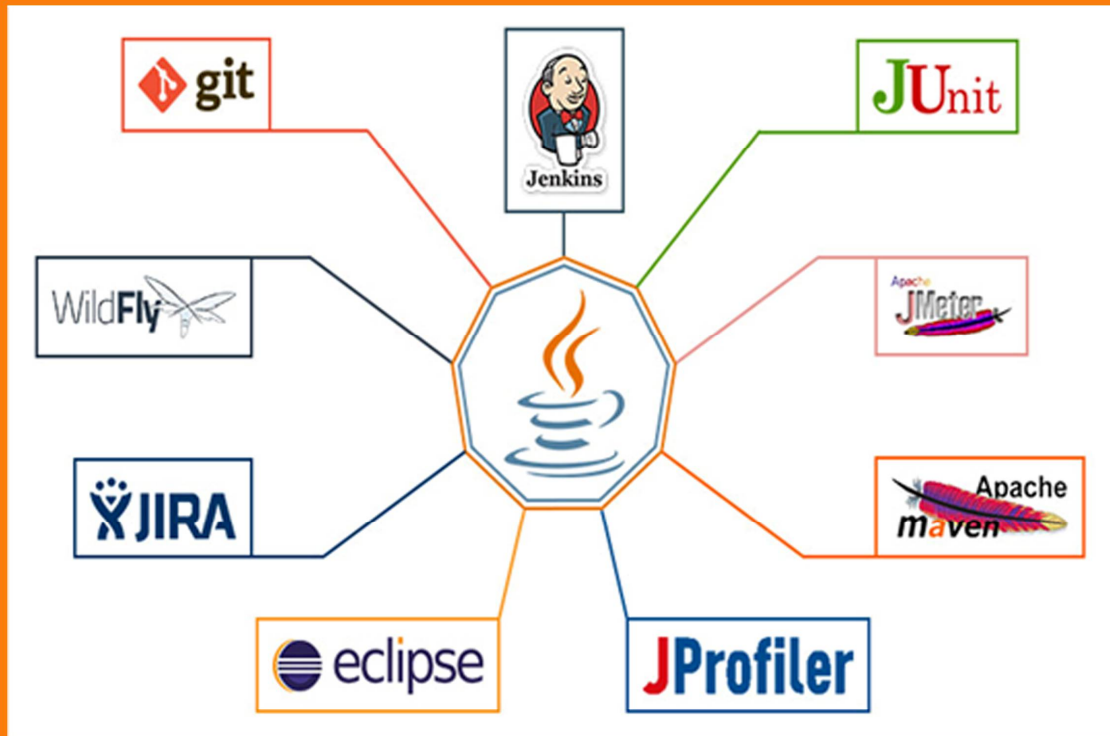


**JAVA means DURGA SOFT**

# Java Real Time Tools

## Ajax



India's No.1 Software Training Institute

# DURGASOFT

[www.durgasoft.com](http://www.durgasoft.com) Ph: 9246212143 ,8096969696

## AJAX

### Introduction to AJAX:

Ajax stands for Asynchronous JavaScript and XML.

AJAX is not a new programming language, but a new way to use existing standards.

AJAX is the art of exchanging data with a server, and update parts of a web page - without reloading the whole page.

AJAX is about updating parts of a web page, without reloading the whole page.

AJAX applications are browser- and platform-independent.

Ajax can be used for creating rich, web-based applications that look and works like a desktop application

With AJAX you can create better, faster, and more user-friendly web applications.

AJAX is based on java script and HTTP requests.

AJAX is a type of programming made popular in 2005 by Google(with Google Suggest).

Ajax is the use of non standard XMLHttpRequest object to communicate with server side scripts. It can send as well as receive information in a variety of formats including XML, HTML, and even text files. Ajax's most appealing characteristic, however, is its "asynchronous" nature, which means that it can do all of this without having to refresh the page. this allows you to update portions of a page based upon user events.

**LEARN FROM EXPERT & DIAMOND FACULTIES OF AMEERPET...**  
**JAVA MEANS DURGASOFT**  
**INDIA'S NO. 1 SOFTWARE TRAINING INSTITUTE**

AN ISO 9001:2008 CERTIFIED  
**DURGA**  
SOFTWARE SOLUTIONS

#202 2<sup>nd</sup> FLOOR  
www.durgasoft.com

040-64512786  
+91 9246212143  
+91 8096969696

## What is AJAX?

Ajax is a set of technologies, supported by a web browser, including these elements:

- HTML and CSS for presenting.
- JavaScript (ECMAScript) for local processing, and DOM (Document Object Model) to access data inside the page or to access elements of XML file read on the server (with the getElementByTagName method for example)
- The XMLHttpRequest class reads or sends data on the server asynchronously.

Optionally...

- The DomParser class may be used
- JSP or another scripting language may be used
- XML and XSLT to process the data if returned in XML form.
- 
- SOAP may be used to dialog with the server.

The "Asynchronous" word, means that the response of the server while be processed when available, without to wait and to freeze the display of the page.

AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.



**Java Real Time Tools**

**JAVA TOOLS** Means **DURGASOFT**  
Multiple Faculty Members only For **JAVA TOOLS**

**Online Training**      **Class Room Training**

**DURGA SOFTWARE SOLUTIONS**  
www.durgasoftonlinetraining.com    durgasoftonlinetraining@gmail.com    Ph: +91- 8885252627 +91- 7207212428

## **Why to use Ajax?**

It is used mainly to build a fast, dynamic website, but also to save resources.

For improving sharing of resources, it is better to use the power of all the client computers rather than just an unique server and network. Ajax allows performing processing on client computer (in JavaScript) with data taken from the server.

The processing of web page formerly was only server-side, using web services or PHP scripts, before the whole page was sent within the network.

But Ajax can selectively modify a part of a page displayed by the browser, and update it without the need to reload the whole document with all images, menus, etc...

For example, fields of forms, choices of user, may be processed and the result displayed immediately into the same page.

The main purpose of Ajax is to provide a simple and standard means for a webpage to communicate with the server without a complete page refresh

To illustrate the above scenario in detail, consider a simple registration form. You have very likely experienced the frustration of having to try multiple user names when registering for some new website. You fill out entire form, hit the submit button, wait for the second or so, and then get the same form right back with a message saying that the user name you have entered is not available. You try another easy to remember username and find it is also not available. You repeat the same process for several times until you pick some obscure username.

The above process wouldn't be nearly as bad if you didn't have to wait for the entire page to refresh each time you tried a new username.

The above problem can be resolved by using AJAX.



**www.durgasoftonlinetraining.com**

**Online Training**  
**Pre Recorded Video**  
**Classes Training**  
**Corporate Training**

**Ph: +91-8885252627, 7207212427**  
**+91-7207212428**

 **USA Ph : 4433326786**

**E-mail : durgasoftonlinetraining@gmail.com**

### How does AJAX work?

Ajax uses a programming model with display and events. These events are user actions they call functions associated to elements of the web page.

Interactivity is achieved with forms and buttons. DOM allows to link elements of the page with actions and also to extract data from XML files provided by the server.

The keystone of AJAX is the **XMLHttpRequest** object.

#### **XMLHttpRequest Object:**

All modern browsers support the **XMLHttpRequest** object (IE5 and IE6 uses an **ActiveXObject**).

The **XMLHttpRequest** object is used to exchange data with a server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.

#### **How to create an XMLHttpRequest Object?**

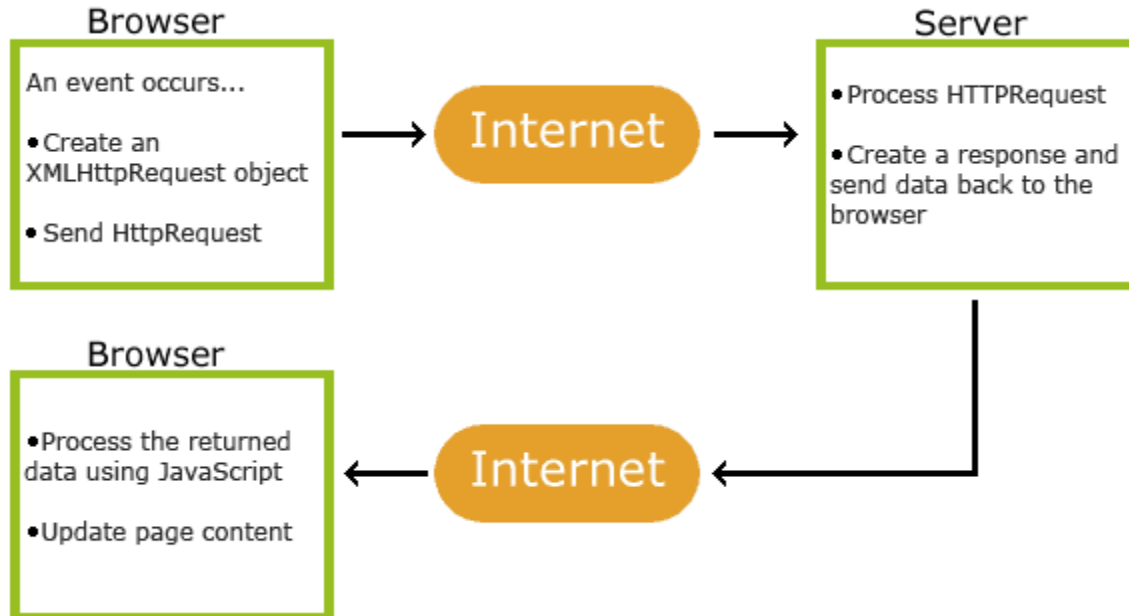
All modern browsers (**IE7+, Firefox, Chrome, Safari, and Opera**) have a built-in **XMLHttpRequest** object.

Syntax for creating an **XMLHttpRequest** object:

```
variable=new XMLHttpRequest();
```

Old versions of Internet Explorer (IE5 and IE6) use an ActiveX Object:

```
variable=new ActiveXObject("Microsoft.XMLHTTP");
```



To work with Ajax, XMLHttpRequest object (**Ajax engine**) provides us **Four methods** and **Five Properties**.

The two methods provided by XMLHttpRequest object are:

1. open ("GET/POST", url, true/false)
2. send () or send (String)
3. setRequestHeader (header\_name,header\_value)
4. overrideMimeType("text/xml");

**FREE TRAINING VIDEOS**  
**You Tube** **3000+**  
**VIDEOS**  
[www.youtube.com/durgasoftware](http://www.youtube.com/durgasoftware)

The following tabular form gives the description of the above two methods.

Method	Description
<b>open(http_method, url, async)</b>	Specifies the type of request, the URL, and if the request should be handled asynchronously or not.  <b>method:</b> the type of request: <b>GET</b> or <b>POST</b> <b>url:</b> the location of the file on the server <b>async:</b> <b>true</b> (asynchronous) or <b>false</b> (synchronous)
<b>send(string)</b>	Sends the request off to the server.  <b>string:</b> Only used for <b>POST</b> requests
<b>setRequestHeader(header, value)</b>	Adds HTTP headers to the request.  <b>header:</b> specifies the header name <b>value:</b> specifies the header value

### GET or POST?

GET is simpler and faster than POST, and can be used in most cases.

However, always use POST requests when:

- A cached file is not an option (update a file or database on the server)
- Sending a large amount of data to the server (POST has no size limitations)

## JAVA Means DURGA SOFT

- Sending user input (which can contain unknown characters), POST is more robust and secure than GET

Ex:- A simple GET request is shown below.

```
xmlhttp.open("GET","demo_get.jsp",true);  
xmlhttp.send();
```

If you want to send information with the GET method, add the information to the URL

```
xmlhttp.open("GET","demo_get2.jsp?fname=sachin&lname=Tendulkar", true);  
xmlhttp.send();
```

The Ajax given 5 properties are listed below:

- 1. onreadystatechange**
- 2. readyState**
- 3.status**
- 4. responseText**
- 5. responseXML**

### **1. onreadystatechange:**

This property holds (or) contains the name of the javascript function to be called whenever response from the server is received by the Ajax engine.



## JAVA Means DURGA SOFT

While working with Ajax, we need to take the support of minimum of two java script functions

One for sending the request to the server and the other for receiving response from the server.

Ex:-

```
function fun1()
{
  var x=new XMLHttpRequest("Microsoft.XMLHTTP");
  x.onreadystatechange=fun2;
  -----
  x.open (-,-,-);
  x.send ();
}
```



**www.durgajobs.com**  
*Continuous Job Updates for every hour*

**Fresher Jobs**   **Govt Jobs**   **Bank Jobs**  
**Walk-ins**   **Placement Papers**   **IT Jobs**  
**Interview Experiences**

*Complete Job information across India*

### 2. readyState:

- ❖ This property holds the status of the request
- ❖ The values of this property will be dynamically changed.
- ❖ readyState property values are varied from 0 to 4.

**0: request not initialized:** - means that **AJAX engine is created**,  
But **open (-,-,-)** is not called.

- 1: **server connection established:** - Means that **open (-,-,-)** is called.
- 2: **request received:** - means that **send ()** is called
- 3: **processing request:** - Means that request is under processing.
- 4: **request finished and response is ready:** Means that response for the Request is generated by the server.

### 3. status:

**status** attribute holds different status codes to know the status of the response returned by the server

#### The following status codes represent different states of the response:

- **100-199:** represents **info**
- **200-299:** represents **success response**
- **300-399:** represents **redirection**
- **400-499:** represents **incomplete web resource**
- **500-599:** represents **server error**

**Note:** When **readyState** is **4** and **status** is **200**, the response is ready

### 4. responseText:

- This property holds the response data returned by the server for the given request.
- This property holds data(response) if the MIME type of the response is **Other than "text/xml"**.

### 5.responseXML:

- This property holds the response generated by the server for the given request.
- This property holds the data(response) only if the MIME type is **"text/xml"**.

Procedure to develop Ajax code to interact with the web resource program of web application:

**Step 1: make web page generate javascript call based request.**

**Ex: Enter country:**`<input type="text" name="t1" onKeyUp="f1()"/>`

When the key of keyboard is pressed and released to type a letter in the text box, “**onKeyUp**” event is raised and it calls the javascript function “**f1()**”.

## Java Real Time Tools



**JAVA TOOLS** Means **DURGASOFT**  
Multiple Faculty Members only For **JAVA TOOLS**  
**Online Training**      **Class Room Training**

## **DURGA SOFTWARE SOLUTIONS**

[www.durgasoftonlinetraining.com](http://www.durgasoftonlinetraining.com) [durgasoftonlinetraining@gmail.com](mailto:durgasoftonlinetraining@gmail.com) Ph: +91- 8885252627 +91-7207212428

**Step2: keep Ajax engine ready in the JavaScript function definition.**

```
function f1 ()
{
    if (window.XMLHttpRequest) // Object of the current window
    {
        var aeng = new XMLHttpRequest(); // code for IE7+, Firefox,
```

```
    }  
else if (window.ActiveXObject) // ActiveX version  
{  
    var aeng = new ActiveXObject("Microsoft.XMLHTTP"); // code for  
                                                    IE6, IE5  
}  
-----  
-----  
}
```

**Step3: specify the name of the call back function to Ajax engine .**

**aeng.onreadystatechange=connection;** ---> **connection** is a callback  
Function

The function that executes automatically based on the events that are raised is called **callback function**.

The above specified “**connection**” JavaScript function will be called automatically by Ajax engine for each change that takes place in **readyState** property value like 0-1, 1-2, 2-3, 3-4.

In the definition of callback function, we keep the logic to write the response received by Ajax engine to webpage, but we make that logic execute only when **readyState** property value is “4”.

**Step4: prepare request url having query string.**

```
//read form data  
var strng=document.f1.t1.value;  
// frame request url  
var url="test.jsp?q="+strng;
```

**Step5: make Ajax engine send request to target web resource program  
Asynchronously.**

aeng.open (“GET”, url, true);

**true:** makes Ajax engine send **asynchronous request** to the target web

resource program

**false:** makes Ajax engine send **synchronous request** to the target web resource program

**Step6: develop target web resource program like servlet or jsp in the Web application of server to process the request and generate the response.**

**Step7: define callback function definition (java script function) having logic to update webpage content when Ajax engine's "responseState" is "4" and http response "status" code is "200".**

Ex:

```
function connection()
{
  If(aeng.readyState==4 || aeng.readyState=="complete")
  {
    If(aeng.status==200)
    {
      -----
      -----
    }
  }
}
```

**Ajax code to activate Ajax engine having cross-browser compatibility:**

**var aengg;**

**if (window.XMLHttpRequest) // code for IE7+, Firefox, Chrome, Opera, Safari**

```
{
  aeng = new XMLHttpRequest();
}
```

**else if (window.ActiveXObject) // code for IE6-**

```
{
  try{
    aeng = new ActiveXObject("Msxml2.XMLHTTP"); //for IE4+ to IE6
  }
```

```
  catch (Exception e)
```

```
  {
    aeng= new ActiveXObject("Microsoft.XMLHTTP");//for old IE
  }
```

```

    }
}

```

### Example Application 1:

The given below example describes how exactly Ajax works. And also it displays all the matching countries while you type characters in the text box provided on the form page of this application. In the process of displaying all the matching countries, it asynchronously communicates with the server and updates the page without causing the complete page reloading.

#### index.html

```

<html>
<head>
<script>
    var xh;
function fun1()
{
    if(window.ActiveXObject)
    {
        xh=new ActiveXObject("Microsoft.XMLHTTP"); // code for IE6, IE5
    }
else if(window.XMLHttpRequest)
{
    xh=new XMLHttpRequest();// code for IE7+, Firefox, Chrome, Opera, Safari
}

    strng =document.f1.country.value;
    alert(strng);
    xh.onreadystatechange=connection;
    xh.open("GET" , "test.jsp?q="+strng , true);

```

## JAVA Means DURGA SOFT

```
xh.send();
}

function connection()
{
    if(xh.readyState==4)
        {
            document.getElementById("d1").innerHTML=xh.responseText;
        }
}
</script>
</head>
<body>
    <form name="f1">
        country:<input type="text" name="country" onkeyup="fun1()"/>
    </form>
    <div id="d1"></div>
</body>
</html>
```

### **test.jsp**

```
<%@ page import="java.util.*" %>
<%
    System.out.println("entered into jsp");
    ArrayList choices=new ArrayList();
    String input=request.getParameter("q");
    System.out.println(input);
    Locale[] locales=Locale.getAvailableLocales();
    System.out.println("locales created.....");
    for(int i=0;i<locales.length;i++)
```

## JAVA Means DURGA SOFT

```
{  
  
    Locale locale=locales[i];  
  
    String country=locale.getDisplayCountry();  
  
    if(country.toUpperCase().startsWith(input.toUpperCase()))  
  
        choices.add(country);  
  
}  
  
Iterator it=choices.iterator();  
  
while(it.hasNext())  
  
{  
  
    String str=it.next().toString();  
  
    out.println(str+"<br>");  
  
    System.out.println("in while loop");  
  
}  
  
>
```

### Example Application2:

This example is also about updating the current webpage without reloading the entire webpage by taking the support of Ajax. In this example, when you select your state, all the districts belonging to that state will be added dynamically to the second select box. And also when you select your district all your mandals belonging to your selected district will be dynamically added to the third select box

**LEARN FROM EXPERT & DIAMOND FACULTIES OF AMEERPET...**  
**JAVA MEANS DURGASOFT**  
**INDIA'S NO. 1 SOFTWARE TRAINING INSTITUTE**

AN ISO 9001:2008 CERTIFIED  
**DURGA**  
SOFTWARE SOLUTIONS

# 202 2<sup>nd</sup> FLOOR  
www.durgasoft.com

040-64512786  
+91 9246212143  
+91 8096969696



**Index.html:**

```
<html>
<head>
<script language="javascript">
var ajaxeng;
function fun1()
{
    if(window.ActiveXObject)
    {
        ajaxeng=new ActiveXObject("Microsoft.XMLHTTP");
    }
    else if(window.XMLHttpRequest)
    {
        ajaxeng=new XMLHttpRequest();
    }
    strng =document.f1.sel1.value;
    ajaxeng.onreadystatechange=fun2;
    ajaxeng.open("GET","test.jsp?p1=" + strng , true);
    ajaxeng.send();
}

function fun2()
{
    var z=0;
    removeAll();
    if(ajaxeng.readyState==4 )
    {
        var xml=ajaxeng.responseXML;
        var m=xml.documentElement;
```

## JAVA Means DURGA SOFT

```
var res=m.getElementsByTagName("dname")[z].childNodes[0].nodeValue;
while(res!=null)
{
    addOption(res);
    z++;
    res= m.getElementsByTagName("dname")[z].childNodes[0].nodeValue;
}
}
```

```
function removeAll()
{
    var x1=document.f1.sel2.length;
    for(i=x1;i>=0;i--)
    {
        document.f1.sel2.options[i]=null;
    }
}
```

```
function addOption(result)
{
    var opt=document.createElement("OPTION");
    opt.text=result;
    opt.value=result;
    document.f1.sel2.options.add(opt);
}
```

```
function fun3()
{
    if(window.ActiveXObject)
```

# Java Real Time Tools



**JAVA TOOLS** Means **DURGASOFT**

Multiple Faculty Members only For **JAVA TOOLS**

**Online Training**

**Class Room Training**

## **DURGA SOFTWARE SOLUTIONS**

[www.durgasoftonlinetraining.com](http://www.durgasoftonlinetraining.com) [durgasoftonlinetraining@gmail.com](mailto:durgasoftonlinetraining@gmail.com) Ph: +91-8885252627 +91-7207212428

```
{
    ajaxeng=new XMLHttpRequest("Microsoft.XMLHTTP");
}
else if(window.XMLHttpRequest)
{
    ajaxeng=new XMLHttpRequest();
}

strng =document.f1.sel2.value;

alert("name="+strng);

ajaxeng.onreadystatechange=fun4;
ajaxeng.open("GET","test1.jsp?p1=" + strng , true);
ajaxeng.send();
}

function fun4()
{
    var z=0;
    removeAll2();
    if(ajaxeng.readyState==4 )
    {
        var xml=ajaxeng.responseXML;
        varm=xml.documentElement;
        var res=m.getElementsByTagName("mname")[z].childNodes[0].nodeValue;
        while(res!=null)
        {
            addOption2(res);
            z++;
            res=m.getElementsByTagName("mname")[z].childNodes[0].nodeValue;
        }
    }
}
```

```
    }  
  }  
}
```

```
function removeAll2()
```

```
{  
  var x1=document.f1.sel3.length;  
  for(i=x1;i>=0;i--)  
  {  
    document.f1.sel3.options[i]=null;  
  }  
}
```

```
function addOption2(result)
```

```
{  
  var opt=document.createElement("OPTION");  
  opt.text=result;  
  opt.value=result;  
  document.f1.sel3.options.add(opt);  
}
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<form name="f1">
```

```
Select your State:&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<select name="sel1" onchange="fun1()" >
```

```
<option value="10">Andhr Pradesh</option>
```

```
<option value="20" selected>Tamil Nadu</option>
```

```
<option value="30">Kerala</option>
```

```
<option value="40">Karnataka</option>
```

## JAVA Means DURGA SOFT

```
        </select></br></br>Select your  
District:&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<select name="sel2" onchange="fun3()">  
        <option>-----select-----</option>  
    </select></br></br>  
Mandals of your selected Distict:<select name="sel3">  
        <option>-----select-----</option>  
    </select>  
  
</body>  
</html>
```



### Test.jsp:

```
%@ page import="java.sql.*" %>  
<%  
    try  
    {  
        response.setContentType("text/xml");  
        String select=request.getParameter("p1");  
        int v=Integer.parseInt(select);  
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");  
        Connection con=DriverManager.getConnection("jdbc:odbc:ajax","ajax","ajax");  
        Statement st=con.createStatement();  
        ResultSet rs=st.executeQuery("select dname from Districts where distid="+v);
```

## JAVA Means DURGA SOFT

```
out.println("<root>");
while(rs.next())
{
    out.println("<lname>"+rs.getString(1)+"</lname>");
}
out.println("</root>");
con.close();
}catch(Exception e)
{
    e.printStackTrace();
}
%>
```

### **test1.jsp:**

```
<%@ page import="java.sql.*" %>
<%
try
{
    response.setContentType("text/xml");
    String select=request.getParameter("p1");
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    Connection con=DriverManager.getConnection("jdbc:odbc:ajax","ajax","ajax");
    Statement st=con.createStatement();
    ResultSet rs=st.executeQuery("select lname from Mandals where
                                lname='"+select+"'");

    out.println("<root>");

    while(rs.next())
    {
        out.println("<lname>"+rs.getString(1)+"</lname>");
    }
    out.println("</root>");
    con.close();
}
catch(Exception e)
{
    e.printStackTrace();
}
%>
```

**Example Application3:**

This example describes what exactly asynchronous communication is and how that can be achieved with the help of Ajax.

**Links.html:**

```
<html>
<head>
<script language="javascript">
var xh1,xh2;
function fun1()
{
    if(window.ActiveXObject)
    {
        xh1=new ActiveXObject("Microsoft.XMLHTTP");
    }
    else if(window.XMLHttpRequest)
    {
        xh1=new XMLHttpRequest();
    }

    xh1.onreadystatechange=result1;
    xh1.open("GET" , "srv1" , true);
    xh1.send();
}
function fun2()
{
    if(window.ActiveXObject)
    {
        xh2=new ActiveXObject("Microsoft.XMLHTTP");
    }
    else if(window.XMLHttpRequest)
    {
```





## JAVA Means DURGA SOFT

```
{  
    public void doGet(HttpServletRequest req, HttpServletResponse res)  
        throws ServletException, IOException  
    {  
        res.setHeader("cache-control", "no-cache");  
        res.setContentType("text/html");  
        PrintWriter pw=res.getWriter();  
        try{  
            Thread.sleep(10000);  
        }  
        catch(Exception e)  
        {  
            //e.printStackTrace();  
            System.out.println("exception has come");  
        }  
        pw.println("Result of link1");  
        pw.close();  
    }  
}
```



### SSLinkServlet2:

```
import javax.servlet.*;  
import javax.servlet.http.*;  
import java.io.*;  
public class LinkServlet2 extends HttpServlet  
{  
    public void doGet(HttpServletRequest req, HttpServletResponse res)  
        throws ServletException, IOException  
    {  
    }  
}
```

## JAVA Means DURGA SOFT

```
        res.setHeader("cache-control","no-cache");
        res.setContentType("text/html");
        PrintWriter pw=res.getWriter();
        pw.println("Result of link2");
        pw.close();
    }
}
```

### **web.xml:**

```
<web-app>
  <servlet>
    <servlet-name>dum1</servlet-name>
    <servlet-class>LinkServlet1</servlet-class>
  </servlet>

  <servlet>
    <servlet-name>dum2</servlet-name>
    <servlet-class>LinkServlet2</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>dum1</servlet-name>
    <url-pattern>/srv1</url-pattern>
  </servlet-mapping>

  <servlet-mapping>
    <servlet-name>dum2</servlet-name>
    <url-pattern>/srv2</url-pattern>
  </servlet-mapping>
</web-app>
```



**www.durgasoftonlinetraining.com**

**Online Training**  
**Pre Recorded Video**  
**Classes Training**  
**Corporate Training**

**Ph: +91-8885252627, 7207212427**  
**+91-7207212428**

 **USA Ph : 4433326786**

**E-mail : durgasoftonlinetraining@gmail.com**

**Drawbacks of AJAX:-**

- ❖ If JavaScript is not activated, Ajax can't work. The user must be asked to set JavaScript from within options of the browser, with the "noscript" tag.
- ❖ Since data to display are loaded dynamically, they are not part of the page, and the keywords inside are not used by search engines.
- ❖ The asynchronous mode may change the page with delays (when the processing on the server take some times), this may be disturbing.
- ❖ The back button may be deactivated (this is not the case in examples provided here). This may be overcome.

JAVA Means DURGA SOFT

**www.durgajobs.com**

*Continuous Job Updates for every hour*

Fresher Jobs

Govt Jobs

Bank Jobs

Walk-ins

Placement Papers

IT Jobs

Interview Experiences

*Complete Job information across India*

LEARN FROM EXPERT & DIAMOND FACULTIES OF AMEERPET...

**JAVA MEANS DURGASOFT**

**INDIA'S NO. 1 SOFTWARE TRAINING INSTITUTE**

AN ISO 9001:2008 CERTIFIED

**DURGA**  
SOFTWARE SOLUTIONS

#202 2<sup>nd</sup> FLOOR  
www.durgasoft.com

040-64512786  
+91 9246212143  
+91 8096969696

JAVA Means DURGA SOFT

LEARN FROM EXPERTS ...

**COMPLETE JAVA**

CORE JAVA, ADV. JAVA, ORACLE, STRUTS, HIBERNATE, SPRING, WEB SERVICES,...

**COMPLETE .NET**

C#.NET, ASP.NET, SQL SERVER, MVC 5 & WCF

**TESTING TOOLS**

MANUAL + SELENIUM

**ORACLE | D2K**

**MSBI | SHARE POINT**

**HADOOP | ANDROID**

**C, C++, DS, UNIX**

**CRT & APTITUDE TRAINING**

AN ISO 9001:2008 CERTIFIED

**DURGA**  
Software Solutions®

# 202, 2nd Floor, HUDA Maitrivanam,  
Ameerpet, Hyd. Ph: 040-64512786,  
**9246212143, 8096969696**

[www.durgasoft.com](http://www.durgasoft.com)

**Java Real Time Tools**



**JAVA TOOLS** Means **DURGASOFT**

Multiple Faculty Members only For **JAVA TOOLS**

**Online Training**

**Class Room Training**

**DURGA SOFTWARE SOLUTIONS**

[www.durgasoftonlinetraining.com](http://www.durgasoftonlinetraining.com) [durgasoftonlinetraining@gmail.com](mailto:durgasoftonlinetraining@gmail.com) Ph: +91- 8885252627 +91- 7207212428