AJAX Application

Agenda

1 AJAX Application

Objectives

At the end of this module, you will be able to:

- Learn to develop a simple Suggest Application using AJAX

Ajax Application




Prerequisites for running Ajax Application

- AJAX does not have huge API requirements compared to other technologies
- For an AJAX Application to execute, the browser must support AJAX
- For Microsoft Internet Explorer, IE 6.0 or above is recommended, as the earlier versions does not support AJAX
- As we are going to fetch response from the server, a web server at least is required
- Since the request and response are carried out simultaneously, it is preferable that the RAM is at least 512MB or more

Suggest Application

Case Study : Create an application that takes city names as input from the user.

- While taking the input, for each key press it is going to provide a list of city names starting with the entered keywords, as suggestion
- Use AJAX to generate the suggestion from a servlet having an array of city names

Steps to follow

For creating the application we have to follow the following steps:

1. Create a **HTML file** with a **text box** and suggestion **div**

2. Create a **Javascript file** with AJAX code

3. Create a **Server-Side code (Servlet)** for processing the application request and generating the response

Step 1- CitySuggest.html

```
<html>
    <head>
       <title>City Suggest</title>
         <script src="CityApps.js">
       </script>
    </head>
    <body>
       <!-- Text box to type city names -->
       Enter City <input type="text" name="city_name"
  onkeyup="sendRequest(this.value)"/>
       <p><strong>Suggestions:</strong></p>
       <!-- Div where the application response shall be displayed -->
       <div id="city_suggest"></div>
    </body>
</html>
```

Step 2- CityApps.js (Javascript)

```
var req;//global variable
//function to get the keyword and generate request object
function sendRequest(cname)
{
//for firefox/safari/opera/google chrome
if (window.XMLHttpRequest) {
req = new XMLHttpRequest( );
}
else if (window.ActiveXObject)//for IE
{req = new ActiveXObject("Microsoft.XMLHTTP");}
//concatenate the city name as parameter value to url
var url = "NameSuggest?city_n="+cname;
//check server request state and invoke getResponse()
req.onreadystatechange = getResponse;
req.open("GET", url, true);//send request to server
req.send(null);
}
```

Step 2- CityApps.js (Javascript) (Contd.).

```
//function to get the response and display in the specific area
function getResponse()
{
if (req.readyState==4) //if request is complete
{
if (req.status == 200) //if target page is found
{
//write the response text in the div area
document.getElementById("city_suggest").innerHTML = req.responseText;;
}
}
}
```

Step 3- NameSuggest.java (Servlet)

```
/*
 * Servlet Class to process a keyword
 * and return matching city names as suggestion
 * from a list of city names
 */
package com.wipro;

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
```

Step 3- NameSuggest.java (Servlet) (Contd.).

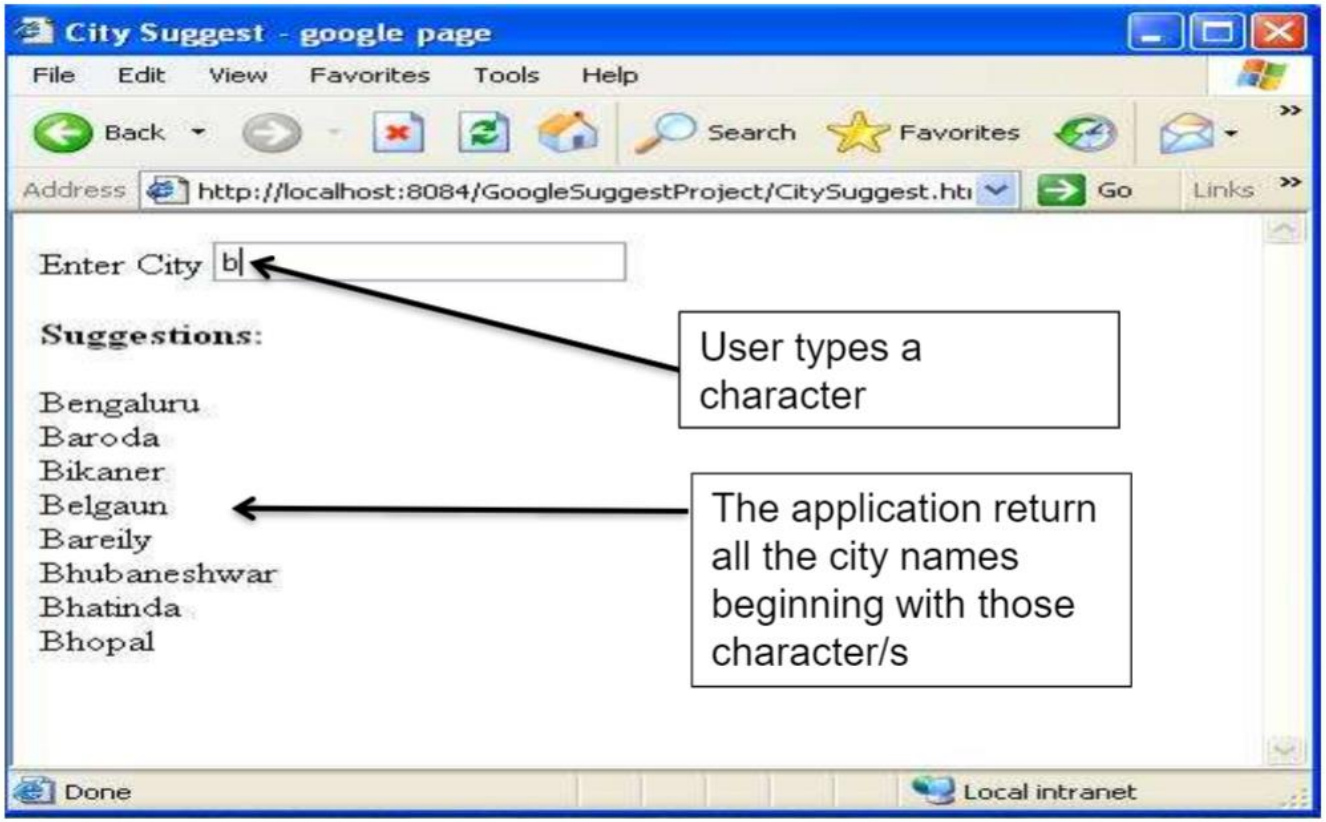
```
public class NameSuggest extends HttpServlet {
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
  response) throws ServletException, IOException {
  try {
       response.setContentType("text/html;charset=UTF-8");
       PrintWriter out = response.getWriter();
       String suggestion="";
         //Array containing city names
         String cities[]={***see declaration from notes***}
         //fetch the city name from application request
       String city_name=request.getParameter("city_n");
         if(city_name.length()>0)
            {
                for(int i=0;i<cities.length;i++)
                {
```

Step 3- NameSuggest.java (Servlet) (Contd.).

```
/*convert the array index value and the request parameter value to uppercase and
  then check if the array index value(city name) is prefixed with the supplied
  string */


    if(cities[i].toUpperCase().startsWith
    (city_name.toUpperCase()))
        {
    suggestion=suggestion+cities[i]+"<br>";
        }
      }
    }
    out.println(suggestion);
    }
  catch(Exception e)
  {e.printStackTrace();}
  }
}
```

Suggest Application Output



Summary

In this module, you were able to:

- Develop simple suggest application using AJAX

References

- w3schools.com (2012). AJAX Introduction. Retrieved April 30, 2012, from, *http://www.w3schools.com/ajax/default.asp*

- Greg Murray (2005). Asynchronous JavaScript Technology and XML(Ajax) With the Java Platform. Retrieved April 30, 2012, from, *http://www.oracle.com/technetwork/articles/javaee/ajax-135201.html*

- Adaptive path (2012). Ajax: A New Approach to Web Applications. Retrieved May 2, 2012, from, *http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications*

Thank You