**ListBox**

The ListBox control enables you to display a list of items to the user that the user can select by clicking.

Constructors: ListBox

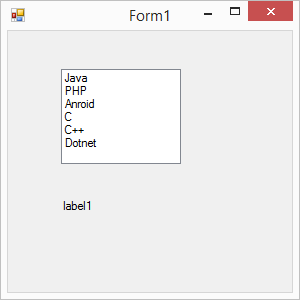
Properties: Items, SelectedIndex, SelectedItem, SelectedValue, size, sorted

Methods: contains, Dispose, FindString, GetItemText, GetSelected

**program 1**

**1) add items to listbox constant**

**2) select any item form the listbox and display in label**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

label1.Text = listBox1.SelectedItem.ToString();

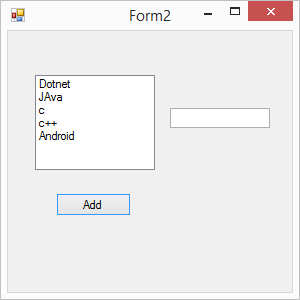
}

}

}

**Program 2**

**1) add items to listbox using listclass**



using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form2 : Form

{

public Form2()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

// List class

List<string> Courses = new List<string>();

Courses.Add("Dotnet");

Courses.Add("JAva");

Courses.Add("c");

Courses.Add("c++");

Courses.Add("Android");

// add the List class values into listbox

foreach(string st in Courses)

{

listBox1.Items.Add(st);

}

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

textBox1.Text = listBox1.SelectedItem.ToString();

}

}

}

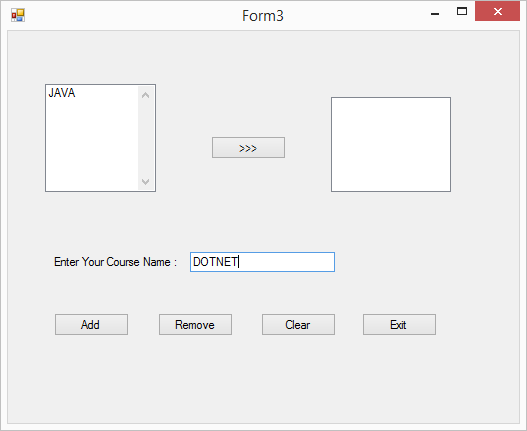
Program 3

1) add items to list box in dynamically

2) remove items to listbox in dynamically

3) clear the items listbox dynamically

4) add the items from one listbox to another listbox



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form3 : Form

{

public Form3()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

if (textBox1.Text != "")

{

// check couse name in listbox

if (listBox1.Items.Contains(textBox1.Text) == true)

{

MessageBox.Show(" This course Already added....");

}

else

{

listBox1.Items.Add(textBox1.Text);

textBox1.Clear();

textBox1.Focus();

}

}

else

{

MessageBox.Show(" Please Enter Your Course Name....");

textBox1.Focus();

}

}

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

// only take the Letters

if(char.IsLetter(e.KeyChar) || e.KeyChar==8 || e.KeyChar==13)

{

e.Handled = false;

if(e.KeyChar==13)

{

if (textBox1.Text != "")

{

if (listBox1.Items.Contains(textBox1.Text) == true)

{

MessageBox.Show(" This course Already added....");

}

else

{

listBox1.Items.Add(textBox1.Text);

textBox1.Clear();

textBox1.Focus();

}

}

else

{

MessageBox.Show(" Please Enter Your course Name...");

textBox1.Focus();

}

}

}

else

{

e.Handled = true;

}

}

private void button2\_Click(object sender, EventArgs e)

{

if (listBox1.SelectedIndex >= 0)

{

listBox1.Items.RemoveAt(listBox1.SelectedIndex);

MessageBox.Show("Your Item Remove Sucessfully...");

}

else

{

MessageBox.Show(" Please Select Your Deleted Course Name....");

}

}

private void button3\_Click(object sender, EventArgs e)

{

listBox1.Items.Clear();

listBox2.Items.Clear();

}

private void button5\_Click(object sender, EventArgs e)

{

if (listBox2.Items.Contains(listBox1.SelectedItem.ToString()) == false)

{

listBox2.Items.Add(listBox1.SelectedItem.ToString());

}

else

{

MessageBox.Show(" This course already added.....");

}

}

private void button4\_Click(object sender, EventArgs e)

{

Application.Exit();

}

}

}

**Combox box :**  It is an editable textbox with dorp down listbox of permitted values

Constructors: comboBox

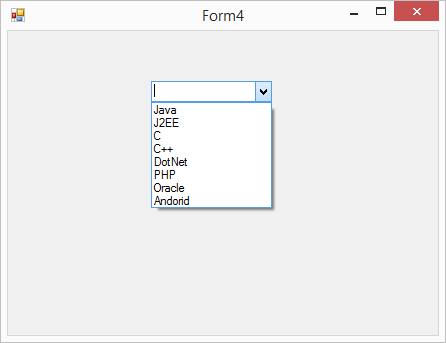
Properties: Items, SelectedIndex, SelectedItem , SelectedText ,SelectedValue, Sorted

Methods: Dispose, FindString

**program 4**

**1) add items to combobox constant**

**2) select any item form the combobox and display in label**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form4 : Form

{

public Form4()

{

InitializeComponent();

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

label1.Text = comboBox1.SelectedItem.ToString();

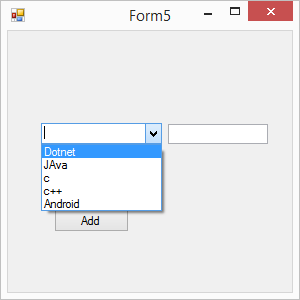
}

}

}

**Program 5**

1. **add items to Combobox using listclass**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form5 : Form

{

public Form5()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

// List class

List<string> Courses = new List<string>();

Courses.Add("Dotnet");

Courses.Add("JAva");

Courses.Add("c");

Courses.Add("c++");

Courses.Add("Android");

// add the List class values into listbox

foreach (string st in Courses)

{

comboBox1.Items.Add(st);

}

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

textBox1.Text = comboBox1.SelectedItem.ToString();

}

}

}

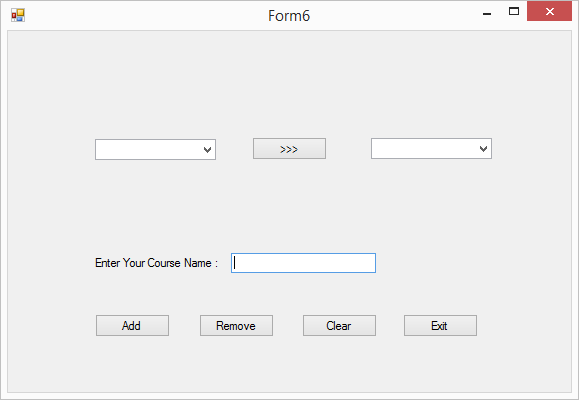
**Program 6**

**1) add items to combo box in dynamically**

**2) remove items to combobox in dynamically**

**3) clear the items combobox dynamically**

**4) add the items from one combobox to another combobox**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form6 : Form

{

public Form6()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

if (textBox1.Text != "")

{

// check couse name in listbox

if (comboBox1.Items.Contains(textBox1.Text) == true)

{

MessageBox.Show(" This course Already added....");

}

else

{

comboBox1.Items.Add(textBox1.Text);

textBox1.Clear();

textBox1.Focus();

}

}

else

{

MessageBox.Show(" Please Enter Your Course Name....");

textBox1.Focus();

}

}

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

// only take the Letters

if (char.IsLetter(e.KeyChar) || e.KeyChar == 8 || e.KeyChar == 13)

{

e.Handled = false;

if (e.KeyChar == 13)

{

if (textBox1.Text != "")

{

if (comboBox1.Items.Contains(textBox1.Text) == true)

{

MessageBox.Show(" This course Already added....");

}

else

{

comboBox1.Items.Add(textBox1.Text);

textBox1.Clear();

textBox1.Focus();

}

}

else

{

MessageBox.Show(" Please Enter Your course Name...");

textBox1.Focus();

}

}

}

else

{

e.Handled = true;

}

}

private void button2\_Click(object sender, EventArgs e)

{

if (comboBox1.SelectedIndex >= 0)

{

comboBox1.Items.RemoveAt(comboBox1.SelectedIndex);

MessageBox.Show("Your Item Remove Sucessfully...");

}

else

{

MessageBox.Show(" Please Select Your Deleted Course Name....");

}

}

private void button3\_Click(object sender, EventArgs e)

{

comboBox1.Items.Clear();

comboBox2.Items.Clear();

}

private void button4\_Click(object sender, EventArgs e)

{

Application.Exit();

}

private void button5\_Click(object sender, EventArgs e)

{

if (comboBox2.Items.Contains(comboBox1.SelectedItem.ToString()) == false)

{

comboBox2.Items.Add(comboBox1.SelectedItem.ToString());

}

else

{

MessageBox.Show(" This course already added.....");

}

}

}

}

**Program 7**

**1) Auto Complete Mode = SuggestAppend**

**2) Auto Complete Source = ListItems**

**Check box**

A CheckBox control allows users to select a single or multiple options from a list of options.

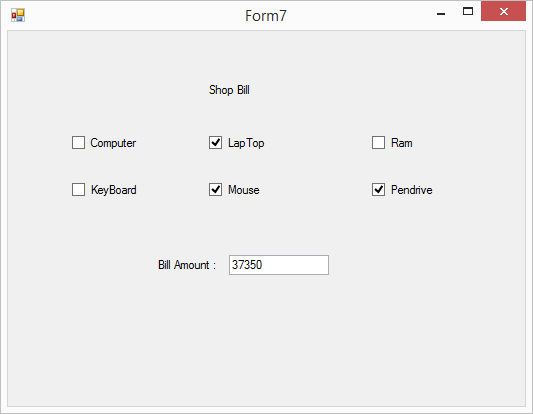
Constructors : checkBox

Properties : Checked

Methods: Dispose.Add

**Program 8**

**1) Display the number of products using check box and find bill amount when select the chekbox**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form7 : Form

{

// variable

double billamount;

public Form7()

{

InitializeComponent();

}

private void checkBox1\_CheckedChanged(object sender, EventArgs e)

{

if(checkBox1.Checked==true)

{

billamount = billamount + 25000;

}

else

{

billamount = billamount - 25000;

}

textBox1.Text = billamount.ToString();

}

private void checkBox2\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox2.Checked == true)

{

billamount = billamount + 35000;

}

else

{

billamount = billamount - 35000;

}

textBox1.Text = billamount.ToString();

}

private void checkBox3\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox3.Checked == true)

{

billamount = billamount + 2500;

}

else

{

billamount = billamount - 2500;

}

textBox1.Text = billamount.ToString();

}

private void checkBox4\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox4.Checked == true)

{

billamount = billamount + 250;

}

else

{

billamount = billamount - 250;

}

textBox1.Text = billamount.ToString();

}

private void checkBox5\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox5.Checked == true)

{

billamount = billamount + 350;

}

else

{

billamount = billamount - 350;

}

textBox1.Text = billamount.ToString();

}

private void checkBox6\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox6.Checked == true)

{

billamount = billamount + 2000;

}

else

{

billamount = billamount - 2000;

}

textBox1.Text = billamount.ToString();

}

}

}

**Checked List box**

The CheckedListBox control gives you all the capability of a list box and also allows you to display a check mark next to the items in the list box.

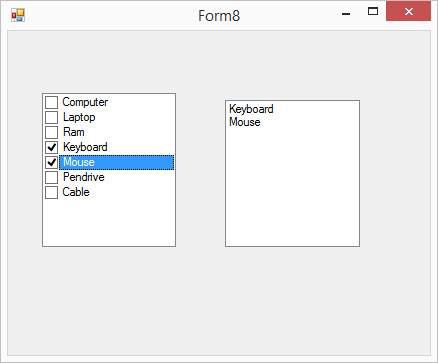
Constructors : checkedListBox

Properties : Checked,items..etc

Methods : Dispose,Add.

**Program 9**

1. **Take the Checked Listbox Select Multiple Options and display one listbox**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form8 : Form

{

public Form8()

{

InitializeComponent();

}

private void checkedListBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

//listBox1.Items.Clear();

//for (int i = 0; i < checkedListBox1.CheckedItems.Count; i++)

//{

// listBox1.Items.Add(checkedListBox1.CheckedItems[i].ToString());

//}

}

private void checkedListBox1\_MouseUp(object sender, MouseEventArgs e)

{

listBox1.Items.Clear();

for (int i = 0; i < checkedListBox1.CheckedItems.Count; i++)

{

listBox1.Items.Add(checkedListBox1.CheckedItems[i].ToString());

}

}

}

}

**Radio button**

Enables the user to select a single option from a group of choices when paired with other RadioButton controls

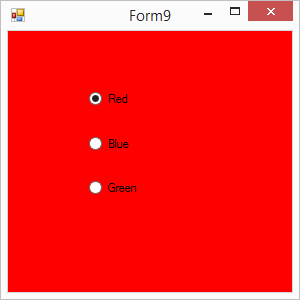
Constructors : RadioButton

Properties : Checked

Methods : Dispose

**Program 10**

**1) Change the backcolor using radio button**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Windows\_Day\_2

{

public partial class Form9 : Form

{

public Form9()

{

InitializeComponent();

}

private void radioButton1\_CheckedChanged(object sender, EventArgs e)

{

if(radioButton1.Checked==true)

{

this.BackColor = Color.Red;

}

}

private void radioButton2\_CheckedChanged(object sender, EventArgs e)

{

if(radioButton2.Checked==true)

{

this.BackColor = Color.Blue;

}

}

private void radioButton3\_CheckedChanged(object sender, EventArgs e)

{

if (radioButton3.Checked == true)

{

this.BackColor = Color.Green;

}

}

}

}