# AJAX

# What is Ajax, Introduction to Ajax

Ajax is not a programming language, its the script.  Ajax means, Asynchronous java script and xml, and this is the type of programming made popular in 2005 by Google suggest [Google search, while we are typing some thing in Google search box it will show you some suggestions automatically right that is Ajax]

With Ajax we can move the data from our web page to the server, and we can update the part of our web page with out refreshing the entire page, cool na :-)

And mates,  Ajax is the widely using scripting concept now a days in the real time projects.

And its highly recommended to know JavaScript and CSS, xml a little before we start Ajax programs

## What is Synchronous & Asynchronous

Actually, Synchronous means at a time we can send single request and we need to wait for the response before send the second request, and Asynchronous means we can send the second request before we get the response of first request, Ajax is the example of this Asynchronous type.

**Platform:** AJAX applications are platform independent, we can expect the exact working style in all the platforms :-)

# Ajax First Program With Explanation

<html>

<head>

<script type="text/javascript">

function fun1()

{

var a;

    if (window.XMLHttpRequest)

    {// If the browser if IE7+[or]Firefox[or]Chrome[or]Opera[or]Safari

      a=new XMLHttpRequest();

    }

   else

    {//If browser is IE6, IE5

      a=new ActiveXObject("Microsoft.XMLHTTP");

    }

a.onreadystatechange=function()

{

  if (a.readyState==4 && a.status==200)

  {

    document.getElementById("myDiv").innerHTML=a.responseText;

   }

}

a.open("POST","java4s.txt",true);

a.send();

} // fun1() close

</script>

</head>

<body>

<div id="myDiv" style="width: 300px; height: 30px;">Click on the button below</div>

<button type="button" onclick="fun1()">Change Content</button>

</body>

</html>

## ****Explanation****

* Once the document loaded then immediately we will be able to see one button **Change Content**, and see i have been given onclick=”fun1()” [ line number 33 ] means once we click on this button controller will go to line number 4 and will starts execute that fun1()
* At line number 6, i have taken one variable with name **a**
* For any ajax program, we must create one request object to send our ajax request to the server, that ajax object is nothing but XMLHttpRequest object
* See line number 8,  i have written window.XMLHttpRequest means am checking whether my browser supports XMLHttpRequest object or not, if yes assigning XMLHttpRequest object into a [ a=new XMLHttpRequest(); ] else i mean if our web browser doesnt supports XMLHttpRequest object then we can assign ActiveXObject   [ which supports old web browsers ] into our variable a [  a=new ActiveXObject(“Microsoft.XMLHTTP”); ]
* So from line numbers 8 -15 request object creation work been done
* Now controller directly jumps to line number 25,  and opens the connection and send the request to java4s.txt (this is my notepad file), see actually the 3rd parameter i have given as **true** means Asynchronous data transfer will be activated
* Finally request will be sent in line number 26, that’s it.
* If server sends the response back to our application then controller will automatically execute from line number 17 – 23, you may get one doubt like why its not executed initially… ? there is a reason actually this a.onreadystatechange=function() executes only when we get the response from the server, hope you are clear ;)

## java4s.txt

welcome to java4s.com

That’s it mates just put these **2** files, in one folder and test the example of your own, and i will explain more about this open() and send() methods in the next sessions, just don’t think much about this :-)

# Ajax Request, open() and send() methods

In Ajax if we want to send the request to the server,  we have 2 methods in XMLHttpRequest object to do this work,  those are open() and send().

Actually open() method will opens the connection with the server and send will sends our request object to the server.

Let we took the XMLHttpRequest object into one variable called **obj** then

**obj**.open(” POST “, ” destination URL “,true);  
**obj**.send();

* open() method has **3** parameters, first parameter having the values POST/GET this is depends on our requirement, my choice is always POST, because its having security than GET
* Second parameter is the destination, to where we need to send the request.  It may be any file path or url or url patterns [ in java ] or what ever
* Third parameter having the values true/false,  actually true means we are opening Asynchronous data transfer, and false means Synchronous

Actually, Synchronous means at a time we can send single request and we need to wait for the response before send the second request, and Asynchronous means we can send the second request before we get the response of first request, Ajax is the example of this Asynchronous type.

Finally send() method will send the request object to the server.

# Ajax Server Response, responseText and responseXML

Once we send the request to the destination [ server ], we will get the response from the server in two formats either in Text or XML, i mean we can get the response from the server by using responseText and responseXML propertys of our XMLHttpRequest object

<html>

<head>

<script type="text/javascript">

function fun1()

{

var a;

    if (window.XMLHttpRequest)

    {// If the browser if IE7+[or]Firefox[or]Chrome[or]Opera[or]Safari

      a=new XMLHttpRequest();

    }

   else

    {//If browser is IE6, IE5

      a=new ActiveXObject("Microsoft.XMLHTTP");

    }

a.onreadystatechange=function()

{

  if (a.readyState==4 && a.status==200)

  {

    document.getElementById("myDiv").innerHTML=a.responseText;

   }

}

a.open("POST","my.txt",true);

a.send();

} // fun1() close

</script>

</head>

<body>

<div id="myDiv" style="width: 300px; height: 30px;">

Click on the button below

</div>

<button type="button" onclick="fun1()">Change Content</button>

</body>

</html>

## Explanation

* see we are getting the response in the form of text [ see line number **21** ]
* once the response is ready, our div content will be replaced with the response text [ see line number **32** ]

Will see more while discussing Ajax with Servlets

# Ajax onreadystatechange Event Of XMLHttpRequest Object

Once the request been sent to the server,  changes will happen in the current webpage based on the response only.  The onreadystatechange event will be triggered every time the readyState changes.

Here the properties of XMLHttpRequest object

## onreadystatechange

A function will be stored and will be called automatically each time readyState property changes

## readyState

0 – request not initialized  
1 – connection been established with the server  
2 – request received from the server  
3 – request being processing  
4 – request finished and response is ready

## status

200 : ok  
404 : page not found, it’s normal 404 error

So, readyState = 4  **&** status = 200 then every thing will be perfect, i mean we got the response from the server, and  our webpage will be modified with the response text/xml.

# Ajax Onchange Fetch The Data From The Database

Let us see how to fetch the data from database onchange of drop down, actually this is the real time scenario, am using jsp  you can integrate with any type of frame works ( in .java files ), concept is same.

## Index.html

<html>

<head>

<script type="text/javascript">

function loadXMLDoc()

{

var xmlhttp;

var keys=document.dummy.sele.value

var urls="http://www.java4s.com:2011/On\_select\_from\_database\_dropdown/db\_fetch.jsp?ok="+keys

if (window.XMLHttpRequest)

{// code for IE7+, Firefox, Chrome, Opera, Safari

xmlhttp=new XMLHttpRequest();

}

else

{// code for IE6, IE5

xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");

}

xmlhttp.onreadystatechange=function()

{

if (xmlhttp.readyState==4)

{

var some=xmlhttp.responseXML.documentElement;

document.getElementById("a").innerHTML=some.getElementsByTagName("empno")[0].childNodes[0].nodeValue;

document.getElementById("b").innerHTML=some.getElementsByTagName("empname")[0].childNodes[0].nodeValue;

document.getElementById("c").innerHTML=some.getElementsByTagName("empaddr")[0].childNodes[0].nodeValue;

}

}

xmlhttp.open("GET",urls,true);

xmlhttp.send();

}

</script>

</head>

<body>

<form name="dummy">

<select name="sele" onchange="loadXMLDoc()">

<option>value</option>

<option value="100">100</option>

<option value="101">101</option>

</select>

</form>

id: <span id="a"></span><br>

name: <span id="b"></span><br>

address: <span id="c"></span>

</body>

</html>

## db\_fetch.jsp

<%@ page import="java.io.\*,java.sql.\*" %>

<%@ page contentType="text/html" pageEncoding="UTF-8"%>

<%

response.setContentType("text/xml");

String sn=request.getParameter("ok");

int i=Integer.parseInt(sn);

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con =DriverManager.getConnection("jdbc:oracle:thin:@www.java4s.com:1521:XE","system","admin");

Statement st=con.createStatement();

ResultSet rs = st.executeQuery("select \* from emp where empno="+i);

if(rs.next())

{

out.println("<emp>");

out.println("<empno>"+rs.getInt(1)+"</empno>");

out.println("<empname>"+rs.getString(2)+"</empname>");

out.println("<empaddr>"+rs.getString(3)+"</empaddr>");

out.println("</emp>");

}

rs.close();

st.close();

con.close();

%>

# Country State Dropdown Example With AJAX/Servlets

Folks this is similar to the country/state drop down using Ajax, i believe we have lot of examples regarding this country/state drop down so just would like to do little changes, so i converted into employee number and name. But total concept is exactly same…!

## Index.html

<html>

<head>

<script type="text/javascript">

function loadXMLDoc()

{

var xmlhttp;

var keys=document.dummy.sele.value

var urls="http://www.java4s.com:2011/On\_select\_from\_database\_dropdown\_Update\_2/db\_fetch.jsp?ok="+keys

if (window.XMLHttpRequest)

  {

  xmlhttp=new XMLHttpRequest();

  }

else

  {

  xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");

  }

xmlhttp.onreadystatechange=function()

  {

removeall();

  if (xmlhttp.readyState==4)

    {

            z=0;

            var roott=xmlhttp.responseXML.documentElement;

            var ress=roott.getElementsByTagName("empname")[z].childNodes[0].nodeValue;

            while(ress!=null)

            {

                    addoptions(ress)

                    z++

             var ress=roott.getElementsByTagName("empname")[z].childNodes[0].nodeValue;

            }

    }

    function removeall()

    {

        var ct=document.dummy.sele2.length;

        for(i=ct; i>=0; i--)    {

            document.dummy.sele2.options[i]=null;

             }

    }

    function addoptions(reslt)

    {

        var ct1=document.createElement("OPTION");

        ct1.text=reslt;

        ct1.value=reslt;

        document.dummy.sele2.options.add(ct1);

    }

}

xmlhttp.open("GET",urls,true);

xmlhttp.send();

}

</script>

</head>

<body>

<form name="dummy">

<font face="verdana" size="2">

Employee Number:    <select name="sele" onchange="loadXMLDoc()">

            <option>select</option>

            <option value="100">100</option>

            <option value="101">101</option>

    </select>

Employee Name:    <select name="sele2">

            <option>--choose--</option>

    </select>

</form>

</body>

</html>

## db\_fetch.jsp

<%@ page import="java.io.\*,java.sql.\*" %>

<%@ page contentType="text/html" pageEncoding="UTF-8"%>

<%

            response.setContentType("text/xml");

            String sn=request.getParameter("ok");

            int i=Integer.parseInt(sn);

                    Class.forName("oracle.jdbc.driver.OracleDriver");

                    Connection con =DriverManager.getConnection("jdbc:oracle:thin:@www.java4s.com:1521:XE","system","admin");

                    Statement st=con.createStatement();

                    ResultSet rs = st.executeQuery("select empname from emp where empno="+i);

                    out.println("<emp>");

                    while(rs.next())

                    {

                        out.println("<empname>"+rs.getString(1)+"</empname>");

                    }

                    out.println("</emp>");

rs.close();

st.close();

con.close();

%>

# Ajax,jQuery Auto Fill Dropdown On Page Load

## Files Required

* index.html
* db\_fetch.jsp
* jquery-1.2.6.js (Add this jquery file at db\_fetch.jsp location, same  folder)
* web.xml

## Index.html

<html>

<head>

<script src="jquery-1.2.6.js" type="text/javascript"></script>

<script type="text/javascript">

$(document).ready(function(){

    var xmlhttp;

    var urls="http://www.java4s.com:2011/On\_select\_from\_database\_dropdown\_jquery/db\_fetch.jsp"

    if (window.XMLHttpRequest)

    {

         xmlhttp=new XMLHttpRequest();

    }

    else

    {

        xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");

    }

    xmlhttp.onreadystatechange=function()

   {

      z=0;

      removeall();

    if (xmlhttp.readyState==4)

    {

            var roott=xmlhttp.responseXML.documentElement;

            var ress=roott.getElementsByTagName("empname")[z].childNodes[0].nodeValue;

            while(ress!=null)

            {

                    addoptions(ress)

                    z++

              var ress=roott.getElementsByTagName("empname")[z].childNodes[0].nodeValue;

            }

     }

    function removeall()

    {

        var ct=document.dummy.sele2.length;

        for(i=ct; i>=0; i--)    {

            document.dummy.sele2.options[i]=null;

             }

    }

    function addoptions(reslt)

    {

        var ct1=document.createElement("OPTION");

        ct1.text=reslt;

        ct1.value=reslt;

        document.dummy.sele2.options.add(ct1);

    }

 }    //onreadystatechnge function end

xmlhttp.open("GET",urls,true);

xmlhttp.send();

});

</script>

</head>

<body>

<form name="dummy">

EName:    <select name="sele2">

            <option>--choose--</option>

    </select>

</form>

</body>

</html>

**Notes**: See line number 6 in index.html ,$(document).ready(function() means we no need to send any request like onchange or onclick, once the page loads automatically the code in this $(document).ready(function() block will be executed, so indirectly we are calling this function on page load hope you are clear.

## db\_fetch.jsp

<%@ page import="java.io.\*,java.sql.\*" %>

<%@ page contentType="text/html" pageEncoding="UTF-8"%>

<%

            response.setContentType("text/xml");

                    Class.forName("oracle.jdbc.driver.OracleDriver");

                    Connection con =DriverManager.getConnection("jdbc:oracle:thin:@www.java4s.com:1521:XE","system","admin");

                    Statement st=con.createStatement();

                    ResultSet rs = st.executeQuery("select empname from emp");

                    out.println("<emp>");

                    while(rs.next())

                    {

                        out.println("<empname>"+rs.getString(1)+"</empname>");

                    }

                    out.println("</emp>");

rs.close();

st.close();

con.close();

%>

# Checking User Name Availability With AJAX – Google Style

We will see how to check the user name availability in the database with ajax,  this is some thing what we can find out at time of gmail signup

## Index.html

<html>

<head>

<script type="text/javascript">

function loadXMLDoc()

{

var xmlhttp;

var k=document.getElementById("id2").value;

var urls="pass.jsp?ver="+k;

if (window.XMLHttpRequest)

  {

  xmlhttp=new XMLHttpRequest();

  }

else

  {

  xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");

  }

xmlhttp.onreadystatechange=function()

  {

  if (xmlhttp.readyState==4)

    {

        //document.getElementById("err").style.color="red";

        document.getElementById("err").innerHTML=xmlhttp.responseText;

    }

  }

xmlhttp.open("GET",urls,true);

xmlhttp.send();

}

</script>

</head>

<body>

Enter name: <input type="text" name="id2" id="id2" onkeyup="loadXMLDoc()">

<span id="err"> </span>

</body>

</html>

## pass.jsp

<%@ page import="java.io.\*,java.sql.\*" %>

<%@ page contentType="text/html" pageEncoding="UTF-8"%>

<%

            String sn=request.getParameter("ver");

                    Class.forName("oracle.jdbc.driver.OracleDriver");

                    Connection con =DriverManager.getConnection("jdbc:oracle:thin:@www.java4s.com:1521:XE","system","admin");

                    Statement st=con.createStatement();

                    //ResultSet rs = st.executeQuery("select \* from emp where empno="+sn);

                    ResultSet rs = st.executeQuery("select \* from emp where empname='"+sn+"'");  // this is for name

                    if(rs.next())

                    {

                        out.println("<font color=red>");

                        out.println("Name already taken");

                        out.println("</font>");

                    }else {

                        out.println("<font color=green>");

                        out.println("Available");

                        out.println("</font>");

                    }

rs.close();

st.close();

con.close();

%>