

LAB 3

Spring 2011, BESE- 15

Interface building using asp.net

Objective

This is the second lab activity on designing a user interface using ASP.net. In the previous lab we had seen how to use 'Labels', 'Text boxes' and 'Buttons' to an interface. Today we will focus on:

- Dropdown menus
- Picture Box
- File upload control
- Check Boxes
- Radio Buttons
- List Box
- Calendar
- Multi view control

Submission Requirements

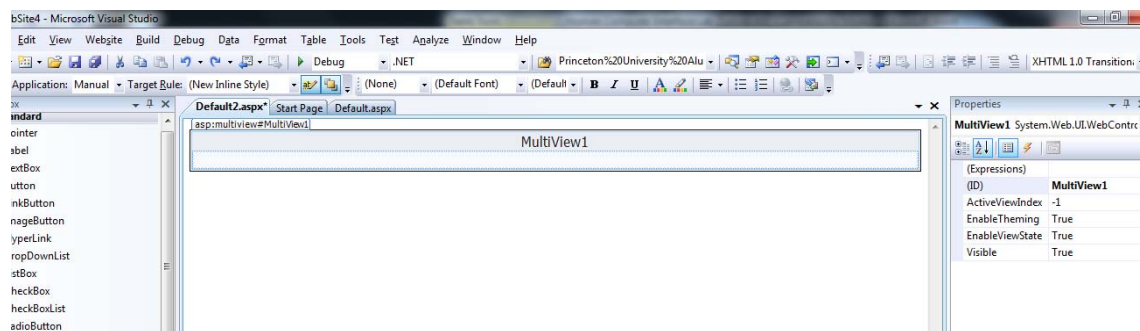
Task 1 comprises practice exercises and you do NOT need to submit any lab report/program code. Task 2 is a home assignment. Along with this you also need to submit Assignment 1 given as such in today's lab folder.

Deadline for submission is: Wednesday 2nd March, 2011.

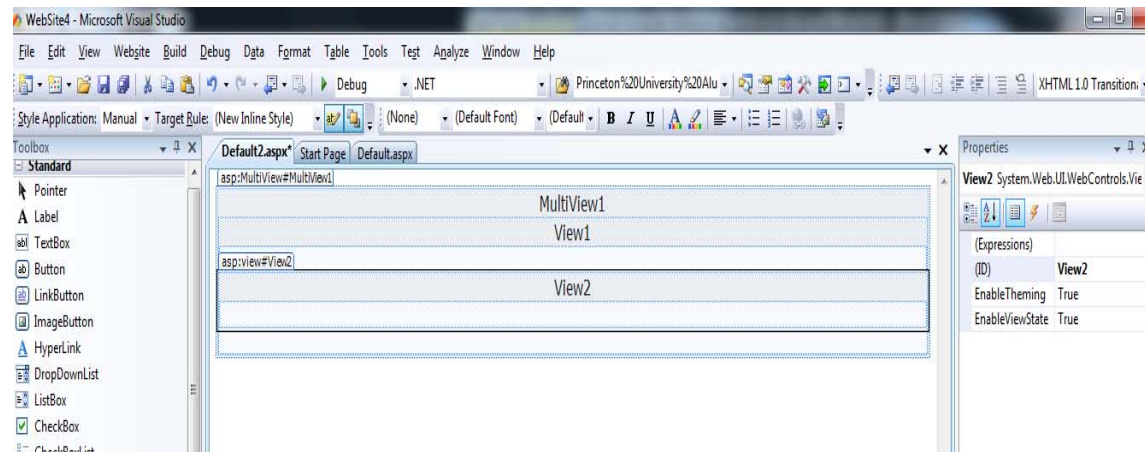
Task 1:

Task 1 is to create a simple Student Registration page that would take from user his/her first and last name, date of birth, Id, password, gender and interest details.

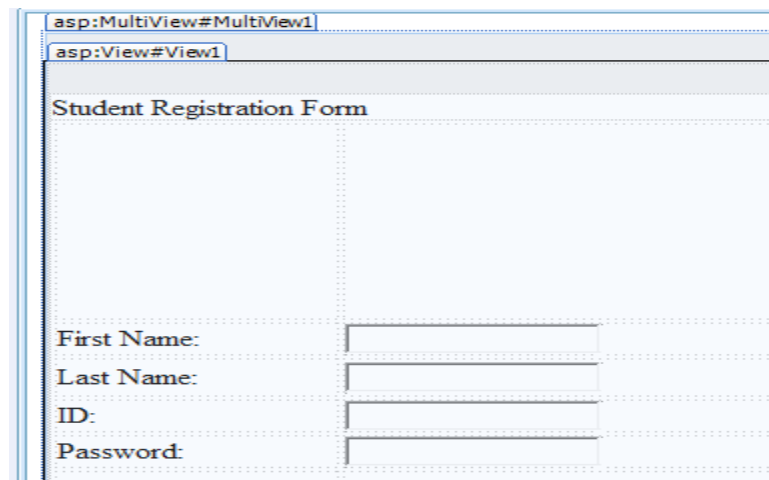
1. Start visual studio. Create a new website with C# as the programming language.
2. Add a multiview control to your (web) page. The MultiView provides multiple views on one page, with one view being displayed at a time. This provides a great advantage when trying to develop a multi-step process, as you are not required to set the visibility of each individual view for each step. Instead, you can just set the currently active view. All controls within each view will be accessible in the code-behind.



3. Add two View controls to the MultiView control .A View control works as a container of common controls and a MutliView control works as a container of several View controls.

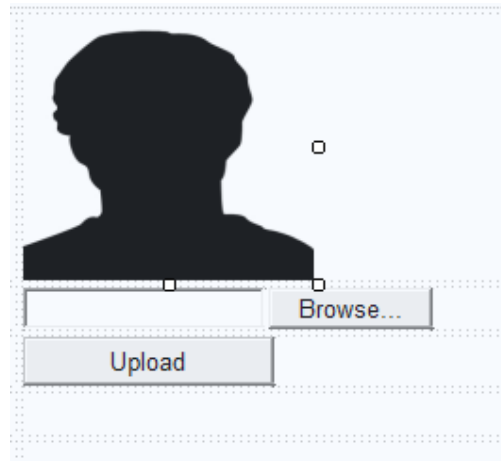


4. On view 1 add an html table (only for easier layout design of page). Add labels and textboxes to the table for
 - a. First name
 - b. Last name
 - c. ID
 - d. Password
 - e.



5. Add an image control to the page.
6. With the image control add a FileUpload control. A file upload control browses the user's computer to upload a file to the server. It has the attribute FileName that is used to get the name and extension of chosen file. The path of file on client computer is not saved. The file can be saved on any directory on the server using the SaveAs function of the FileUpload control.

7. Add a button with these and name it Upload.

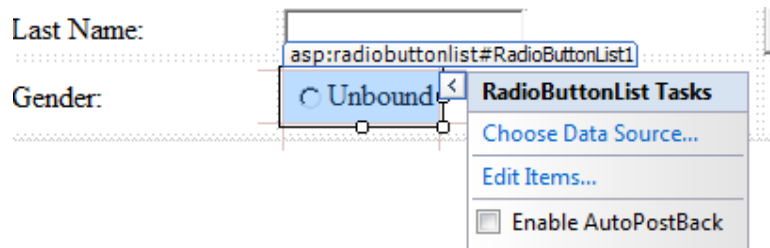


8. The code behind of the button Upload is to save the file chosen by the user to server. The path to this file will be passed to the image control so that the file can be displayed.

```
protected void Button1_Click(object sender, EventArgs e)
{
    string filename1 = FileUpload1.FileName.ToString();
    FileUpload1.SaveAs(@"C:\temp\" + filename1);
    // string SaveLocation = Server.MapPath("Data") + "\\\";

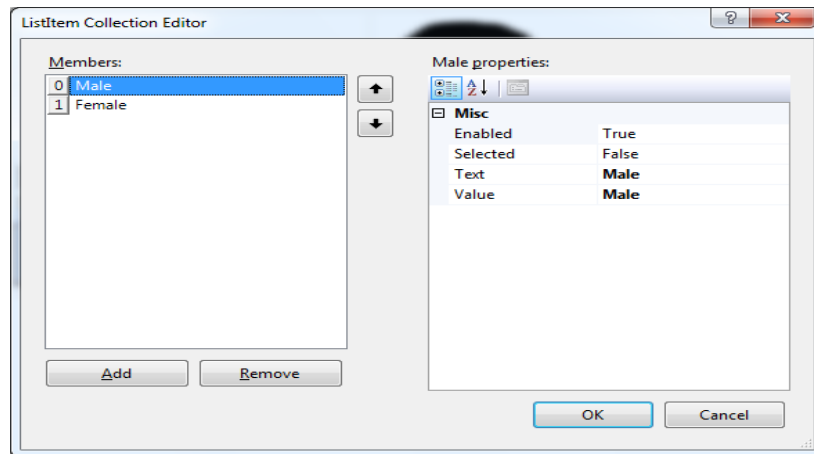
    Image1.ImageUrl = "~/\" + filename1 + "";
}
```

9. Next we require gender information from user. To get user choice we will use a radio button list. Radio buttons allow one of the available options to be selected. Add a radio list control to the page.



10. A small arrow on the top right corner of the control opens up a list of options that can be used with the control. Choose data source is for selecting values from a database and edit items is for entering values without use of database. Click the Edit Items option.

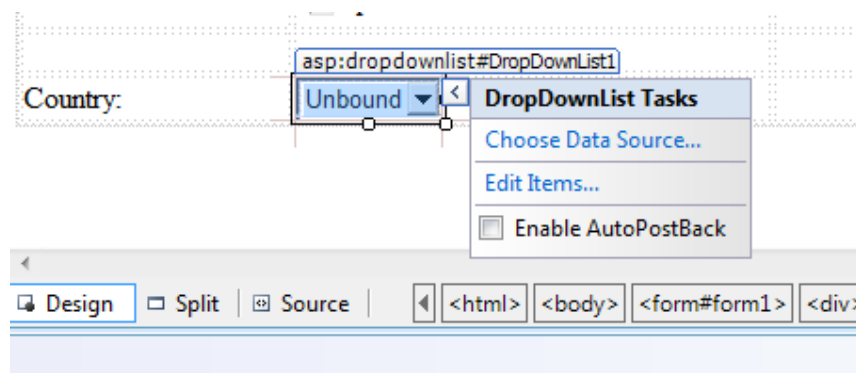
When the next window appears, select the Add button to add items to the list.



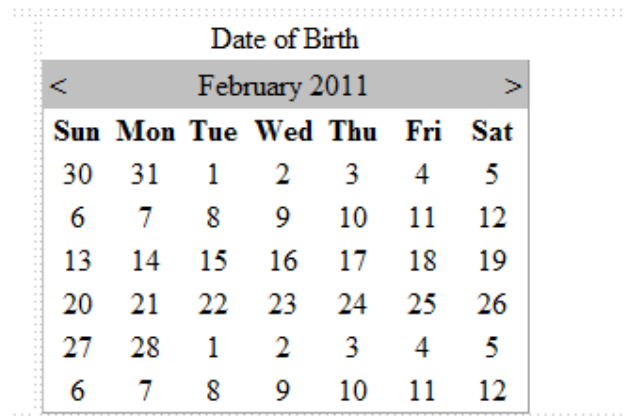
11. Next we require the user to provide information about his/her interests. Since there can be more than one interests so we need to allow the user multiple selection. For multiple selections add a check box list to the page. Add items to it as you did for radio button list.

Gender:	<input type="radio"/> Male <input type="radio"/> Female
Interests:	<input type="checkbox"/> Book Reading <input type="checkbox"/> Web Surfing <input type="checkbox"/> Sports

12. To get user's country information, we will use a dropdown list. The dropdown list is used in all the cases where the list of values is considerably long. It also allows single selection to the user. Add a label for country and a dropdown list for values. Add values to the dropdown list just as you did for radio button list.



13. Next we add a calendar control to the page to get user's date of birth.



14. In the end add a button to the page and name it 'Summary'. We want this control to redirect us to the second view where all the information we just entered is visible and we can thus submit it. For that add the following code to the button.

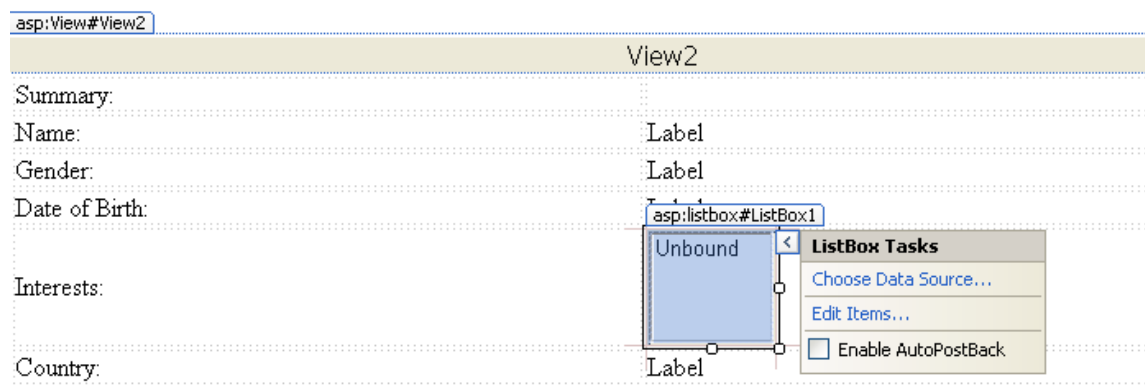
```
protected void Button2_Click(object sender, EventArgs e)
{
    MultiView1.ActiveViewIndex = 1;
}
```

In the code:

The line 1 selects the view that would be visible to user. We have added two views to the page and want one of them to be visible at any particular time. By default the multiview control makes the first view visible, the one with index 0.

So in this line we are switching to second view by making it active.

15. In the second view add appropriate number of labels to display the data entered by the user in the previous section. Since the user can enter more than one 'Interests', we will use a listbox control to show these multiple choices. Leave the listBox unbound and with no items.



16. We need the values of these labels to be updated as soon as the user navigates to this view. The user can get to this view by clicking the summary button on view 1 so the code for changing these labels also has to be added to that control.

```
protected void Button2_Click(object sender, EventArgs e)
{
    MultiView1.ActiveViewIndex = 1;

    Label5.Text = TextBox1.Text + TextBox2.Text;
    Label6.Text = RadioButtonList1.SelectedValue;
    Label7.Text = Calendar1.SelectedDate.ToString();

    Label9.Text = DropDownList1.SelectedValue;
}
```

17. To add the list of selected choices as 'Interests' we need to loop through all of the checkBoxList items and add the selected ones to our ListBox.

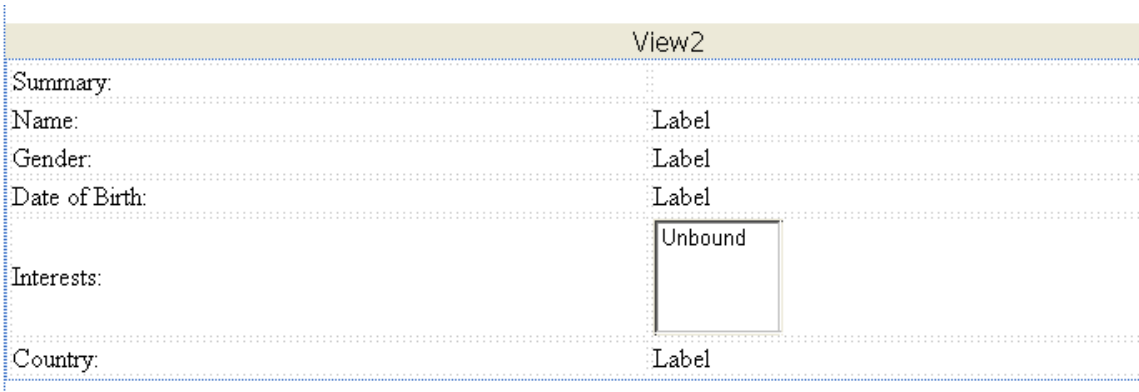
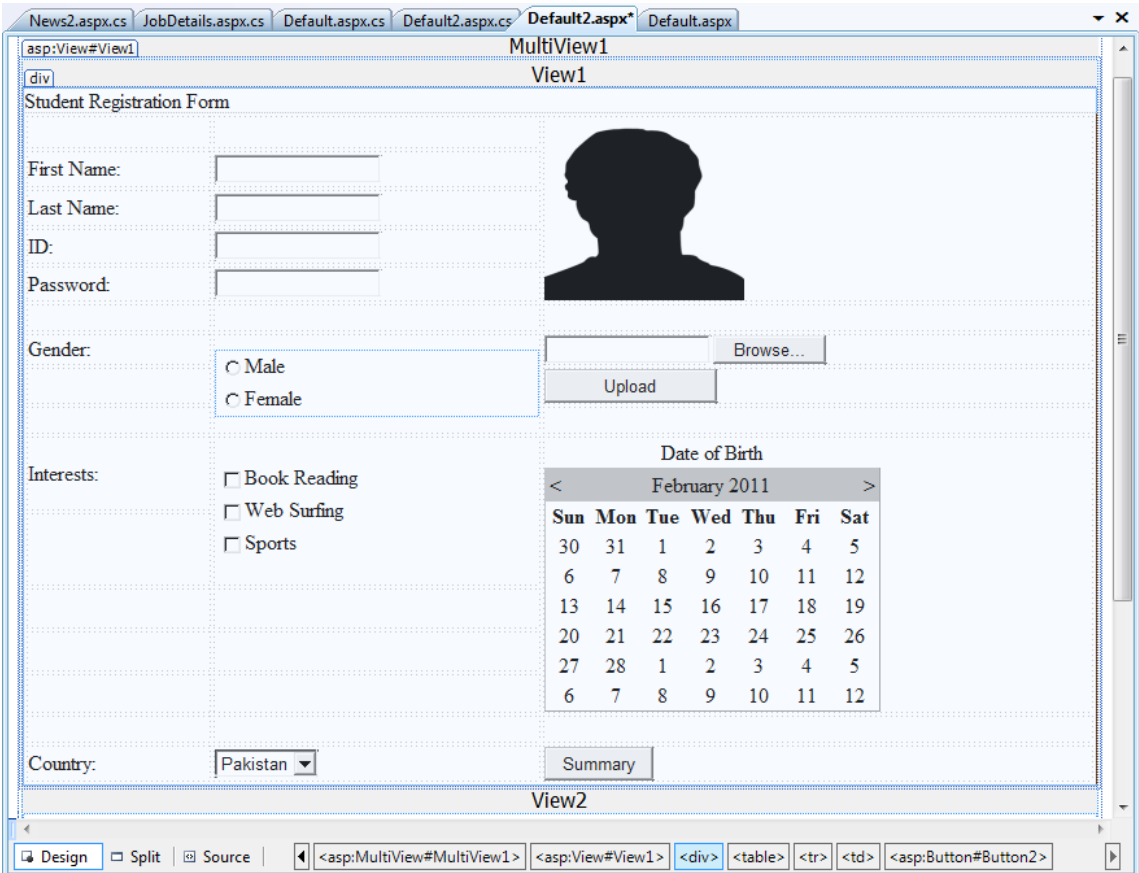
```
for (int i = 0; i < CheckBoxList1.Items.Count; i++)
{
    if (CheckBoxList1.Items[i].Selected == true)
    {
        ListBox1.Items.Add( CheckBoxList1.Items[i].Text);
    }
}
```

18. Add a button to the second view named 'Submit'. This control will be empty. Add another button and name it 'Previous'. This is to allow the user to go back to the previous view. To do that we need to change the active view of the MultiView control. Add the following code to this button:

```
protected void Button3_Click(object sender, EventArgs e)
{
    MultiView1.ActiveViewIndex = 0;
}
```

{

The final application **may** look like this:



Task 2:

You are provided with a link to a computer manufacturer's website. They allow users to customize their orders by choosing individual components of the computer system they wish to order.

At any particular moment a specific class of components is visible to the user. Recreate a webpage similar to this one which provides the user with facility of browsing through multiple views.

Note: Functionality such as dropdown menu does not need to be included in your page at this time. Your webpage should however make use of controls covered so far.

Link:

http://configure.us.dell.com/dellstore/config.aspx?oc=fndon08b&c=us&l=en&s=dhs&cs=19&model_id=inspiron-17r-combo-mod