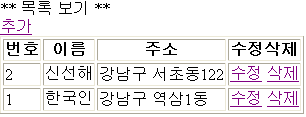
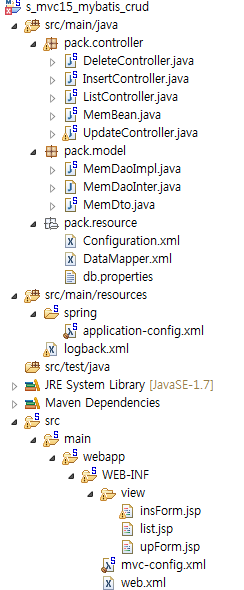
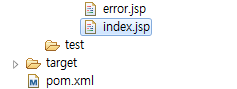
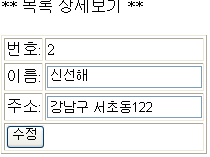
Spring + MyBatis (CRUD) 실습 – Maven





index.jsp

<a href="list">회원목록</a>

web.xml ----<!-- 한글 처리 -->

MemBean.java : 컨트롤러에서 사용

package pack.controller;

public class MemBean {

private String num, name, addr;

getter, setter…

}

MemDto.java : 모델에서 사용

package pack.model;

public class MemDto {

private String num, name, addr;

getter, setter…

}

//--- MyBatis 설정 -------------------------

db.properties

driver=com.mysql.jdbc.Driver

url=jdbc:mysql://localhost:3307/test

username=root

password=1234

DataMapper.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="dev">

<select id="selectDataAll" resultType="dto">

select num,name,addr from mem

</select>

<select id="selectPart" parameterType="string" resultType="dto">

select \* from mem where num=#{num}

</select>

<insert id="insertData" parameterType="formBean">

insert into mem values(#{num},#{name},#{addr})

</insert>

<update id="updateData" parameterType="formBean">

update mem set name=#{name},addr=#{addr}

where num=#{num}

</update>

<delete id="deleteData" parameterType="string">

delete from mem where num=#{num}

</delete>

</mapper>

Configuration.xml

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE configuration

PUBLIC "-//mybatis.org//DTD Config 3.0//EN"

"http://mybatis.org/dtd/mybatis-3-config.dtd">

<configuration>

<typeAliases> <!-- 별명 부여 -->

<typeAlias type="pack.model.MemDto" alias="dto"/>

<typeAlias type="pack.controller.MemBean" alias="formBean"/>

</typeAliases>

<mappers>

<mapper resource="pack/resource/DataMapper.xml" />

</mappers>

</configuration>

application-config.xml

<?xml version="1.0" encoding="UTF-8"?>

<beans …>

<bean class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">

<property name="locations">

<value>classpath:pack/resource/db.properties</value>

<!-- <value>WEB-INF/db.properties</value> -->

</property>

</bean>

<bean id="dataSource"

class="org.springframework.jdbc.datasource.SimpleDriverDataSource">

<property name="driverClass" value="${driver}" />

<property name="url" value="${url}" />

<property name="username" value="${username}" />

<property name="password" value="${password}" />

</bean>

<bean

class="org.mybatis.spring.SqlSessionFactoryBean">

<property name="dataSource" ref="dataSource" />

<property name="configLocation"

value="classpath:pack/resource/Configuration.xml" />

</bean>

</beans>

mvc-config.xml

<?xml version="1.0" encoding="UTF-8"?>

<beans …">

<context:component-scan base-package="pack.model" />

<context:component-scan base-package="pack.controller" />

<mvc:annotation-driven />

<bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="prefix" value="/WEB-INF/view/"/>

<property name="suffix" value=".jsp"/>

</bean>

</beans>

MemDaoInter.java : 인터페이스

package pack.model;

import java.util.List;

import pack.controller.MemBean;

public interface MemDaoInter {

List<MemDto> getDataAll();

MemDto selectPart(String num);

boolean insertData(MemBean bean);

boolean updateData(MemBean bean);

boolean deleteData(String num);

}

MemDaoImpl.java ----------

package pack.model;

import java.util.List;

import javax.annotation.PostConstruct;

…

import pack.controller.MemBean;

@Repository

public class MemDaoImpl extends SqlSessionDaoSupport implements MemDaoInter{

@Autowired

SqlSessionFactory factory;

@PostConstruct

public void setFactory() {

setSqlSessionFactory(factory);

}

public List<MemDto> getDataAll() {

return getSqlSession().selectList("selectDataAll");

}

public boolean insertData(MemBean bean) {

try {

getSqlSession().insert("insertData", bean);

return true;

} catch (Exception e) {

System.out.println("insertData err:" + e);

return false;

}

}

public MemDto selectPart(String num) {

return getSqlSession().selectOne("selectPart", num);

}

public boolean updateData(MemBean bean) {

try {

getSqlSession().update("updateData", bean);

return true;

} catch (Exception e) {

System.out.println("updateData err:" + e);

return false;

}

}

public boolean deleteData(String num) {

int re = getSqlSession().delete("deleteData", num);

if(re > 0 )

return true;

else

return false;

}

}

ListController.java ------------

package pack.controller;

…

import pack.model.MemDaoInter;

@Service

@Controller

public class ListController {

@Autowired

private MemDaoInter daoInter;

@RequestMapping("list")

public ModelAndView list(){

return new ModelAndView("list", "list", daoInter.getDataAll());

}

}

InsertController.java --------

package pack.controller;

…

import pack.model.MemDaoInter;

@Service

@Controller

public class InsertController {

@Autowired

private MemDaoInter daoInter;

@ModelAttribute("command")

public MemBean formBack(){ //기본값으로 처리 시 생략 가능

return new MemBean();

}

@RequestMapping(value="insert", method=RequestMethod.GET)

public String insert(){

return "insForm";

}

@RequestMapping(value="insert", method=RequestMethod.POST)

public String submit(MemBean bean){

boolean b = daoInter.insertData(bean);

if(b) {

return "redirect:/list";

}else{

return "redirect:/error.jsp";

}

}

}

UpdateController.java ---------

package pack.controller;

…

import pack.model.\*;

@Service

@Controller

public class UpdateController {

@Autowired

private MemDaoInter daoInter;

@RequestMapping(value="update", method=RequestMethod.GET)

public ModelAndView up(@RequestParam("num")String num){

MemDto dto = daoInter.selectPart(num);

return new ModelAndView("upForm", "dto", dto);

}

@RequestMapping(value="update", method=RequestMethod.POST)

public String submit(MemBean bean){

boolean b = daoInter.updateData(bean);

if(b) {

return "redirect:/list";

}else{

return "redirect:/error.jsp";

}

}

}

DeleteController.java ---

package pack.controller;

…

import pack.model.MemDaoInter;

@Service

@Controller

public class DeleteController {

@Autowired

private MemDaoInter daoInter;

@RequestMapping("delete")

public String del(@RequestParam("num")String num){

if(daoInter.deleteData(num)){

return "redirect:/list";

}else{

return "redirect:/error.jsp";

}

}

}

--- View 관련 ------------------------------

list.jsp -----

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

<html>

<body>

\*\* 회원 정보 목록(@MVC-MyBatis-CRUD) \*\*<br/>

<a href="insert">추가</a><br/>

<table border="1">

<tr><th>번호</th><th>이름</th><th>주소</th><th>변경</th></tr>

<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<c:forEach var="m" items="${list}">

<tr>

<td>${m.num}</td>

<td>${m.name}</td>

<td>${m.addr}</td>

<td>

<a href="update?num=${m.num}">수정</a>

<a href="delete?num=${m.num}">삭제</a>

</td>

</tr>

</c:forEach>

</table>

</body></html>

insForm.jsp -----

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

<html>

<body>

\*\* 회원 입력 \*\*<br/>

<form action="insert" commandName="command" method="post">

번호 : <input type="text" name="num"><br/>

이름 : <input type="text" name="name"><br/>

주소 : <input type="text" name="addr"><br/>

<input type="submit" value="등록">

</form>

</body>

</html>

error.jsp

<html><body>에러 발생!</body></html>

upForm.jsp -----

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

<html>

<body>\*\* 회원 자료 수정 \*\*<br/>

<form action="update" method="post">

번호 : ${dto.num}<br/>

<input type="hidden" name="num" value="${dto.num}">

이름 : <input type="text" name="name" value="${dto.name}"><br/>

주소 : <input type="text" name="addr" value="${dto.addr}"><br/>

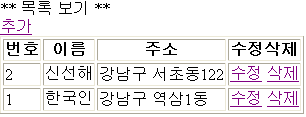
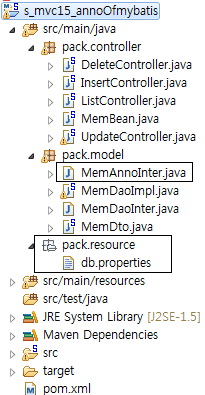
<input type="submit" value="수정">

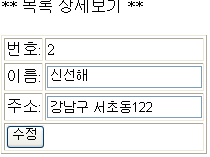
</form>

</body>

</html>

Spring + MyBatis(Annotation으로 SQL기술) - (CRUD) 실습



web.xml -----

MemBean.java : 컨트롤러에서 사용

MemDto.java : 모델에서 사용

db.properties

mvc-config.xml

application-config.xml

<?xml version="1.0" encoding="UTF-8"?>

<beans …>

<bean class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">

<property name="locations">

<value>classpath:pack/resource/db.properties</value>

<!-- <value>WEB-INF/db.properties</value> -->

</property>

</bean>

<bean id="dataSource"

class="org.springframework.jdbc.datasource.SimpleDriverDataSource">

<property name="driverClass" value="${driver}" />

<property name="url" value="${url}" />

<property name="username" value="${username}" />

<property name="password" value="${password}" />

</bean>

<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">

<property name="dataSource" ref="dataSource" />

</bean>

<bean class="org.mybatis.spring.mapper.MapperFactoryBean">

<property name="mapperInterface"

value="pack.model.MemAnnoInter" />

<property name="sqlSessionFactory" ref="sqlSessionFactory" />

</bean>

</beans>

MemAnnoInter.java : 인터페이스 - SQL문을 이 곳에서 맵핑 – Annotation 처리

package pack.model;

import java.util.List;

…

import pack.controller.MemBean;

public interface MemAnnoInter {

@Select("select num,name,addr from mem")

public List<MemDto> getdataAll();

@Select("select \* from mem where num=#{num}")

public MemDto selectPart(String num);

@Insert("insert into mem values(#{num},#{name},#{addr})")

public int insertData(MemBean bean);

@Update("update mem set name=#{name},addr=#{addr} where num=#{num}")

public int updateData(MemBean bean);

@Delete("delete from mem where num=#{num}")

public int deleteData(String num);

}

MemDaoImpl.java

package pack.model;

import java.util.List;

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

import org.springframework.stereotype.Repository;

import pack.controller.MemBean;

@Repository

public class MemDaoImpl implements MemDaoInter{

@Autowired

MemAnnoInter annoInter;

public List<MemDto> getDataAll() {

return annoInter.getdataAll();

}

public boolean insertData(MemBean bean) {

int re = annoInter.insertData(bean);

if(re > 0){

return true;

}else{

return false;

}

}

public MemDto selectPart(String num) {

return annoInter.selectPart(num);

}

public boolean updateData(MemBean bean) {

int re = annoInter.updateData(bean);

if(re > 0){

return true;

}else{

return false;

}

}

public boolean deleteData(String num) {

int re = annoInter.deleteData(num);

if(re > 0 )

return true;

else

return false;

}

}

ListController.java

InsertController.java

UpdateController.java

DeleteController.java

--- View 관련 ------------------------------

앞예제와 동일