

# 1 Ajax And Database

## 3 1. Lab1

5 [META-INF/context.xml]

```
6 <?xml version="1.0" encoding="UTF-8"?>
7 <Context docBase="/0715" path="/0715" reloadable="true">
8   <Resource auth="Container"
9     driverClassName="oracle.jdbc.driver.OracleDriver"
10    maxActive="10" maxIdle="10" maxWait="10000"
11    name="jdbc/myoracle" password="tiger"
12    type="javax.sql.DataSource"
13    url="jdbc:oracle:thin:@192.168.228.128:1521:orcl"
14    username="scott"/>
15 </Context>
```

17 [WEB-INF/web.xml]

```
18 <?xml version="1.0" encoding="UTF-8"?>
19 <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="
http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd" id="WebApp_ID" version="3.0">
20   <display-name>0715</display-name>
21
22   <resource-ref>
23     <description>Oracle JNDI</description>
24     <res-ref-name>jdbc/myoracle</res-ref-name>
25     <res-type>javax.sql.DataSource</res-type>
26     <res-auth>Container</res-auth>
27   </resource-ref>
28
29 </web-app>
```

31 [DBConnection.java]

```
32 package kr.co.javaexpert.libs.db;
33
34 import javax.sql.DataSource;
35 import java.sql.Connection;
36 import java.sql.SQLException;
37 import javax.naming.Context;
38 import javax.naming.InitialContext;
39 import javax.naming.NamingException;
40
41 public class DBConnection {
42   public static Connection getConnection(){
43     Connection conn = null;
44     try {
45       Context context = new InitialContext();
46       Context env = (Context)context.lookup("java:comp/env");
47       DataSource ds = (DataSource)env.lookup("jdbc/myoracle");
48       conn = ds.getConnection();
49     } catch (NamingException e) {
50       System.out.println(e.getMessage());
51     } catch (SQLException e){
52       System.out.println(e.getMessage());
53     }
54     return conn;
55   }
56 }
```

58 [DBCclose.java]

```
59 package kr.co.javaexpert.libs.db;
60
61 import java.sql.SQLException;
62 import java.sql.Connection;
63
64 public class DBClose {
65   public static void close(Connection conn){
```

```

66         if(conn != null)
67             try {
68                 conn.close();
69             } catch (SQLException e) {
70                 System.out.println(e.getMessage());
71             }
72     }
73 }
74
75 [js/httpRequest.js]
76 function getXMLHttpRequest() {
77     if (window.ActiveXObject) {
78         try {
79             return new ActiveXObject("Msxml2.XMLHTTP");
80         } catch(e) {
81             try {
82                 return new ActiveXObject("Microsoft.XMLHTTP");
83             } catch(e1) { return null; }
84         }
85     } else if (window.XMLHttpRequest) {
86         return new XMLHttpRequest();
87     } else {
88         return null;
89     }
90 }
91 var httpRequest = null;
92
93 function sendRequest(url, params, callback, method) {
94     httpRequest = getXMLHttpRequest();
95     var httpMethod = method ? method : 'get';
96     if (httpMethod.toLowerCase() != 'get' && httpMethod.toLowerCase() != 'post') {
97         httpMethod = 'get';
98     }
99     var httpParams = (params == null || params == '') ? null : params;
100    var httpUrl = url;
101    if (httpMethod == 'get' && httpParams != null) {
102        httpUrl = httpUrl + "?" + httpParams;
103    }
104    httpRequest.open(httpMethod, httpUrl, true);
105    httpRequest.setRequestHeader(
106        'Content-Type', 'application/x-www-form-urlencoded');
107    httpRequest.onreadystatechange = callback;
108    httpRequest.send(httpMethod == 'post' ? httpParams : null);
109 }
110
111 [getGugunlist.jsp]
112 <%@ page language="java" contentType="text/html; charset=UTF-8"
113     pageEncoding="UTF-8"%>
114 <%@ page import="kr.co.javaexpert.libs.db.*, java.sql.*" %>
115 <%
116     request.setCharacterEncoding("utf-8");
117     String sido = request.getParameter("sido");
118     Connection conn = DBConnection.getConnection();
119     String sql = "SELECT DISTINCT gugun FROM zipcode ";
120     sql += "WHERE sido = ? ORDER BY gugun";
121     PreparedStatement pstmt = conn.prepareStatement(sql);
122     pstmt.setString(1, sido);
123     ResultSet rs = pstmt.executeQuery();
124 %>
125     <select name="gugun">
126         <option>--구군선택--</option>
127 <% while(rs.next()){
128     String gugun = rs.getString(1);
129 %>
130     <option value="<%=gugun%>"><%=gugun %></option>
131 <% } //while end
132     rs.close();

```

```

133         pstmt.close();
134     %>
135     </select>
136     <% DBClose.close(conn); %>
137
138 [getAddresslist.jsp]
139     <%@ page language="java" contentType="text/html; charset=UTF-8"
140         pageEncoding="UTF-8"%>
141     <%@ page import="kr.co.javaexpert.libs.db.*, java.sql.*" %>
142     <%
143         request.setCharacterEncoding("utf-8");
144         String sido = request.getParameter("sido");
145         String gugun = request.getParameter("gugun");
146         String dong = request.getParameter("dong");
147         Connection conn = DBConnection.getConnection();
148         String sql = null;
149         if(sido == "" && gugun == ""){
150             sql = "SELECT zipcode, sido, gugun, dong, bunji FROM zipcode ";
151             sql += "WHERE dong LIKE '%" + dong + "%' ";
152         }else if(sido != "" && gugun == ""){
153             sql = "SELECT zipcode, sido, gugun, dong, bunji FROM zipcode ";
154             sql += "WHERE sido = '" + sido + "' AND dong LIKE '%" + dong + "%' ";
155         }else if(sido != "" && gugun != ""){
156             sql = "SELECT zipcode, sido, gugun, dong, bunji FROM zipcode ";
157             sql += "WHERE sido = '" + sido + "' AND gugun = '" + gugun + "' AND ";
158             sql += "dong LIKE '%" + dong + "%' ";
159         }
160         Statement stmt = conn.createStatement();
161         ResultSet rs = stmt.executeQuery(sql);
162     %>
163     <table cellpadding="3" cellspacing="0" border="1" width="500">
164     <%
165         while(rs.next()){
166             String zipcode = "[" + rs.getString("zipcode") + "]";
167             String dbsido = rs.getString("sido");
168             String dbgugun = rs.getString("gugun");
169             String dbdong = rs.getString("dong");
170             String bunji = rs.getString("bunji");
171             if(bunji == null) bunji = "";
172             String address = dbsido + " " + dbgugun + " " + dbdong + " " + bunji;
173         %>
174         <tr>
175             <td><%=zipcode %></td>
176             <td><%=address %></td>
177         </tr>
178     <% }
179         rs.close();
180         stmt.close();
181     %>
182     </table>
183     <% DBClose.close(conn); %>
184
185 [zipsearch.jsp]
186     <%@ page language="java" contentType="text/html; charset=UTF-8"
187         pageEncoding="UTF-8"%>
188     <%@ page import="kr.co.javaexpert.libs.db.*, java.sql.*" %>
189     <%
190         Connection conn = DBConnection.getConnection();
191         Statement stmt = conn.createStatement();
192         String sql = "SELECT DISTINCT sido FROM zipcode ORDER BY sido";
193         ResultSet rs = stmt.executeQuery(sql);
194     %>
195     <!DOCTYPE html>
196     <html lang="ko">
197     <head>
198     <meta charset="UTF-8">
199     <title>Ajax Demo</title>

```

[illegible]

```
267 </html>
268 <% DBClose.close(conn); %>
```

## 271 2. Lab2

272  
273 [member.sql]

```
274 CREATE TABLE member (
275     PHONE CHAR(14) NOT NULL PRIMARY KEY,
276     IRUM VARCHAR2(20) NOT NULL,
277     SIDO VARCHAR2(10) NOT NULL,
278     GUGUN VARCHAR2(20) NOT NULL,
279     DONG VARCHAR2(100) NOT NULL,
280     BUNJI VARCHAR2(17) NULL,
281     ZIPCODE CHAR(7) NOT NULL
282 );
283
284 INSERT INTO MEMBER values ('02-123-4567', '홍길동', '서울', '강남구', '개포1동 강남아파트', '', '135-806');
285 INSERT INTO MEMBER values ('02-345-7777', '조성모', '서울', '영등포구', '대림3동', '718~774', '150-815');
286 INSERT INTO MEMBER values ('02-974-4567', '이미자', '서울', '성동구', '옥수동', '', '133-100');
287 INSERT INTO MEMBER values ('02-3482-6777', '이영애', '서울', '강남구', '개포1동 강남아파트', '', '135-806');
288 INSERT INTO MEMBER values ('02-203-4444', '박찬호', '서울', '동대문구', '휘경2동', '', '130-092');
289 INSERT INTO MEMBER values ('02-888-5555', '선동열', '서울', '노원구', '중계1동 신안아파트', '', '139-922');
290 INSERT INTO MEMBER values ('02-244-2424', '한지민', '서울', '동작구', '노량진1동', '13~112', '156-800');
291 INSERT INTO MEMBER values ('02-111-1111', '공유', '서울', '구로구', '오류1동 동부골든아파트', '', '152-783');
292 INSERT INTO MEMBER values ('02-222-2222', '윤은혜', '서울', '광진구', '군자동', '345~374', '143-840');
293 INSERT INTO MEMBER values ('02-333-3333', '나훈아', '서울', '강동구', '천호3동', '', '134-023');
```

294  
295 [lookupCustomer.jsp]

```
296 <%@ page language="java" contentType="text/html; charset=utf-8"
297     pageEncoding="utf-8" import="java.sql.*" %>
298 <jsp:directive.page import="kr.co.javaexpert.libs.db.*" />
299 <%
300     String phone = request.getParameter("phone");
301     Connection conn = DBConnection.getConnection();
302     String sql = "SELECT irum, sido, gugun, dong, bunji, zipcode FROM member ";
303     sql += "WHERE phone = '" + phone + "' ";
304     Statement stmt = conn.createStatement();
305     ResultSet rs = stmt.executeQuery(sql);
306     String msg = null;
307     if(rs.next()){
308         msg = rs.getString("irum") + "\n";
309         msg += rs.getString("sido") + " " + rs.getString("gugun") + " " + rs.getString("dong");
310         if(rs.getString("bunji") == null ) msg += " ";
311         else msg += " " + rs.getString("bunji");
312         msg += " #" + rs.getString("zipcode");
313     }
314     out.println(msg);
315     rs.close();
316     stmt.close();
317     DBClose.close(conn);
318 %>
```

319  
320 [pizza.html]

```
321 <!DOCTYPE html>
322 <html>
323 <head>
324 <meta charset="UTF-8">
325 <title>Break Neck Pizza Delivery</title>
326 <link rel="stylesheet" type="text/css" href="css/breakneck.css" media="screen" />
327 <script src="js/httpRequest.js"></script>
328 <script>
329     var xmlObj;
330     function getCustomerInfo(){
331         var phone = document.getElementById("phone").value;
332         var param = "phone=" + escape(phone);
333         //xmlObj = sendRequest("lookupCustomer.jsp", param, updatePage, "get");
```

```

334         xmlObj = new XMLHttpRequest();
335         xmlObj.onreadystatechange = updatePage;
336         xmlObj.open("get", "lookupCustomer.jsp?" + param);
337         xmlObj.send(null);
338     }
339     function updatePage(){
340         if(xmlObj.readyState == 4 && xmlObj.status == 200){
341             var customerAddress = xmlObj.responseText;
342             document.getElementById("address").value = customerAddress;
343         }
344     }
345 </script>
346 </head>
347 <body>
348     <div>
349         <p></p>
350         <form method="POST" action="placeOrder.jsp">
351             <p><label for="phone">Enter your phone number:</label>
352             <input type="text" size="30" name="phone" id="phone" onChange="getCustomerInfo()"/>
353             </p>
354             <p><label>Your order will be delivered to:</label></p>
355             <p><textarea name="address" rows="4" cols="60" id="address"></textarea></p>
356             <p>Type your order in here:</p>
357             <p><textarea name="order" rows="6" cols="60" id="order"></textarea></p>
358             <p><input type="submit" value="Order Pizza" id="submit" /></p>
359         </form>
360     </div>
361 </body>
362 </html>
363
364 [breakneck.css]
365     div {
366         color: darkred;
367         font-family: sans-serif;
368         margin: auto;
369         padding: 1em;
370         border: double darkred;
371         width:615px;
372     }
373
374     .customer-info {
375         font-weight: bold;
376     }
377
378     .customer-data {
379         color: black;
380     }
381 }
382
383 [placeOrder.jsp]
384 <%@ page language="java" contentType="text/html; charset=utf-8"
385     pageEncoding="utf-8"%>
386 <%
387     request.setCharacterEncoding("utf-8");
388     String address = request.getParameter("address");
389     String order = request.getParameter("order");
390 %>
391 <!DOCTYPE html>
392 <html lang="en">
393 <head>
394     <meta charset="utf-8">
395     <title>Break Neck Pizza Delivery</title>
396     <link rel="stylesheet" type="text/css" href="css/breakneck.css" />
397 </head>
398 <body>
399     <p>
400         

```

```
401         </p>
402         <p>Your order will be delivered to:</p>
403         <p class="customer-data">
404     <%
405         out.println(address + "<br />");
406     %>
407     </p>
408     <p>We have your order down as:</p>
409     <p class="customer-data">
410 <%
411         out.println(order + "<br />");
412     %>
413     </p>
414 </body>
415 </html>
```