**Introduction to ASP.NET**

ASP.NET is a web development platform, which provides a programming model, a comprehensive software infrastructure and various services required to build up robust web applications for PC, as well as mobile devices.

ASP.NET works on top of the HTTP protocol, and uses the HTTP commands and policies to set a browser-to-server bilateral communication and cooperation.

ASP.NET is a part of Microsoft .Net platform. ASP.NET applications are compiled codes, written using the extensible and reusable components or objects present in .Net framework. These codes can use the entire hierarchy of classes in .Net framework.

The ASP.NET application codes can be written in any of the following languages:

Visual Basic.Net

Jscript

C#

J#

ASP.NET is used to produce interactive, data-driven web applications over the internet. It consists of a large number of controls such as text boxes, buttons, and labels for assembling, configuring, and manipulating code to create HTML pages.

ASP.NET Web Forms Model

ASP.NET web forms extend the event-driven model of interaction to the web applications. The browser submits a web form to the web server and the server returns a full mark-up page or HTML page in response.

All client-side user activities are forwarded to the server for stateful processing. The server processes the output of the client actions and triggers the reactions.

Now, HTTP is a stateless protocol. ASP.NET framework helps in storing the information regarding the state of the application, which consists of:

Page state

Session state

The page state is the state of the client, i.e., the content of various input fields in the web form. The session state is the collective information obtained from various pages the user visited and worked with, i.e., the overall session state. To clear the concept, let us take an example of a shopping cart.

User adds items to a shopping cart. Items are selected from a page, say the items page, and the total collected items and price are shown on a different page, say the cart page. Only HTTP cannot keep track of all the information coming from various pages. ASP.NET session state and server-side infrastructure keep track of the information collected globally over a session.

The ASP.NET runtime carries the page state to and from the server across page requests while generating ASP.NET runtime codes, and incorporates the state of the server-side components in hidden fields.

This way, the server becomes aware of the overall application state and operates in a two-tiered connected way.

Button Controls

ASP.NET provides three types of button control:

Button: It displays text within a rectangular area.

Link Button: It displays text that looks like a hyperlink.

Image Button: It displays an image.

When a user clicks a button, two events are raised: Click and Command.

Basic syntax of button control:

<asp:Button ID="Button1" runat="server" onclick="Button1\_Click" Text="Click" />

Common properties of the button control:

Property Description

Text The text displayed on the button. This is for button and link button controls only.

ImageUrl For image button control only. The image to be displayed for the button.

AlternateText For image button control only. The text to be displayed if the browser cannot display the image.

CommandName A string value that is passed to the command event when a user clicks the button.

CommandArgument A string value that is passed to the command event when a user clicks the button.

PostBackUrl The URL of the page that is requested when the user clicks the button.

**Text Boxes and Labels**

Text box controls are typically used to accept input from the user. A text box control can accept one or more lines of text depending upon the settings of the TextMode attribute.

Label controls provide an easy way to display text which can be changed from one execution of a page to the next. If you want to display text that does not change, you use the literal text.

Basic syntax of text control:

<asp:TextBox ID="txtstate" runat="server" ></asp:TextBox>

Common Properties of the Text Box and Labels:

Property Description

TextMode Specifies the type of text box. SingleLine creates a standard text box, MultiLIne creates a text box that accepts more than one line of text and the Password causes the characters that are entered to be masked. The default is SingleLine.

Text The text content of the text box.

MaxLength The maximum number of characters that can be entered into the text box.

Wrap It determines whether or not text wraps automatically for multi-line text box; default is true.

ReadOnly Determines whether the user can change the text in the box; default is false, i.e., the user cannot change the text.

Columns The width of the text box in characters. The actual width is determined based on the font that is used for the text entry.

Rows The height of a multi-line text box in lines. The default value is 0, means a single line text box.

The mostly used attribute for a label control is 'Text', which implies the text displayed on the label.

**Check Boxes and Radio Buttons**

A check box displays a single option that the user can either check or uncheck and radio buttons present a group of options from which the user can select just one option.

To create a group of radio buttons, you specify the same name for the GroupName attribute of each radio button in the group. If more than one group is required in a single form, then specify a different group name for each group.

If you want check box or radio button to be selected when the form is initially displayed, set its Checked attribute to true. If the Checked attribute is set to true for multiple radio buttons in a group, then only the last one is considered as true.

Basic syntax of check box:

<asp:CheckBox ID= "chkoption" runat= "Server">

</asp:CheckBox>

Basic syntax of radio button:

<asp:RadioButton ID= "rdboption" runat= "Server">

</asp: RadioButton>

Common properties of check boxes and radio buttons:

Property Description

Text The text displayed next to the check box or radio button.

Checked Specifies whether it is selected or not, default is false.

GroupName Name of the group the control belongs to.

List Controls

ASP.NET provides the following controls

Drop-down list

List box

Radio button list

Check box list

Bulleted list.

These controls let a user choose from one or more items from the list. List boxes and drop-down lists contain one or more list items. These lists can be loaded either by code or by the ListItemCollection editor.

Basic syntax of list box control:

<asp:ListBox ID="ListBox1" runat="server" AutoPostBack="True" OnSelectedIndexChanged="ListBox1\_SelectedIndexChanged">

</asp:ListBox>

Basic syntax of drop-down list control:

<asp:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True" OnSelectedIndexChanged="DropDownList1\_SelectedIndexChanged">

</asp:DropDownList>

Common properties of list box and drop-down Lists:

Property Description

Items The collection of ListItem objects that represents the items in the control. This property returns an object of type ListItemCollection.

Rows Specifies the number of items displayed in the box. If actual list contains more rows than displayed then a scroll bar is added.

SelectedIndex The index of the currently selected item. If more than one item is selected, then the index of the first selected item. If no item is selected, the value of this property is -1.

SelectedValue The value of the currently selected item. If more than one item is selected, then the value of the first selected item. If no item is selected, the value of this property is an empty string ("").

SelectionMode Indicates whether a list box allows single selections or multiple selections.

Common properties of each list item objects:

Property Description

Text The text displayed for the item.

Selected Indicates whether the item is selected.

Value A string value associated with the item.

**The ListItemCollection**

The ListItemCollection object is a collection of ListItem objects. Each ListItem object represents one item in the list. Items in a ListItemCollection are numbered from 0.

When the items into a list box are loaded using strings like: lstcolor.Items.Add("Blue"), then both the Text and Value properties of the list item are set to the string value you specify. To set it differently you must create a list item object and then add that item to the collection.

The ListItemCollection Editor is used to add item to a drop-down list or list box. This is used to create a static list of items. To display the collection editor, select edit item from the smart tag menu, or select the control and then click the ellipsis button from the Item property in the properties window.

Common properties of ListItemCollection:

Property Description

Item(integer) A ListItem object that represents the item at the specified index.

Count The number of items in the collection.

Common methods of ListItemCollection:

Methods Description

Add(string) Adds a new item at the end of the collection and assigns the string parameter to the Text property of the item.

Add(ListItem) Adds a new item at the end of the collection.

Insert(integer, ListItem) Inserts the item at the specified index location in the collection.

Remove(string) Removes the item with the text value same as the string.

Remove(ListItem) Removes the specified item.

RemoveAt(integer) Removes the item at the specified index as the integer.

Clear Removes all the items of the collection.

FindByValue(string) Returns the item whose value is same as the string.

FindByValue(Text) Returns the item whose text is same as the string.

**Radio Button list and Check Box list**

A radio button list presents a list of mutually exclusive options. A check box list presents a list of independent options. These controls contain a collection of ListItem objects that could be referred to through the Items property of the control.

Basic syntax of radio button list:

<asp:RadioButtonList ID="RadioButtonList1" runat="server" AutoPostBack="True"

OnSelectedIndexChanged="RadioButtonList1\_SelectedIndexChanged">

</asp:RadioButtonList>

Basic syntax of check box list:

<asp:CheckBoxList ID="CheckBoxList1" runat="server" AutoPostBack="True"

OnSelectedIndexChanged="CheckBoxList1\_SelectedIndexChanged">

</asp:CheckBoxList>

Bulleted lists and Numbered lists

The bulleted list control creates bulleted lists or numbered lists. These controls contain a collection of ListItem objects that could be referred to through the Items property of the control.

Basic syntax of a bulleted list:

<asp:BulletedList ID="BulletedList1" runat="server">

</asp:BulletedList>

Common properties of the bulleted list:

Property Description

BulletStyle This property specifies the style and looks of the bullets, or numbers.

RepeatDirection It specifies the direction in which the controls to be repeated. The values available are Horizontal and Vertical. Default is Vertical.

RepeatColumns It specifies the number of columns to use when repeating the controls; default is 0.