

(3) 두번째 예제를 위한 테이블 생성

-- 과목테이블 생성

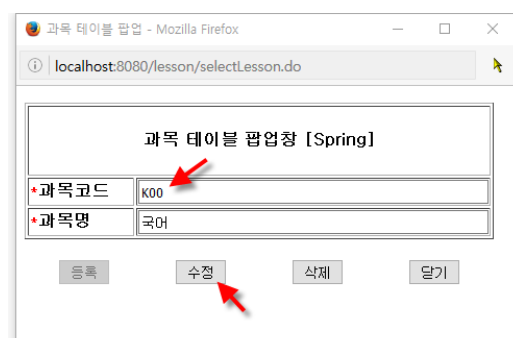
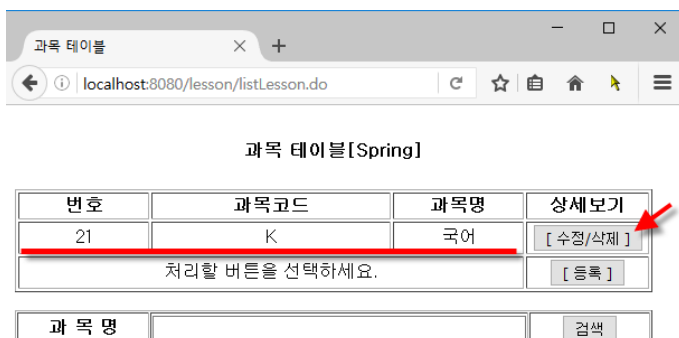
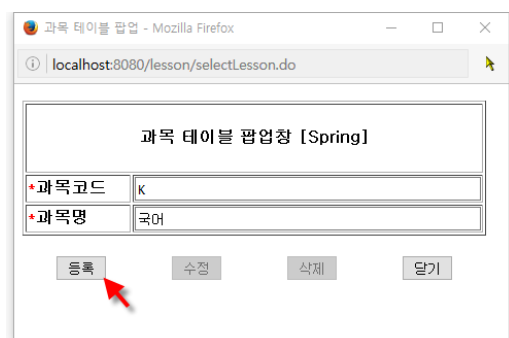
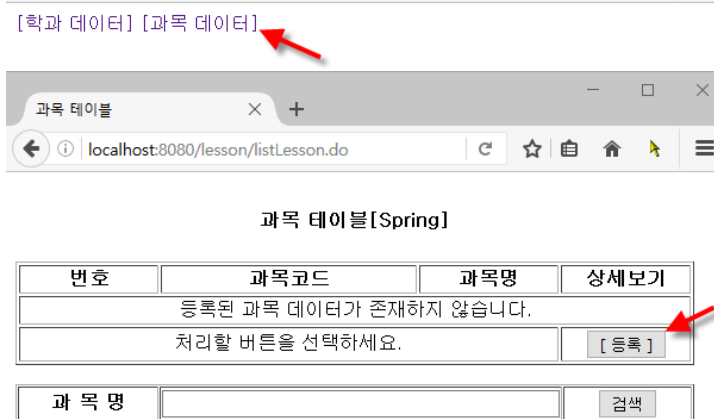
```
create table lesson(  
    no      number      not null,  
    l_num   varchar2(10) not null,  
    l_name  varchar2(30) not null,  
    constraint lesson_pk primary key(no)  
);
```

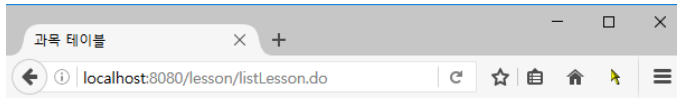
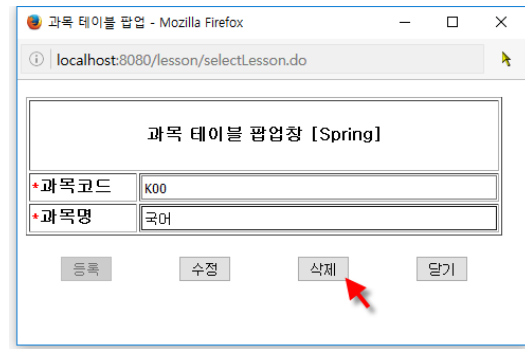
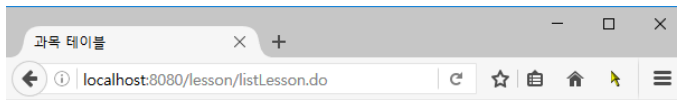
```
comment on table   lesson is '과목테이블 정보';  
comment on column lesson.no is '순번';  
comment on column lesson.l_num is '과목번호';  
comment on column lesson.l_name is '과목명';
```

-- 과목테이블 순번 시퀀스 생성

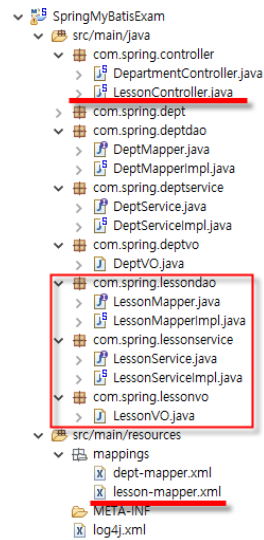
```
create sequence lesson_seq  
start with 1  
increment by 1  
cache 2  
nocycle;
```

· 코딩 후 결과화면





프로그램의 기본 구조는 다음과 같다.



root-context.xml 추가 수정

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:context="http://www.springframework.org/schema/context"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd
http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context-4.1.xsd" >
```

...
생략

```
...
<!-- <context:component-scan base-package="com.spring.deptdao"> </context:component-scan>
<context:component-scan base-package="com.spring.deptservice"> </context:component-scan>-->
<context:component-scan base-package="com.spring.lessondao"> </context:component-scan>
<context:component-scan base-package="com.spring.lessonservice"> </context:component-scan>
</beans>
```

LessonVO.java

```
package com.spring.lessonvo;

public class LessonVO {
    private int no;
    private String lnum;
    private String lname;

    public int getNo() {
        return no;
    }
    public void setNo(int no) {
        this.no = no;
    }
    public String getLnum() {
        return lnum;
    }
    public void setLnum(String lnum) {
        this.lnum = lnum;
    }
    public String getLname() {
        return lname;
    }
    public void setLname(String lname) {
        this.lname = lname;
    }
}
```

LessonController.java

```
package com.spring.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
```

```

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;

import com.spring.lesson.service.LessonService;
import com.spring.lesson.vo.LessonVO;

/* 컨트롤러 */
@Controller
@RequestMapping(value="/lesson")
public class LessonController {
    private static final String CONTEXT_PATH="lesson";

    @Autowired
    private LessonService lessonService;

    /* 전체 조회 */
    @RequestMapping("/listLesson")
    public ModelAndView listLesson(@ModelAttribute LessonVO param) {

        List<LessonVO> list = lessonService.listLesson(param);

        ModelAndView mav = new ModelAndView();
        mav.addObject("lessonList", list);
        mav.addObject("paramName", param.getLname());
        mav.setViewName(CONTEXT_PATH + "/lesson");

        return mav;
    }

    /* 상세 정보 보기 */
    @RequestMapping("/selectLesson")
    public ModelAndView selectLesson(@RequestParam("no") int no) {
        ModelAndView mav = new ModelAndView();
        if(no==0) {
            mav.addObject("mode", "insert");
        } else {
            mav.addObject("lessonVO", lessonService.selectLesson(no));
            mav.addObject("mode", "update");
        }
    }
}

```

```

        mav.setViewName(CONTEXT_PATH + "/lesson_pop");
        return mav;
    }

    /* 레코드 추가 */
    @RequestMapping("/insertLesson")
    public ModelAndView insertLesson(@ModelAttribute LessonVO param) {
        String resultStr = "";
        int result = lessonService.insertLesson(param);

        if(result > 0) resultStr = "등록 완료이 완료되었습니다.";
        else resultStr = "등록에 문제가 있어 완료하지 못하였습니다.";

        ModelAndView mav = new ModelAndView();
        mav.addObject("result", resultStr);
        mav.setViewName("/result");

        return mav;
    }

    /* 레코드 수정 */
    @RequestMapping("/updateLesson")
    public ModelAndView updateLesson(@ModelAttribute LessonVO param) {
        String resultStr = "";
        int result = lessonService.updateLesson(param);

        if(result > 0) resultStr = "수정 완료이 완료되었습니다.";
        else resultStr = "수정에 문제가 있어 완료하지 못하였습니다.";

        ModelAndView mav = new ModelAndView();
        mav.addObject("result", resultStr);
        mav.setViewName("/result");

        return mav;
    }

    /* 레코드 삭제 */
    @RequestMapping("/deleteLesson")
    public ModelAndView deleteLesson(@RequestParam("no") int no) {
        String resultStr = "";
        int result = lessonService.deleteLesson(no);
    }

```

```

        if(result > 0) resultStr = "삭제 완료이 완료되었습니다.";
        else resultStr = "삭제에 문제가 있어 완료하지 못하였습니다.";

        ModelAndView mav = new ModelAndView();
        mav.addObject("result", resultStr);
        mav.setViewName("/result");

        return mav;
    }
}

```

LessonService.java

```

package com.spring.lessonservice;

import java.util.List;
import com.spring.lessonvo.LessonVO;

public interface LessonService {
    public List<LessonVO> listLesson(LessonVO param);
    public LessonVO selectLesson(int no);
    public int insertLesson(LessonVO param);
    public int updateLesson(LessonVO param);
    public int deleteLesson(int no);
}

```

LessonServiceImpl.java

```

package com.spring.lessonservice;

import java.util.ArrayList;
import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import org.springframework.transaction.annotation.Transactional;
import com.spring.lessondao.LessonMapper;
import com.spring.lessonvo.LessonVO;

@Service
@Transactional
public class LessonServiceImpl implements LessonService {

    @Autowired

```

```
private LessonMapper lessonMapper;
```

```
@Override
```

```
public List<LessonVO> listLesson(LessonVO param) {  
    List<LessonVO> list = new ArrayList<LessonVO>();  
    list = lessonMapper.listLesson(param);  
    return list;  
}
```

```
@Override
```

```
public LessonVO selectLesson(int no) {  
    return lessonMapper.selectLesson(no);  
}
```

```
@Override
```

```
public int insertLesson(LessonVO param) {  
    return lessonMapper.insertLesson(param);  
}
```

```
@Override
```

```
public int updateLesson(LessonVO param) {  
    return lessonMapper.updateLesson(param);  
}
```

```
@Override
```

```
public int deleteLesson(int no) {  
    return lessonMapper.deleteLesson(no);  
}
```

```
}
```

LessonMapper.java

```
package com.spring.lessondao;
```

```
import java.util.List;
```

```
import com.spring.lessonvo.LessonVO;
```

```
public interface LessonMapper {
```

```
    public List<LessonVO> listLesson(LessonVO param);
```

```
    public LessonVO selectLesson(int no);
```

```
    public int insertLesson(LessonVO param);
```

```
    public int updateLesson(LessonVO param);
```

```
    public int deleteLesson(int no);
```

```
}
```

LessonMapperImpl.java

```

package com.spring.lessondao;

import java.util.List;
import javax.inject.Inject;
import org.apache.ibatis.session.SqlSession;
import org.springframework.stereotype.Repository;
import com.spring.lessonvo.LessonVO;

@Repository
public class LessonMapperImpl implements LessonMapper {

    @Inject
    private SqlSession sqlSession;

    private static final String namespace = "com.spring.lessondao.LessonMapper";

    @Override
    public List<LessonVO> listLesson(LessonVO param) {
        return sqlSession.selectList(namespace + ".listLesson", param);
    }

    @Override
    public LessonVO selectLesson(int no) {
        return sqlSession.selectOne(namespace + ".selectLesson", no);
    }

    @Override
    public int insertLesson(LessonVO param) {
        return sqlSession.insert(namespace + ".insertLesson", param);
    }

    @Override
    public int updateLesson(LessonVO param) {
        return sqlSession.update(namespace + ".updateLesson", param);
    }

    @Override
    public int deleteLesson(int no) {
        return sqlSession.delete(namespace + ".deleteLesson", no);
    }

}

```

lesson-mapper.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-mapper.dtd">

```



```

<mapper namespace="com.spring.lessondao.LessonMapper">

    <resultMap type="com.spring.lessonvo.LessonVO" id="lessonVO">
        <result property="no"    column="NO" />
        <result property="lnum"  column="L_NUM" />
        <result property="lname" column="L_NAME" />
    </resultMap>

    <select          id="listLesson"          parameterType="com.spring.lessonvo.LessonVO"
resultMap="lessonVO">
        /* Mapper - listLesson */
        SELECT NO, L_NUM, L_NAME
        FROM   LESSON
        <where>
            <if test="lname != null and lname != ''">
                L_NAME LIKE '%' || #{lname} || '%'
            </if>
        </where>
        ORDER BY NO DESC
    </select>

    <select id="selectLesson" parameterType="int" resultMap="lessonVO">
        /* Mapper - selectLesson */
        SELECT  NO, L_NUM, L_NAME
        FROM    LESSON
        WHERE   NO = #{no}
    </select>

    <insert id="insertLesson" parameterType="com.spring.lessonvo.LessonVO">
        /* Mapper - insertLesson */
        <selectKey keyProperty="no" resultType="int" order="BEFORE">
            select lesson_seq.nextval from dual
        </selectKey>
        INSERT INTO LESSON
            (NO, L_NUM, L_NAME )
        VALUES
            (#{no}, #{lnum}, #{lname})
    </insert>

    <update id="updateLesson" parameterType="com.spring.lessonvo.LessonVO">
        /* Mapper - updateLesson */
        UPDATE LESSON

```

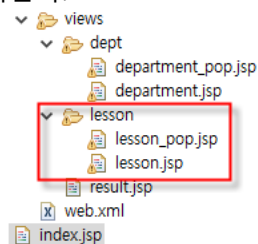
```

        SET    L_NUM    = #{lnum},
              L_NAME    = #{lname}
        WHERE NO        = #{no}
    </update>

    <delete id="deleteLesson" parameterType="int">
        /* Mapper - deleteLesson */
        DELETE FROM LESSON WHERE NO = #{no}
    </delete>
</mapper>

```

⑧ "views" 폴더에 "lesson"폴더를 생성한다. 폴더를 선택하고 [New] - [JSP file]을 선택하고, 파일명으로 "lesson.jsp" 입력한 후, [Finish] 버튼을 클릭한다.



views/lesson/lesson.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>과목 테이블</title>
<script type="text/javascript" src="http://code.jquery.com/jquery-latest.js"> </script>
<script type="text/javascript">
    function insertPopup() {
        $("#no").val(0);
        window.open("", "pop", "width=480, height=250");
        $("#popupForm").attr("action", "/lesson/selectLesson.do");
        $("#popupForm").attr("target", "pop");
        $("#popupForm").submit();
    }

    function updatePopup(no) {
        $("#no").val(no);
        window.open("", "pop", "width=480, height=250");
    }

```

```

        $("#popupForm").attr("action", "/lesson/selectLesson.do");
        $("#popupForm").attr("target", "pop");
        $("#popupForm").submit();
    }

    function listSearch() {
        $("#searchForm").attr("action", "/lesson/listLesson.do");
        $("#searchForm").submit();
    }
</script>
<style type="text/css">
    table{width:570px}
    #searchForm table th{width:114px}
    #searchForm table td{width:104px; text-align:center; }
    #lname{width:328px}
</style>
</head>
<body>
    <form id="popupForm" name="popupForm" method="post">
        <input type="hidden" id="no" name="no"/>
    </form>
    <div>
        <table border="0" cellpadding="1" cellspacing="1">
            <tr>
                <th width="650"> <h4> 과목 테이블[Spring] </h4> </th>
            </tr>
        </table>
    </div>
    <div>
        <table border="1">
            <colgroup>
                <col width="120px" />
                <col width="220px" />
                <col width="120px" />
                <col width="110px"/>
            </colgroup>
            <thead>
                <tr>
                    <th>번호</th>
                    <th>과목코드</th>
                    <th>과목명</th>
                    <th>상세보기</th>
                </tr>
            </thead>
        </table>
    </div>

```

```

        </tr>
    </thead>
    <tbody>
        <c:if test="${empty lessonList}">
            <tr>
                <td colspan="5" align="center">
                    등록된 과목 데이터가 존재하지 않습니다.</td>
            </tr>
        </c:if>
        <c:forEach items="${lessonList}" var="row">
            <tr align="center">
                <td>${row.no}</td>
                <td>${row.lnum}</td>
                <td>${row.lname}</td>
                <td><input type="button" value="[ 수정/삭제 ]"
                    onclick="updatePopup(${row.no});"/></td>
            </tr>
        </c:forEach>
        <tr>
            <td colspan="3" align="center">처리할 버튼을 선택하세요.</td>
            <td align="center">
                <input type="button" value="[ 등록 ]"
                onclick="insertPopup()"/></td>
        </tr>
    </tbody>
</table>
</div>
<p></p>
<div><!-- search -->
    <form id="searchForm" name="searchForm" method="post">
        <div>
            <table border="1" cellpadding="1" cellspacing="1">
                <tr>
                    <th width="100px">과 목 명</th>
                    <td><input type="text" id="lname" name="lname" /></td>
                    <td><input type="button" value="검색"
                        onclick="listSearch()" /></td>
                </tr>
            </table>
        </div>
    </form>
</div>

```

```
</body>
</html>
```

[New] - [JSP file]을 선택하고, 파일명으로 "lesson_pop.jsp" 입력한 후, [Finish] 버튼을 클릭한다.

view/lesson/lesson_pop.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>과목 테이블 팝업</title>
<style type="text/css">
    .required{color:red;}
    table{width:450px;}
    table th{width:120px; text-align:left;}
    input[type="text"]{width:330px;}
</style>
<script type="text/javascript" src="http://code.jquery.com/jquery-latest.js"> </script>
<script type="text/javascript">
    $(document).ready(function(){
        var mode = "${mode}";
        if(mode == "insert") {
            $('#insertData').attr('disabled',false);
            $('#updateData').attr('disabled',true);
            $('#deleteData').attr('disabled',true);
        } else if(mode == "update") {
            $('#insertData').attr('disabled',true);
            $('#updateData').attr('disabled',false);
            $('#deleteData').attr('disabled',false);
        }

        $("#insertData").click(function(){
            if(!validateForm()) return;
            if($("#no").val() == "") $("#no").val(0);
            if(confirm('등록을 진행할까요?')) {
                $("#lessonForm").attr("action", "/lesson/insertLesson.do");
                $("#lessonForm").submit();
            }
        });
    });
```

```

        $("#updateData").click(function(){
            if(!validateForm()) return;
            if(confirm('수정을 진행할까요?')) {
                $("#lessonForm").attr("action", "/lesson/updateLesson.do");
                $("#lessonForm").submit();
            }
        });

        $("#deleteData").click(function(){
            if(confirm('삭제를 진행할까요?')) {
                $("#lessonForm").attr("action", "/lesson/deleteLesson.do");
                $("#lessonForm").submit();
            }
        });

        $("#closeWindow").click(function(){
            window.close();
        });
    });

    function validateForm() {
        if($("#lnum").val().replace(/Ws/g,"") == "") {
            alert('과목코드를 입력해주세요.');
```

return false;

```
        }
        if($("#lname").val().replace(/Ws/g,"") == "") {
            alert('과목명을 입력해주세요.');
```

return false;

```
        }
        return true;
    }
</script>
</head>
<body>
    <p></p>
    <div>
        <form id="lessonForm" name="lessonForm" method="post">
            <input type="hidden" id="no" name="no" value="{lessonVO.no}" />
            <table border="1">
                <thead>
                    <tr>
```

```

        <td colspan="2" align="center">
            <h4>과목 테이블 팝업창 [Spring] </h4> </td>
        </tr>
    </thead>
    <tbody>
    <tr>
        <th><span class="required">*</span>과목코드</th>
        <td><input type="text" id="lnum"
            name="lnum" value="{lessonVO.lnum}" /> </td>
    </tr>
    <tr>
        <th><span class="required">*</span>과목명</th>
        <td><input type="text" id="lname"
            name="lname" value="{lessonVO.lname}" /> </td>
    </tr>
    </tbody>
</table>
</form>
</div>
<div>
<p></p>
<table border="0">
    <tr align="center">
        <td>
            <input type="button" id="insertData" value="등록" /> </td>
        <td>
            <input type="button" id="updateData" value="수정" /> </td>
        <td>
            <input type="button" id="deleteData" value="삭제" /> </td>
        <td>
            <input type="button" id="closeWindow" value="닫기" /> </td>
    </tr>
</table>
</div>
</body>
</html>

```

[New] - [JSP file]을 선택하고, 파일명으로 "index.jsp" 입력한 후, [Finish] 버튼을 클릭한다.

index.jsp (기존 파일에 추가작업)

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>

```

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>스프링 MVC 데이터베이스 연동</title>
<style type="text/css">
    a{text-decoration:none;}
</style>
</head>
<body>
<div>
    <a href="department/listDepartment.do">[학과 데이터]</a>
    <a href="lesson/listLesson.do">[과목 데이터]</a>
</div>
</body>
</html>
```

수정 후 실행본다.