**What is AJAX?**

Ajax stands for Asynchronous Javascript & XML. It is a web technology through which a postback from a client (browser) to the server goes partially, which means that instead of a complete postback, a partial postback is triggered by the Javascript XmlHttpRequest object. In such a scenario, web-application users won't be able to view the complete postback progress bar shown by the browser. In an AJAX environment, it is Javascript that starts the communication with the web server. Ajax technology in a website may be implemented by using plain Javascript and XML. Code in such a scenario may tend to look little complex, for which the AJAX Framework in .NET can be embedded in ASP.NET web applications. In addition to XML & Javascript, AJAX is also based on DOM - the Document Object Model technology of browsers through which objects of the browser can be accessed through the memory heap using their address. JSON - Javascript Object Notation is also one of the formats used in AJAX, besides XML. So basically, in an AJAX-based web application, the complete page does not need to reload, and only the objects in context of ajaxification are reloaded. Ajax technology avoids the browser flickering.

**What are limitations of Ajax?**

An Ajax Web Application tends to confused end users if the network bandwidth is slow, because there is no full postback running. However, this confusion may be eliminated by using an UpdateProgress control in tandem.

An Ajax Web Application is little bit confused to end users because if there is huge amout of data to show in grid like structure than it may not be worked that why we were put a progress bar to know the end user that there is something going on.

**How do I get the XMLHttpRequest object?**

Depending upon the browser... if (window.ActiveXObject) { // Internet Explorer http\_request = new ActiveXObject("Microsoft.XMLHTTP"); } else if...

if(window.XMLHttpRequest)  
{  
xmlhttpobj=new XMLHttpRequest();  
return xmlhttpobj;  
  
}  
else  
{  
try  
{  
  
xmlhttpobj=new ActiveXObject("Microsoft.XMLHTTP");  
}catch(e)  
{  
xmlhttpobj=new ActiveXObject("Msxml2.XMLHTTP");  
}  
}  
}

**What is Dojo?**

Dojo is a third-party javascript toolkit for creating rich featured applications. Dojo is an Open Source DHTML toolkit written in JavaScript. It builds on several contributed code bases (nWidgets, Burstlib, f(m)), which is why we refer to it sometimes as a "unified" toolkit. Dojo aims to solve some long-standing historical problems with DHTML which prevented mass adoption of dynamic web application development.

**Should I use XML or text, JavaScript, or HTML as a return type?**

It depends. Clearly the 'X' in AJAX stands for XML, but several AJAX proponents are quick to point out that nothing in AJAX, per se, precludes using other types of payload, such as, JavaScript, HTML, or plain text.  
  
\* XML - Web Services and AJAX seem made for one another. You can use client-side API's for downloading and parsing the XML content from RESTful Web Services. (However be mindful with some SOAP based Web Services architectures the payloads can get quite large and complex, and therefore may be inappropriate with AJAX techniqes.)  
\* Plain Text - In this case server-generated text may be injected into a document or evaluated by client-side logic.  
\* JavaScript - This is an extension to the plain text case with the exception that a server-side component passes a fragment of JavaScript including JavaScript object declarations. Using the JavaScript eval() function you can then create the objects on the client. JavaScript Object Notation (JSON), which is a JavaScript object based data exchange specification, relies on this technique.  
\* HTML - Injecting server-generated HTML fragments directly into a document is generally a very effective AJAX technique. However, it can be complicated keeping the server-side component in sync with what is displayed on the client.  
  
Mashup is a popular term for creating a completely new web application by combining the content from disparate Web Services and other online API's. A good example of a mashup is housingmaps.com which graphically combines housing want-ads from craiglist.org and maps from maps.google.com.

**Can we override the EnablePartialRendering property of the ScriptManager class?**

Yes. But this has to be done before the init event of the page (or during runtime after the page has already loaded). Otherwise an InvalidOperationException will be thrown.

**How to handle multiple or concurrent requests in Ajax?**

For concurrent requests, declare separate XmlHttpRequest objects for each request. For example, for request to get data from an SQL table1, use something like this...  
xmlHttpObject1.Onreadystatechange = functionfromTable1();  
and to get data from another table (say table2) at the same time, use  
xmlHttpObject2.Onreadystatechange = functionfromTable2();  
Ofcourse, the XmlHttpObject needs to be opened & parameters passed too, like as shown below...  
xmlHTTPObject1.open("GET","http://"localhost// " + "Website1/Default1.aspx" true);  
Note that the last parameter "true" used above means that processing shall carry on without waiting for any response from the web server. If it is false, the function shall wait for a response.

**How to create an AJAX website using Visual Studio?**

Using Visual Studio Web Developer Express 2005 & versions above it, Ajax based applications may easily be created. Note that the Ajax Framework & Ajax Extensions should be installed (In case of VS 2005). If using Visual Studio 2008 Web Developer Express or above, Ajax comes along with it (so no need of a separate installation).  
Steps: Start Visual Studio, Click on File -> New Website -> Under Visual Studio Installed templates -> Select ASP.NET Ajax-Enabled Site. Enter a location & select OK.

**What is the main difference between Client side Java Script and and Server side Java Script how actully they run on both side with Example**

Clientside Javascript encompasses the core language plus extras such as the predefined objects, only relevant to running Javasript in a browser.The clientside javascript is embedded directly in the HTML pages and is interpreted by the browser completly at the run time.  
  
Serverside javascript also encompasses the core language plus extas as the predefined objects and functions only relevant to running Javasript in a server.The serverside javascripts are compiled before they are deployed.

**What browsers support AJAX?**

Internet Explorer 5.0 and above, Opera 7.6 and above, Netscape 7.1 and above, Firefox 1.0 and above, Safari 1.2 and above, among others.

**How do I create a thread to do AJAX polling?**

JavaScript does not have threads. JavaScript functions are called when an event happens in a page such as the page is loaded, a mouse click, or a form element gains focus. You can create a timer using the setTimeout which takes a function name and time in milliseconds as arguments. You can then loop by calling the same function as can be seen in the JavaScript example below.  
  
function checkForMessage() {  
// start AJAX interaction with processCallback as the callback function  
}  
  
// callback for the request  
function processCallback() {  
  
// do post processing  
  
setTimeout("checkForMessage()", 10000);  
}  
  
Notice that the checkForMessage will continue to loop indefinitely. You may want to vary the increment the interval based on activity in the page or your use cases. You may also choose to have logic that would break out of the loop based on some AJAX response processing condition.

**When should I use an Java applet instead of AJAX?**

Applets provide a rich experience on the client side and there are many things they can do that an AJAX application cannot do, such as custom data streaming, graphic manipulation, threading, and advanced GUIs. While DHTML with the use of AJAX has been able to push the boundaries on what you can do on the client, there are some things that it just cannot do. The reason AJAX is so popular is that it only requires functionality built into the browser (namely DHTML and AJAX capabilities). The user does not need to download and/or configure plugins. It is easy to incrementally update functionality and know that that functionality will readily available, and there are not any complicated deployment issues. That said, AJAX-based functionality does need to take browser differences into consideration. This is why we recommend using a JavaScript library such as Dojo which abstracts browser differences. So the "bottom line" is: If you are creating advanced UIs where you need more advanced features on the client where you want UI accuracy down to the pixel, to do complex computations on the client, use specialized networking techniques, and where you know that the applet plugin is available for your target audience, applets are the way to go. AJAX/DHTML works well for applications where you know the users are using the latest generation of browsers, where DHTML/AJAX "good enough" for you, and where your developers have JavaScript/DHTML/AJAX skills. Many amazing things can be done with AJAX/DHTML but there are limitations. AJAX and applets can be used together in the same UIs with AJAX providing the basic structure and applets providing more advanced functionality. The Java can communicate to JavaScript using the Live-Connect APIs. The question should not be should framed as do I use AJAX or applets, but rather which technology makes the best sense for what you are doing. AJAX and applets do not have to be mutually exclusive.

**How do I handle the back and forward buttons in AJAX?**

While you could go out and create a custom solution that tracks the current state on your application I recommend you leave this to the experts. Dojo addresses the navigation in a browser neutral way as can be seen in the JavaScript example below.  
  
function updateOnServer(oldId, oldValue,  
itemId, itemValue) {  
var bindArgs = {  
url: "faces/ajax-dlabel-update",  
method: "post",  
content: {"component-id": itemId, "component-value":  
itemValue},  
mimetype: "text/xml",  
load: function(type, data) {  
processUpdateResponse(data);  
},  
backButton: function() {  
alert("old itemid was " + oldId);  
},  
forwardButton: function(){  
alert("forward we must go!");  
}  
};  
dojo.io.bind(bindArgs);  
}  
  
The example above will update a value on the server using dojo.io.bind() with a function as a property that is responsible for dealing with the browser back button event. As a developer you are capable of restoring the value to the oldValue or taking any other action that you see fit.  
  
AJAX: How to Handle Bookmarks and Back Buttons details this problem and provides a JavaScript library Really Simple History framework (RSH) that focuses just on the back and forward issue.

**What are the properties of the XmlHttpRequest object? What are the different types of readyStates in Ajax?**

i) onreadyStateChange - This function is used to process the reply from the web server.<br>ii) readyState - This property holds the response status of the web server. There are 5 states:<br>1 - request not yet initialized<br>2 - request now set<br>3 - request sent<br>4 - request processing<br>5 - request completes

**What is the difference between proxied and proxyless calls?**

Proxied calls are made through stub objects that mimic your PHP classes on the JavaScript side. E.g., the helloworld class from the Hello World example.  
Proxyless calls are made using utility javascript functions like HTML\_AJAX.replace() and HTML\_AJAX.append().

**How do I submit a form or a part of a form without a page refresh?**

When creating a form make sure that the "form" element "onSubmit" attribute is set to a JavaScript function that returns false.  
<form onSubmit="doAJAXSubmit();return false;" >  
<input type="text" id="tf1" />  
<input type="submit" id="submit1" value="Update"/>  
</>  
  
You can also submit data by associating a function with a form button in a similar way.  
  
<form onSubmit="doAJAXSubmit();return false;" >  
<input type="text" id="tf1" />  
<input type="button" id="button1" onClick="doAJAXSubmit()" value="Update"/>  
</>  
  
Note that the form "onSubmit" attribute is still set. If the user hits the enter key in the text field the form will be submitted so you still need to handle that case.  
When updating the page it is recommend you wait to make sure that the AJAX update of the form data was successful before updating the data in the page. Otherwise, the data may not properly update and the user may not know. I like to provide an informative message when doing a partial update and upon a successful AJAX interaction I will then update the page.

How to create an Object in JavaScript ?

1) var obj = new Object();   
2) var ob = {};

Write a way by which you can do something on the close of the window ?

call onUnload on the body tag and write your javascript code there   
  
e.g.

<body onUnload=''alert('thanks for visiting us !!')">

How can you set position of the page (Top and Left) to 0 using Javascript in one line code?

Well, there is an inbuilt function in JavaScript named **scroll()** which takes 2 arguments x and y.   
  
If you want to set top and left to 0 then call this.

window.scroll(0,0);

What is other equivalent option of document.getElementById() when you are working with Ajax?

The other equivalent option of document.getElementById() is **$get()** ;.   
  
For example.

var x = $get('<%= upnlScoping.ClientID %>');

var y = document.getElementById('<%= upnlScoping.ClientID %>');

Is it possible make a call to server side event of any button using javascript?

Yes, it's possible. You can use \_\_doPostBack() function to call server side event.

What is undefined variable?

Variables which are declared but not assigned any value is called undefined variable.

How to get the Scroll height and Width in JavaScript !!

Width : window.pageXOffset ||   
document.body.scrollLeft ||   
document.documentElement.scrollLeft;   
  
Height: window.pageYOffset ||   
document.body.scrollTop ||   
document.documentElement.scrollTop;

Can Javascript code be broken in different lines?

Yes,   
Breaking is possible within a string statement by using a backslash "\" at the end .   
Ex:   
document.write("Good Morning. \   
I am Mr. John");   
  
But it is not possible within any other javascript statement.   
Ex :   
is possible but not   
document.write \   
("Good Morning. I am Mr. John");

What is === operator in JavaScript and how it is different from == operator?

=== operator checks for the equality. It gives true when both the compared values are having same value and same data type As === does not perform any type conversion while comparing.   
  
It is different from == as == does the type conversion while comparing.   
  
Thanks,   
Virendra Dugar

What is undefined value means in JavaScript?

There can be multiple reasons of having undefined values   
1. Object does not exist. Like you create an object of a control which does not exists in your page and when you access it, it is undefined.   
2. No value is assigned to the object.   
3. If you are trying to access a property which does not exists for the object.

What is the result of below given line of code in Java Script? 5+4+'7'

The answer is 97.   
  
As 5 and 4 are integer so total becomes 9 and it's add '7' to it as string so it becomes 97.

I have regular expression for date: ^[0-1]{1,1}\d{1,1}/[0-3]{1,1}\d{1,1}/\d{4}$.   
what will be the correct output according to the above regular expression.

1. 20/10/2009
2. **10/20/2009**
3. 2009/10/20

Name the DataTypes of JavaScript?

**1)Integer**

**2)String**

**3)Boolean**

**4)Null**

Object can be assigned property by...

1. testobj.testproperty = "val"
2. testobj["testproperty"] = "val"
3. **Both**
4. none

Which method is used to Clear an array using JavaSctipt?

1. **Clear()**
2. Remove()
3. ClearAll()
4. Delete()

What will be the Output of the following using JavaScript   
document.Write(Math.Round(6.7));

1. **7**
2. 6
3. 6.7
4. 6.700

Which method is used to convert a string to uppercase letters?

1. **toUpperCase()**
2. UpperCase()
3. Upper()
4. all the above

String Operations:   
What will be the Output of the Following Code   
var ContentURL="http://MySite1/online-tests/1111-First-questions";   
ContentURL= ContentURL.substring(ContentURL.lastIndexOf("/")+1,ContentURL.length);   
var LastIndex;   
if(ContentURL.indexOf('-')>-1)   
LastIndex=ContentURL.indexOf('-');   
else   
LastIndex=ContentURL.length;   
ContentURL= ContentURL.substring(ContentURL.lastIndexOf("/")+1,LastIndex);

1. online
2. tests
3. **1111**
4. questions

What does the <noscript> tag do?

1. Stops execution of <JavaScript> on the page.
2. **Displayed the text enclosed within <noscript> tag when JavaScript is disabled.**
3. Shows that this page does not contain any script.
4. None of the above.

JavaScript is interpreted by \_\_\_\_\_\_\_\_\_

1. **Client**
2. Server
3. Object
4. None of the above

For generating random numbers we have to used the following function?

1. **Math.random()**
2. Math.round()
3. Math.ceilng
4. math.count()

Which method is used to clear an Array in javascript?

1. **Claea()**
2. push()
3. Append()
4. removeByElement()

Who Developed JavaScript?

1. **Netscape**
2. Microsoft
3. OPeara
4. Sun (java)

What does is Nan function do?

1. **Return true if the argument is not a number**
2. Return true if the argument is a number
3. Return true if the argument is divided by zero
4. Return true if the argument is a perfect square.

What Boolean operators are not supported by JavaScript?

1. && (AND)
2. || (OR)
3. ! (NOT)
4. **None of the above.**

How to get value from dropdown (select) control?

Write following code   
  
alert(document.getElementById('dropdown1').value);

How to toggle display an HTML element?

Call following function

function ToggleFollowingText(id)

{

document.getElementById(id).style.display == '' ? document.getElementById(id).style.display = 'none' : document.getElementById(id).style.display = '';

}

In above function you need to pass id of the html element as the parameter.   
If the control is already displaying then it will hide otherwise it will be shown.

How to get value from RadioButtonList control?

Here id is the name property of the RadioButtonList

function GetRadioButtonValue(id)

{

var radio = document.getElementsByName(id);

for (var ii = 0; ii < radio.length; ii++)

{

if (radio[ii].checked)

alert(radio[ii].value);

}

}

For more details, click http://www.dotnetfunda.com/articles/article72.aspx

How to get value from a textbox?

Write following code   
  
alert(document.getElementById('txtbox1').value);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Ajax

what is Ajax?

Ajax is asynchronous JavaScript and XML.it can use only those control in your page that post back to server side.it can avid UN usable control that can every time auto post back to server side.Ajax is time consuming .   
  
ex.In your web form 5 control only you have 2 control can auto post back every time at server side. you can add that control in update panel.

What role “#&&” plays in a querysting?

# works as a fragment delimeter in a querysting. With which you can delimit history state.   
While && preceeds, state information in query string.

Is there any property names “IsNavigating”?

Yes, it is available when you are managing browser history.   
With this property of “IsNavigating”, you can determine if postback is occurred for navigation purpose or for some other.   
Its set to true if its navigation call.

With IE, enabling history point will render additional,

1. Webpart
2. **iFrame**
3. table
4. DIV

Behavior varies with browser while enable history points.

Saving browser history point or navigating through page raises event,…

1. **Sys.Application.navigate**
2. Sys.Application.navigation
3. Sys.Application.navigatePoint
4. Sys.Application.navigationPoint

What is AjaxControl ToolKit?

AjaxControl Toolkit is an open source project built on top of ajax framework. It has more than 30 controls which are really cool and useful.   
  
Visit this site to find out more about AjaxControlToolKit.   
  
http://www.asp.net/(S(fu2l2uzphr2u3u45q2dnez55))/ajax/AjaxControlToolkit/Samples/

To create browser history point using client script, we make call method…

1. Sys.Application.createHistoryPoint
2. **Sys.Application.addHistoryPoint**
3. Sys.Application.saveHistoryPoint
4. Sys.Application.makeHistoryPoint

Default value of EnableHistory attribute in scriptmanager is,

1. True
2. **False**
3. User can’t set this
4. None

Process of browser history management consumes much memory and occupies more space. To make things simpler the default value is kept "False"