```
1 Using Connection Pooling with JNDI
   1. JNDI(Java Naming And Directory Interface)
 4
     1)Naming service 및 directory service에 접근하기 위한 API
 5
     2)필요한 data를 name/value 형식으로 저장하고, 필요한 data는 name 값을 이용하여 value값을 얻는 형태
     3)대표적인 경우가 DNS
 6
 7
     4)Web browser에서 영문형식으로 domain name으로 ip값을 얻어오는 방식이 JNDI 와 유사
     5)Database용 JNDI는 "jdbc/데이타베이스버전" 형식으로 지정
 8
 9
10
11 2. Lab
12
     1)server.xml
13
       <Context docBase="JNDIDemo" path="/JNDIDemo" reloadable="true"
       source="org.eclipse.jst.jee.server:JNDIDemo">
         <Resource auth="Container" driverClassName="oracle.idbc.driver.OracleDriver"</pre>
14
         maxActive="10" maxIdle="10"
15
             maxWait="1000" name="jdbc/myoracle" password="tiger"
             type="javax.sql.DataSource"
             url="jdbc:oracle:thin:@localhost:1521:XE" username="scott"/>
16
17
       </Context>
18
19
     2)web.xml
20
       <resource-ref>
21
         <description>JNDI Test</description>
22
         <res-ref-name>jdbc/myoracle</res-ref-name>
23
         <res-type>javax.sql.DataSource</res-type>
24
         <res-auth>Container</res-auth>
25
       </resource-ref>
26
27
     3)src/com.example.dao.DBConnection.java
28
       package com.example.dao;
29
30
       import java.sql.Connection;
31
       import java.sql.SQLException;
32
33
       import javax.naming.Context;
34
       import javax.naming.InitialContext;
35
       import javax.naming.NamingException;
36
       import javax.sql.DataSource;
37
38
       public class DBConnection {
         public static Connection getConnection(){
39
40
           Connection conn = null;
41
           try {
42
             Context context = new InitialContext();
43
             Context env = (Context)context.lookup("java:comp/env");
44
             DataSource ds = (DataSource)env.lookup("jdbc/myoracle");
45
             conn = ds.getConnection();
           } catch (NamingException e) {
46
47
             System.out.println(e.getMessage());
48
           } catch(SQLException e){
49
             System.out.println(e.getMessage());
50
51
           return conn;
```

```
}
 52
 53
 54
 55
      4)Create Stored Procedure sp_select
 56
        CREATE OR REPLACE PROCEDURE sp_select
 57
 58
           v deptno IN employees.department id%TYPE,
 59
           employee records OUT SYS REFCURSOR
 60
         )
        ÁS
 61
 62
         BEGIN
 63
           OPEN employee records FOR
 64
           SELECT employee id, first name, salary,
                 TO_CHAR(hire_date, 'YYYY-MM-DD') AS hiredate,
 65
 66
                 department_name, city, e.department_id AS deptno
 67
           FROM employees e INNER JOIN departments d ON e.department id = d.department id
 68
                       INNER JOIN locations | ON d.location_id = l.location_id
 69
           WHERE e.department_id = v_deptno;
 70
         END:
 71
        /
 72
 73
      5)Build Path에 oracle driver 추가하기
 74
         -project > right-click > Build Path > Configure Build Path...
 75
         -Libraries tab > Click [Add External JARs...]
 76
         -Select ojdbc6.jar > Click [Apply and Close]
 77
 78
      6)WebContent/WEB-INF/lib에 jar file 추가
 79
         -oidbc6.iar
         -taglibs-standard-impl-1.2.5.jar
 80
 81
         -taglibs-standard-spec-1.2.5.jar
 82
 83
      7)src/com.example.vo.EmployeeVO.java
 84
         package com.example.vo;
 85
 86
        public class EmployeeVO {
 87
           private int employee_id;
           private String first name;
 88
           private double salary;
 89
 90
           private String hiredate;
 91
           private String department_name;
 92
           private String city;
 93
          private int departno;
 94
 95
           public EmployeeVO() {}
 96
 97
           public int getEmployee id() {
 98
            return employee id;
 99
           }
100
           public void setEmployee id(int employee id) {
101
            this.employee id = employee id;
102
103
104
105
           public String getFirst name() {
```

```
106
             return first name;
107
108
109
           public void setFirst_name(String first_name) {
             this.first_name = first_name;
110
111
112
113
           public double getSalary() {
114
             return salary;
115
           }
116
117
           public void setSalary(double salary) {
118
             this.salary = salary;
119
120
121
           public String getHiredate() {
122
             return hiredate;
123
124
125
           public void setHiredate(String hiredate) {
             this.hiredate = hiredate;
126
127
           }
128
129
           public String getDepartment_name() {
130
             return department name;
131
132
133
           public void setDepartment name(String department name) {
134
             this.department_name = department_name;
135
           }
136
137
           public String getCity() {
138
             return city;
139
140
141
           public void setCity(String city) {
142
             this.city = city;
143
144
145
           public int getDepartno() {
146
             return departno;
147
148
           public void setDepartno(int departno) {
149
             this.departno = departno;
150
151
152
153
           @Override
154
           public String toString() {
             return "EmployeeVO [employee id=" + employee id + ", first name=" + first name + ",
155
             salary=" + salary
                 + ", hiredate=" + hiredate + ", department_name=" + department_name + ", city="
156
                 + city
                 + ", department id=" + departno + "]";
157
```

```
158
           }
159
160
         }
161
162
       8)src/com.example.dao.EmployeeDao.java
         package com.example.dao;
163
164
165
         import java.sql.CallableStatement;
166
         import java.sql.Connection;
167
         import java.sql.ResultSet;
168
         import java.sql.SQLException;
         import java.util.ArrayList;
169
170
171
         import com.example.vo.EmployeeVO;
172
173
         public class EmployeeDao {
           public static ArrayList<EmployeeVO> selectAll(int deptno) throws SQLException{
174
175
             ArrayList<EmployeeVO> list = new ArrayList<EmployeeVO>();
176
             Connection conn = DBConnection.getConnection();
             CallableStatement cstmt = conn.prepareCall("{ call sp_select(?, ?) }");
177
178
             cstmt.setInt(1, deptno);
179
             cstmt.registerOutParameter(2, oracle.jdbc.OracleTypes.CURSOR);
180
             cstmt.executeUpdate();
181
             ResultSet rs = (ResultSet)cstmt.getObject(2);
182
             while(rs.next()) {
183
               EmployeeVO emp = new EmployeeVO();
               emp.setEmployee_id(rs.getInt("employee_id"));
184
185
               emp.setFirst name(rs.getString("first name"));
186
               emp.setSalary(rs.getDouble("salary"));
               emp.setHiredate(rs.getString("hiredate"));
187
               emp.setDepartment name(rs.getString("department name"));
188
189
               emp.setCity(rs.getString("city"));
190
               emp.setDepartno(rs.getInt("deptno"));
191
               list.add(emp);
192
193
             if(rs != null) rs.close();
194
             if(cstmt != null) cstmt.close();
             return list;
195
        }
196
197
198
199
       9)src/com.example.service.EmployeeService.java
200
         package com.example.service;
201
         import java.sql.SQLException;
202
203
         import java.util.ArrayList;
204
205
         import com.example.dao.EmployeeDao;
206
         import com.example.vo.EmployeeVO;
207
208
         public class EmployeeService {
209
           private int deptno;
210
           private ArrayList<EmployeeVO> list;
211
```

```
public void setDeptno(int deptno) {
212
213
          this.deptno = deptno;
214
         }
215
216
         public ArrayList<EmployeeVO> getList() {
          ArrayList<EmployeeVO> list = null;
217
218
          try {
219
            list = EmployeeDao.selectAll(this.deptno);
220
          }catch(SQLException ex) {
221
            System.out.println(ex);
222
223
          return list;
224
        }
225
       }
226
227
     10) dbtest.jsp
228
       <@ page language="java" contentType="text/html; charset=UTF-8"
       pageEncoding="UTF-8"%>
       < @ page import="java.util.ArrayList, com.example.vo.EmployeeVO" %>
229
230
       <@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
       <jsp:useBean id="service" class="com.example.service.EmployeeService" />
231
       <c:set target="${service}" property="deptno" value="${empty param.deptno? 10:
232
       param.deptno}"/>
       <!DOCTYPE html>
233
       <html>
234
235
         <head>
236
           <meta charset="UTF-8">
237
          <title>사원명단</title>
238
         </head>
239
         <body>
240
          <h1>사원 명단(부서번호: <c:out value="${empty param.deptno? 10: param.deptno}"
          />)</h1>
241
          242
            <thead>
243
              244
                사원번호사원이름봉급
245
               입사일자부서이름<부서위치</th>
               부서번호
246
247
              248
            </thead>
249
            250
              <c:forEach items="${service.list}" var="emp">
251
               ${emp['employee_id']}${emp['first_name']}${emp.sal
252
               ary}
253
               ${emp.hiredate}${emp['department name']}${emp.ci
               ty}
254
               ${emp.departno}
255
              256
              </c:forEach>
257
            258
          259
         </body>
260
       </html>
```