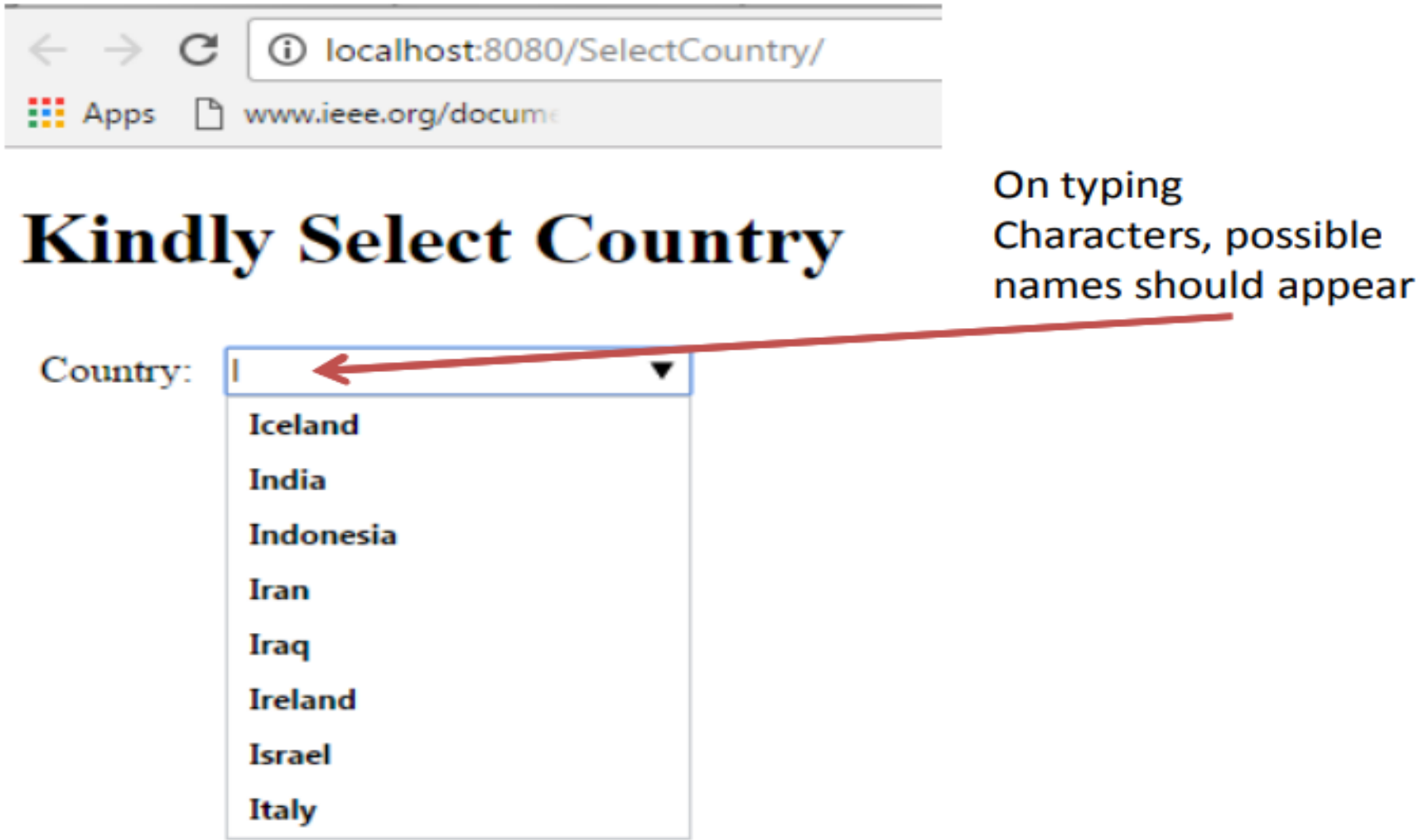
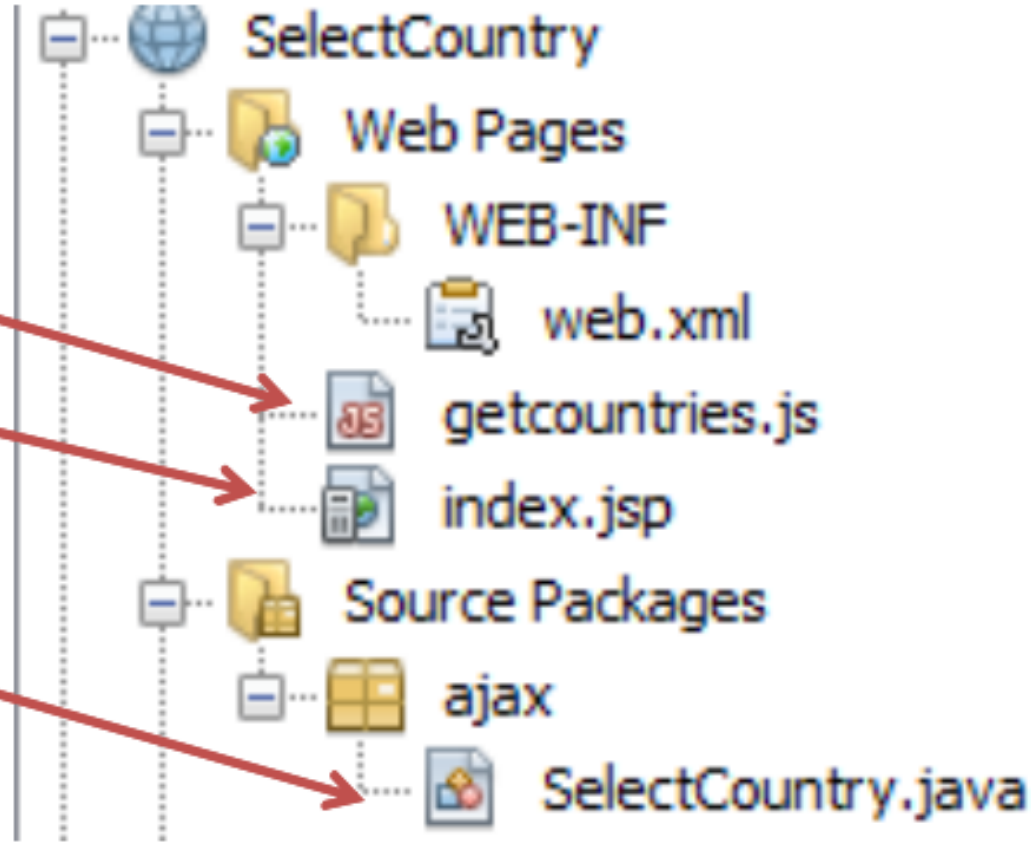
AJAX

AJAX Application – Auto complete country name



AJAX Application – Auto complete country name

- Client side programming
  - JavaScript
  - Form
- Server side programming
  - Servlet



Create servlet (SelectCountry.java)

```
package ajax;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
```

Create servlet (SelectCountry.java)

```
public class SelectCountry extends HttpServlet {
    private static final String[] COUNTRIES = new String[] {
        "Afghanistan", "Albania", "Algeria", "American Samoa",
    "Andorra",
        "Angola", "Anguilla", "Antarctica", "Antigua and Barbuda",
    "Argentina",
        "Armenia", "Aruba", "Australia", "Austria", "Azerbaijan",
        "Bahrain", "Bangladesh", "Barbados", "Belarus", "Belgium",
        "Belize", "Benin", "Bermuda", "Bhutan", "Bolivia",
        "Bosnia and Herzegovina", "Botswana", "Bouvet Island", "Brazil",
    "British Indian Ocean Territory",
        "British Virgin Islands", "Brunei", "Bulgaria", "Burkina Faso",
    "Burundi",
```

Create servlet (SelectCountry.java)

"Cote d'Ivoire", "Cambodia", "Cameroon", "Canada", "Cape Verde",
"Cayman Islands", "Central African Republic", "Chad", "Chile", "China",
"Christmas Island", "Cocos (Keeling) Islands", "Colombia", "Comoros", "Congo",
"Cook Islands", "Costa Rica", "Croatia", "Cuba", "Cyprus", "Czech Republic",
"Democratic Republic of the Congo", "Denmark", "Djibouti", "Dominica", "Dominican Republic",
"East Timor", "Ecuador", "Egypt", "El Salvador", "Equatorial Guinea", "Eritrea",
"Estonia", "Ethiopia", "Faeroe Islands", "Falkland Islands", "Fiji", "Finland",
"Former Yugoslav Republic of Macedonia", "France", "French Guiana", "French Polynesia",
"French Southern Territories", "Gabon", "Georgia", "Germany", "Ghana", "Gibraltar",

Create servlet (SelectCountry.java)

"Greece", "Greenland", "Grenada", "Guadeloupe", "Guam", "Guatemala",
"Guinea", "Guinea-Bissau",
"Guyana", "Haiti", "Heard Island and McDonald Islands", "Honduras",
"Hong Kong", "Hungary",
"Iceland", "India", "Indonesia", "Iran", "Iraq", "Ireland", "Israel", "Italy",
"Jamaica",
"Japan", "Jordan", "Kazakhstan", "Kenya", "Kiribati", "Kuwait",
"Kyrgyzstan", "Laos",
"Latvia", "Lebanon", "Lesotho", "Liberia", "Libya", "Liechtenstein",
"Lithuania", "Luxembourg",
"Macau", "Madagascar", "Malawi", "Malaysia", "Maldives", "Mali",
"Malta", "Marshall Islands",
"Martinique", "Mauritania", "Mauritius", "Mayotte", "Mexico",
"Micronesia", "Moldova",
"Monaco", "Mongolia", "Montserrat", "Morocco", "Mozambique",
"Myanmar", "Namibia",

Create servlet (SelectCountry.java)

"Nauru", "Nepal", "Netherlands", "Netherlands Antilles", "New Caledonia", "New Zealand",

"Nicaragua", "Niger", "Nigeria", "Niue", "Norfolk Island", "North Korea", "Northern Marianas",

"Norway", "Oman", "Pakistan", "Palau", "Panama", "Papua New Guinea", "Paraguay", "Peru",

"Philippines", "Pitcairn Islands", "Poland", "Portugal", "Puerto Rico", "Qatar",

"Reunion", "Romania", "Russia", "Rwanda", "Sqo Tome and Principe", "Saint Helena",

"Saint Kitts and Nevis", "Saint Lucia", "Saint Pierre and Miquelon",

"Saint Vincent and the Grenadines", "Samoa", "San Marino", "Saudi Arabia", "Senegal",

"Seychelles", "Sierra Leone", "Singapore", "Slovakia", "Slovenia", "Solomon Islands",

Create servlet (SelectCountry.java)

```
"Somalia", "South Africa", "South Georgia and the South Sandwich Islands", "South
    Korea",
      "Spain", "Sri Lanka", "Sudan", "Suriname", "Svalbard and Jan Mayen",
    "Swaziland", "Sweden",
      "Switzerland", "Syria", "Taiwan", "Tajikistan", "Tanzania", "Thailand", "The
    Bahamas",
      "The Gambia", "Togo", "Tokelau", "Tonga", "Trinidad and Tobago", "Tunisia",
    "Turkey",
      "Turkmenistan", "Turks and Caicos Islands", "Tuvalu", "Virgin Islands", "Uganda",
      "Ukraine", "United Arab Emirates", "United Kingdom",
      "United States", "United States Minor Outlying Islands", "Uruguay",
    "Uzbekistan",
      "Vanuatu", "Vatican City", "Venezuela", "Vietnam", "Wallis and Futuna",
    "Western Sahara",
      "Yemen", "Yugoslavia", "Zambia", "Zimbabwe"
     };
```

```
protected void processRequest(HttpServletRequest request,
    HttpServletResponse response)
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    response.setContentType("text/xml");
    response.setHeader("Cache-control", "no-cache, no-store");
    response.setHeader("Pragma", "no-cache");
    response.setHeader("Expires", "-1");
```

Create servlet (SelectCountry.java)

```
String queryToken=request.getParameter("queryToken");
    List<String> matchingCountries=getCountries(queryToken);
    StringBuffer sBuff=new StringBuffer();
    Iterator<String> iterator = matchingCountries.iterator();
    sBuff.append("<countries>");
    while(iterator.hasNext()) {
       String country = (String)iterator.next();
       sBuff.append("<country>");
       sBuff.append(country);
       sBuff.append("</country>");
    }
    sBuff.append("</countries>");
    out.write(sBuff.toString());
  }
```

Create servlet (SelectCountry.java)

```
private List<String> getCountries(String queryToken){
    String country = null;
    int totalCountries=COUNTRIES.length;
    queryToken = queryToken.toLowerCase();
    List<String> matchingCountries = new ArrayList<String>();
    for(int i=0; i<totalCountries; i++) {
       country = COUNTRIES[i].toLowerCase();
       if(country.startsWith(queryToken)) {
          matchingCountries.add(COUNTRIES[i]);
       }
    }
    return matchingCountries;
 }
```

```
@Override
    protected void doGet(HttpServletRequest request,
     HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
@Override
    protected void doPost(HttpServletRequest request,
     HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
}
```

Testing the servlet

- Localhost:8080/SelectCountry/country?queryToken=I

Form (index.jsp)

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <script type="text/javascript"
     src="getcountries.js"></script>
        <meta http-equiv="Content-Type" content="text/html;
     charset=UTF-8">
        <title>Select Country</title>
    </head>
    <body onload="init()">
        <h1>Kindly Select Country</h1>
```

Form (index.jsp)

```
<form name="registrationForm" action="register">
        <table border="0" cellpadding="5">
          <tr>
           <td>Country:</td>
           <td><input type="text" name="country" id="country"
   oninput="getCountries();" list="countryList"/>
              <datalist id="countryList">
              </datalist>
           </td>
          </tr>
        </table>
     </form>
  </body>
</html>
```

At runtime, we will add <option /> for each country
E.g., <option value="India" />

A datalist is a newly introduced tag in HTML5.

JavaScript (getcountries.js)

```
var xhr;
function init() {
   countryField = document.getElementById("country");
   countryListField = document.getElementById("countryList");
}
function getCountries() {
      var queryToken=countryField.value;
      if(queryToken!=""){
         var url = "country?queryToken=" + escape(countryField.value);
         xhr = createXHRObject();
         xhr.open("GET", url, true);
         xhr.onreadystatechange = callback;
         xhr.send(null);
      }
```

JavaScript (getcountries.js)

```
function createXHRObject() {
    var xhrObject;
    try{
        // Opera 8.0+, Firefox, Safari
        xhrObject = new XMLHttpRequest();
    }catch (e){
        // Internet Explorer Browsers
        try{
            xhrObject = new
    ActiveXObject("Msxml2.XMLHTTP");
        }catch (e) {
```

JavaScript (getcountries.js)

```
function createXHRObject() {
    var xhrObject;
    try{
        // Opera 8.0+, Firefox, Safari
        xhrObject = new XMLHttpRequest();
    }catch (e){
        // Internet Explorer Browsers
        try{
          xhrObject = new ActiveXObject("Msxml2.XMLHTTP");
        }catch (e) {
            try{
            xhrObject = new ActiveXObject("Microsoft.XMLHTTP");
            }catch (e){
```

JavaScript (getcountries.js)

```
            // Something went wrong
            alert("Your browser broke!");
            return false;
        }
    }
}
return xhrObject;
}
```

JavaScript (getcountries.js)

```
function callback() {
    if (xhr.readyState == 4) {
        if (xhr.status == 200) {
            setMessaage(xhr.responseXML);
        }
    }
}
```

JavaScript (getcountries.js)

```
function setMessaage(responseXML){
   var countriesElement=responseXML.getElementsByTagName("country");
   var sb="";
   for (var i = 0, max = countriesElement.length; i < max; i++) {
      var country = countriesElement[i].childNodes[0].nodeValue;
      sb=sb+"<option value=\""+country+"\" />";
   }
   countryListField.innerHTML = sb;
}
```

Running the application