Spring board(list)

Test

0. Spring FrameWork 빌드 버전 체크사항

java-version : 1.8
org.springframework-version : 4.3.3.RELEASE
<u>Junit : 4.12</u>

spring-jdbc : ${org.springframework-version}
Ojdbc6 : 11.2.0.3

Mybatis : 3.4.4
mybatis-spring : 1.3.1
mybatis-spring : ${org.springframework-version}

```
<!-- 사설 저장소 등록 -->
<!-- (추가)OJDBC -->
<repositories>
<repository>
    <id>codelds</id>

<url>https://code.lds.org/nexus/content/g
roups/main-repo</url>
    </repository>
</repositories>
```

1. root-context.xml

Namespaces

Configure Namespaces

Select XSD namespaces to use in the configuration file

- ☐ aop - http://www.springframework.org/schema/aop
- ☑ beans - http://www.springframework.org/schema/beans
- ☐ c - http://www.springframework.org/schema/c
- ☐ cache - http://www.springframework.org/schema/cache
- ☑ context - http://www.springframework.org/schema/context
- ☑ jdbc - http://www.springframework.org/schema/jdbc
- ☐ jee - http://www.springframework.org/schema/jee
- ☐ lang - http://www.springframework.org/schema/lang
- ☐ mvc - http://www.springframework.org/schema/mvc
- ☑ mybatis-spring - http://mybatis.org/schema/mybatis-spring
- ☐ p - http://www.springframework.org/schema/p
- ☐ task - http://www.springframework.org/schema/task
- ☐ tx - http://www.springframework.org/schema/tx
- ☐ util - http://www.springframework.org/schema/util

```xml
12    <!-- Root Context: defines shared resources visible to all other web components -->
13    <!-- 오라클 DB와 연동 -->
14    <bean class="org.springframework.jdbc.datasource.DriverManagerDataSource" id="dataSource">
15      <property value="oracle.jdbc.driver.OracleDriver" name="driverClassName" />
16      <property value="jdbc:oracle:thin:@192.168.0.241:1521:xe" name="url" />
17      <!-- 오라클 사용자 이름 -->
18      <property value="T2_DBA" name="username" />
19      <!-- 오라클 사용자 비밀번호 -->
20      <property value="1234" name="password" />
21    </bean>
22
23    <bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
24        <property name="dataSource" ref="dataSource"/>
25        <property name="configLocation"
26        value="classpath:/mybatis-config.xml"/>
27        <property name="mapperLocations"
28        value="classpath:mappers/**/*.xml"/>
29    </bean>
30
31    <bean id="sqlSession" class="org.mybatis.spring.SqlSessionTemplate"
32        destroy-method="clearCache">
33        <constructor-arg name="sqlSessionFactory"   ref="sqlSessionFactory"/>
34    </bean>
```

<!– mySQL 8.0 DB와 연동 -->

```xml
15  <bean id="dataSource"
16      class="org.springframework.jdbc.datasource.DriverManagerDataSource">
17      <property name="driverClassName"
18          value="com.mysql.jdbc.Driver" />
19      <property name="url"
20          value="jdbc:mysql://localhost:3306/scott" />
21      <property name="username" value="root" />
22      <property name="password" value="1234" />
23  </bean>
24
25  <bean id="sqlSessionFactory"
26      class="org.mybatis.spring.SqlSessionFactoryBean">
27      <property name="dataSource" ref="dataSource" />
28      <property name="configLocation"
29          value="classpath:/mybatis-config.xml" />
30      <property name="mapperLocations"
31          value="classpath:employee-mapper.xml" /> <!-- FIXME -->
32  </bean>
33
34  <bean id="sqlSession"
35      class="org.mybatis.spring.SqlSessionTemplate"
36      destroy-method="clearCache">
37      <constructor-arg name="sqlSessionFactory"
38          ref="sqlSessionFactory" />
39  </bean>
```
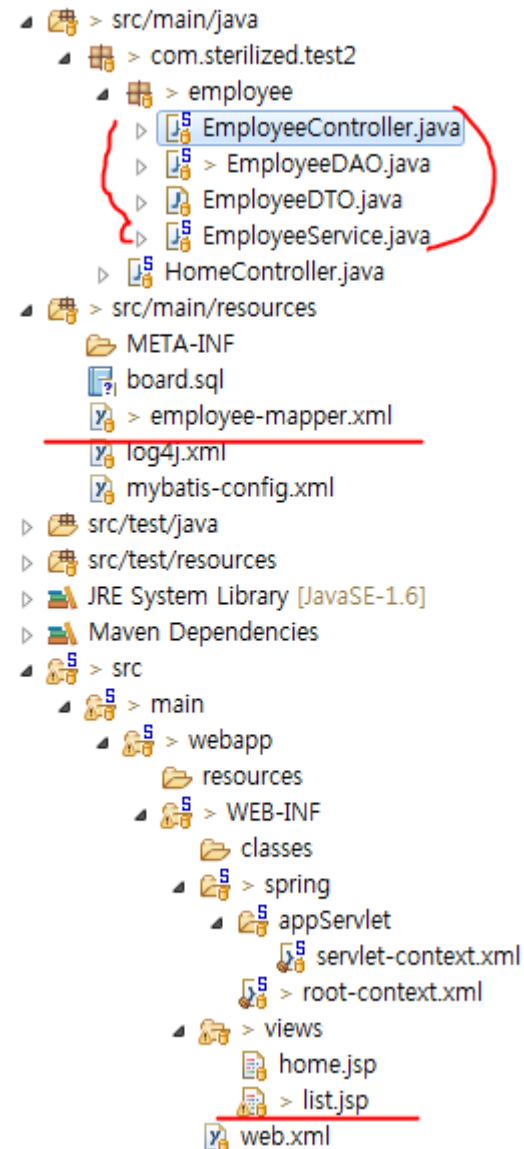
2. / board 테이블 만들기

CREATE TABLE if not exist `board` (
  `id` **int(11) NOT NULL AUTO_INCREMENT,**
  `name` **varchar(45) DEFAULT NULL,**
  PRIMARY KEY (`id`),
  UNIQUE KEY `id_UNIQUE` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=13 **DEFAULT**
CHARSET=utf8;

기본데이터 10개 정도 입력해놓기

Result Grid	
id	name
1	강
2	김
3	조
4	추
5	유
6	김2
7	이
8	허
9	양
10	한
11	최
12	워
NULL	NULL

Information

Table: board

Columns:
  <u>id</u>   int(11) AI PK
  name   varchar(45)

3. Java 파일 4개 생성(employee 패키지)
4. employee-mapper.xml 생성
5. employee/list.j네 파일 생성
 (그림오류)

```
▲ 🏛 > src/main/java
  ▲ 🏛 > com.sterilized.test2
    ▲ 🏛 > employee
      ▷ 📄 EmployeeController.java
      ▷ 📄 > EmployeeDAO.java
      ▷ 📄 EmployeeDTO.java
      ▷ 📄 EmployeeService.java
    ▷ 📄 HomeController.java
▲ 🏛 > src/main/resources
    📂 META-INF
    📄 board.sql
    📄 > employee-mapper.xml
    📄 log4j.xml
    📄 mybatis-config.xml
  ▷ 🏛 src/test/java
  ▷ 🏛 src/test/resources
  ▷ 📚 JRE System Library [JavaSE-1.6]
  ▷ 📚 Maven Dependencies
  ▲ 📁 > src
    ▲ 📁 > main
      ▲ 📁 > webapp
        📂 resources
        ▲ 📁 > WEB-INF
          📂 classes
          ▲ 📁 > spring
            ▲ 📁 appServlet
              📄 servlet-context.xml
            📄 > root-context.xml
          ▲ 📁 > views
            📄 home.jsp
            📄 > list.jsp
        📄 web.xml
```

```java
package com.sterilized.test2.employee;

public class EmployeeDTO {
    private int id;
    private String name;

    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}
```

```java
1 package com.sterilized.test2.employee;
2
3 import javax.inject.Inject;
8
9 @Controller
10 @RequestMapping("/employee")
11 public class EmployeeController {
12
13     @Inject
14     private EmployeeService service;
15
16     @RequestMapping("list")
17     public String employeeListController(Model model) {
18         model.addAttribute("employeeList", service.getEmployeeList());
19         return "list";
20     }
21 }
```

```java
package com.sterilized.test2.employee;

import java.util.ArrayList;

@Service
public class EmployeeService {

    @Inject
    private EmployeeDAO dao;

    public ArrayList<EmployeeDTO> getEmployeeList() {
        return dao.getEmployeeList();
    }
}
```

```java
1  package com.sterilized.test2.employee;
2
3⊕ import java.util.ArrayList;
10
11 @Repository
12 public class EmployeeDAO {
13
14⊖     @Inject
15     private SqlSession session;
16     private static final String namespace = "com.sterilized.test2.employee-mapper";
17
18⊖     public ArrayList<EmployeeDTO> getEmployeeList() {
19         List<EmployeeDTO> list = session.selectList(namespace + ".selectList");
20         return (ArrayList<EmployeeDTO>) list;
21     }
22
23 }
```

```xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE mapper  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
3
4⊖ <mapper namespace="com.sterilized.test2.employee-mapper">
5⊖     <select id="selectList" resultType="com.sterilized.test2.employee.EmployeeDTO">
6         SELECT * FROM board ORDER BY RAND()
7     </select>
8  </mapper>
9
```

```jsp
1  <%@ page language="java" contentType="text/html; charset=UTF-8"
2      pageEncoding="UTF-8"%>
3  <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
4  <!DOCTYPE html>
5  <html>
6  <head>
7  <meta charset="UTF-8">
8  <title>List of Employee</title>
9  </head>
10 <body>
11
12 <table border="1px solid black" style="padding : 3px;">
13     <tr>
14         <th>ID</th>
15         <th>Name</th>
16     </tr>
17     <c:forEach var="emp" items="${employeeList}">
18         <tr>
19             <td>${emp.id}</td>
20             <td>${emp.name}</td>
21         </tr>
22     </c:forEach>
23 </table>
24
25 </body>
26 </html>
```

```java
 1  package com.sterilized.test2;
 2
 3⊕ import java.util.Locale;
 9
10⊖ /**
11   * Handles requests for the application home page.
12   */
13  @Controller
14  public class HomeController {
15
16⊖     @RequestMapping(value = "/", method = Reque
17      public String home(Locale locale, Model mod
18          return "forward:/employee/list";
19      }
20
21  }
22
```

List of Employee ×

← → C | ⓘ localhost:8080/test2/

⠿ 앱 | 응용SW기초활용 | 무료영어사이트

ID	Name
3	조
12	원
11	최
6	김2
8	허
10	함
5	윤
7	이
2	김
1	강
4	추
9	양