

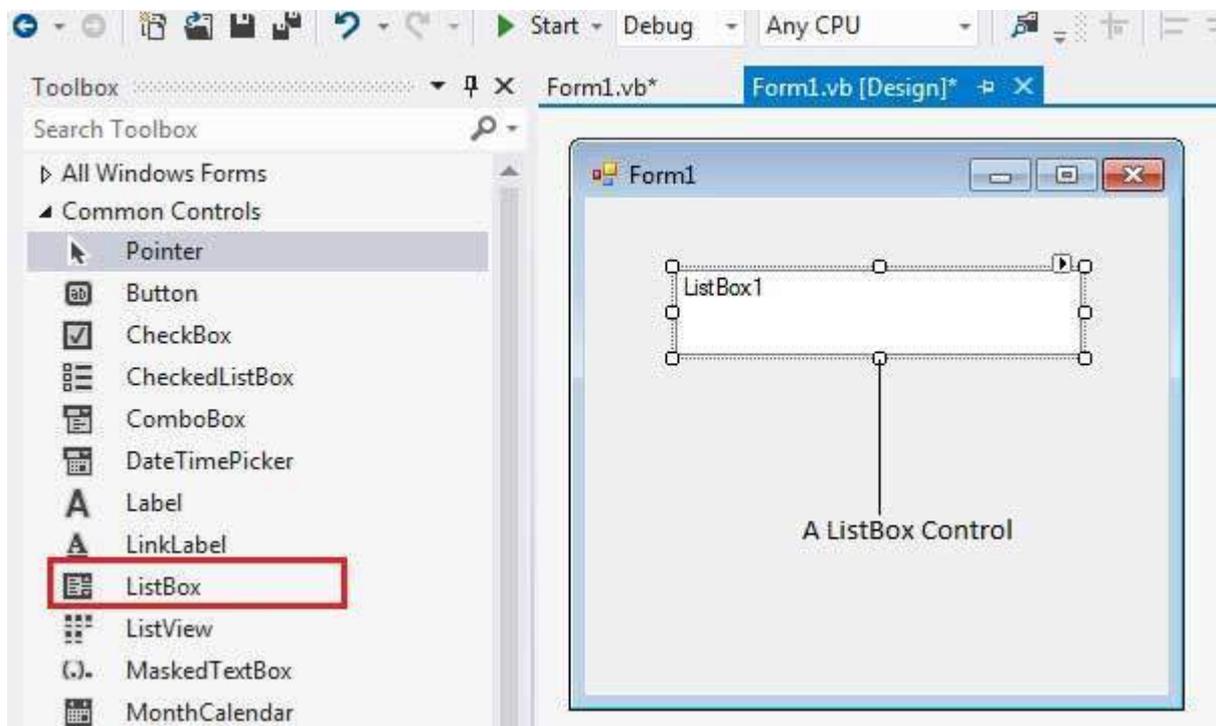
VB.NET - LISTBOX CONTROL

http://www.tutorialspoint.com/vb.net/vb.net_listbox.htm

Copyright © tutorialspoint.com

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.



You can populate the list box items either from the properties window or at runtime. To add items to a ListBox, select the ListBox control and get to the properties window, for the properties of this control. Click the ellipses ... button next to the Items property. This opens the String Collection Editor dialog box, where you can enter the values one at a line.

Properties of the ListBox Control

The following are some of the commonly used properties of the ListBox control:

S.N	Property	Description
1	AllowSelection	Gets a value indicating whether the ListBox currently enables selection of list items.
2	BorderStyle	Gets or sets the type of border drawn around the list box.
3	ColumnWidth	Gets or sets the width of columns in a multicolumn list box.
4	HorizontalExtent	Gets or sets the horizontal scrolling area of a list box.
5	HorizontalScrollBar	Gets or sets the value indicating whether a horizontal scrollbar is displayed in the list box.
6	ItemHeight	Gets or sets the height of an item in the list box.
7	Items	Gets the items of the list box.
8	MultiColumn	Gets or sets a value indicating whether the list box supports multiple columns.

9	ScrollAlwaysVisible	Gets or sets a value indicating whether the vertical scroll bar is shown at all times.
10	SelectedIndex	Gets or sets the zero-based index of the currently selected item in a list box.
11	SelectedIndices	Gets a collection that contains the zero-based indexes of all currently selected items in the list box.
12	SelectedItem	Gets or sets the currently selected item in the list box.
13	SelectedItems	Gets a collection containing the currently selected items in the list box.
14	SelectedValue	Gets or sets the value of the member property specified by the ValueMember property.
15	SelectionMode	Gets or sets the method in which items are selected in the list box. This property has values: <ul style="list-style-type: none"> • None • One • MultiSimple • MultiExtended
16	Sorted	Gets or sets a value indicating whether the items in the list box are sorted alphabetically.
17	Text	Gets or searches for the text of the currently selected item in the list box.
18	TopIndex	Gets or sets the index of the first visible item of a list box.

Methods of the ListBox Control

The following are some of the commonly used methods of the ListBox control:

S.N	Method Name & Description
1	BeginUpdate Prevents the control from drawing until the EndUpdate method is called, while items are added to the ListBox one at a time.
2	ClearSelected Unselects all items in the ListBox.
3	EndUpdate Resumes drawing of a list box after it was turned off by the BeginUpdate method.
4	FindString Finds the first item in the ListBox that starts with the string specified as an argument.
5	FindStringExact

Finds the first item in the ListBox that exactly matches the specified string.

6

GetSelected

Returns a value indicating whether the specified item is selected.

7

SetSelected

Selects or clears the selection for the specified item in a ListBox.

8

OnSelectedIndexChanged

Raises the SelectedIndexChanged event.

8

OnSelectedValueChanged

Raises the SelectedValueChanged event.

Events of the ListBox Control

The following are some of the commonly used events of the ListBox control:

S.N	Event	Description
1	Click	Occurs when a list box is selected.
2	SelectedIndexChanged	Occurs when the SelectedIndex property of a list box is changed.

Consult Microsoft documentation for detailed list of properties, methods and events of the ListBox control.

Example 1

In the following example, let us add a list box at design time and add items on it at runtime.

Take the following steps:

- Drag and drop two labels, a button and a ListBox control on the form.
- Set the Text property of the first label to provide the caption "Choose your favourite destination for higher studies".
- Set the Text property of the second label to provide the caption "Destination". The text on this label will change at runtime when the user selects an item on the list.
- Click the listbox and the button controls to add the following codes in the code editor.

```
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")
    End Sub
End Class
```

```

    ListBox1.Items.Add("Japan")
    ListBox1.Items.Add("Russia")
    ListBox1.Items.Add("China")
    ListBox1.Items.Add("India")
End Sub

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

```

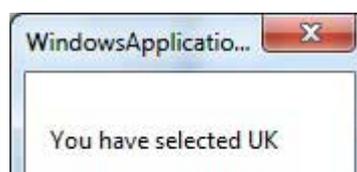
When the above code is executed and run using Start button available at the Microsoft Visual Studio tool bar, it will show the following window:

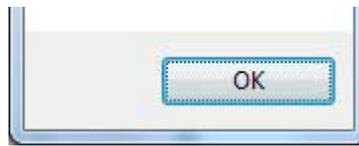


When the user chooses a destination, the text in the second label changes:



Clicking the Select button displays a message box with the user's choice:

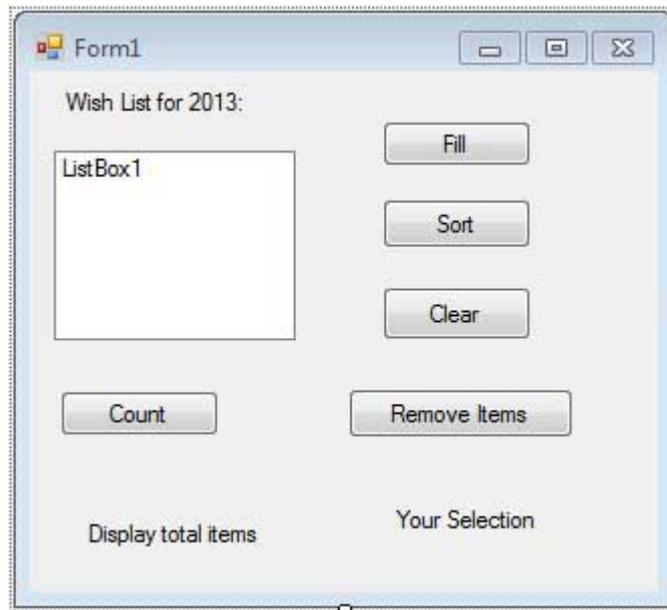




Example 2

In this example, we will fill up a list box with items, retrieve the total number of items in the list box, sort the list box, remove some items and clear the entire list box.

Design the Form:



Add the following code in the code editor window:

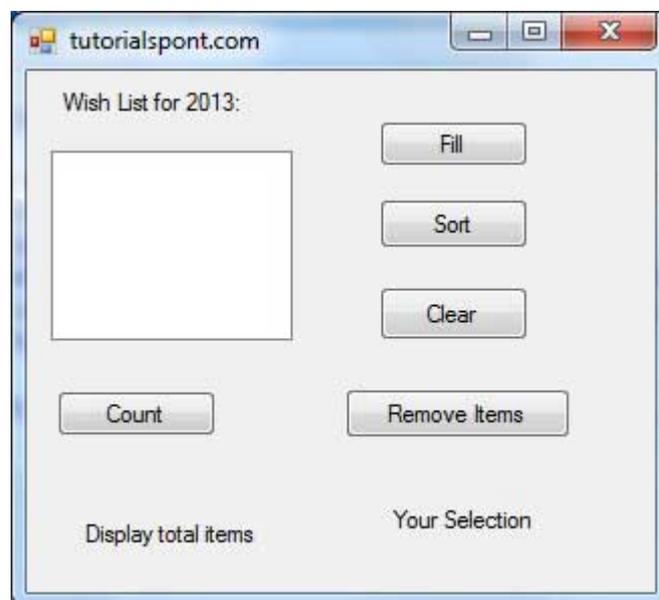
```
Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        ' Set the caption bar text of the form.
        Me.Text = "tutorialspont.com"
        ' creating multi-column and multiselect list box
        ListBox1.MultiColumn = True
        ListBox1.SelectionMode = SelectionMode.MultiExtended
    End Sub
    'populates the list
    Private Sub Button1_Click_1(sender As Object, e As EventArgs) _
        Handles Button1.Click
        ListBox1.Items.Add("Safety")
        ListBox1.Items.Add("Security")
        ListBox1.Items.Add("Governance")
        ListBox1.Items.Add("Good Music")
        ListBox1.Items.Add("Good Movies")
        ListBox1.Items.Add("Good Books")
        ListBox1.Items.Add("Education")
        ListBox1.Items.Add("Roads")
        ListBox1.Items.Add("Health")
        ListBox1.Items.Add("Food for all")
        ListBox1.Items.Add("Shelter for all")
        ListBox1.Items.Add("Industrialisation")
        ListBox1.Items.Add("Peace")
        ListBox1.Items.Add("Liberty")
        ListBox1.Items.Add("Freedom of Speech")
    End Sub
    'sorting the list
    Private Sub Button2_Click(sender As Object, e As EventArgs) _
        Handles Button2.Click
        ListBox1.Sorted = True
    End Sub
    'clears the list
```

```

Private Sub Button3_Click(sender As Object, e As EventArgs) _
    Handles Button3.Click
    ListBox1.Items.Clear()
End Sub
'removing the selected item
Private Sub Button4_Click(sender As Object, e As EventArgs) _
    Handles Button4.Click
    ListBox1.Items.Remove(ListBox1.SelectedItem.ToString)
End Sub
'counting the numer of items
Private Sub Button5_Click(sender As Object, e As EventArgs) _
    Handles Button5.Click
    Label1.Text = ListBox1.Items.Count
End Sub
'displaying the selected item on the third label
Private Sub ListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) _
    Handles ListBox1.SelectedIndexChanged
    Label3.Text = ListBox1.SelectedItem.ToString()
End Sub
End Class

```

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:



Fill the list and check workings of other buttons:

