

Using ListBox and ComboBox Controls In Visual Basic 6:

The ListBox will display a single column of strings, referred to as **items**. The items to appear initially can either be specified at design time with the List property or set with code in the Form_Load event procedure. Then code is used to access, add, or delete items from the list. If the number of items exceed the value that be displayed, scroll bars will automatically appear on the control. These scroll bars can be scrolled up and down or left to right through the list.

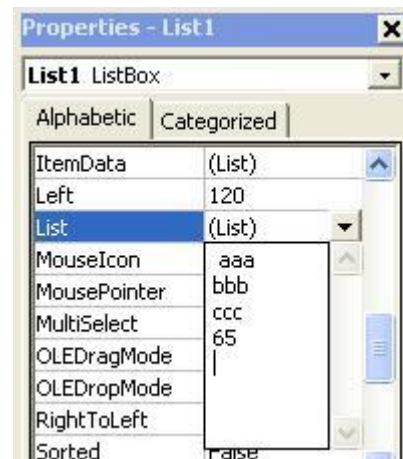
A ComboBox is best through of as a text box with a help list attached. With an ordinary textbox, the user must type information into the box. With a combobox, the user has the option of either typing in information or just selecting the appropriate piece of information from a list

Property	Description
Properties	
Enabled	By setting this property to True or False user can decide whether user can interact with this control or not
List	String array. Contains the strings displayed in the drop-down list. Starting array index is 0. Use CTRL+Enter to insert values.
Sorted	Boolean. Specifies whether the ListBox &ComboBox items are sorted or not.
Style	Integer. Specifies the style of the ListBox &ComboBox appearance
Text	String. Specifies the selected item in the ComboBox.
Visible	Boolean. Specifies whether ListBox &ComboBox is visible or not at run time
Event Procedures	
Change	Called when text in ComboBox is changed
Click	Called when the ListBox &ComboBox is clicked

Methods	Description	Example
AddItem	Add an item to the ListBox &ComboBox	List1.additem item:Combo1.additem item
ListCount	Integer. Contains the number of drop-down list items	X=List1.listcount: Y=Combo1.listcount
ListIndex	Integer. Contains the index of the selected ListBox &ComboBox item. If an item is not selected, ListIndex is -1	X=List1.ListIndex : Y=Combo1.ListIndex
List	String array. Contains the strings displayed in the drop-down list. Starting array index is 0.	X=List1.List(1): Y=Combo1.List(4) X=List1.List(List1.ListIndex)
Text	String. Specifies the selected item in the ListBox &ComboBox.	X=List1.Text: Y=Combo1.Text
Clear	Removes all items from the ListBox &ComboBox	List1.Clear: Combo1.Clear
RemoveItem	Removes the specified item from the ListBox &ComboBox	List1.RemoveItem 1: Combo1.RemoveItem 5 List1.RemoveItem List1.ListIndex
NewIndex	Integer. Index of the last item added to the ListBox &ComboBox. If the ComboBox does not contain any items , NewIndex is -1	X=list1.NewIndex: Y= Combo1.NewIndex

Adding items to a ListBox &ComboBox : It is possible to populate the list at design time or run time

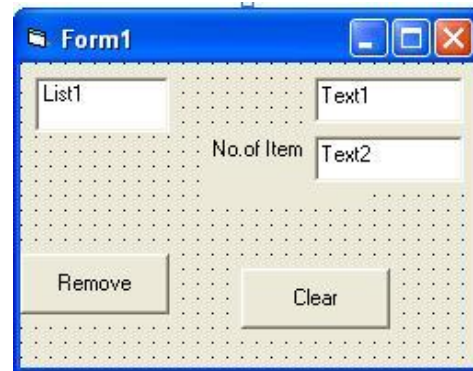
- **Design Time :** To add items to a list at design time, click on List property in the property box and then add the items. Press CTRL+ENTER after adding each item as shown below.



Example 1: Design a form with one list box, two textboxes and two command buttons.

Write code for the following-events.

- 1- Form_Load event, to add items.(3)
- 2- In click event of listbox,to add item to text1 from list box if item is selected
- 3- In click event of command1 (Remove), to remove item from list box if item is selected and display the number of items in the listbox into text2.
- 4- In click event of command2 (Clear), to clear items from list box.



Solution:

```

Private Sub
Form_Load()
List1.AddItem "Ail"
List1.AddItem "Ahmed"
List1.AddItem "Sara"
Text2.Text=List1.ListCount
End Sub
-----
Private Sub List1_Click()
Text1.text=List1.Text or Text1.Text=List1.List (List1.ListIndex).
End Sub
-----
Private Sub Command1_Click()
List1.RemoveItem
List1.ListIndex End Sub
-----
Private Sub Command2_Click()
List1.Clear
End Sub
    
```

The two most useful types of combobox are denoted as style property combobox as shown in Figure below.



With a style 1 combo box, the list is always visible. With style 0 or 2 combobox, the list drops down when the user clicks on the arrow. In either case, when an item from the list is highlighted, the item automatically appears in the text box at the top and its value is assigned to the text property of the combo box. The items to appear initially can either be specified at design time with the combo property or set with `combo_change()` event procedure directly.

ComboBoxes have essentially the same properties, event and methods as ListBoxes

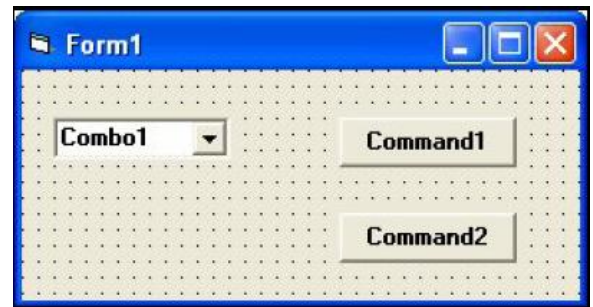
Example 2 c inside the combobox when click on the one of command button and clear any item when you click another command

Solution:

```
Private Sub Form_Load()
    Command1.Caption = "Additem"
    Command2.Caption = "Removeitem"
End Sub
```

```
Private Sub Command1_Click()
    Combo1.AddItem Combo1.Text: Combo1.Text = ""
End Sub
```

```
Private Sub Command2_Click()
    Combo1.RemoveItem Combo1.ListIndex
End Sub
```



Example 3: Design a form with one command button for change background the form after selected the color from combo box

Solution:

```
Private Sub Form_Load()
    Command1.Caption = "Set Backcolor"
    Combo1.AddItem "vbred": Combo1.AddItem "vbblue"
    Combo1.AddItem "vbgreen": Combo1.AddItem "vbcyan"
    Combo1.AddItem "vbyellow": Combo1.AddItem "vbwhite"
End Sub
```

```
Private Sub Command1_Click()
    Form1.BackColor = QBColor(Combo1.ListIndex)
End Sub
```

التسلسل حسب اللون اختيار
المركب الصندوق داخل
الألوان أسماء حسب وليس
المكتوبة