

University of Anbar
College of Computer Science
and Information Technology
Computer Network Systems Department



Visual Programming I

Lecture Twelve
Third Stage

First Course 2021 - 2022

Seddiq Qais Abd Al-Rahman

MSc Computer Science

co.sedeikaldossary@uoanbar.edu.iq

Visual Programming

List Box

A list box control displays a list of string items in a box. It provides a number of facilities including the ability to select an item in a list by clicking on it, add items and delete items. The following are the properties and methods of the list box that you will commonly use:

Properties

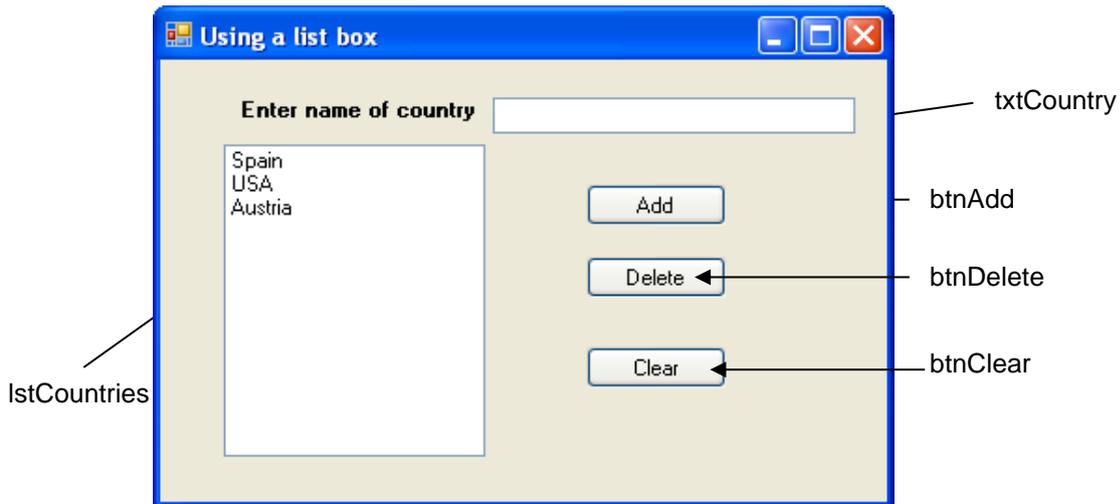
Items	Contains all of the items in the list box. Each item is indexed. The first item is indexed as 0, the next 1 and so on.
Items.Count	Contains the number of items in the list box.
SelectedIndex	Contain the index or number of the item currently selected. For example, if the 6th item is selected it will have the value of 5 (since the first item is numbered as 0). If no item is selected its value is -1.
Sorted	Has a value of either true or false and specifies whether or not the items are sorted for display.
Text	Contains the currently selected item from the list box.

Methods

Items.Add	Adds an item to the list box.
Items.Clear	Removes all items from the list box.
Items.RemoveAt	Removes an item from the list box. We must supply the index value of the item to be removed.

Specification: Allow the user to enter the name of a country into a text box. Provide three buttons to add the name to a list box and to display all of the names alphabetically, to delete a selected country and to delete all of the countries.

1. Open a new project. Sets the form's text property to Using a List Box.
2. On the form place a label, three buttons, a list box and a text box onto the form and position them as shown below.



- Set the properties of these controls as listed in the table below. The purpose of the Items property in design mode is to populate the list box before the program starts. To do this click the ellipsis button in the Items property and type the three countries listed into the String Collection Editor.

Control	Property	Property Setting
Label	Text	Enter name of country
Text box	Name Text	txtCountryt blank (i.e. delete <i>Text1</i>)
Button	Name Text	btnAdd Add
Button	Name Text	btnDelete Delete
Button	Name Text	btnClear Clear
List box	Name Sorted Items	lstCountries true Germany France Japan

Coding

A lot of the code from now on will contain comments. These help other programmers (and sometimes yourself!) to understand the code that has been written. The compiler will ignore anything on a line after two forward slashes.

- Double click the Add button to bring up the template for the Click event method. Complete the code by adding the following code between the curly braces. The Items.Add method is used to add an item to the list box. Whatever the user has typed into the text box will be added as a new item in the list box.

```
lstCountries.Items.Add(txtCountry.Text); //add contents of text
//box to list box display
txtCountry.Text = ""; //clear the text box
txtCountry.Focus(); //put cursor in text box ready for the next
```

```
//country
```

2. Double click the Delete button and add the lines of code below. The `Items.RemoveAt` method is used to delete the item currently selected. The property `SelectedIndex` holds the index value of this item. If the third item is selected it holds the value 2 because indexing starts from 0.

```
//delete selected item from list of displayed items  
lstCountries.Items.RemoveAt(lstCountries.SelectedIndex);
```

3. Double click the Clear button and add the lines of code below.

```
//delete all items displayed in the list box  
lstCountries.Items.Clear();
```

Change a Picture box's BackColor using scroll bars

Specification: Use three horizontal scroll bars so that a Picture box's colour changes as the user moves the scroll bars.

To understand how this program works you need to understand how the C# `Color.FromArgb` method works. This method returns a colour structure by mixing proportions of the three primary colours red, green and blue. Each of the colours can have a value from 0 to 255. So the code

```
pic1.BackColor = Color.FromArgb(255,0,0) //max red, no green or blue  
pic2.BackColor = Color.FromArgb(0,255,0) //max green, no red or blue  
pic3.BackColor = Color.FromArgb(125,125,125) //equal mixtures
```

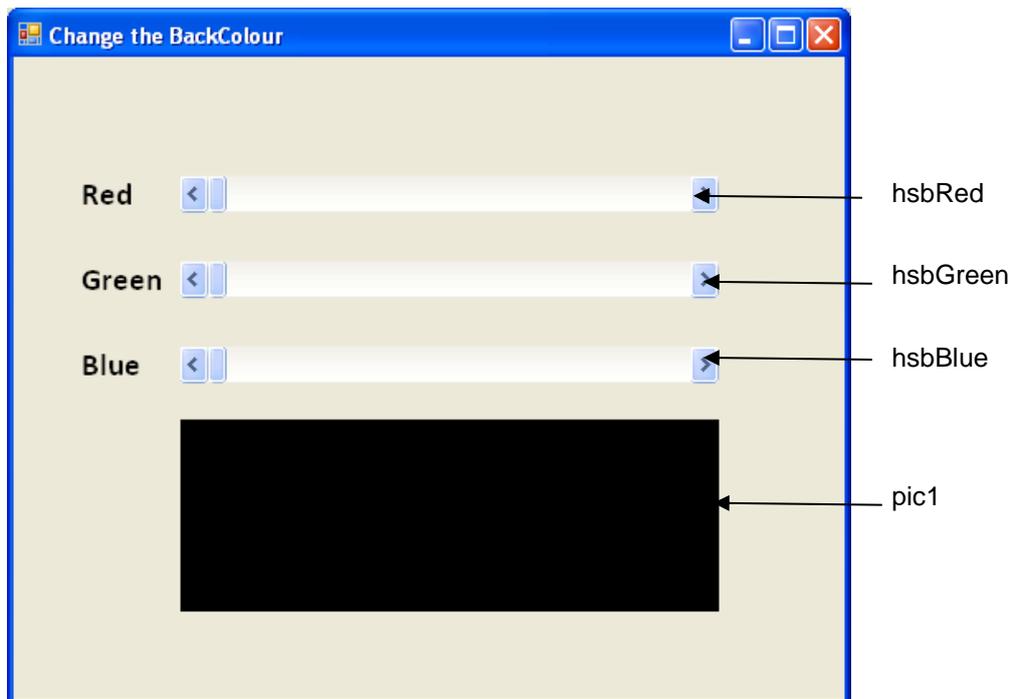
would set the `BackColor` property of `pic1` to red and `pic2` to green. The third `PictureBox` gets equal amounts of the three colours to produce a grey. The three numbers given to the method are examples of method arguments.

1. Open a new project. Sets the form's text property to Change the BackColor.
2. Place three labels, three horizontal scroll bars and a picturebox onto the form as shown in the figure below.
3. Change the Text properties of the labels to Red, Green, and Blue as appropriate.
4. Name the scroll bars `hsbRed`, `hsbGreen` and `hsbBlue`.
5. Name the picturebox `pic1`.
6. Because the default `Minimum` property of a scroll bar is 0 leave this alone. Set the `Maximum` property to 255. To set all three at once select one of them, and with the `Shift` key pressed down select the other two. When you have two or more controls selected, the `Properties` window lists those properties that the controls have in common.
7. Set the `BackColor` property of the picture box to `Black` and the `BorderStyle` property to `FixedSingle`.
8. Double click the red scroll bar to bring up its default `Scroll` event template. Type in the code to set the picturebox colour as follows:

```
pic1.BackColor = Color.FromArgb(hsbRed.Value, hsbGreen.Value, hsbBlue.Value);
```

The line of code is split over two lines only because it will not fit across the page in the handout. The numbers used as arguments to the FromArgb method are read from the Value property of each of the scroll bars.

9. Double click the green scroll bar and type in the same code (you can use copy and paste)
10. Repeat the previous step for the blue scroll bar.
11. Run the program and as you move the scroll bars the colour of the picture box will change. Because each scroll bar has 256 possible values you can set the colour to $256 \times 256 \times 256 = 16,777,216$ colours!



References:

- Tony Gaddis, “Starting out with Visual C#.”, Fourth edition, Boston, Pearson Inc., 2017.
- Salvatore A. Buono, “C# and Game Programming: A Beginner's Guide.” Second Edition, Boca Raton, CRC Press Inc., 2019.
- Eric Butow and Tommy Ryan, "C#: Your Visual Blueprint for Building .NET Applications.", Hungry Minds Inc., New York, 2002.
- Faraz Rasheed, “Programmer's Heaven: C# School.”, First Edition, Fuengirola, Synchron Data, 2006.