ASP.NET Controls: CheckBox, RadioButton, ListBox, Textbox, Label

In this tutorial, you will learn-

Adding ASP.Net Controls to Web Forms
Label Control
Textbox
List box
RadioButton
Checkbox
Button
Event Handler in ASP.Net

Adding ASP.Net Controls to Web Forms

ASP.Net has the ability to add controls to a form such as textboxes and labels.

Let's look at the other controls available for Web forms and see some of their common properties.

In our example, we will create one form which will have the following functionality.

1. The ability for the user to enter his name.
2. An option to choose the city in which the user resides in
3. The ability for the user to enter an option for the gender.
4. An option to choose a course which the user wants to learn. There will be choices for both C# (/c-tutorial.html) and ASP.Net

Let's look at each control in detail. Let's add them to build the form with the above-mentioned functionality.

Step 1) The first step is to open the Forms Designer for the Demo web form. Once you do this, you will be able to drag controls from the toolbox to the Web form.
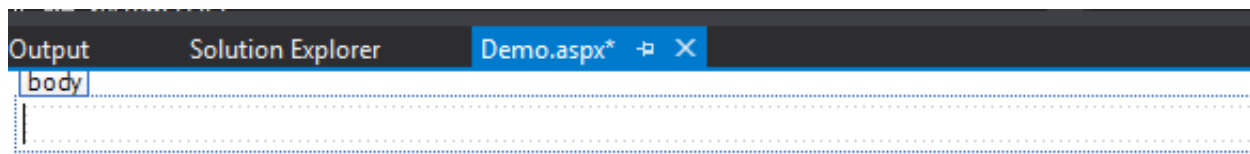
To open the Designer web form,

- Right-click the Demo.aspx file in the Solution Explorer and
- Choose the menu option View Designer.
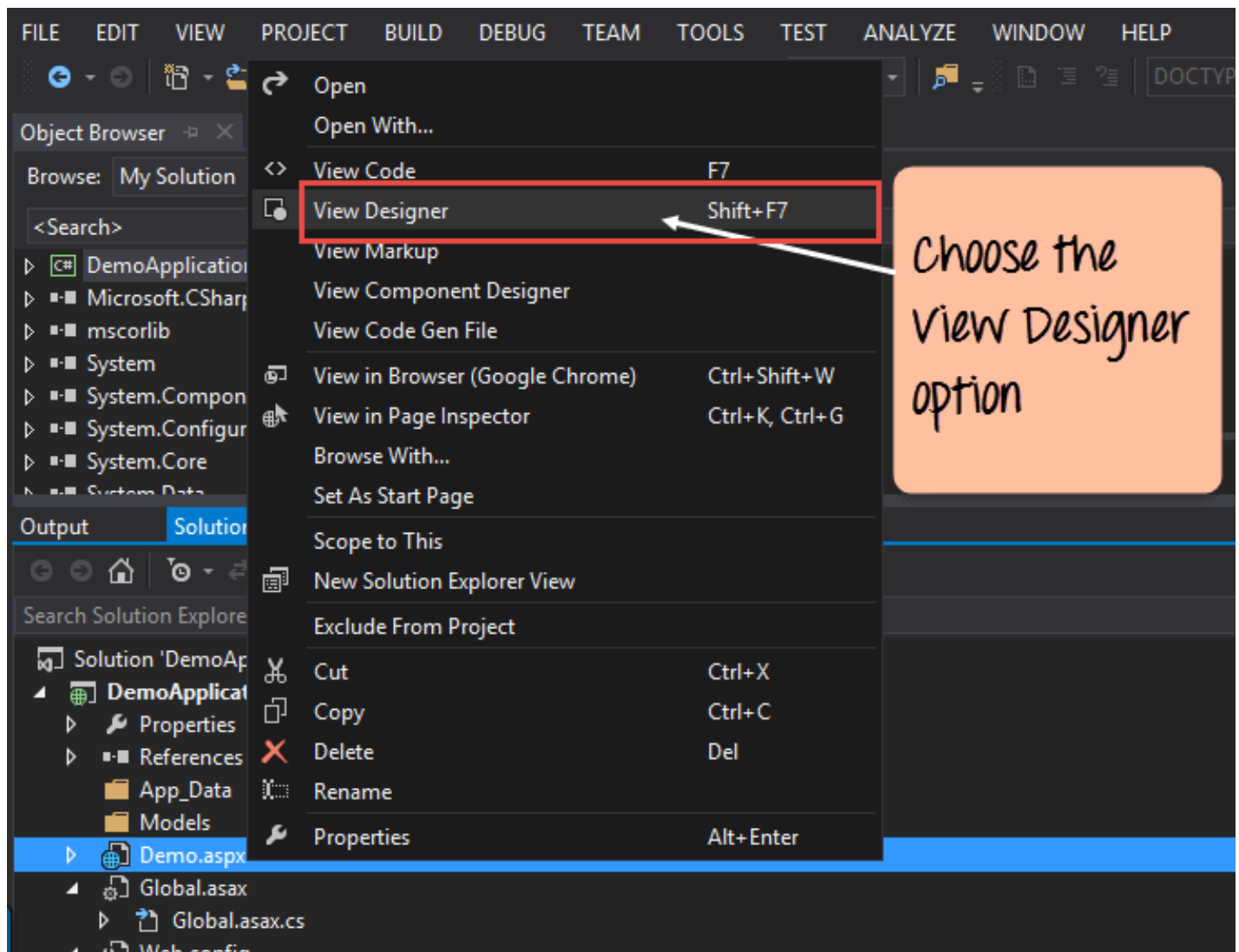
(/images/asp-net/061516_0807_ASPNetIntro12.png)

Once you perform the above step, you will be able to see your Form Designer as shown below.

(/images/asp-net/061516_0807_ASPNetIntro13.png)

Now let's start adding our controls one by one

Label Control

The label control is used to display a text or a message to the user on the form. The label control is normally used along with other controls. Common examples is wherein a label is added along with the textbox control. The label gives an indication to the user on what is expected to fill up in the textbox. Let's see how we can implement this with an example shown below. We will use a label called 'name.' This will be used in conjunction with the textbox controls, which will be added in the later section.
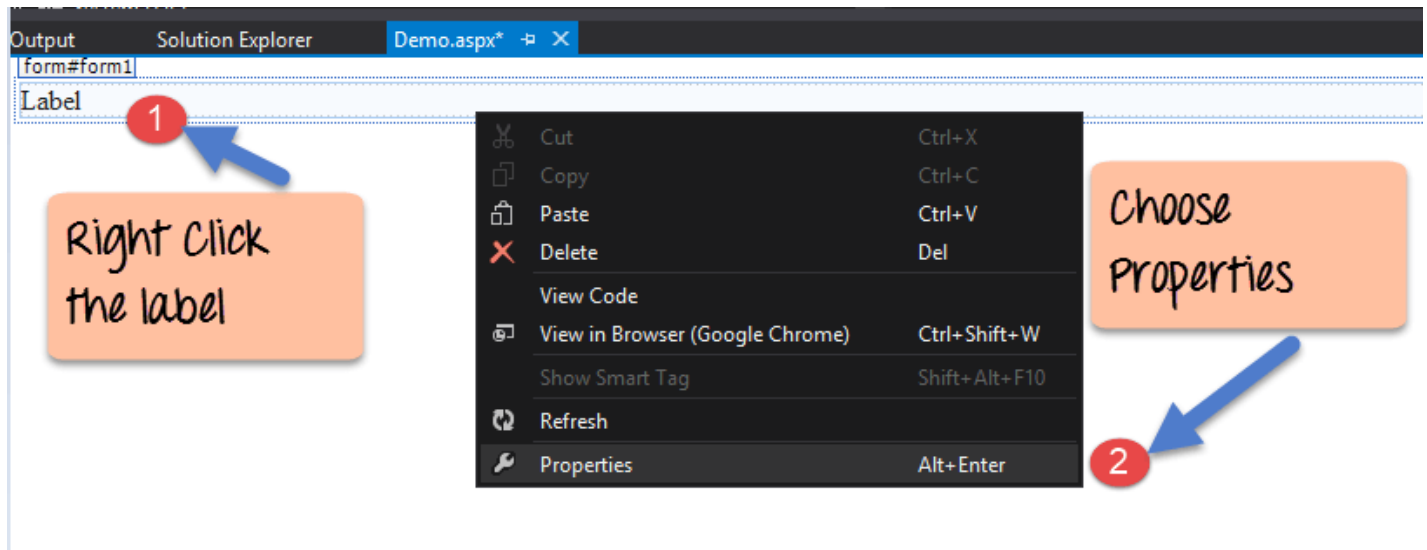
Step 1) The first step is to drag the 'label' control on to the Web Form from the toolbox as shown below.
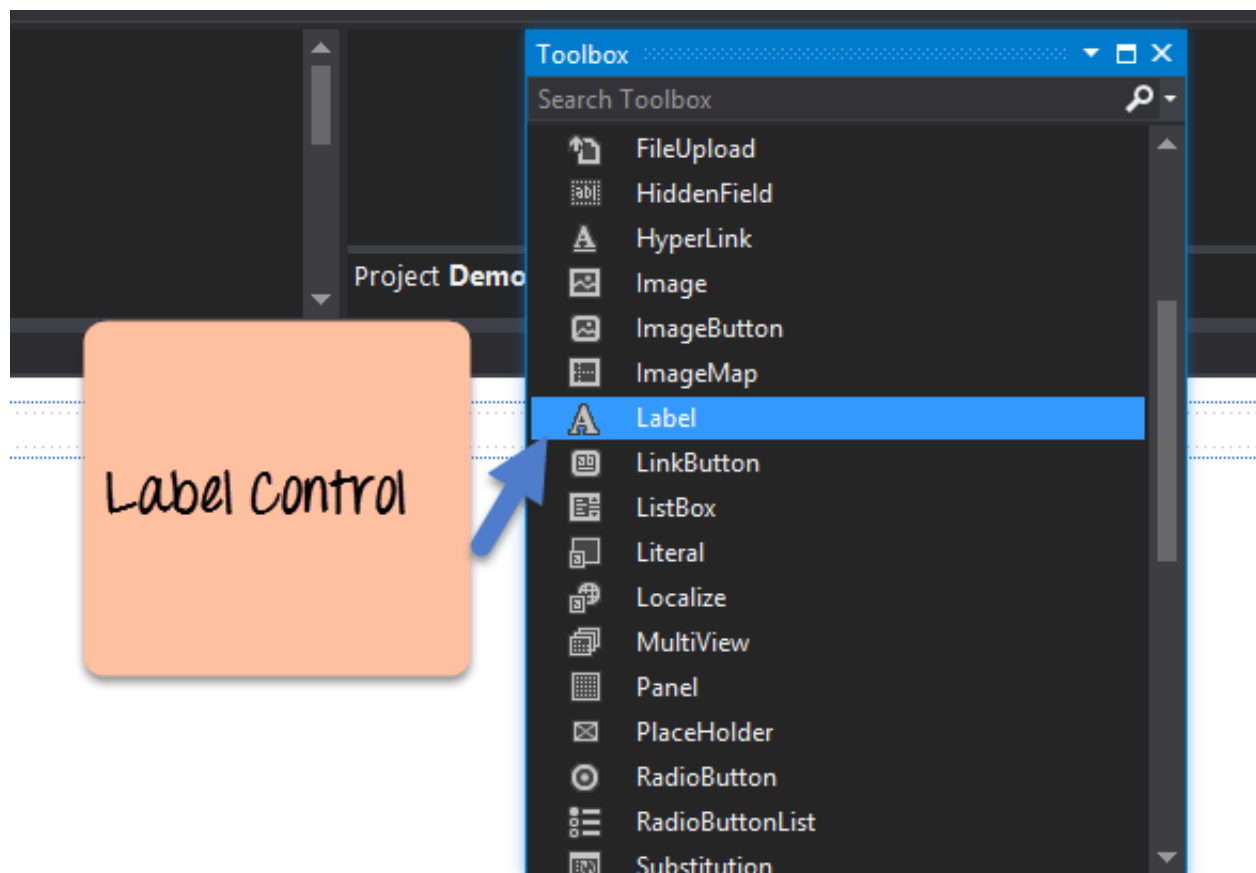
Label Control
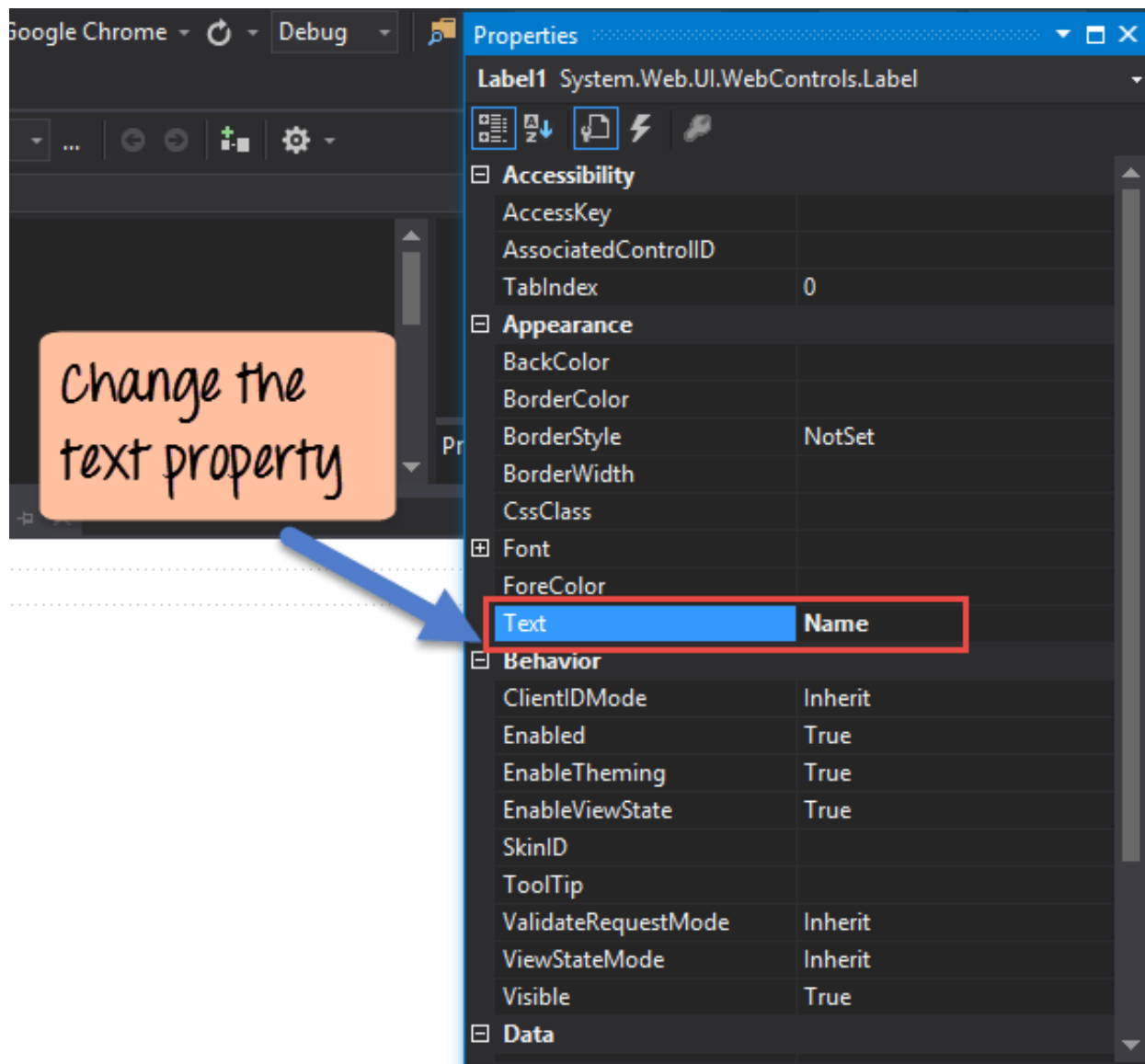
(/images/asp-net/061516_0807_ASPNetIntro14.png)

Step 2) Once the label has been added, follow the following steps.

1. Go to the properties window by right-clicking on the label control
2. Choose the Properties menu option

(/images/asp-net/061516_0807_ASPNetIntro15.png)

Step 3) From the properties window, change the name of the Text property to Name

[(/images/asp-net/061516_0807_ASPNetIntro16.png)](/images/asp-net/061516_0807_ASPNetIntro16.png)

Similarly, also change the ID property value of the control to lblName. By specifying a meaningful ID to controls, it becomes easier to access them during the coding phase. This is shown below.
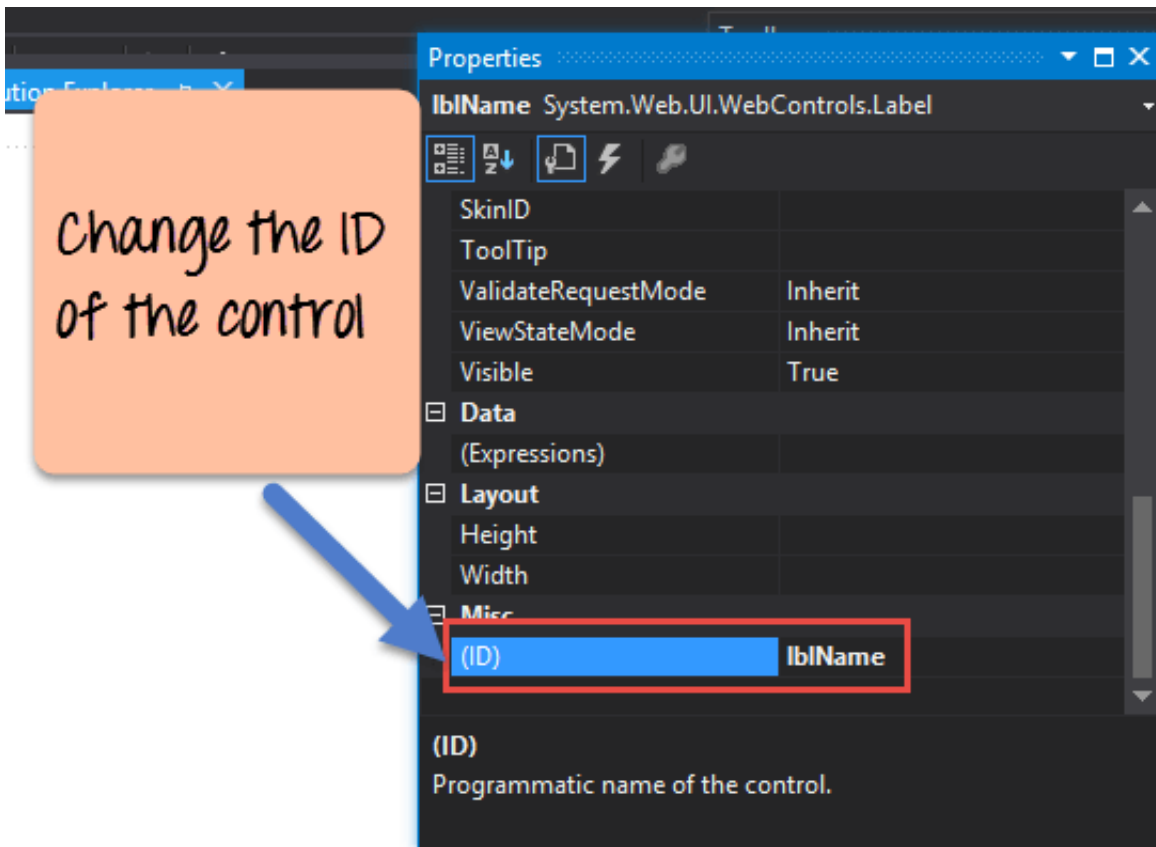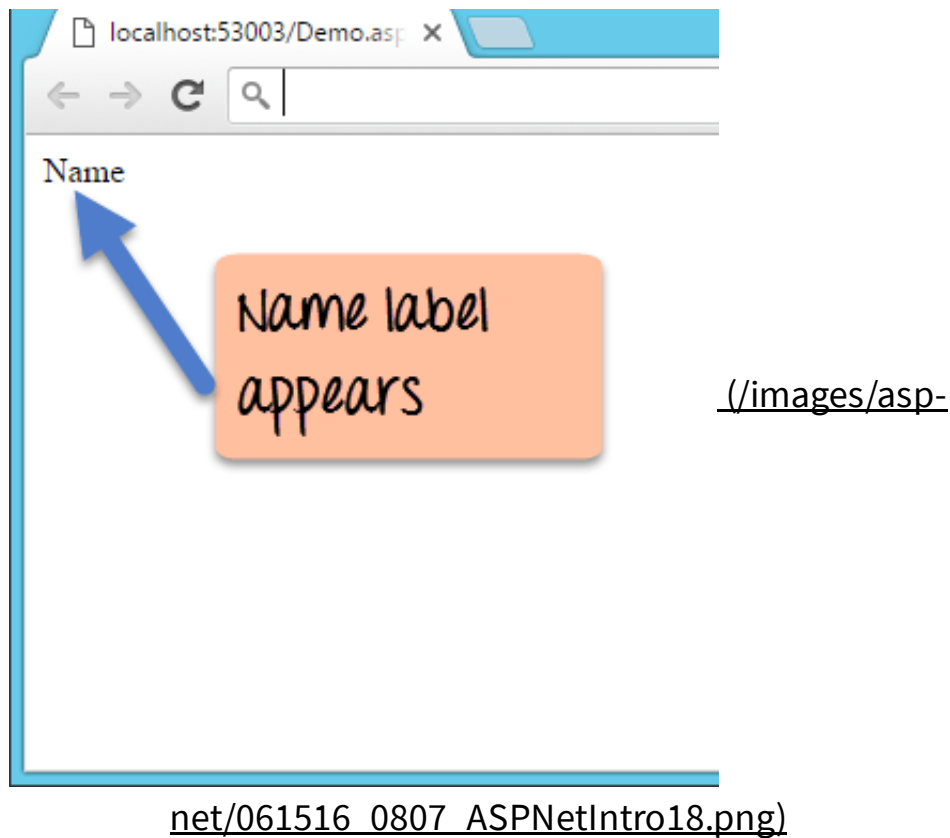
(/images/asp-net/061516_0807_ASPNetIntro17.png)

Once you make the above changes, you will see the following output

Output:-

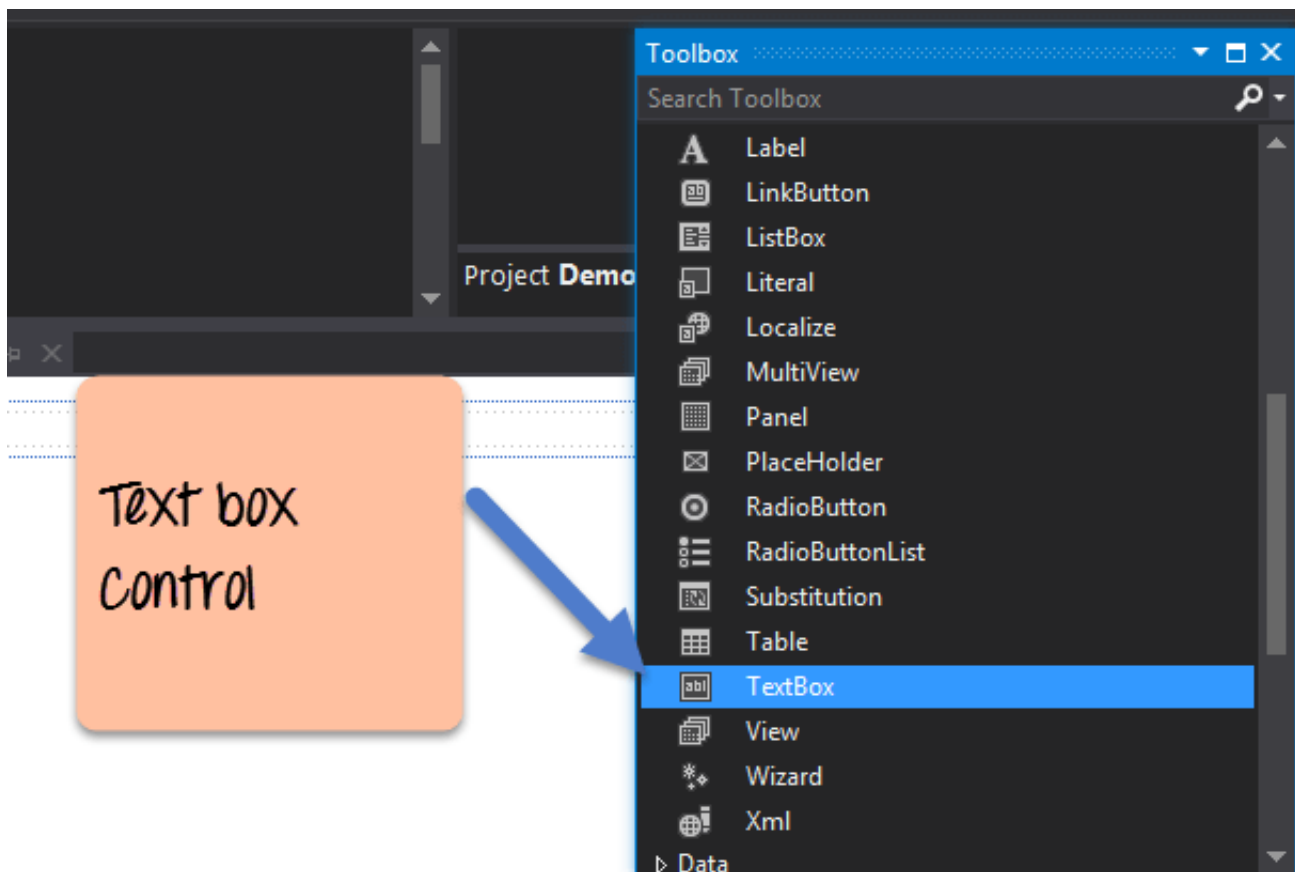(/images/asp-net/061516_0807_ASPNetIntro18.png)
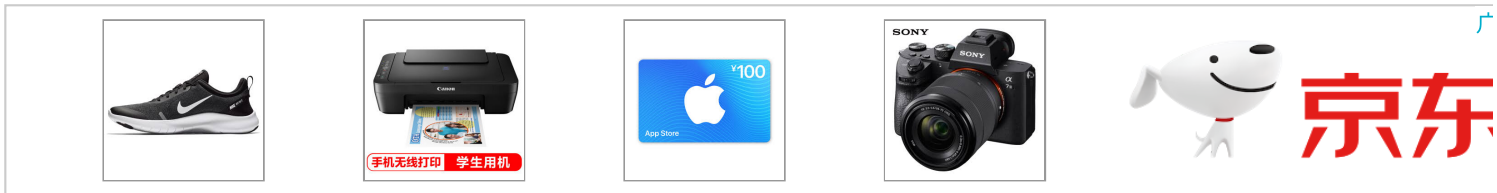
You will see that the Name label appears on the Web Form.
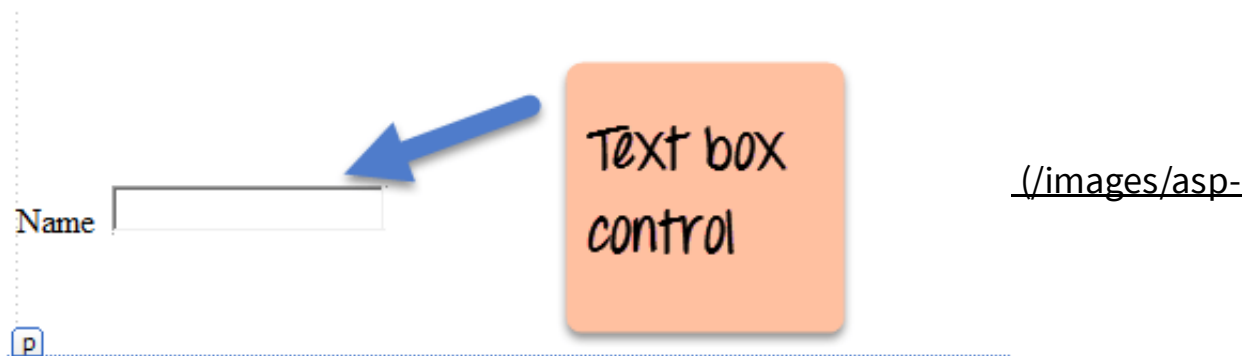
Textbox

A text box is used for allowing a user to enter some text on the Web form application. Let's see how we can implement this with an example shown below. We will add one textbox to the form in which the user can enter his name.

Step 1) The first step is to drag the textbox control onto the Web Form from the toolbox as shown below

(/images/asp-net/061516_0807_ASPNetIntro19.png)
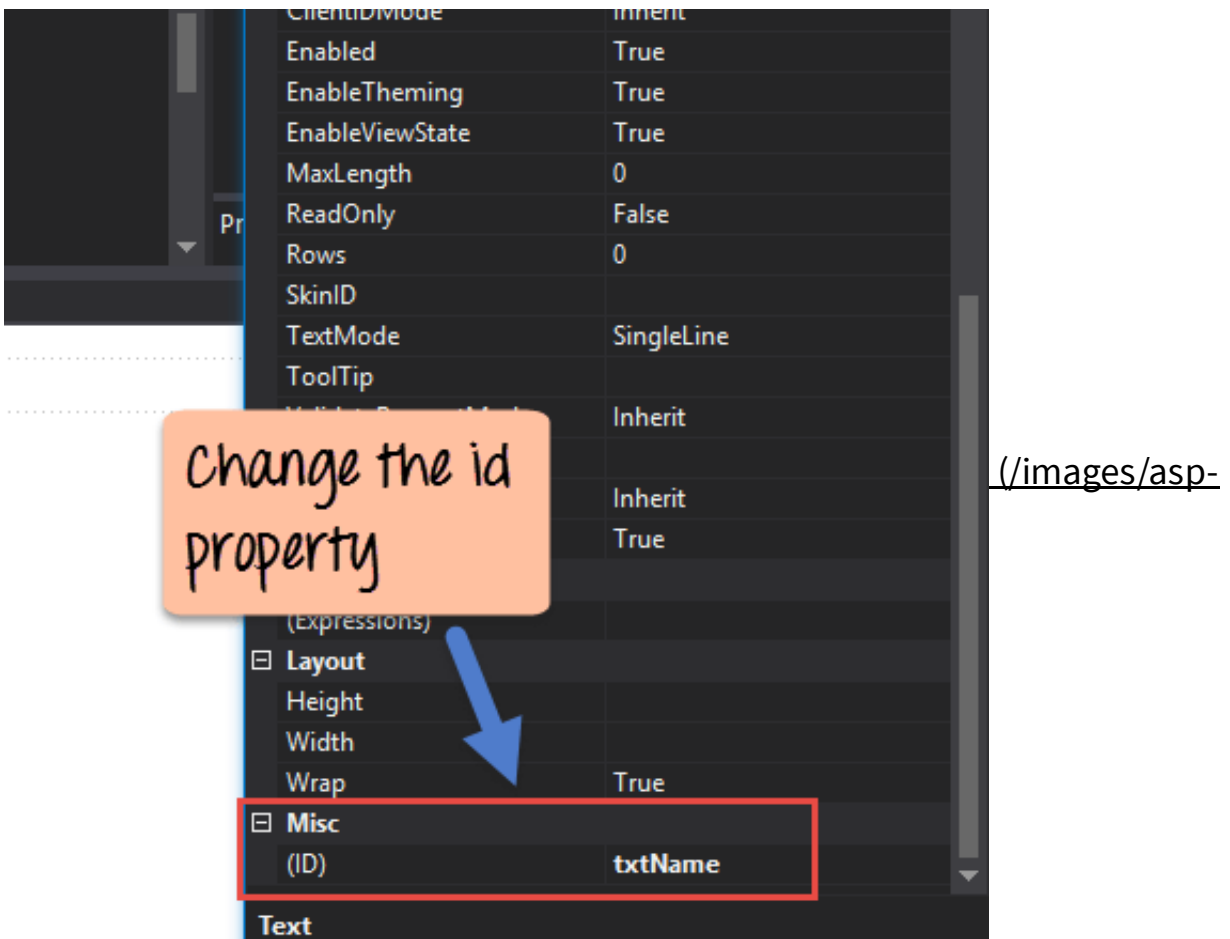
Below is how this would look in the forms designer once the Textbox control is on the form

(/images/asp-net/061516_0807_ASPNetIntro20.png)

Step 2) Once the Textbox has been added, you have to change the ID property.

- Go to the properties window by right-clicking on the Textbox control and
- Choose properties then
- Change the id property of the textbox to txtName.

[(/images/asp-net/061516_0807_ASPNetIntro21.png)](/images/asp-net/061516_0807_ASPNetIntro21.png)

Once you make the above changes, you see the following output.

Output:-

(/images/asp-
net/061516_0807_ASPNetIntro22.png)

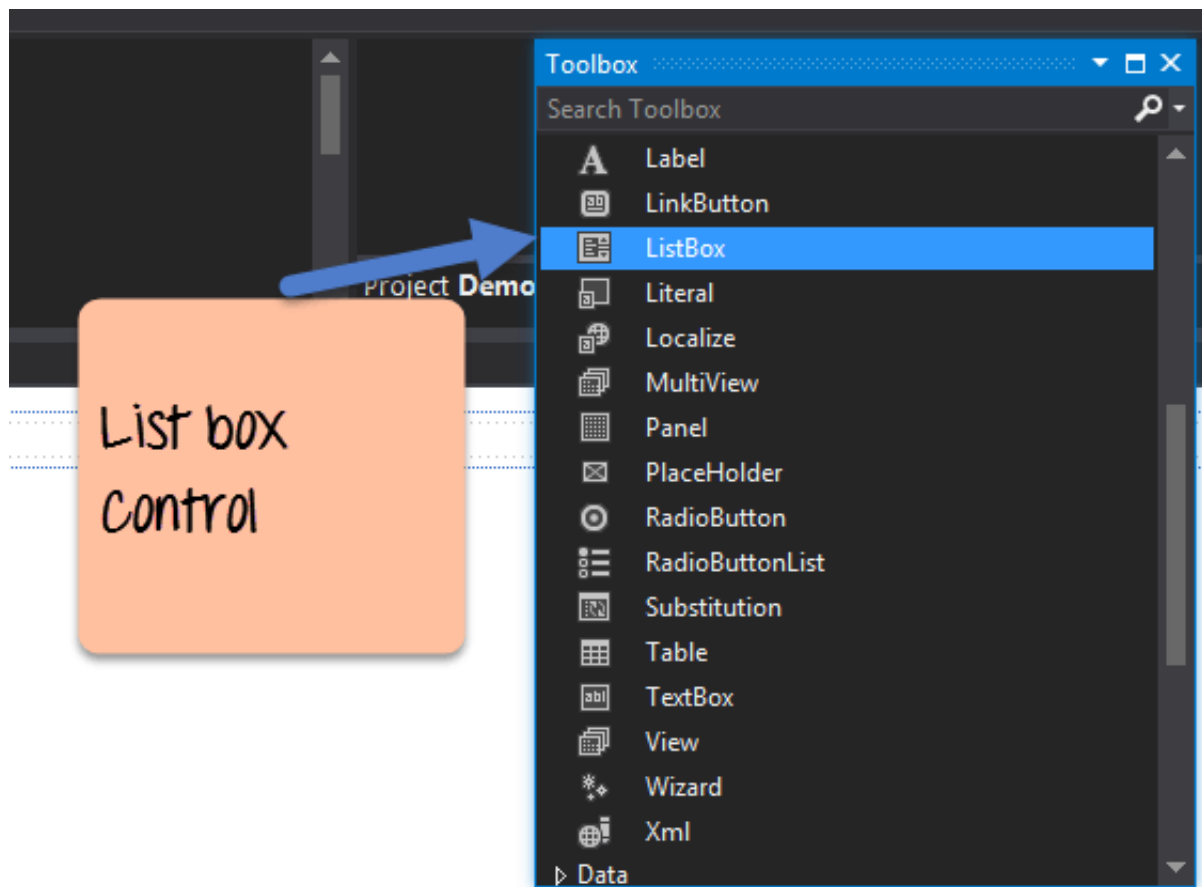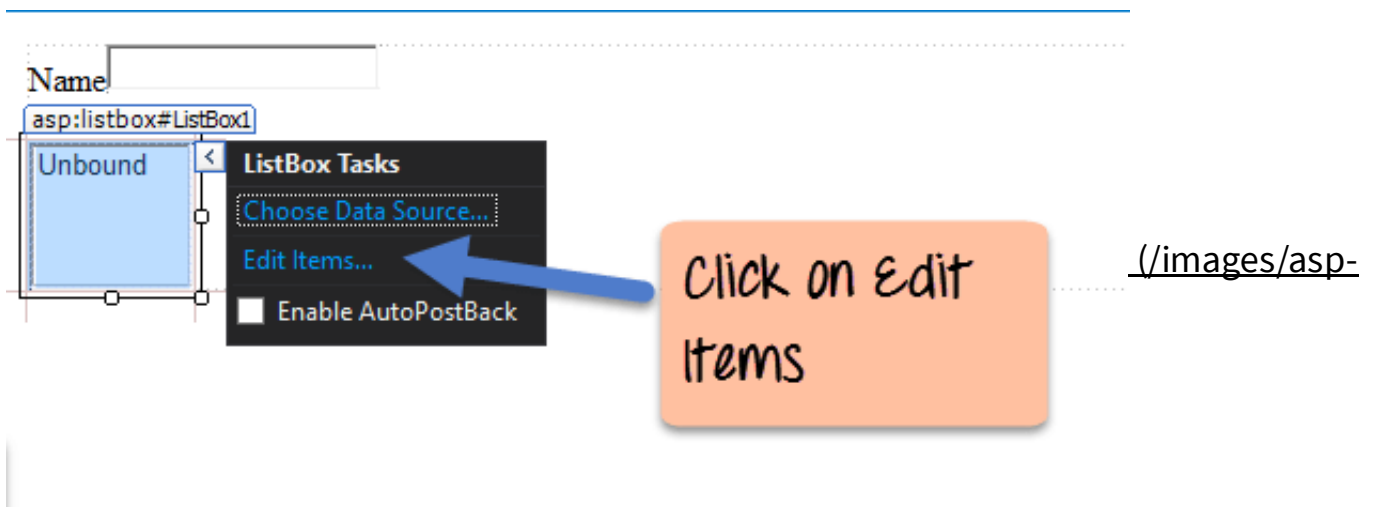# List box

A Listbox is used to showcase a list of items on the Web form. Let's see how we can implement this with an example shown below. We will add a list box to the form to store some city locations.

Step 1) The first step is to drag the list box control on to the Web Form from the toolbox as shown below

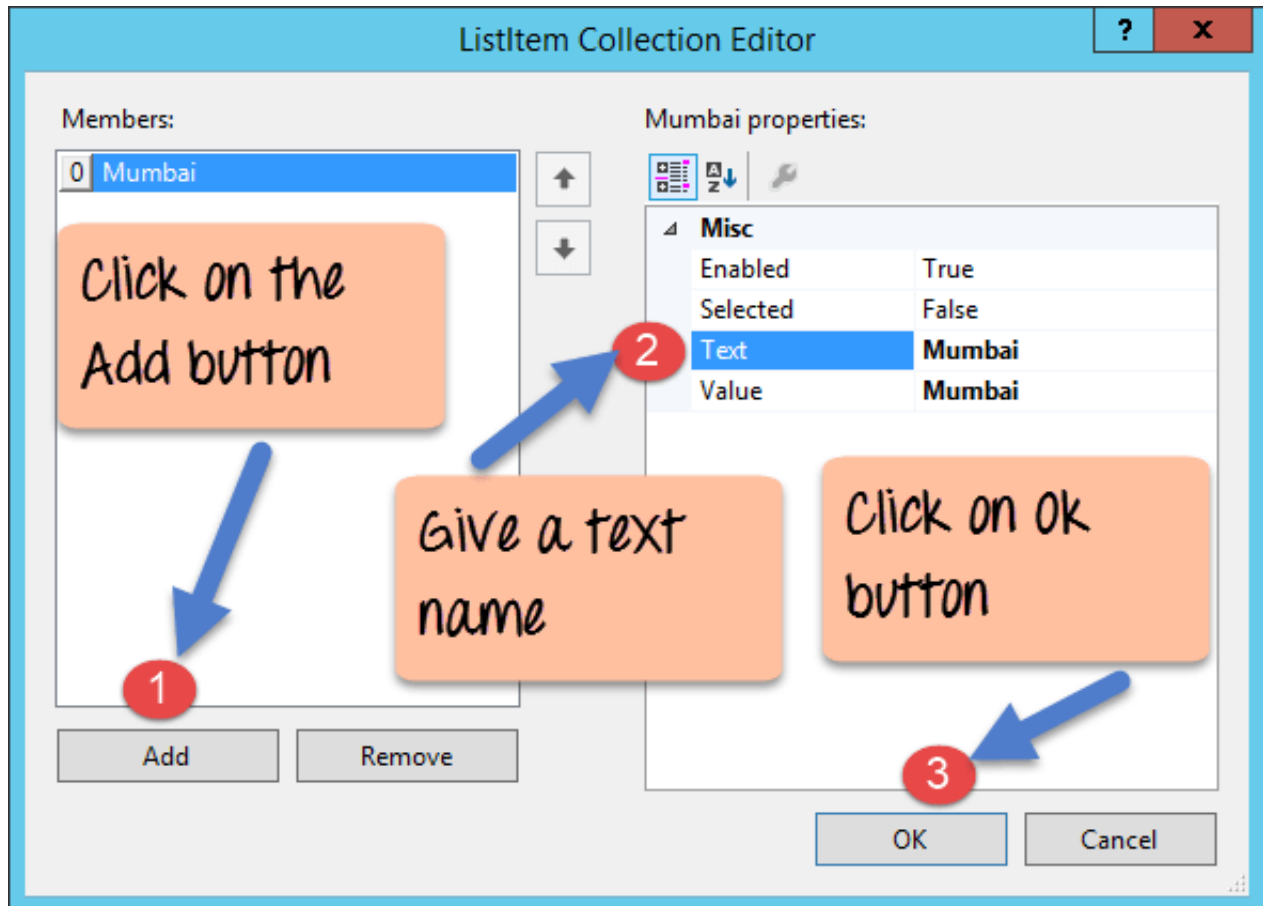(/images/asp-net/061516_0807_ASPNetIntro23.png)

Step 2) Once you drag the listbox to the form, a separate side menu will appear. In this menu choose the 'Edit Items' menu.

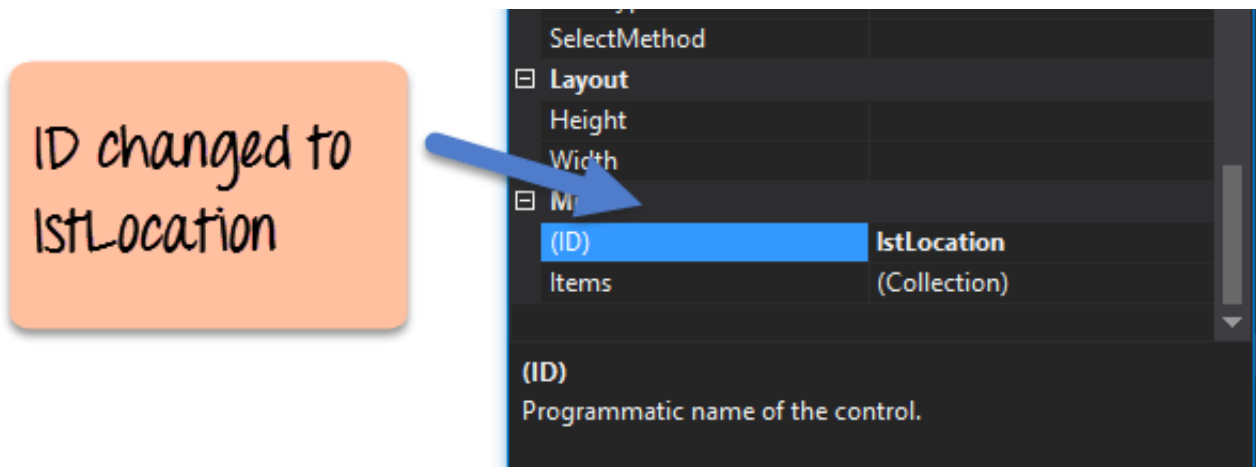(/images/asp-net/061516_0807_ASPNetIntro24.png)

Step 3) You will now be presented with a dialog box in which you can add the list items to the listbox.

1. Click on the Add button to add a list item.
2. Give a name for the text value of the list item – In our case Mumbai. Repeat steps 1 and 2 to add list items for Mangalore and Hyderabad.
3. Click on the OK button

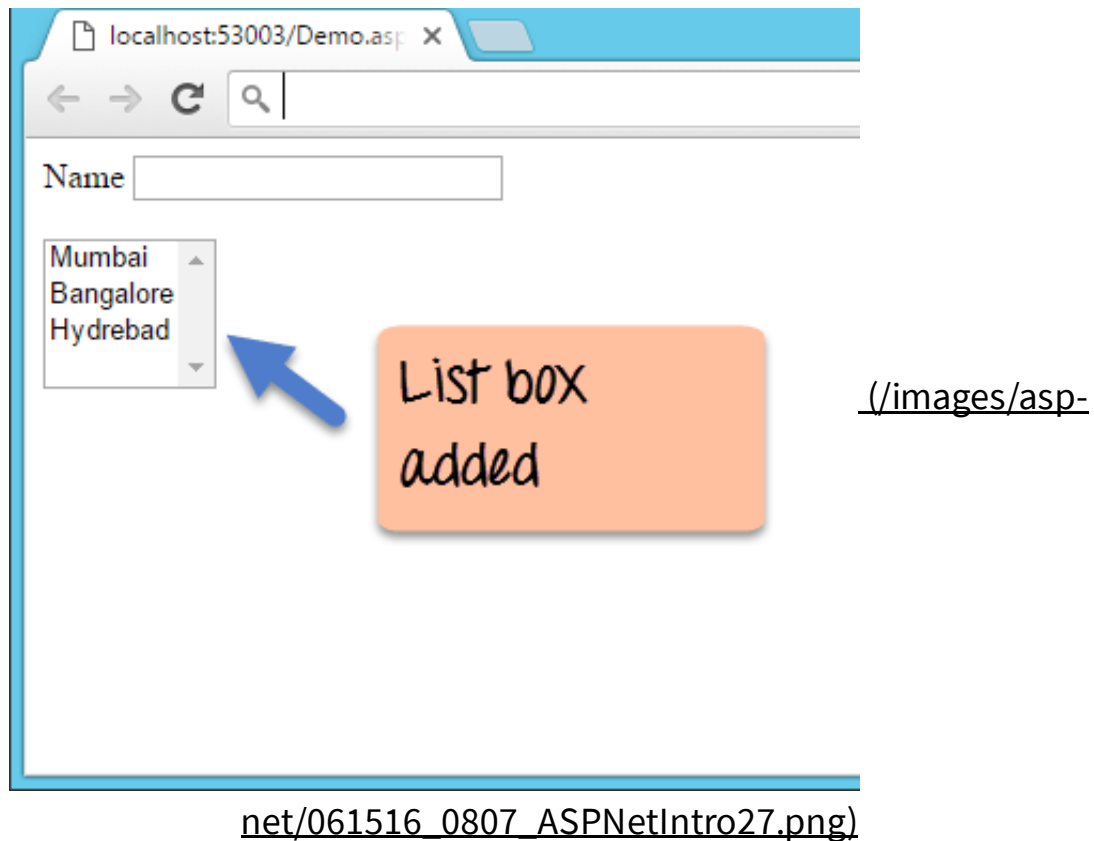
(/images/asp-net/061516_0807_ASPNetIntro25.png)

Step 4) Go to the properties window and change the ID property value of the control to lstLocation.

[(/images/asp-net/061516_0807_ASPNetIntro26.png)](/images/asp-net/061516_0807_ASPNetIntro26.png)

Once you make the above changes, you will see the following output

Output:-

[(/images/asp-net/061516_0807_ASPNetIntro27.png)](/images/asp-net/061516_0807_ASPNetIntro27.png)

From the output, you can clearly see that the Listboxes was added to the form.

RadioButton

A Radio button is used to showcase a list of items out of which the user can choose one. Let's see how we can implement this with an example shown below. We will add a radio button for a male/female option.

Step 1) The first step is to drag the 'radiobutton' control onto the Web Form from the toolbox. (see image below). Make sure to add 2 radio buttons, one for the option of 'Male' and the other for 'Female.'
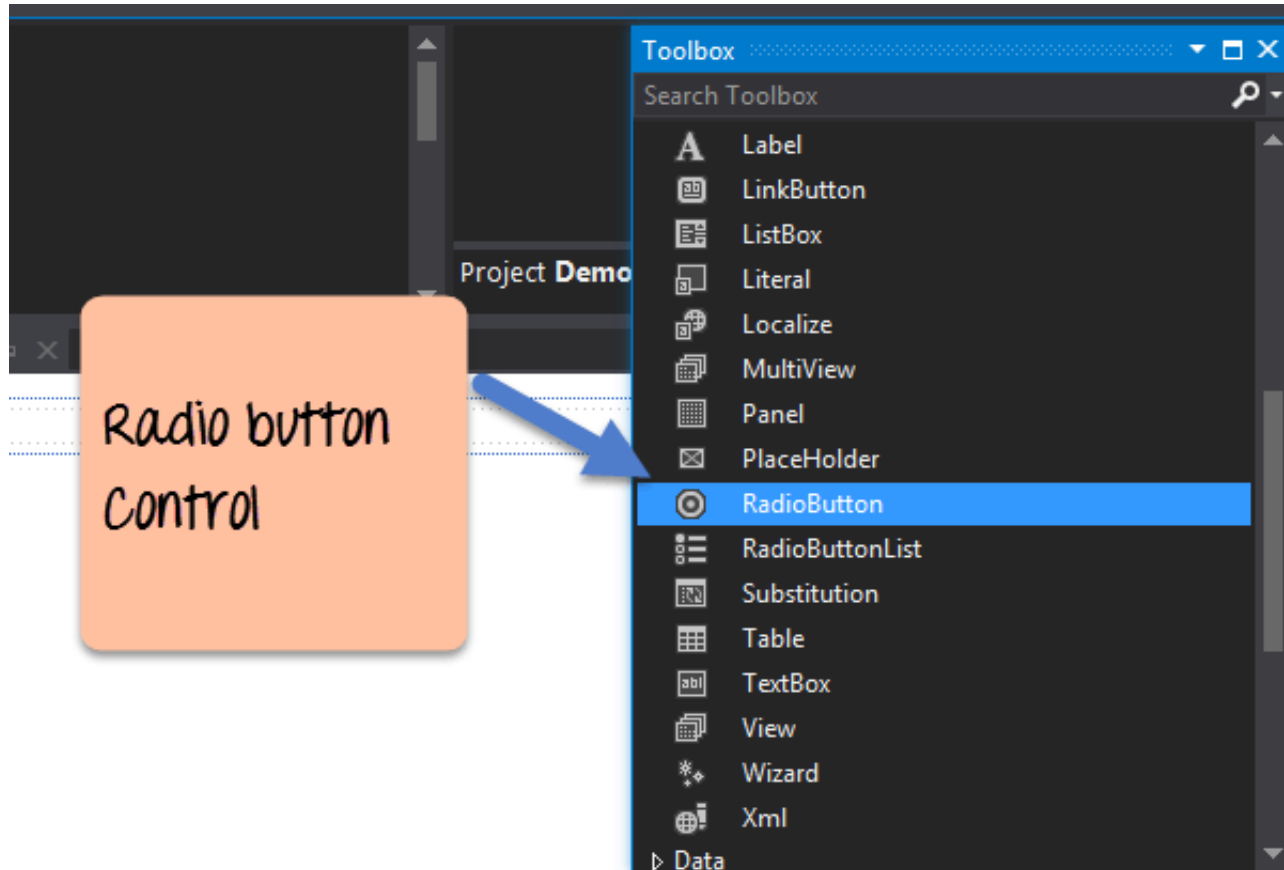
[(/images/asp-net/061516_0807_ASPNetIntro28.png)](/images/asp-net/061516_0807_ASPNetIntro28.png)

Step 2) Once the Radiobutton has been added, change the 'text' property.

- Go to the properties window by clicking on the 'Radiobutton control'.
- Change the text property of the Radio button to 'Male'.
- Repeat the same step to change it to 'Female.'
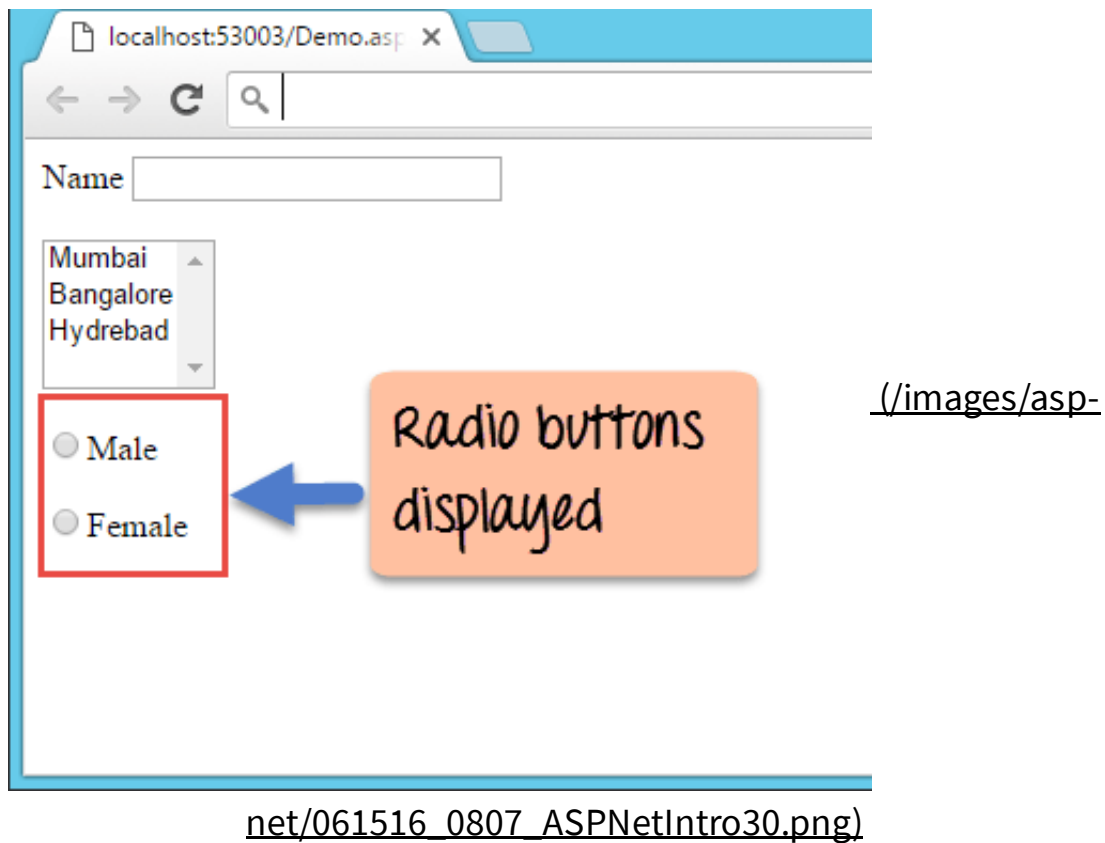- Also, change the ID properties of the respective controls to rdMale and rdFemale.
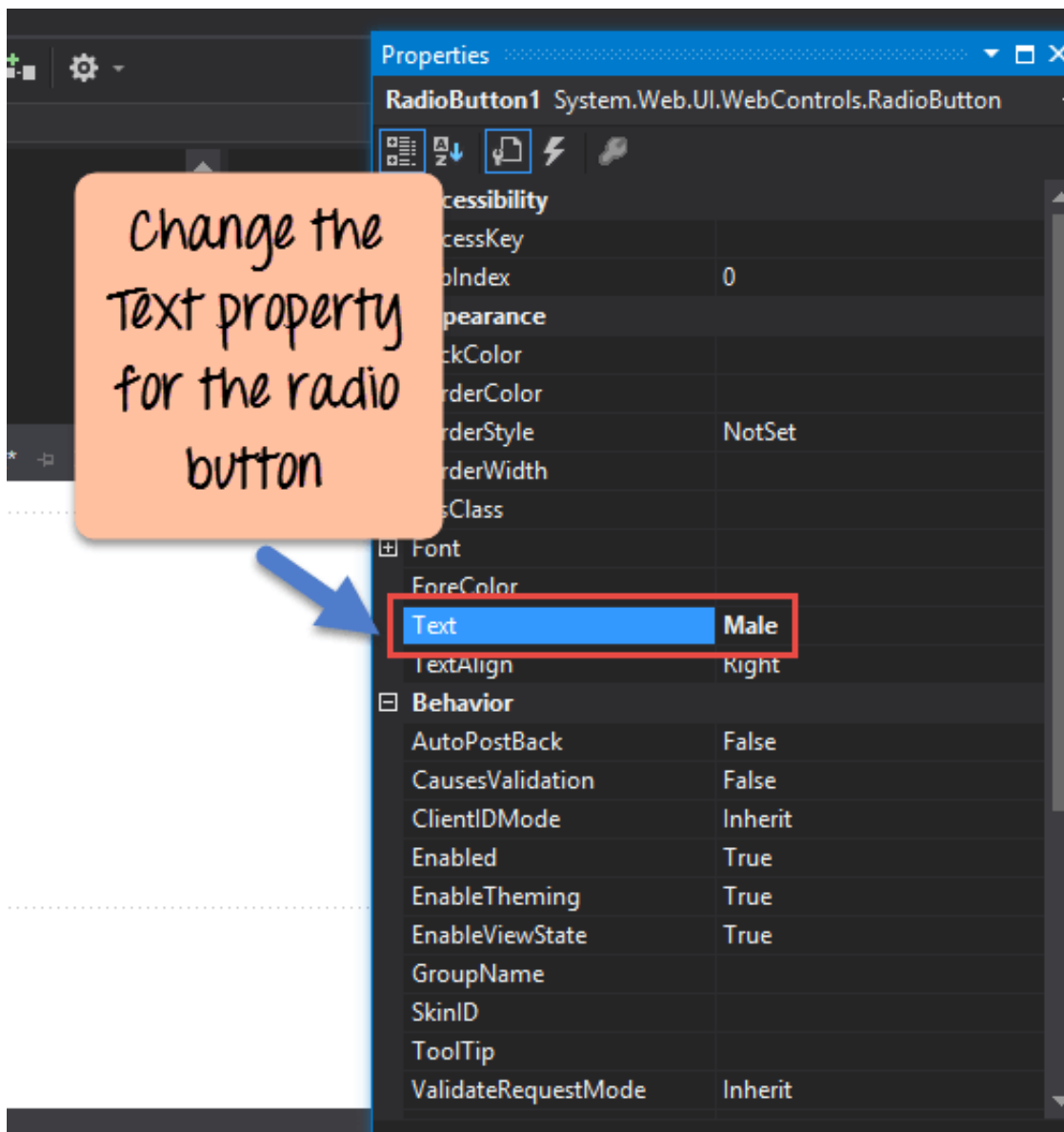
(/images/asp-net/061516_0807_ASPNetIntro29.png)

Once you make the above changes, you will see the following output

Output:-

From the output, you can clearly see that the radio button was added to the form

Checkbox

A checkbox is used to provide a list of options in which the user can choose multiple choices. Let's see how we can implement this with an example shown below. We will add 2 checkboxes to our Web forms. These checkboxes will provide an option to the user on whether they want to learn C# (/c-tutorial.html)or ASP.Net.

Step 1) The first step is to drag the checkbox control onto the Web Form from the toolbox as shown below
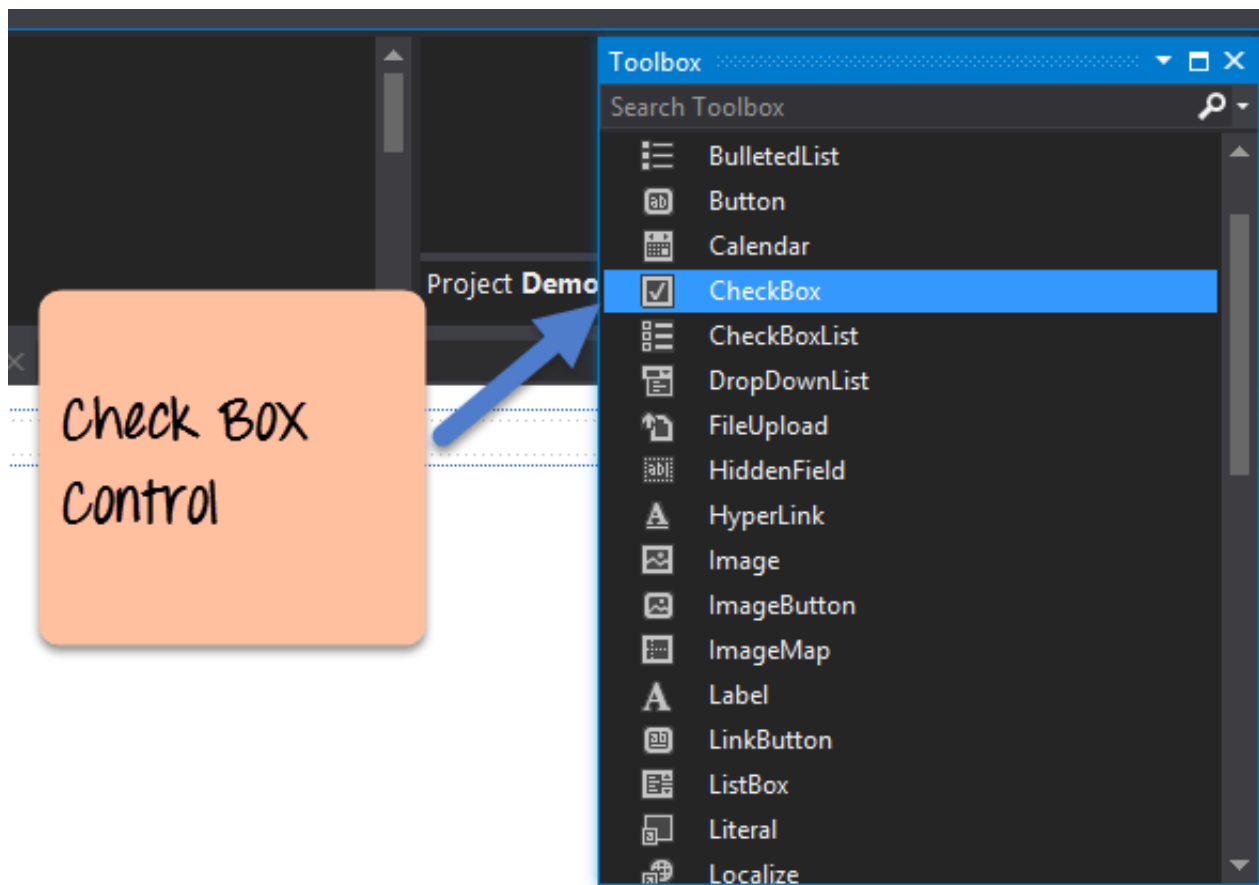
1. Create Form

2. How to Create Own

3. Create a Web Page

4. Registration Form

5. Free Form Samples

6. Free Feedback Form

[(/images/asp-net/061516_0807_ASPNetIntro31.png)](/images/asp-net/061516_0807_ASPNetIntro31.png)

Step 2) Once the Checkboxes have been added, change the checkbox id property to 'chkASP'.

- Go to the properties window by clicking on the Checkbox control.
- Change the ID properties of the respective controls to 'chkC' and 'chkASP'.

Also, change the text property of the Checkbox control to 'C#'. Do the same for the other Checkbox control and change it to 'ASP.Net'.

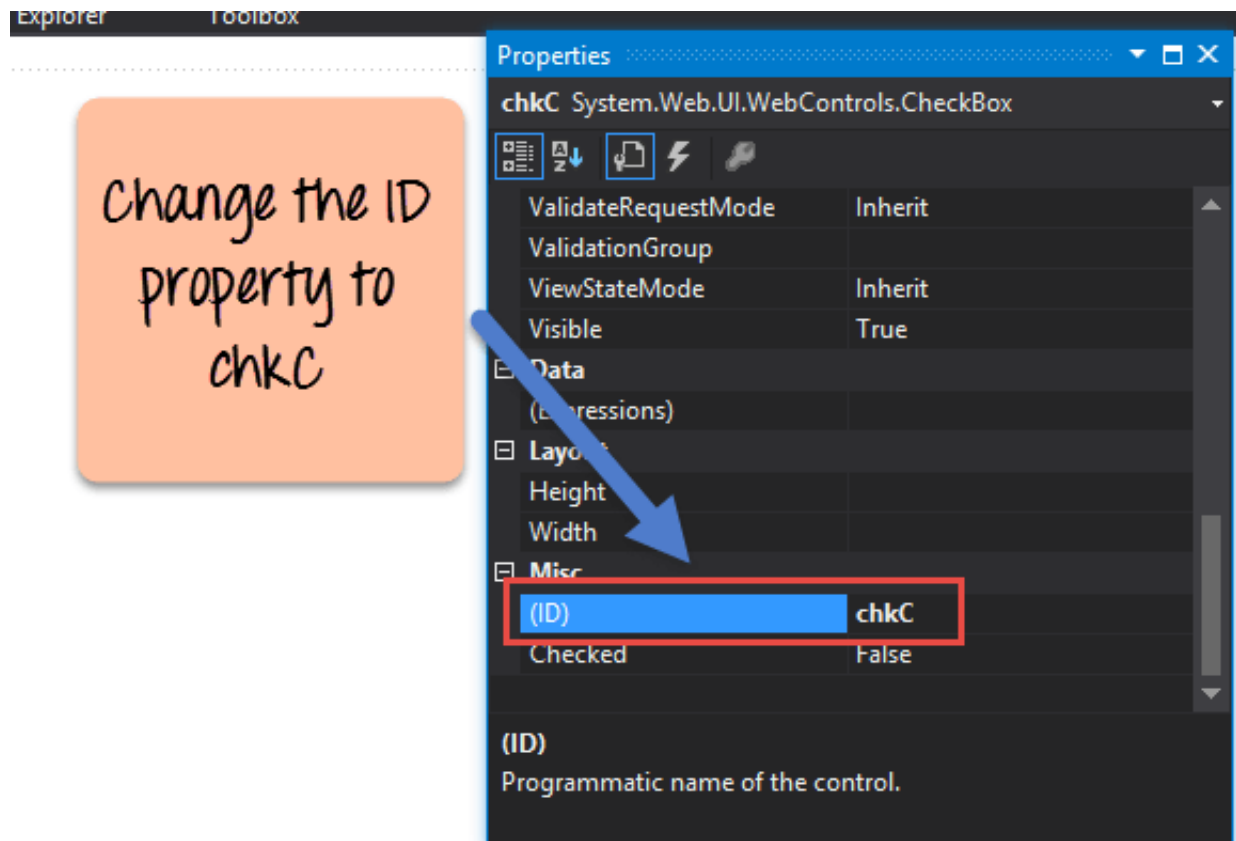1. Change the ID property of the checkbox to 'chkASP'
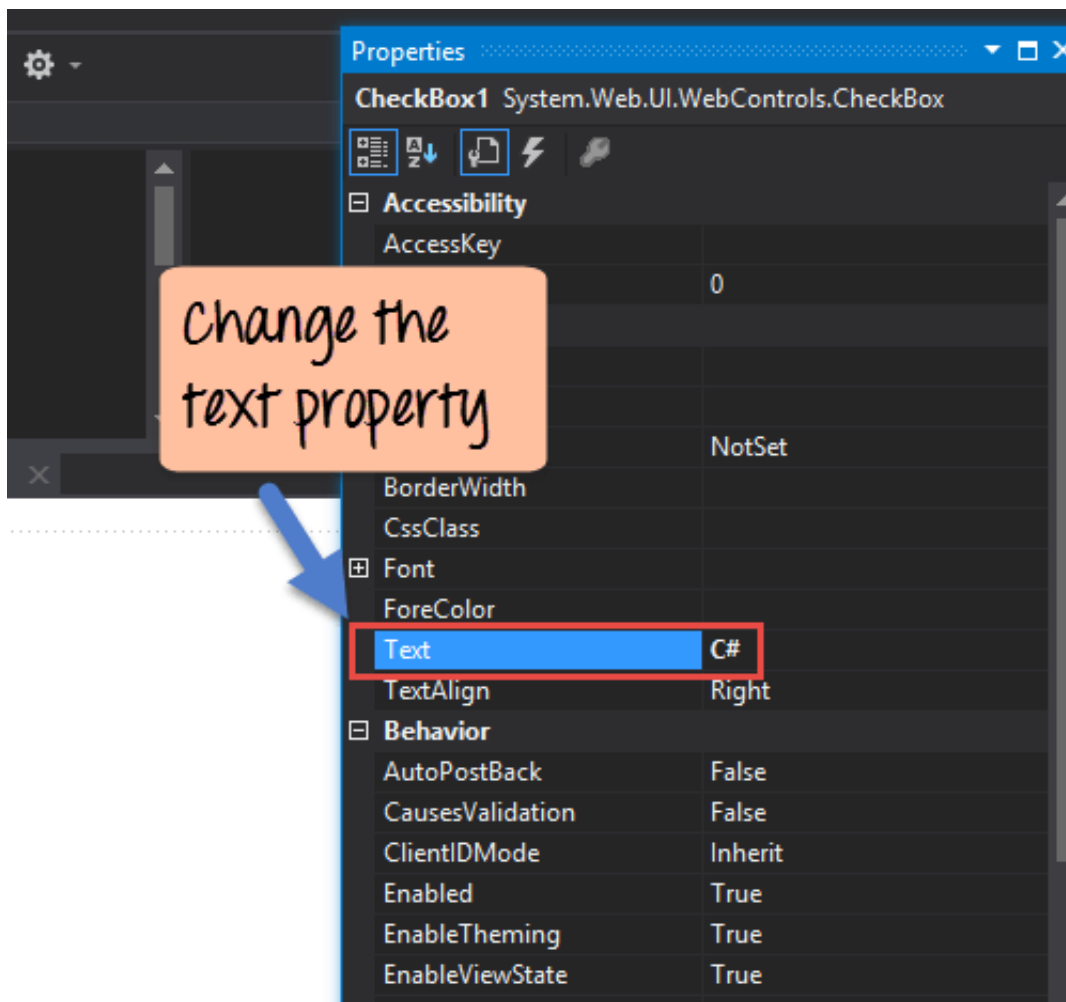
Change the ID property to chkASP
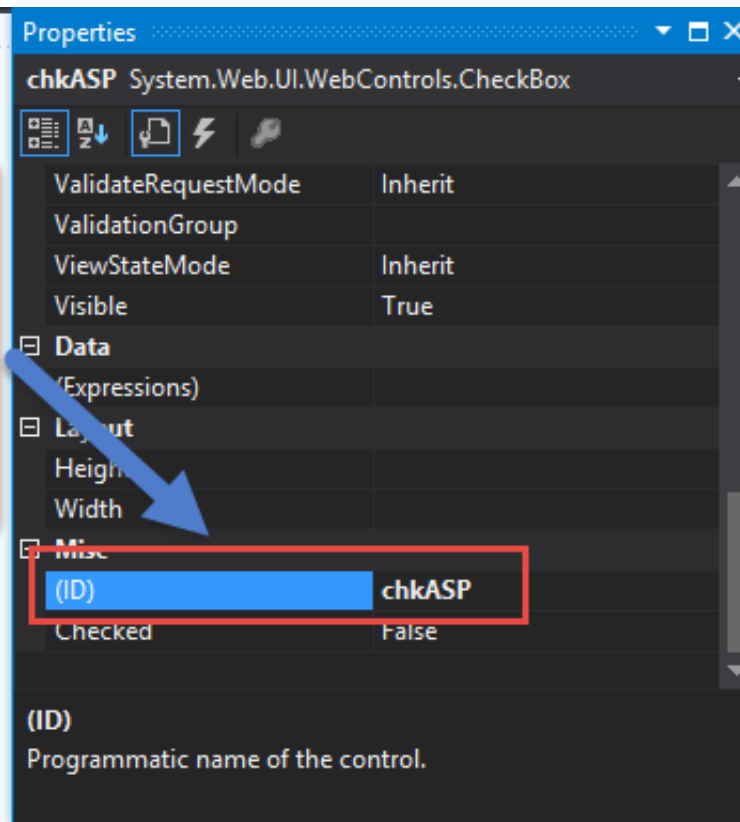
(/images/asp-net/061516_0807_ASPNetIntro32.png)

2. Change the ID property of the checkbox to chkC

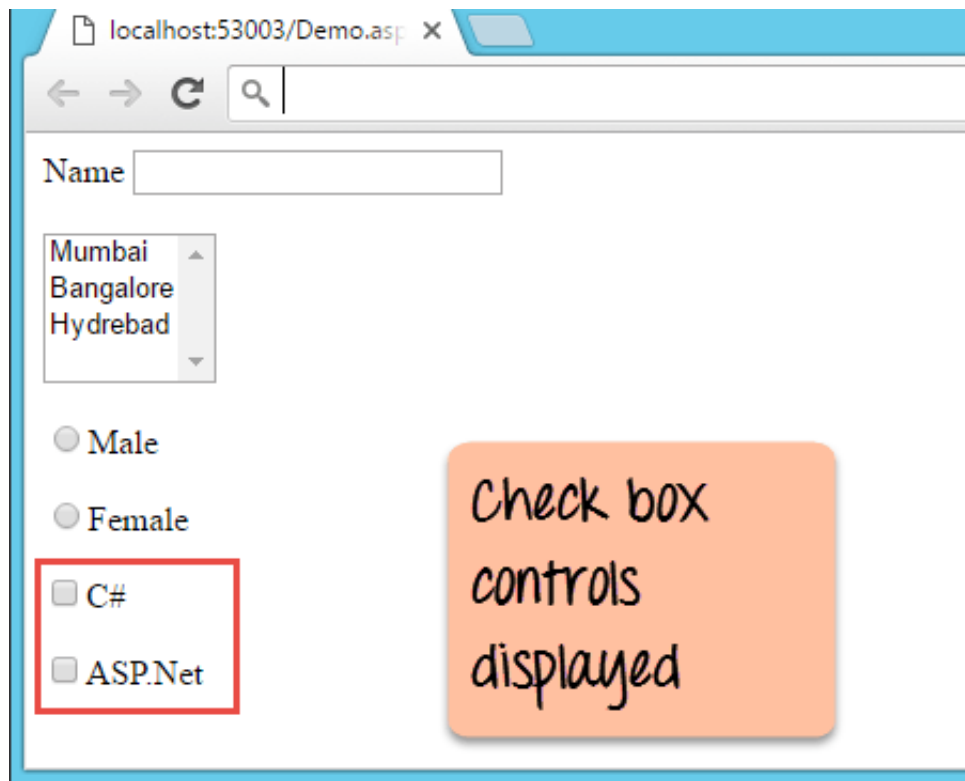Change the ID property to chkC

(/images/asp-net/061516_0807_ASPNetIntro33.png)

(/images/asp-net/061516_0807_ASPNetIntro34.png)

Once you make the above changes, you will see the following output

Output:-

(/images/asp-
net/061516_0807_ASPNetIntro35.png)

From the output, you can clearly see that the Checkboxes was added to the form.
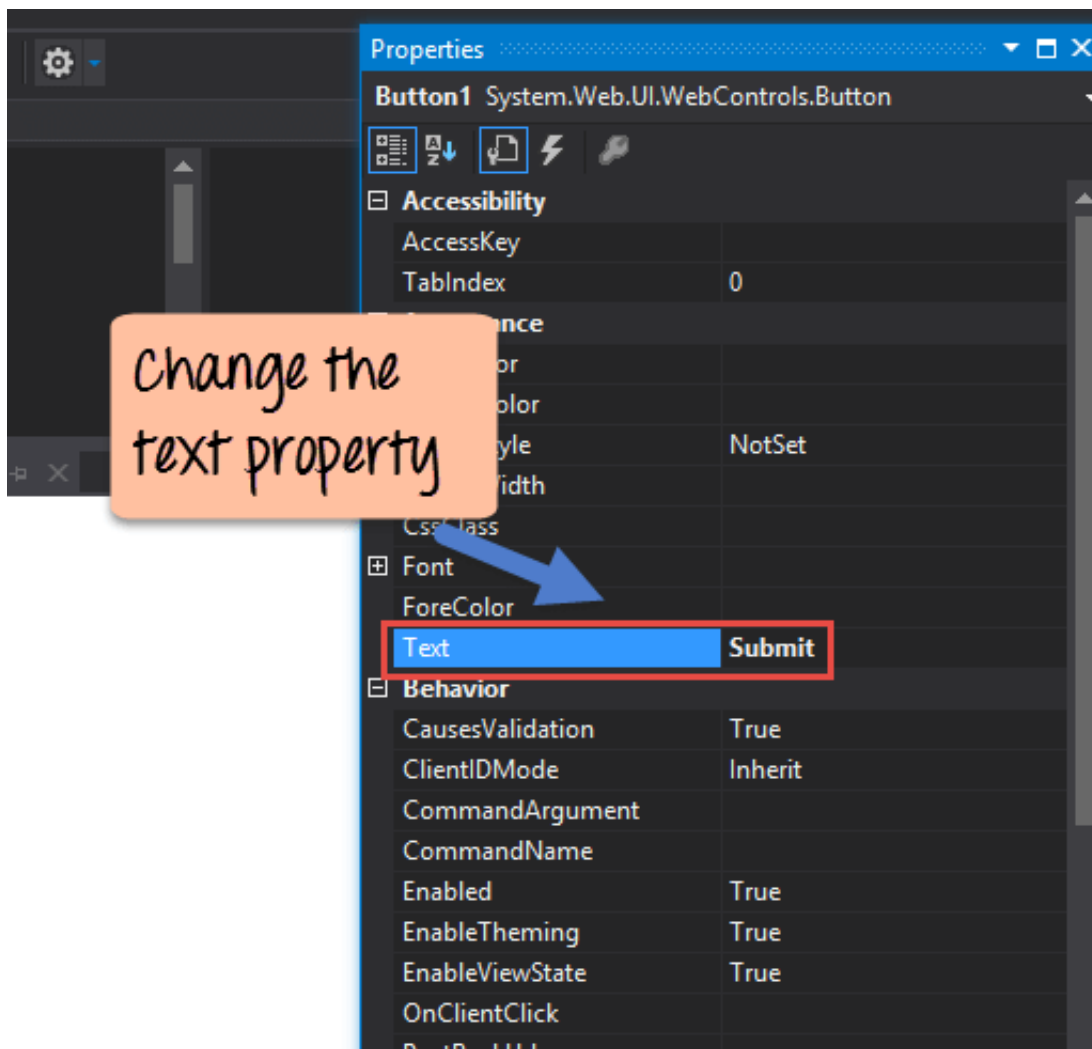
Button

A button is used to allow the user to click on a button which would then start the processing of the form. Let's see how we can implement this with our current example as shown below. We will add a simple button called 'Submit' button. This will be used to submit all the information on the form.

Step 1) The first step is to drag the button control onto the Web Form from the toolbox as shown below
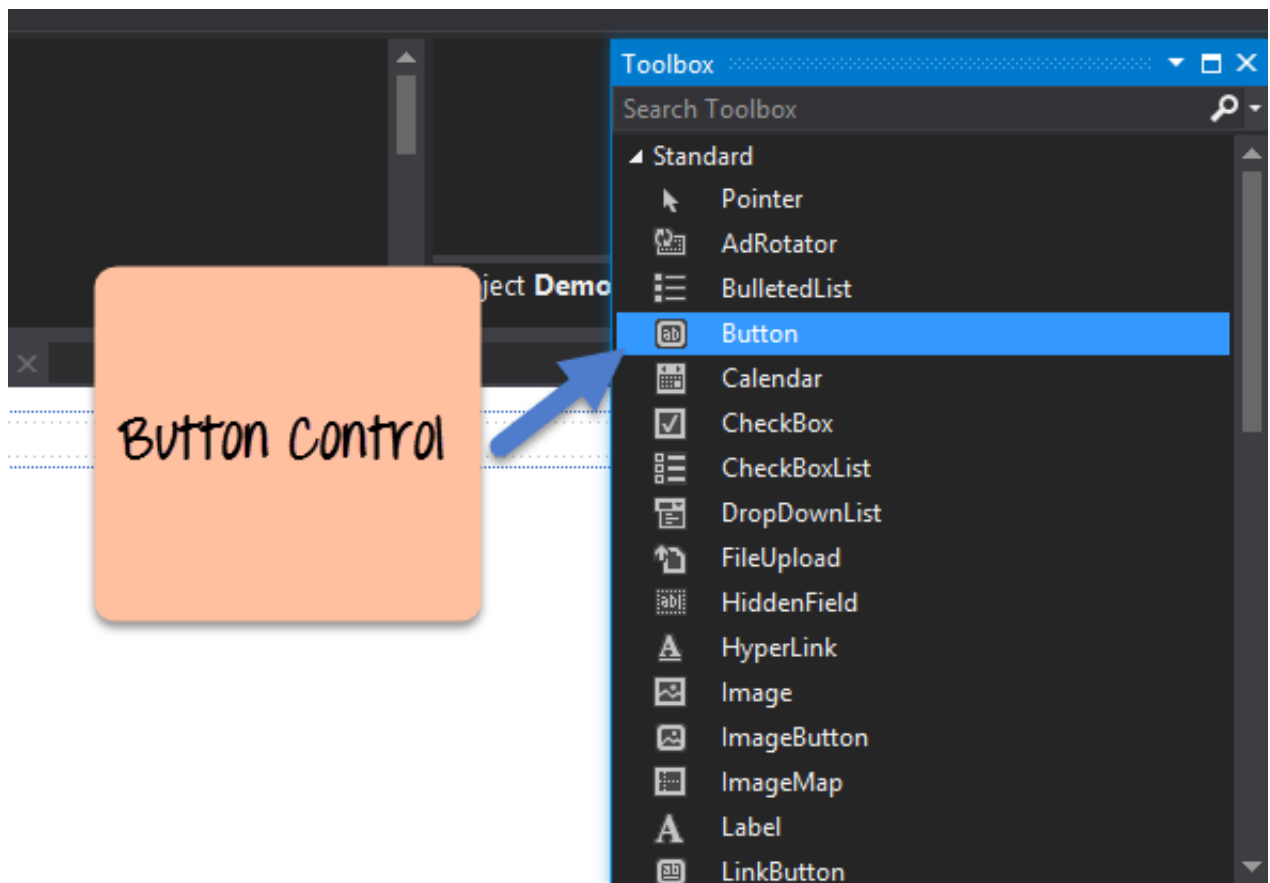
Step 2) Once the button has been added, go to the properties window by clicking on the button control. Change the text property of the button control to Submit. Also, change the ID property of the button to 'btnSubmit'.

(/images/asp-net/061516_0807_ASPNetIntro37.png)

Once you make the above changes, you will see the following output

Output:-

(/images/asp-net/061516_0807_ASPNetIntro38.png)
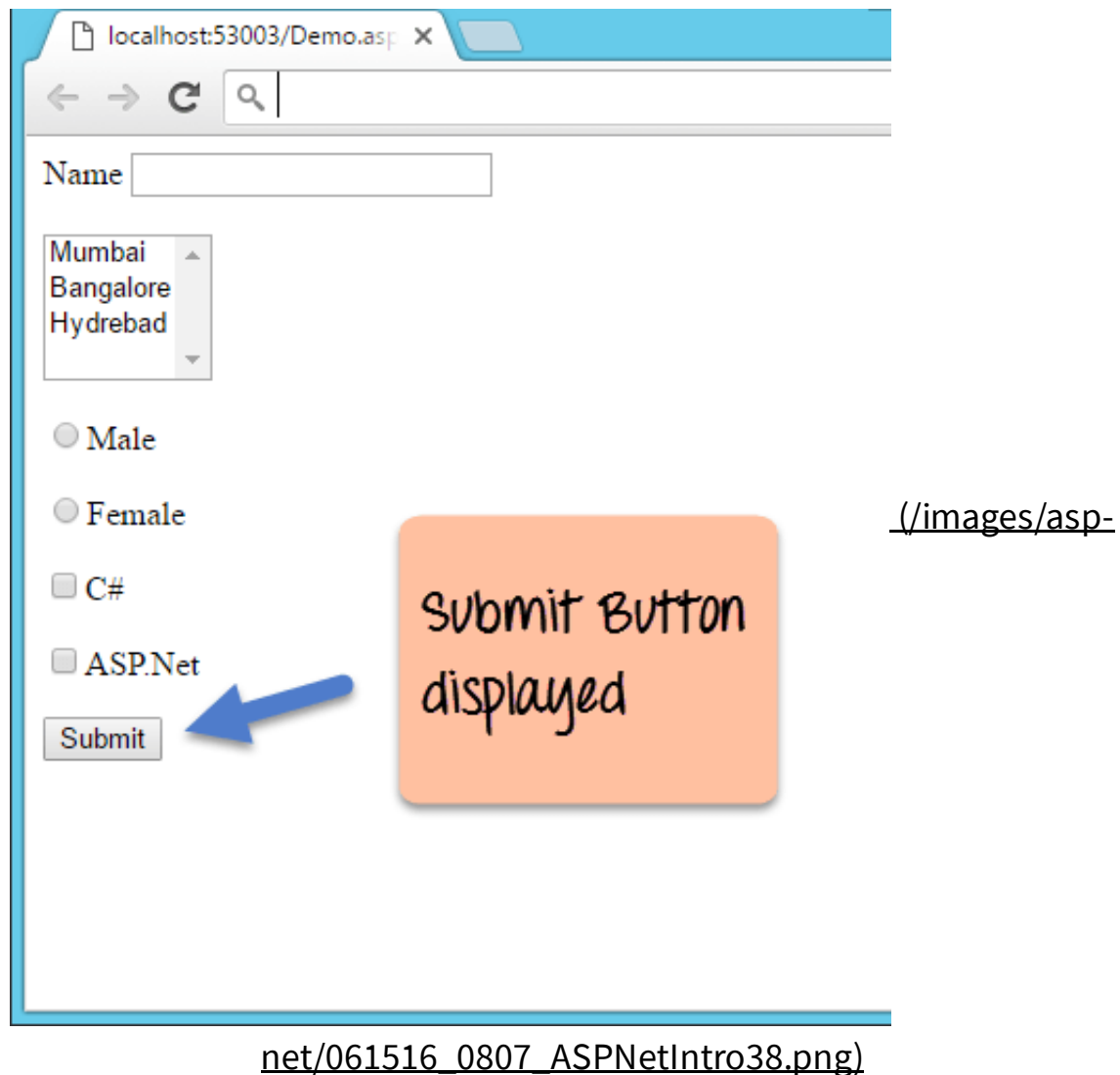
From the output, you can clearly see that the button was added to the form.

Event Handler in ASP.Net

When working with a web form, you can add events to controls. An event is something that happens when an action is performed. Probably the most common action is the clicking of a button on a form.

In web forms, you can add code to the corresponding aspx.cd file. This code can be used to perform certain actions when a button is pressed on the form. This is generally the most common event in Web Forms. Let's see how we can achieve this.

We are going to make this simple. Just add an event to the button control to display the name which was entered by the user. Let's follow the below steps to achieve this.

Step 1) First you have to double-click the Button on the Web Form. This will bring up the event code for the button in Visual Studio.


```
protected void btnSubmit_Click(object sender, EventArgs e)
{


}
```
Event Handler for Button
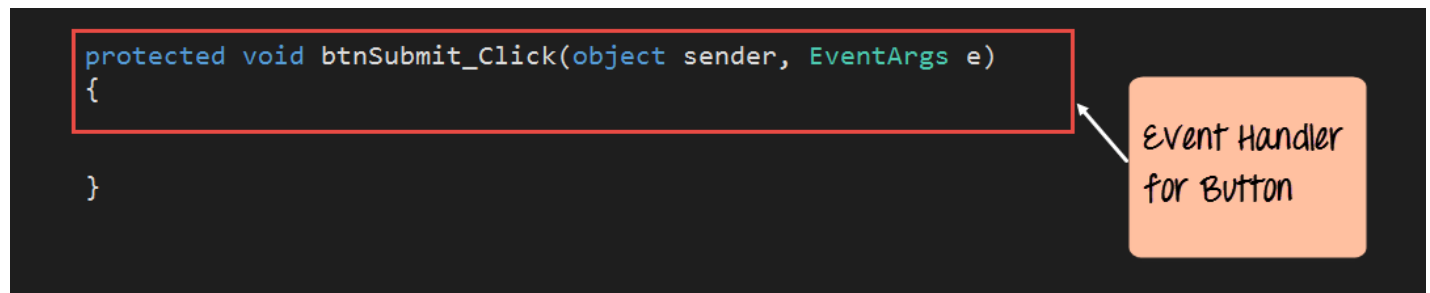
The btnSubmit_Click event is automatically added by Visual Studio when you double click the button in the web forms designer.

Step 2) Let's now add code to the submit event to display the name textbox value and the location chosen by the user.
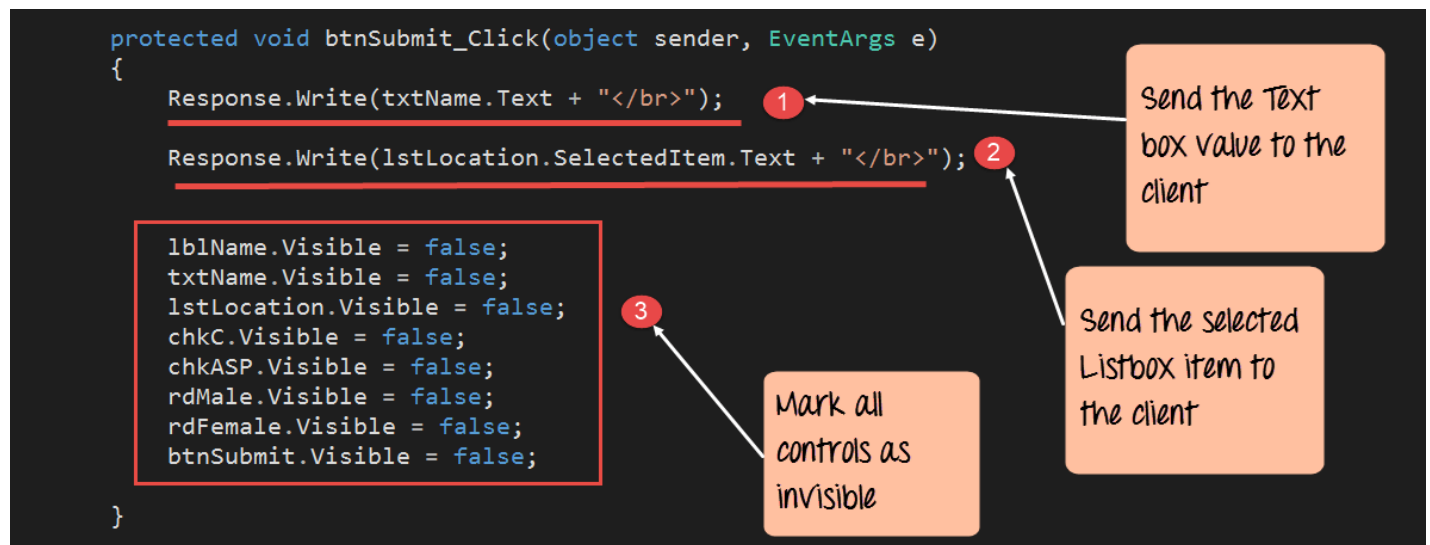

```
protected void btnSubmit_Click(object sender, EventArgs e)
{
    Response.Write(txtName.Text + "</br>");          1

    Response.Write(lstLocation.SelectedItem.Text + "</br>"); 2

    lblName.Visible = false;
    txtName.Visible = false;
    lstLocation.Visible = false;          3
    chkC.Visible = false;
    chkASP.Visible = false;
    rdMale.Visible = false;
    rdFemale.Visible = false;
    btnSubmit.Visible = false;

}
```
Send the Text box value to the client

Send the selected Listbox item to the client

Mark all controls as invisible

```
protected void btnSubmit_Click(object sender,EventArgs e)
{
        Response.Write(txtName.Text + "</br>");

        Response.Write(lstLocation.SelectedItem.Text + "</br>");

        lblName.Visible = false;
        txtName.Visible = false;
        lstLocation.Visible = false;
        chkC.Visible = false;
        chkASP.Visible = false;
        rdFemale.Visible = false;
        btnSubmit.Visible = false;
}
```

Code Explanation:-

1. The above line of code does the simplest thing. It takes the value of the Name textbox control and sends it to the client via the Response object. So if you want to enter the string "Guru99" in the name text box, the value of txtName. A text would be 'Guru99'.

2. The next line of code takes the selected value of the listbox via the property 'lstLocation.SelectedItem.text'. It then writes this value via the Response.Write method back to the client.

3. Finally, we make all the controls on the form as invisible. If we don't do this, all the controls plus our response values will be displayed together.
   Normally, when a person enters all the information on the form such as the Name, location, Gender, etc. The next page shown to the user should only have the information which was not entered. The user does not want to see the Name, Gender, location controls again. But ASP.Net does not know this, and hence, by default, it will again show all the controls when the user clicks the Submit button. Hence, we need to write code to ensure all the controls are hidden so that the user just sees the desired output.

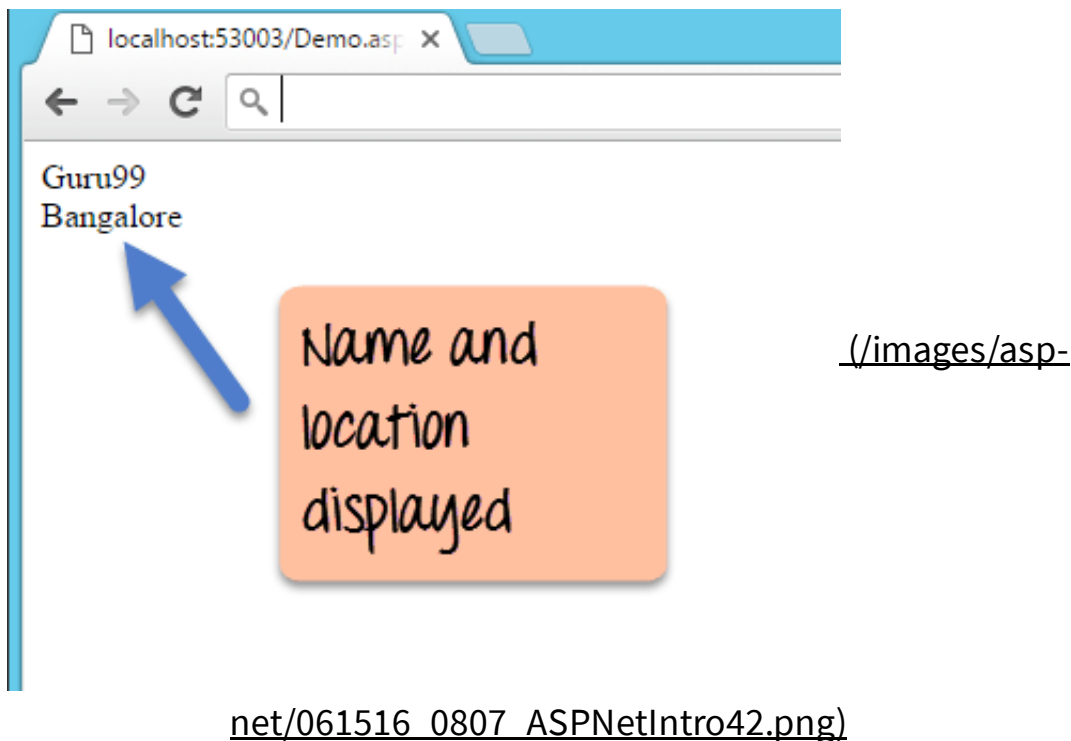One you make the above changes, you will see the following output
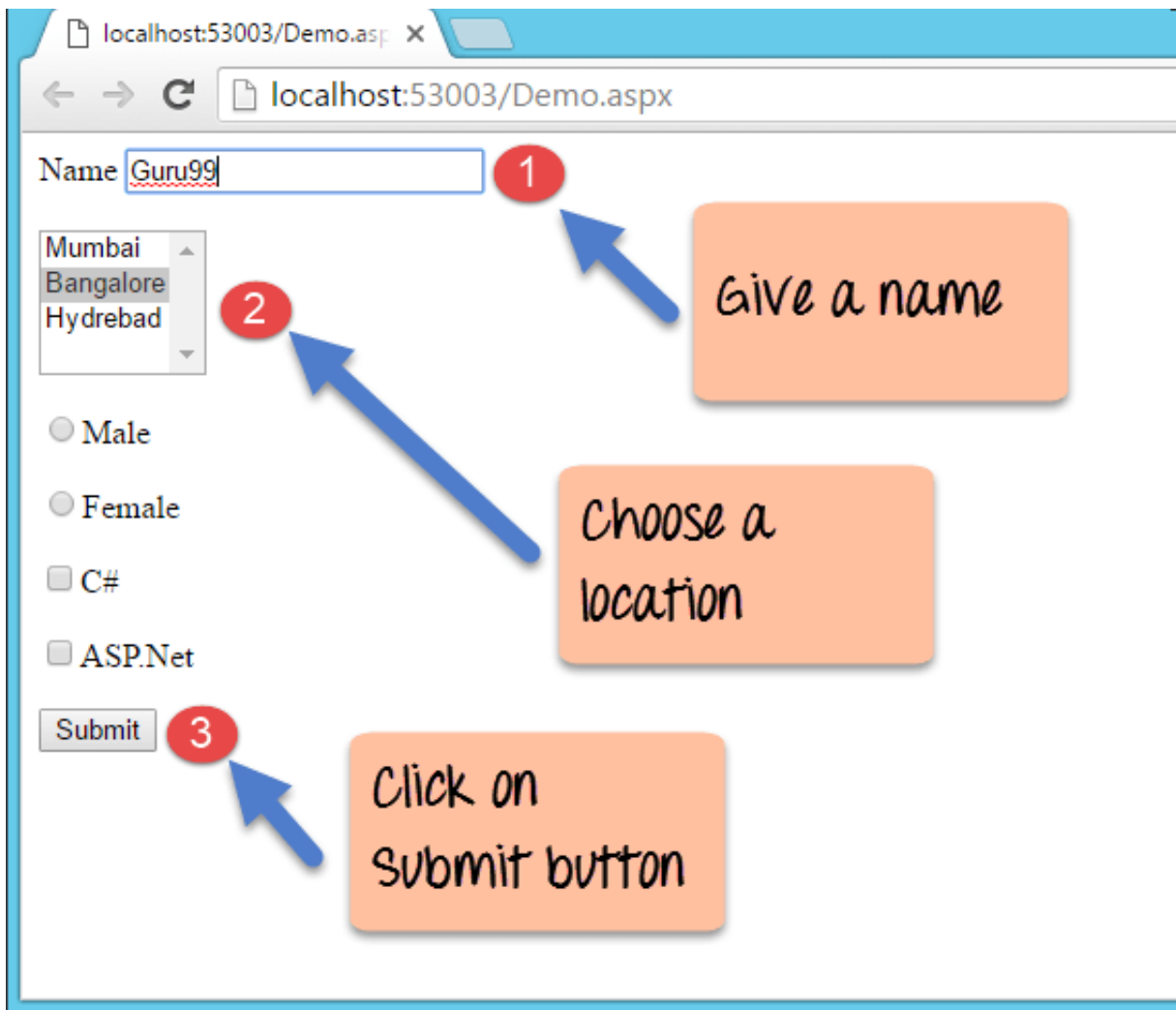
Output:-

[(/images/asp-net/061516_0807_ASPNetIntro41.png)](/images/asp-net/061516_0807_ASPNetIntro41.png)

In the Output screen, carry out the following steps

1. Give a name of Guru99 in the name textbox
2. Choose a location in the listbox of Bangalore
3. Click on the Submit button

Once you do this, you will see 'Guru99' and the location 'Bangalore' is displayed on the page.

Summary:

- In ASP.Net, you can add the standard controls to a form such as labels, textboxes, listboxes, etc.
- Each control can have an event associated with it. The most common event is the button click event. This is used when information needs to be submitted to the web server.

Report a Bug

YOU MIGHT LIKE:

ASP.NET	ASP.NET	ASP.NET
(/asp-net-tracing-	(/asp-net-unit-testing-	(/deploying-website-

debugging-error-handling.html) (/asp-net-tracing-

debugging-error-handling.html)
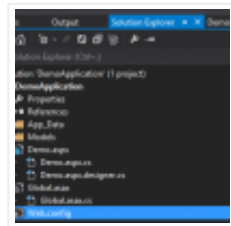
Asp.Net Page Level Tracing, Debugging, Error Handling [Example]

(/asp-net-tracing-debugging-error-handling.html)

project.html) (/asp-net-unit-testing-

project.html)
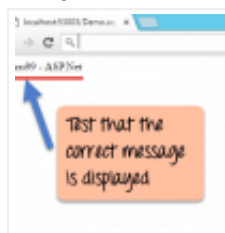
UNIT TESTING in Asp.Net: Complete Tutorial

(/asp-net-unit-testing-project.html)

iis.html)

(/deploying-website-iis.html)

How to Host a Website on IIS: Setup & Deploy Web Application

(/deploying-website-iis.html)

ASP.NET

(/what-is-asp-dot-net.html)

(/what-is-asp-dot-net.html)

What is ASP.NET? and it's ARCHITECTURE

(/what-is-asp-dot-net.html)

ASP.NET

(/asp-net-intro-life-cycle-hello.html) (/asp-net-intro-life-cycle-

hello.html)

ASP.NET Application & PAGE Life Cycle

(/asp-net-intro-life-cycle-hello.html)

ASP.NET
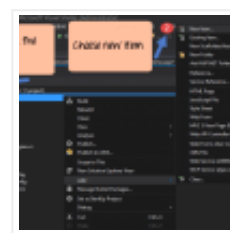
(/asp-net-controls-webforms.html) (/asp-net-controls-

webforms.html)

ASP.NET Web Forms Tutorial: User Controls Examples

(/asp-net-controls-webforms.html)

ASP-Net. Tutorial

1) What is ASP.NET? (/what-is-asp-dot-net.html)

 (https://www.facebook.com/guru99com/)
(https://twitter.com/guru99com)
(https://www.youtube.com/channel/UC19i1XD6k88KqHlET8atqFQ)

(https://forms.aweber.com/form/46/724807646.htm)

About

About US (/about-us.html)
Advertise with Us (/advertise-us.html)
Write For Us (/become-an-instructor.html)
Contact US (/contact-us.html)

Career Suggestion

SAP Career Suggestion Tool (/best-sap-module.html)
Software Testing as a Career (/software-testing-career-complete-guide.html)
Certificates (/certificate-it-professional.html)

Interesting

Books to Read! (/books.html)

Suggest a Tutorial

Blog (/blog/)

Quiz (/tests.html)

eBook (/ebook-pdf.html)

Execute online

Execute Java Online (/try-java-editor.html)

Execute Javascript (/execute-javascript-online.html)

Execute HTML (/execute-html-online.html)

Execute Python (/execute-python-online.html)