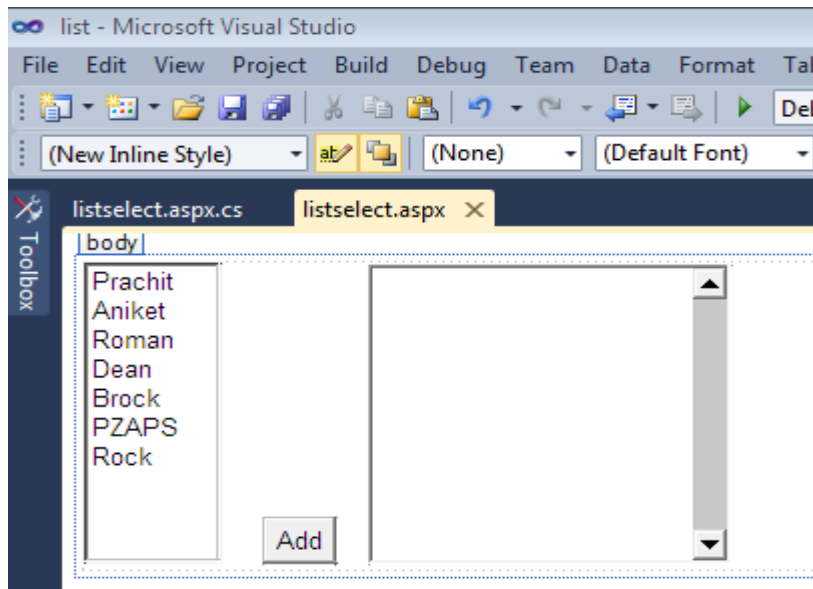


AIM: List of employees is available in listbox. Write an application to add selected or all records from listbox (assume multi-line property of textbox is true).

DESIGN:

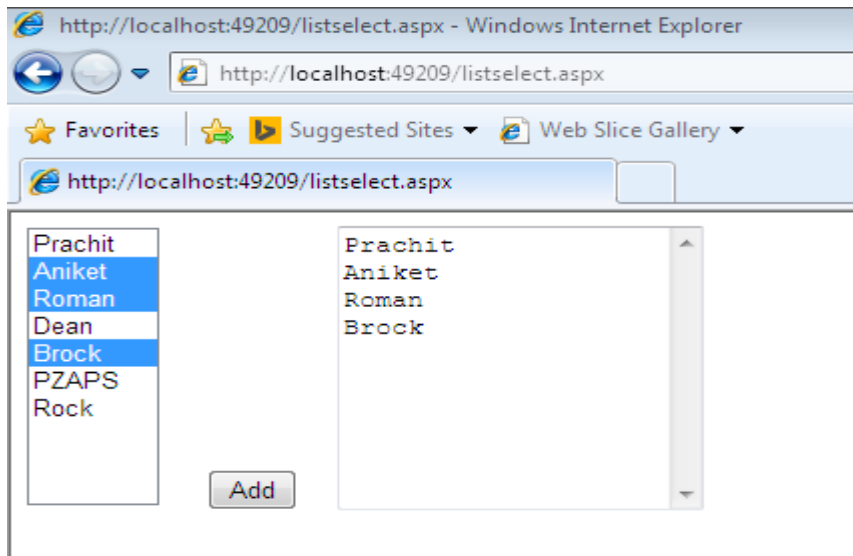


PROPERTIES TABLE:

CODE:

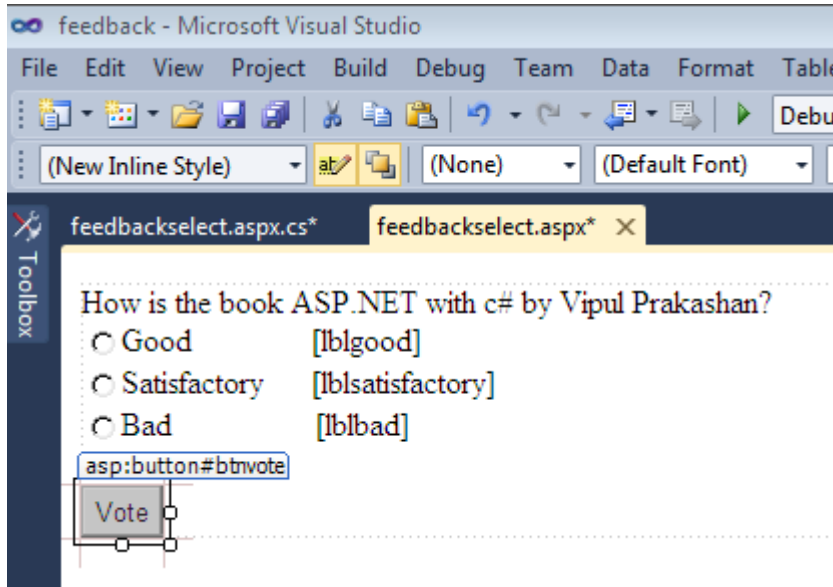
```
using System;
namespace list
{
    public partial class listselect : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }
        protected void btnAdd_Click(object sender, EventArgs e)
        {
            int i;
            for (i = 0; i < lstEmployee.Items.Count; i++)
            {
                if (lstEmployee.Items[i].Selected == true)
                    txtEmployee.Text += lstEmployee.Items[i].Text + "\n";
            }
        }
    }
}
```

BROWSER OUTPUT:



AIM: “How is the book ASP.NET with c# by Vipul Prakashan?” Give the user three choice :
i)Good ii)Satisfactory iii)Bad. Provide a VOTE button. After user votes, present the result in percentage using labels next to the choices.

DESIGN:



PROPERTIES TABLE:

Control	Property	Value
Label1	ID	lbltxt1
	Text	How is the Book ASP.NET with c# Vipul Prakashan
RadioButton1	ID	rdogood
	Text	Good
RadioButton2	ID	rdosatisfactory
	Text	Satisfactory
RadioButton3	ID	rdobad
	Text	Bad
Label2	ID	lblgood
	Text	
Label3	ID	blsatisfactory
	Text	
Label4	ID	lblbad
	Text	
Button	ID	btnvote
	Text	Vote

CODE:

```
using System;
namespace feedback
{
    public partial class feedbackselect : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }
        protected void btnvote_Click(object sender, EventArgs e)
        {
            if (rdogood.Checked == true)
            {
                int goodCount;
                if (ViewState["gcount"] != null)
                    goodCount = Convert.ToInt32(ViewState["gcount"]) + 1;
                else
                    goodCount = 1;
                ViewState["gcount"] = goodCount;
            }

            if (rdosatisfactory.Checked == true)
            {
                int satisfactoryCount;
                if (ViewState["scount"] != null)
                    satisfactoryCount = Convert.ToInt32(ViewState["scount"]) + 1;
                else
                    satisfactoryCount = 1;
                ViewState["scount"] = satisfactoryCount;
            }
            if (rdobad.Checked == true)
            {
                int badCount;
                if (ViewState["bcount"] != null)
                    badCount = Convert.ToInt32(ViewState["bcount"]) + 1;
                else
                    badCount = 1;
                ViewState["bcount"] = badCount;
            }
            int totalCount;
            if (ViewState["count"] != null)
                totalCount = Convert.ToInt32(ViewState["count"]) + 1;
            else
                totalCount = 1;
            ViewState["count"] = totalCount;
            double gper = (Convert.ToDouble(ViewState["gcount"]) /
                Convert.ToDouble(ViewState["count"])) * 100.0f;
```

ASP.NET WITH C#

```
lblgood.Text = gper.ToString() + "%";  
    double sper = (Convert.ToDouble(ViewState["scount"]) /  
Convert.ToDouble(ViewState["count"])) * 100.0f;  
    lblsatisfactory.Text = sper.ToString() + "%";  
    double bper = (Convert.ToDouble(ViewState["bcount"]) /  
Convert.ToDouble(ViewState["count"])) * 100.0f;  
    lblbad.Text = bper.ToString()+"%";  
  
    } } }
```

BROWSER OUTPUT:

