get and post method

Http is the most common protocol used for communication between a web server and a client.

The 2 most commonly used HTTP methods are GET and POST.

GET Request - is generally used to get data from the web server. A GET request is generally issued,

1. When you click on a hyperlink

2. When Response.Redirect() statement is executed

3. When you type URL in the address bar and hit enter

POST Request - is generally used to submit data to the server. A POST request is generally issued,

1. When you click on a submit button

2. When AUTOPOST back is set true and when a selection in the DropDownList is changed

Difference between GET and POST method

1. GET method appends data to the URL, where as with the POST method data can either be appended to the URL or in the message body.

2. As GET request rely on querystrings to send data to the server, there is a length restriction, where as POST requests have no restrictions on data length.

3. While it is possible to change the state of data in database using GET request, they should only be used to retrieve data.

Get Method-

Default.aspx

<form id="form1" method="get" action="Dfg.aspx" runat="server">

<div>

User ID :<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

<br/>

<br />

Password :<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>

<br />

<br />

<br />

<asp:Button ID="Button1" runat="server" Text="Submit" Width="97px" />

</div>

</form>

...........

Dfg.aspx

protected void Page\_Load(object sender, EventArgs e)

{

string id = Request.QueryString["TextBox1"].ToString();

string password = Request.QueryString["TextBox2"].ToString();

Response.Write("Id : " + id + "<br/>");

Response.Write("Password : " + password + "<br/>");

}

When using get method , we need to use Request.QueryString because data will be append into the url of the web page.

It is very important to understand that QueryString takes the reference of name of the control not id

-------------------------------------------------------------

Post method-

<form id="form1" method="post" action="Dfg.aspx" runat="server">

<div>

User ID :<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

<br/>

<br />

Password :<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>

<br />

<br />

<br />

<asp:Button ID="Button1" runat="server" Text="Submit" Width="97px" />

</div>

</form>

.............

Dfg.aspx-

protected void Page\_Load(object sender, EventArgs e)

{

string id = Request.Form["TextBox1"].ToString();

string password = Request.Form["TextBox2"].ToString();

Response.Write("Id : " + id + "<br/>");

Response.Write("Password : " + password + "<br/>");

}

page navigation technique----

What are the different page navigation techniques in asp.net?

OR

How do you move from one webform to another webform in asp.net?

OR

How do you link pages in an application?

1-Hyperlink control-

It is used to navigate to another page. The page you want to navigate to is specified by the NavigateURL property.

Default.aspx

<asp:HyperLink ID="HyperLink1" runat="server" NavigateUrl="~/Default2.aspx">HyperLink</asp:HyperLink>

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

2-Response.Redirect-

Response.Redirect is similar to clicking on a hyperlink.The Hyperlink control does not expose any server side events

when Response.Redirect is used the URL in the address bar changes and the browser history is maintained.

Response.Redirect() can be used to navigate pages/websites on the same web server or on a different web server.

Response.Redirect("~/Default2.aspx");

//Response.Redirect(Page.Request.HttpMethod); //it is used to find out whether the method is get or post

3-Server.Transfer-

The following are the differences between Server.Transfer and Response.Redirect

1. Just like hyperlink and Response.Redirect, Server.Transfer is used to navigate to other pages/sites running on the same web server.

2. Server.Transfer cannot be used to navigate to sites/pages on a different web server.

3. Server.Transfer does not change the URL in the address bar

4. Server.Transfer is faster than Response.Redirect as the redirection happens on the server in one Request/Response cycle. Response.Redirect() involves 2 Request/Response cycles.

5. With Server.Transfer the Form Variables from the original request are preserved.

// Server.Transfer("~/Default2.aspx");

4-Server.Execute-

Server.Transfer terminates the execution of the current page and starts the execution of the new page,

where as Server.Execute process the second Web form without leaving the first Web form.

5-Cross-Page postback

6-Window.Open

.......

Default2.aspx

Response.Write(Page.Request.HttpMethod);