, e As
    EventArgs) Handles MyBase.Load
' Set the caption bar text of the form.
Me.Text = "tutorialspont.com"
ListBox1.Items.Add("Canada")
ListBox1.Items.Add("USA")
ListBox1.Items.Add("UK")
ListBox1.Items.Add("Japan")
ListBox1.Items.Add("Russia")
ListBox1.Items.Add("China")
ListBox1.Items.Add("India")
End Sub
```

Code Part 2:

```
Private Sub Button1_Click(sender As Object,
e As EventArgs) Handles Button1.Click
MsgBox("You have selected " +
ListBox1.SelectedItem.ToString())
End Sub

Private Sub
ListBox1_SelectedIndexChanged(sender As
Object, e As EventArgs)      Handles
ListBox1.SelectedIndexChanged
Label2.Text =
ListBox1.SelectedItem.ToString()
End Sub
End Class
```

Programming Concepts

If Statement: executes code based on a condition, the condition must evaluate true for the code to execute.

Syntax:

```
If condition Then
   statement
End If
```

If Else Statement: The If Else Statement works similar to the if statement, however if the first condition is false the else condition will execute.

Syntax:

```
If condition Then
   Statement
Else
   Statement
End If
```

Programming Concepts

Operators:

Operator	Description	Example
<	Less than operator	if 19 < 20 Then
>	Greater than operator	if 20 > 19 Then
=	Equal to operator	if a = b Then
<>	Not equal to operator	if a <> b Then
<=	Less than or equal to operator	if 19 <= b Then
>=	Greater than or equal to operator	if 19 >= b Then

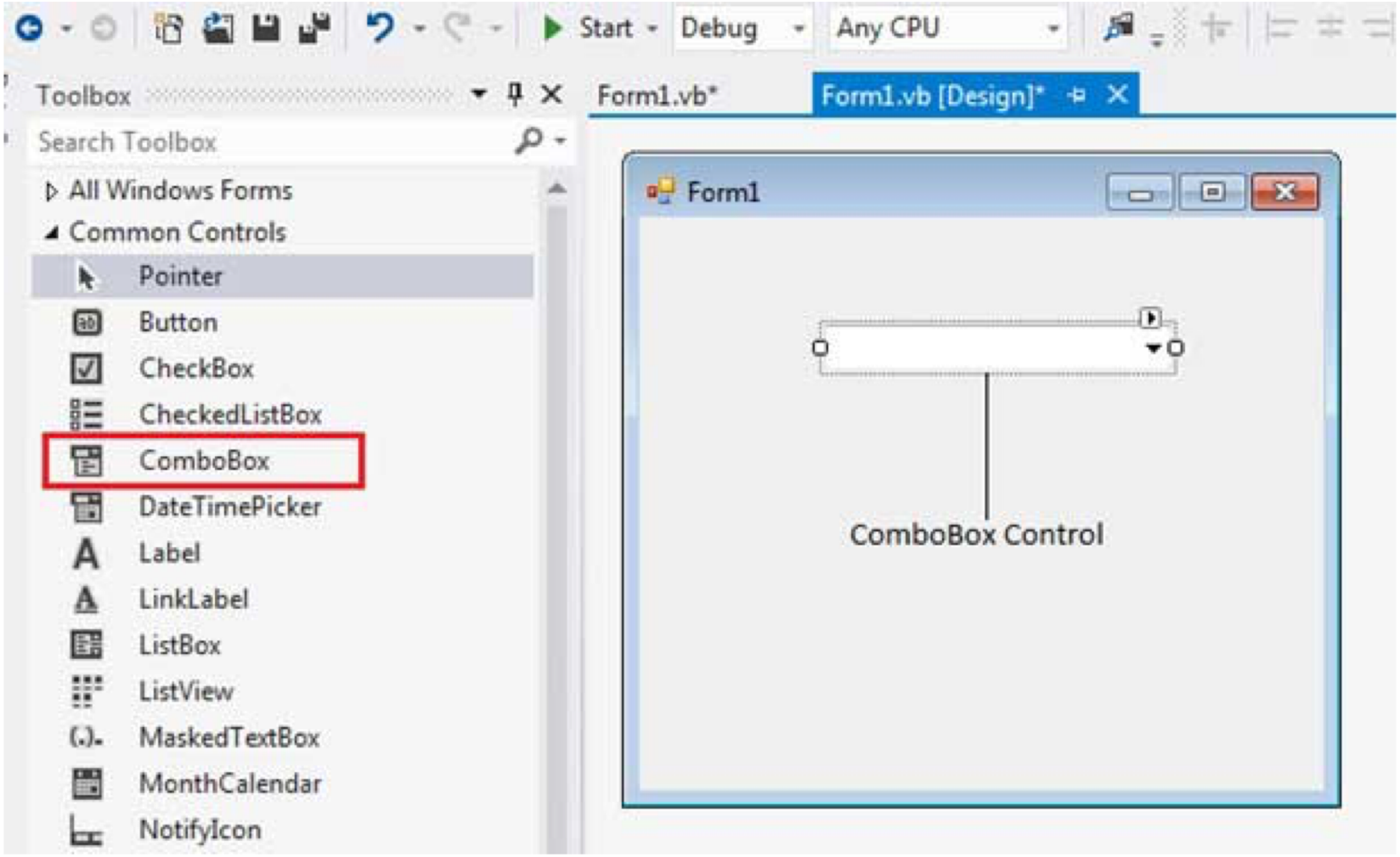
جامعة نجران
NAJRAN UNIVERSITY

ComboBox

- The ComboBox control is used to display a drop-down list of various items. It is a combination of a text box in which the user enters an item and a drop-down list from which the user selects an item.
- Let's create a combo box by dragging a ComboBox control from the Toolbox and dropping it on the form.

جامعة نجران
NAJRAN UNIVERSITY

ComboBox



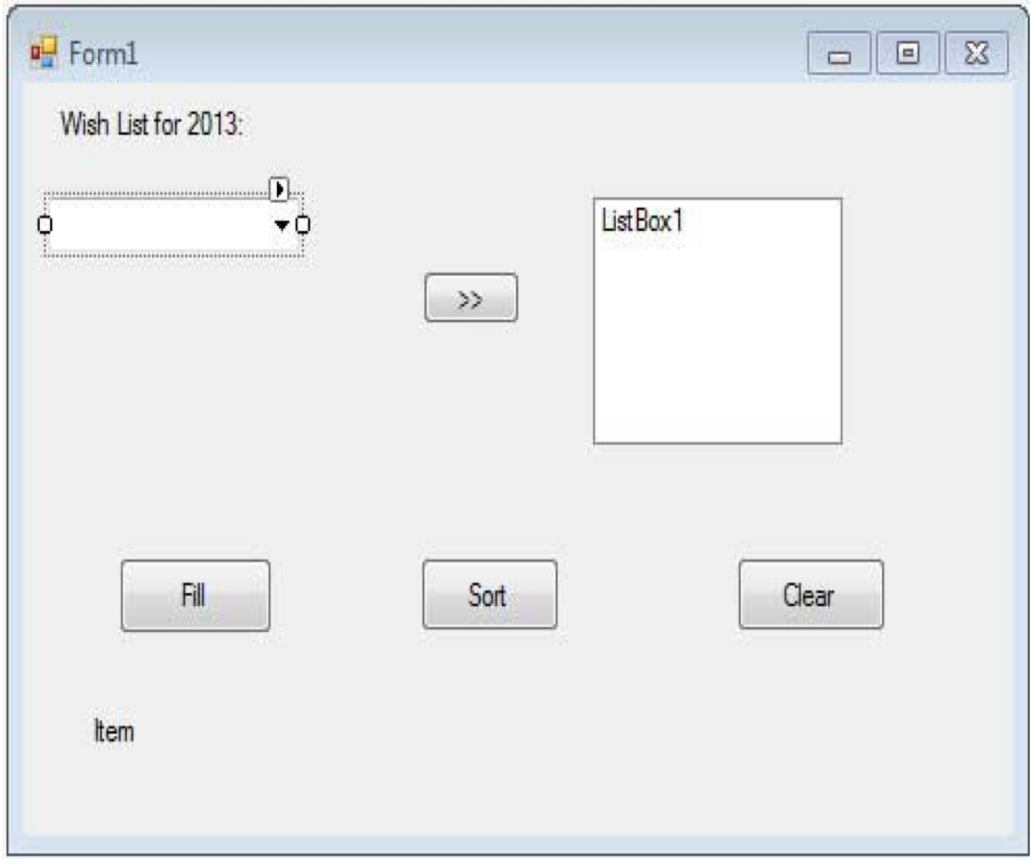
جامعة نجران
NAJRAN UNIVERSITY

ComboBox Properties

Property	Description
DataBindings	Gets the data bindings for the control.
DataManager	Gets the CurrencyManager associated with this control.
DataSource	Gets or sets the data source for this ComboBox.
ItemHeight	Gets or sets the height of an item in the combo box.
Items	Gets an object representing the collection of the items contained in this ComboBox.
MaxLength	Gets or sets the maximum number of characters a user can enter in the editable area of the combo box.
SelectedItem	Gets or sets currently selected item in the ComboBox.
SelectedValue	Gets or sets the value of the member property specified by the ValueMember property.
SelectionStart	Gets or sets the starting index of text selected in the combo box.
Text	Gets or sets the text associated with this control.

ComboBox Example

In this example, let us fill a comboBox with various items, get the selected items in the combo box and show them in a list box and sort the items. Drag and drop a combo box to store the items, a list box to display the selected items, four button controls to add to the list box with selected items, to fill the combo box, to sort the items and to clear the combo box list, respectively. Add a label control that would display the selected item.



ComboBox Example

```
Public Class Form1
  Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    ' Set the caption bar text of the form.
    Me.Text = "tutorialspont.com"
  End Sub
  'sends the selected items to the list box
  Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    If ComboBox1.SelectedIndex > -1 Then
      Dim sindex As Integer
      sindex = ComboBox1.SelectedIndex
      Dim sitem As Object
      sitem = ComboBox1.SelectedItem
      ListBox1.Items.Add(sitem)
    End If
  End Sub
```

ComboBox Example

```
'populates the list
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click

    ComboBox1.Items.Clear()
    ComboBox1.Items.Add("Safety")
    ComboBox1.Items.Add("Security")
    ComboBox1.Items.Add("Governance")
    ComboBox1.Items.Add("Good Music")
    ComboBox1.Items.Add("Good Movies")
    ComboBox1.Items.Add("Good Books")
    ComboBox1.Items.Add("Education")
    ComboBox1.Items.Add("Roads")
    ComboBox1.Items.Add("Health")
     ComboBox1.Items.Add("Food for all")
    ComboBox1.Items.Add("Shelter for all")
    ComboBox1.Items.Add("Industrialisation")
    ComboBox1.Items.Add("Peace")
    ComboBox1.Items.Add("Liberty")
    ComboBox1.Items.Add("Freedom of Speech")
    ComboBox1.Text = "Select from..."
End Sub
```

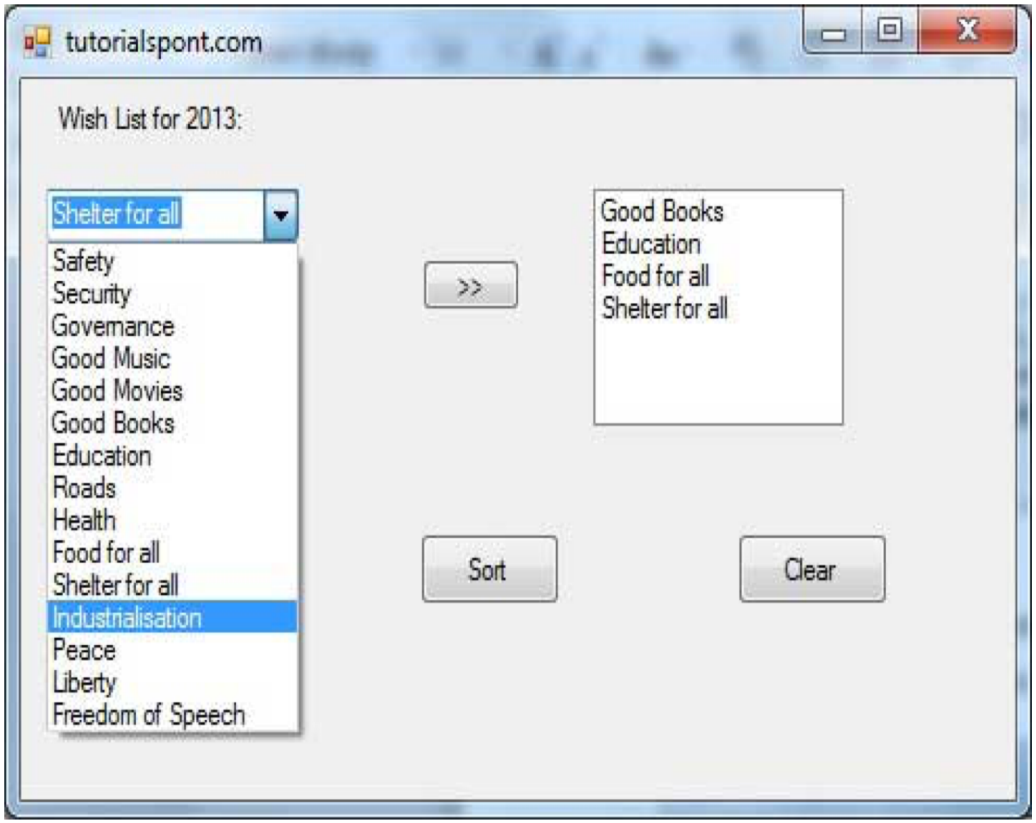
ComboBox Example

```
'sorting the list
 Private Sub Button3_Click(sender As Object, e As EventArgs)
    ComboBox1.Sorted = True
 End Sub


 'clears the list
 Private Sub Button4_Click(sender As Object, e As EventArgs)
    ComboBox1.Items.Clear()
 End Sub


 'displaying the selected item on the label
 Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As EventArgs)
   Handles ListBox1.SelectedIndexChanged
    Label1.Text = ComboBox1.SelectedItem.ToString()
 End Sub
End Class
```

ComboBox Example

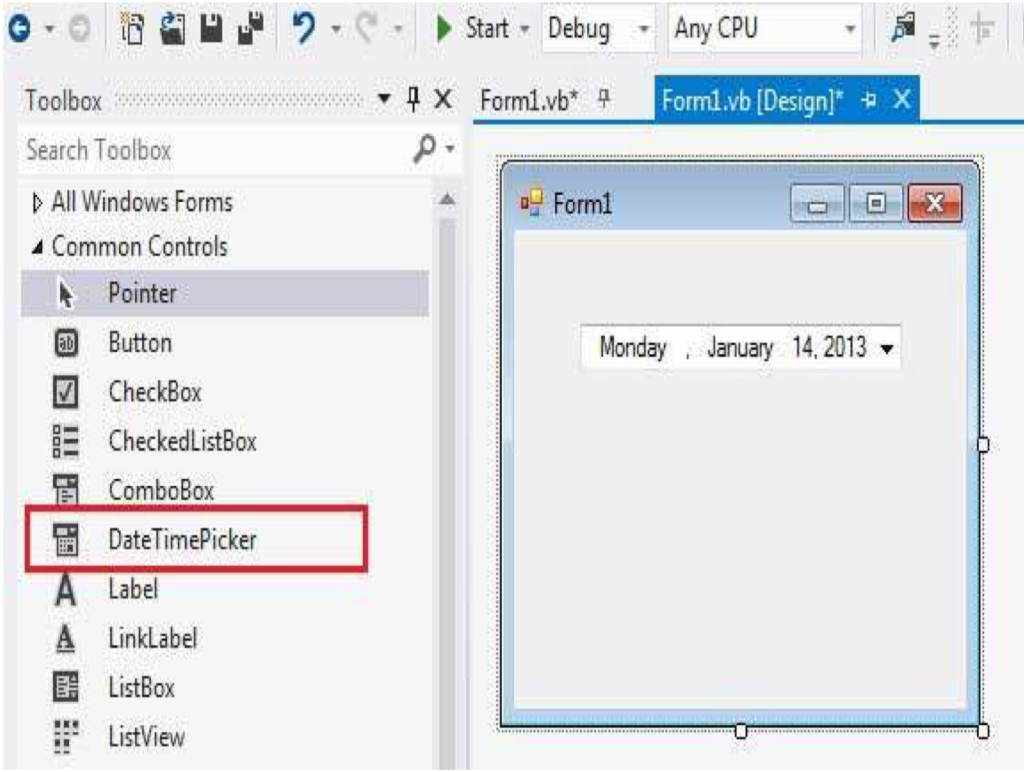
When the above code is executed and run using Start button available at the Microsoft Visual Studio toolbar, it will show the following window:

Click on various buttons to check the actions performed by each:



Date time picker

The DateTimePicker control allows selecting a date and time by editing the displayed values in the control. If you click the arrow in the DateTimePicker control, it displays a month calendar, like a combo box control. The user can make selection by clicking the required date. The new selected value appears in the text box part of the control.



The **MinDate** and the **MaxDate** properties allow you to put limits on the date range.

Properties of the DateTimePicker Control

- **Properties:**
  - **BackColor** Gets or sets a value indicating the background color of the DateTimePicker control.
  - **BackgroundImage** Gets or sets the background image for the control.
  - **BackgroundImageLayout** Gets or sets the layout of the background image of the DateTimePicker control.
  - **CalendarFont** Gets or sets the font style applied to the calendar.
  - **CalendarForeColor** Gets or sets the foreground color of the calendar.
  - **CalendarMonthBackground** Gets or sets the background color of the calendar month.
  - **CalendarTitleBackColor** Gets or sets the background color of the calendar title.
  - **CalendarTitleForeColor** Gets or sets the foreground color of the calendar title.
  - **CalendarTrailingForeColor** Gets or sets the foreground color of the calendar trailing dates.
  - **Checked** Gets or sets a value indicating whether the Value property has been set with a valid date/time value and the displayed value is able to be updated.

Properties of the DateTimePicker Control

- **Properties:**

- **CustomFormat** Gets or sets the custom date/time format string.
- **DropDownAlign** Gets or sets the alignment of the drop-down calendar on the DateTimePicker control.
- **ForeColor** Gets or sets the foreground color of the DateTimePicker control.
- **Format** Gets or sets the format of the date and time displayed in the control.
- **MaxDate** Gets or sets the maximum date and time that can be selected in the control.
- **MaximumDateTime** Gets the maximum date value allowed for the DateTimePicker control.
- **MinDate** Gets or sets the minimum date and time that can be selected in the control.
- **MinimumDateTime** Gets the minimum date value allowed for the DateTimePicker control.
- **PreferredHeight** Gets the preferred height of the DateTimePicker control.
- **RightToLeftLayout** Gets or sets whether the contents of the DateTimePicker are laid out from right to left.
- **ShowCheckBox** Gets or sets a value indicating whether a check box is displayed to the left of the selected date.

Properties of the DateTimePicker Control

- **Properties:**

- **ShowCheckBox** Gets or sets a value indicating whether a check box is displayed to the left of the selected date.
- **ShowUpDown** Gets or sets a value indicating whether a spin button control (also known as an up-down control) is used to adjust the date/time value.
- **Text** Gets or sets the text associated with this control.
- **Value** Gets or sets the date/time value assigned to the control.

Visual Basic Methods Concept

We use methods (or procedures) for reusing code and making code more understandable. A method is just a block of code that you can call, and huge programs have methods.

Syntax:

```
Private Sub methodName()

End Sub
```

- **Methods:**
- **ToString** Returns the string representing the control.

Visual Basic Events Concept

An event is a signal that informs an application that something important has occurred. For example, when a user clicks a control on a form, the form can raise a Click event and call a procedure that handles the event. Events also allow separate tasks to communicate. Say, for example, that your application performs a sort task separately from the main application. If a user cancels the sort, your application can send a cancel event instructing the sort process to stop.

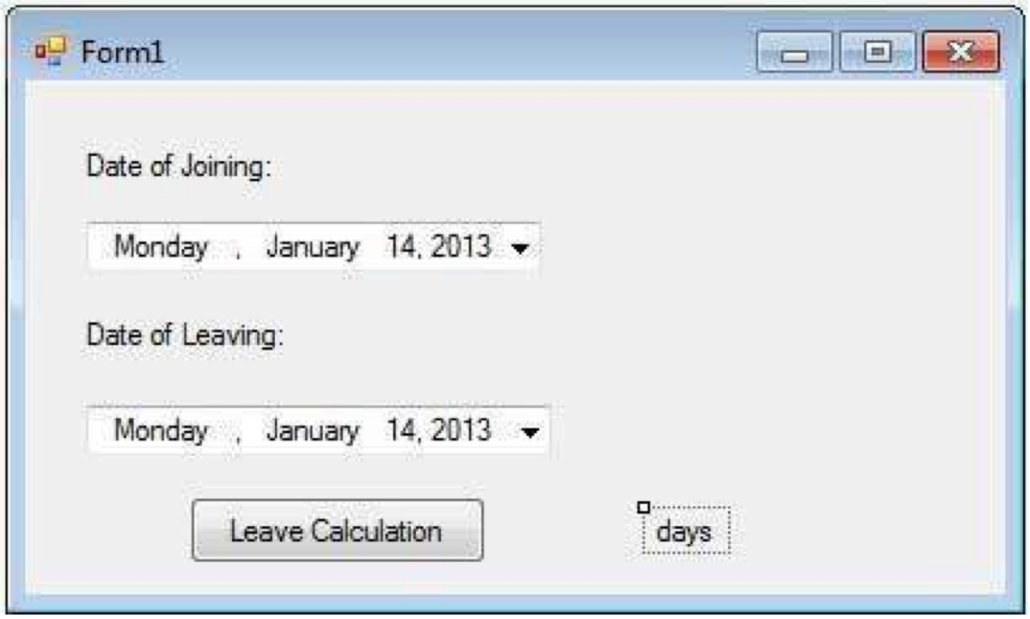
DateTimePicker Control

Events:

- **BackColorChanged** Occurs when the value of the BackColor property changes.
- **BackgroundImageChanged** Occurs when the value of the BackgroundImage property changes.
- **BackgroundImageLayoutChanged** Occurs when the value of the BackgroundImageLayout property changes.
- **Click** Occurs when the control is clicked.
- **CloseUp** Occurs when the drop-down calendar is dismissed and disappears.
- **DoubleClick** Occurs when the control is double-clicked.
- **DragDrop** Occurs when a drag-and-drop operation is completed.
- **ForeColorChanged** Occurs when the value of the ForeColor property changes.
- **FormatChanged** Occurs when the Format property value has changed.
- **MouseClick** Occurs when the control is clicked with the mouse.
- **MouseDoubleClick** Occurs when the control is double-clicked with the mouse.
- **TextChanged** Occurs when the value of the Text property changes.
- **ValueChanged** Occurs when the Value property changes.

DateTimePicker Example

In this example, create a small application for calculating days of leave. Add two DateTimePicker controls on the form, where the user will enter the date of going on leave and the date of joining. Keep a button control for performing the calculation and appropriate label controls for displaying information.

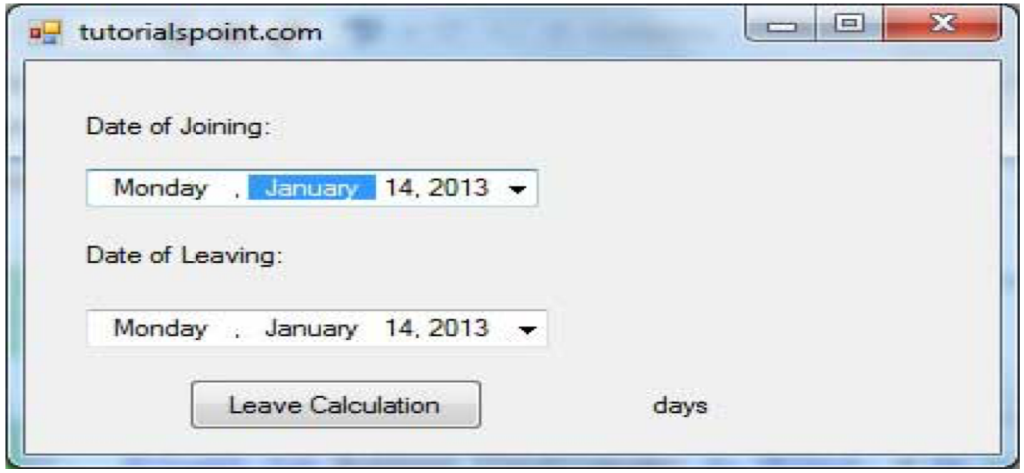


DateTimePicker Example

```
Public Class Form1
  Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    ' Set the caption bar text of the form.
    Me.Text = "tutorialspoint.com"
  End Sub
  Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    Dim d1 As DateTime = DateTimePicker1.Value
    Dim d2 As DateTime = DateTimePicker2.Value
    Dim result As TimeSpan = d1.Subtract(d2)
    Dim days As Integer = result.TotalDays
    Label3.Text = days
  End Sub
End Class
```

DateTimePicker Example

When the above code is executed and run using Start button available at the Microsoft Visual Studio tool bar, it will show the following window:



Select two dates and click on the button for leave calculation: