[ 18 ] 회원인증, fileUpload, MyBatis

1. 요구사항 : 회원가입(가입시 축하메일), 로그인, 로그아웃, 회원정보수정 가능

책이미지를 파일첨부하여 책등록, 책상세보기, 책정보수정, 책리스트(페이징처리)

1. 테이블생성 ; src/main/resources/META-INF/db/initDB.sql

DROP TABLE BOOK CASCADE CONSTRAINTS;

DROP TABLE MEMBER CASCADE CONSTRAINTS;

DROP SEQUENCE BOOK\_SQ;

CREATE SEQUENCE BOOK\_SQ MAXVALUE 99999999 NOCYCLE NOCACHE;

CREATE TABLE BOOK(

bNUM NUMBER(8) PRIMARY KEY,

bTITLE VARCHAR2(100) NOT NULL,

bWRITER VARCHAR2(100) NOT NULL,

bRDATE DATE DEFAULT SYSDATE NOT NULL,

bIMG1 VARCHAR2(100) DEFAULT 'noImg.png' NOT NULL,

bIMG2 VARCHAR2(100) DEFAULT 'noImg.png' NOT NULL,

bINFO VARCHAR2(1000)

);

CREATE TABLE MEMBER(

mID VARCHAR2(100) PRIMARY KEY,

mPW VARCHAR2(100) NOT NULL,

mNAME VARCHAR2(100) NOT NULL,

mMAIL VARCHAR2(100) NOT NULL,

mPOST VARCHAR2(100), -- 우편번호

mADDR VARCHAR2(100) -- 주소

);

1. 더미데이터 입력 ; src/main/resources/META-INF/db/dummyData.sql

-- Book 테이블

INSERT INTO BOOK VALUES

(BOOK\_SQ.NEXTVAL, 'Spring','김스프',sysdate, 'noImg.png','noImg.png','좋은스프링책');

-- Member 테이블

INSERT INTO MEMBER VALUES ('aaa','1','홍길동','yisy0703@naver.com','12345','종로');

commit;

1. Mapper에 들어갈 sql ; src/main/resources/META-INF/db/mappers.sql

-- ★ Book.xml(신규순list, 도서보기(paging), 책갯수, 상세보기, 도서등록, 도서정보수정)

-- Book.xml id = mainList(신규도서순으로 list가져오기)

SELECT \* FROM BOOK ORDER BY BRDATE DESC;

-- Book.xml id = bookList (paging처리해서 도서이름 순으로 list가져오기)

SELECT \* FROM BOOK ORDER BY BTITLE;

SELECT \* FROM (SELECT ROWNUM RN, A.\* FROM (SELECT \* FROM BOOK ORDER BY BTITLE) A)

WHERE RN BETWEEN 1 AND 3;

-- Book.xml id = cntBook (등록된 책 갯수가져오기)

SELECT COUNT(\*) FROM BOOK;

-- Book.xml id = getBook (책번호bnum으로 dto가져오기)

SELECT \* FROM BOOK WHERE BNUM=1;

-- Book.xml id = registerBook (책 등록하기)

INSERT INTO BOOK (bnum, btitle, bwriter, brdate, bimg1, bimg2, binfo) VALUES

(BOOK\_SEQ.NEXTVAL, '오라클','오길동',SYSDATE, 'noImg.png', 'noImg.png','좋은 spring책');

INSERT INTO BOOK (bnum, btitle, bwriter, brdate, bimg1, binfo) VALUES

(BOOK\_SEQ.NEXTVAL, 'Java','윤길동',SYSDATE, 'noImg.png', '좋은 spring책');

INSERT INTO BOOK (bnum, btitle, bwriter, brdate, bimg2, binfo) VALUES

(BOOK\_SEQ.NEXTVAL, 'JSP','제이피',SYSDATE, 'noImg.png','좋은 spring책');

INSERT INTO BOOK (bnum, btitle, bwriter, brdate, binfo) VALUES

(BOOK\_SEQ.NEXTVAL, 'PYTHON','팽길동',SYSDATE, '좋은 spring책');

-- Book.xml id = modifyBook (책정보수정하기)

UPDATE BOOK SET

btitle='jsp servlet',

bwriter='최',

brdate=sysdate,

bimg1='noImg.png',

bimg2='noImg.png',

binfo='좋은 책'

WHERE BNUM=2;

-- ★ Member.xml(id중복체크, 가입, id로 dto가져오기, 정보수정)

-- Member.xml id=idConfirm (해당id가 몇개인지)

SELECT COUNT(\*) FROM MEMBER WHERE MID='bbb';

-- Member.xml id = joinMember (회원가입)

INSERT INTO MEMBER VALUES

('bbb','1','홍길동','hybrid0506@gmail.com', '서울시 종로구','01234');

-- Member.xml id = getMember (mid로 dto가져오기)

SELECT \* FROM MEMBER WHERE MID='bbb';

-- Member.xml id = modifyMember (회원정보수정)

UPDATE MEMBER SET MPW='111',

MNAME='유길동',

MMAIL = 'hybrid0506@gmail.com',

MADDR='서울시 용산구',

MPOST = '01234'

WHERE MID='bbb';

1. pom.xml에 의존추가 부분

<!-- JDBC -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-jdbc</artifactId>

<version>4.1.4.RELEASE</version>

</dependency>

<!-- myBatis -->

<dependency>

<groupId>org.mybatis</groupId>

<artifactId>mybatis</artifactId>

<version>3.2.8</version>

</dependency>

<dependency>

<groupId>org.mybatis</groupId>

<artifactId>mybatis-spring</artifactId>

<version>1.2.2</version>

</dependency>

<!-- file upload -->

<dependency>

<groupId>commons-fileupload</groupId>

<artifactId>commons-fileupload</artifactId>

<version>1.3.1</version>

</dependency>

<!-- Mail sender -->

<dependency>

<groupId>javax.mail</groupId>

<artifactId>mail</artifactId>

<version>1.4</version>

</dependency>

1. web.xml에 한글처리부분 및 \*.do처리 부분

<servlet-mapping>

<servlet-name>appServlet</servlet-name>

<url-pattern>\*.do</url-pattern>

</servlet-mapping>

<!-- 한글처리 -->

<filter>

<filter-name>encodingFilter</filter-name>

<filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>

<init-param>

<param-name>encoding</param-name>

<param-value>UTF-8</param-value>

</init-param>

<init-param>

<param-name>forceEncoding</param-name>

<param-value>true</param-value>

</init-param>

</filter>

<filter-mapping>

<filter-name>encodingFilter</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

1. index.jsp의 body

<body>

<jsp:forward page=*"main.do"*/>

</body>

1. MainController.java

@Controller

**public** **class** MainController {

@RequestMapping(value="main")

**public** String main() {

**return** "main/main";

}

}

1. main/main.jsp

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

pageEncoding=*"UTF-8"*%>

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

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

<c:set var=*"conPath"* value=*"*${pageContext.request.contextPath }*"*/>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<title>Insert title here</title>

<link href=*"*${conPath }*/css/style.css"* rel=*"stylesheet"*>

<style>

*#content* {

width: *800px*; height:*350px*;

margin: *50px auto*;

}

**img**{width: *100px*;}

</style>

</head>

<body>

<c:if test=*"*${**not empty** modifyResult }*"*>

<script> alert("정보 수정이 돤료 되었습니다");</script>

</c:if>

<jsp:include page=*"header.jsp"*/>

<div id=*"content"*>

<table>

<caption>신규 도서 BEST 5</caption>

<tr><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th></tr>

<tr>

<c:forEach var=*"book"* items=*"*${main }*"* begin=*"0"* end=*"4"*>

<td>

${book.btitle }<br>

<a href=*"*${conPath }*/book.do?method=detail&bnum=*${book.bnum}*&pageNum=1"*>

<img src=*"*${conPath }*/bookImgFileUpload/*${book.bimg1}*"*></a><br>

${book.bwriter } 著 <br>

출판일 : ${book.brdate }

</td>

</c:forEach>

</tr>

</table>

</div>

<jsp:include page=*"footer.jsp"*/>

</body>

</html>

1. main/heard.jsp

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

pageEncoding=*"UTF-8"*%>

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

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

<c:set var=*"conPath"* value=*"*${pageContext.request.contextPath }*"* />

<!DOCTYPE html>

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<title>Insert title here</title>

<style>

\* {

padding: *0*; margin: *0*;

}

**body** {

background-color: *#FFF0B5*; font-size: *9pt*;

}

**header** **a** {

text-decoration: *none*; font-weight: *bold*;

}

**header** **li** { list-style: *none*;}

**header** *.gnb* {

width: *100%*; background-color: *#003300*;

}

**header** *.gnb* **ul** {

overflow: *hidden*;

width: *700px*;

height: *30px*;

line-height: *30px*;

margin: *0 auto*;

}

**header** *.gnb* **ul** **li** {

float: *right*; margin-right: *30px*;

}

**header** *.gnb* **a** {

color: *#FFF0B5*;

font-size: *0.9em*;

display: *block*;

padding-left: *15px*;

padding-right: *15px*;

}

**header** *.logo* {

width: *120px*;

height: *20px*;

line-height: *20px*;

text-align: *center*;

margin: *15px auto*;

border: *1px solid #003300*;

cursor: *pointer*;

}

**header** *.lnb* { width: *100%*;}

**header** *.lnb* **ul** {

overflow: *hidden*; width: *300px*; margin: *0 auto*;

}

**header** *.lnb* **ul** **li** {

float: *left*;

padding: *5px 20px*;

line-height: *25px*;

border-top: *1px dashed #003300*;

border-bottom: *1px dashed #003300*;

}

**header** *.lnb* **a** {

height: *1.1em*;

color: *#003300*;

font-size: *1.1em*;

margin-bottom: *20px*;

}

</style>

</head>

<body>

<header>

<div class=*"gnb"*>

<c:if test=*"*${**empty** member}*"*>

<ul>

<li><a href=*"*${conPath }*/member.do?method=joinForm"*>회원가입</a></li>

<li><a href=*"*${conPath }*/member.do?method=loginForm"*>로그인</a></li>

</ul>

</c:if>

<c:if test=*"*${**not empty** member}*"*>

<ul>

<li><a href=*"*${conPath }*/member.do?method=modifyForm"*>정보수정</a></li>

<li><a href=*"*${conPath }*/member.do?method=logout"*>로그아웃</a></li>

<li><a>${member.mname }님</a></li>

</ul>

</c:if>

</div>

<div class=*"logo"* onclick=*"location.href='*${conPath}*/main.do'"*>

LOGO

</div>

<div class=*"lnb"*>

<ul>

<li><a href=*"*${conPath}*/main.do"*>홈</a>

<li><a href=*"*${conPath }*/book.do?method=list&pageNum=1"*>도서보기</a></li>

<li><a href=*"*${conPath }*/book.do?method=registerForm"*>도서등록</a>

</li>

</ul>

</div>

</header>

</body>

</html>

1. main/footer.jsp

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

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<title>Insert title here</title>

</head>

<style>

**footer** {

height:*60px*; background-color: *#003300*;

}

**footer** *#footer\_conts* {

color:*white*;

font-weight: *blod*;

margin:*0 auto*;

height: *60px*;

padding:*10px 30px 0*;

font-size:*0.9em*;

text-align: *center*;

}

</style>

<body>

<footer>

<div id=*"footer\_conts"*>

(주)더조은803 | 전화 : 02-123-1234 | 팩스 : 070-1234-1234 | 메일 : help@tj.com <br>

서울특별시 종로구 관철동

</div>

</footer>

</body>

</html>

1. src/main/resources/META-INF/property/db.properties

db.driverClassName=oracle.jdbc.driver.OracleDriver

db.url=jdbc:oracle:thin:@localhost:1521:xe

db.username=scott

db.password=tiger

1. servlet-context.xml

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

<beans:beans xmlns=*"http://www.springframework.org/schema/mvc"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xmlns:beans=*"http://www.springframework.org/schema/beans"*

xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/mvc https://www.springframework.org/schema/mvc/spring-mvc.xsd*

*http://www.springframework.org/schema/beans https://www.springframework.org/schema/beans/spring-beans.xsd*

*http://www.springframework.org/schema/context https://www.springframework.org/schema/context/spring-context.xsd"*>

<!-- DispatcherServlet Context: defines this servlet's request-processing infrastructure -->

<!-- Enables the Spring MVC @Controller programming model -->

<annotation-driven />

<!-- Handles HTTP GET requests for /resources/\*\* by efficiently serving up static resources in the ${webappRoot}/resources directory -->

<resources mapping=*"/css/\*\*"* location=*"/css/"* />

<resources mapping=*"/js/\*\*"* location=*"/js/"* />

<resources mapping=*"/resources/\*\*"* location=*"/resources/"* />

<!-- Resolves views selected for rendering by @Controllers to .jsp resources in the /WEB-INF/views directory -->

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

<beans:property name=*"prefix"* value=*"/WEB-INF/views/"* />

<beans:property name=*"suffix"* value=*".jsp"* />

</beans:bean>

<context:component-scan base-package=*"com.tj.ch18"* />

<!-- 파일첨부 -->

<beans:bean id=*"multipartResolver"* class=*"org.springframework.web.multipart.commons.CommonsMultipartResolver"*>

<beans:property name=*"maxUploadSize"* value=*"10485760"*/>

</beans:bean>

<!-- 메일발송 -->

<beans:bean id=*"mailSender"*

class=*"org.springframework.mail.javamail.JavaMailSenderImpl"*>

<beans:property name=*"host"* value=*"smtp.gmail.com"*/>

<beans:property name=*"port"* value=*"587"*/>

<beans:property name=*"username"* value=*"hybrid0506"*/>

<beans:property name=*"password"* value=*"0506hybrid"*/>

<beans:property name=*"javaMailProperties"*>

<beans:props>

<beans:prop key=*"mail.smtp.auth"*>true</beans:prop>

<beans:prop key=*"mail.smtp.starttls.enable"*>true</beans:prop>

<beans:prop key=*"mail.debug"*>true</beans:prop>

</beans:props>

</beans:property>

</beans:bean>

</beans:beans>

1. root-context.xml

<?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"*

xmlns:p=*"http://www.springframework.org/schema/p"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans https://www.springframework.org/schema/beans/spring-beans.xsd*

*http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-3.1.xsd"*>

<!-- Root Context: defines shared resources visible to all other web components -->

<context:property-placeholder

location=*"classpath:META-INF/property/db.properties"* />

<bean id=*"dataSource"*

class=*"org.springframework.jdbc.datasource.DriverManagerDataSource"*>

<property name=*"driverClassName"* value=*"${db.driverClassName}"* />

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

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

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

</bean>

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

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

<property name="configLocation" value="classpath:META-INF/sqlmap/config/mybatis-config.xml" />

</bean>

<bean id="sqlSessionTemplate" class="org.mybatis.spring.SqlSessionTemplate">

<constructor-arg index="0" ref="sqlSessionFactory"/>

</bean> -->

</beans>

1. 여기까지 하고 index.jsp 나 프로젝트를 실행한다. 실행되면 다음을 진행
2. BookDto.java

**import** java.sql.Date;

**public** **class** Book {

**private** **int** bnum; // 책번호

**private** String btitle; // 책이름

**private** String bwriter;// 책저자

**private** Date brdate; // 출판일

**private** String bimg1; // 책이미지1 (첨부안하면 noimg.jpg로 할 예정)

**private** String bimg2; // 책이미지2 (첨부안하면 noimg.jpg로 할 예정)

**private** String binfo; // 책소개

**private** **int** startRow;

**private** **int** endRow;

**public** **int** getBnum() {**return** bnum;}

**public** **void** setBnum(**int** bnum) {**this**.bnum = bnum;}

**public** String getBtitle() {**return** btitle;}

**public** **void** setBtitle(String btitle) {**this**.btitle = btitle;}

**public** String getBwriter() {**return** bwriter;}

**public** **void** setBwriter(String bwriter) {**this**.bwriter = bwriter;}

**public** Date getBrdate() {**return** brdate;}

**public** **void** setBrdate(Date brdate) {**this**.brdate = brdate;}

**public** String getBimg1() {**return** bimg1;}

**public** **void** setBimg1(String bimg1) {**this**.bimg1 = bimg1;}

**public** String getBimg2() {**return** bimg2;}

**public** **void** setBimg2(String bimg2) {**this**.bimg2 = bimg2;}

**public** String getBinfo() {**return** binfo;}

**public** **void** setBinfo(String binfo) {**this**.binfo = binfo;}

**public** **int** getStartRow() {**return** startRow;}

**public** **void** setStartRow(**int** startRow) {**this**.startRow = startRow;}

**public** **int** getEndRow() {**return** endRow;}

**public** **void** setEndRow(**int** endRow) {**this**.endRow = endRow;}

@Override

**public** String toString() {

**return** "Book [bnum=" + bnum + ", btitle=" + btitle + ", bwriter=" + bwriter + ", brdate=" + brdate + ", bimg1="

+ bimg1 + ", bimg2=" + bimg2 + ", binfo=" + binfo + ", startRow=" + startRow + ", endRow=" + endRow

+ "]";

}

}

1. MemberDto.java

**public** **class** Member {

**private** String mid;

**private** String mpw;

**private** String mname;

**private** String mmail; //메일

**private** String maddr; //주소

**private** String mpost; //우편번호

**public** String getMid() {**return** mid;}

**public** **void** setMid(String mid) {**this**.mid = mid;}

**public** String getMpw() {**return** mpw;}

**public** **void** setMpw(String mpw) {**this**.mpw = mpw;}

**public** String getMname() {**return** mname;}

**public** **void** setMname(String mname) {**this**.mname = mname;}

**public** String getMmail() {**return** mmail;}

**public** **void** setMmail(String mmail) {**this**.mmail = mmail;}

**public** String getMaddr() {**return** maddr;}

**public** **void** setMaddr(String maddr) {**this**.maddr = maddr;}

**public** String getMpost() {**return** mpost;}

**public** **void** setMpost(String mpost) {**this**.mpost = mpost;}

@Override

**public** String toString() {

**return** "Member [mid=" + mid + ", mpw=" + mpw + ", mname=" + mname + ", mmail=" + mmail + ", maddr=" + maddr

+ ", mpost=" + mpost + "]";

}

}

1. Book.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=*"Book"*>

<resultMap type=*"Book"* id=*"BookResult"*>

<result property=*"bnum"* column=*"bnum"*/>

<result property=*"btitle"* column=*"btitle"*/>

<result property=*"bwriter"* column=*"bwriter"*/>

<result property=*"brdate"* column=*"brdate"*/>

<result property=*"bimg1"* column=*"bimg1"*/>

<result property=*"bimg2"* column=*"bimg2"*/>

<result property=*"binfo"* column=*"binfo"*/>

</resultMap>

<!-- 책들 모든 리스트(입력 순) -->

<select id=*"mainList"* resultMap=*"BookResult"*>

SELECT \* FROM BOOK order by brdate DESC

</select>

<!-- 책이름 가나다순으로 PAGING처리 -->

<select id=*"bookList"* parameterType=*"Book"* resultMap=*"BookResult"*>

SELECT \* FROM (SELECT ROWNUM RN, A.\*

FROM (SELECT \* FROM BOOK order by btitle) A)

WHERE RN BETWEEN #{startRow} AND #{endRow}

</select>

<!-- 페이징 때 쓰려고 등록된 책 갯수-->

<select id=*"cntBook"* resultType=*"int"*>

SELECT COUNT(\*) FROM BOOK

</select>

<select id=*"getBook"* parameterType=*"int"* resultType=*"Book"*>

<![CDATA[

SELECT \* FROM BOOK WHERE BNUM=#{bnum}

]]>

</select>

<insert id=*"registerBook"* parameterType=*"Book"*>

<if test=*"bimg1=='' and bimg2!=''"*>

INSERT INTO BOOK (BNUM, BTITLE, BWRITER, BRDATE, BIMG2, BINFO) VALUES

(BOOK\_SEQ.NEXTVAL, #{btitle},#{bwriter},#{brdate}, #{bimg2},#{binfo})

</if>

<if test=*"bimg1!='' and bimg2==''"*>

INSERT INTO BOOK (BNUM, BTITLE, BWRITER, BRDATE, BIMG1, BINFO) VALUES

(BOOK\_SEQ.NEXTVAL, #{btitle},#{bwriter},#{brdate},#{bimg1},#{binfo})

</if>

<if test=*"bimg1=='' and bimg2==''"*>

INSERT INTO BOOK (BNUM, BTITLE, BWRITER, BRDATE, BINFO) VALUES

(BOOK\_SEQ.NEXTVAL, #{btitle},#{bwriter},#{brdate},#{binfo})

</if>

<if test=*"bimg1!='' and bimg2!=''"*>

INSERT INTO BOOK VALUES

(BOOK\_SEQ.NEXTVAL, #{btitle},#{bwriter},#{brdate}, #{bimg1},#{bimg2},#{binfo})

</if>

</insert>

<update id=*"modifyBook"* parameterType=*"Book"*>

UPDATE BOOK SET

BTITLE=#{btitle},

BWRITER=#{bwriter},

BRDATE=#{brdate},

<if test=*"bimg1!=''"*>BIMG1=#{bimg1},</if>

<if test=*"bimg2!=''"*>BIMG2=#{bimg2},</if>

BINFO=#{binfo}

WHERE BNUM=#{bnum}

</update>

</mapper>

1. Member.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=*"Member"*>

<resultMap type=*"Member"* id=*"MemberResult"*>

<result property=*"mid"* column=*"mid"*/>

<result property=*"mpw"* column=*"mpw"*/>

<result property=*"mname"* column=*"mname"*/>

<result property=*"mmail"* column=*"mmail"*/>

<result property=*"maddr"* column=*"maddr"*/>

<result property=*"mpost"* column=*"mpost"*/>

</resultMap>

<select id=*"idConfirm"* parameterType=*"String"* resultType=*"int"*>

SELECT COUNT(\*) FROM MEMBER WHERE MID=#{mid, jdbcType=VARCHAR}

</select>

<insert id=*"joinMember"* parameterType=*"Member"*>

INSERT INTO MEMBER VALUES

(#{mid},#{mpw},#{mname},#{mmail},#{maddr},#{mpost})

</insert>

<select id=*"getMember"* parameterType=*"String"* resultType=*"Member"*>

SELECT \* FROM MEMBER WHERE MID=#{mid}

</select>

<update id=*"modifyMember"* parameterType=*"Member"*>

UPDATE MEMBER SET

MPW=#{mpw},

MNAME=#{mname},

MMAIL=#{mmail},

MADDR=#{maddr},

MPOST=#{mpost}

WHERE MID=#{mid}

</update>

</mapper>

1. mybatis-config.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>

<typeAlias type=*"com.tj.ch18.dto.Book"* alias=*"Book"*/>

<typeAlias type=*"com.tj.ch18.dto.Member"* alias=*"Member"*/>

</typeAliases>

<mappers>

<mapper resource=*"META-INF/sqlmap/mappers/Book.xml"*/>

<mapper resource=*"META-INF/sqlmap/mappers/Member.xml"*/>

</mappers>

</configuration>

1. root-context.xml

<?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 https://www.springframework.org/schema/beans/spring-beans.xsd*

*http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-3.1.xsd"*>

<!-- Root Context: defines shared resources visible to all other web components -->

<context:property-placeholder

location=*"classpath:META-INF/property/db.properties"* />

<bean id=*"dataSource"*

class=*"org.springframework.jdbc.datasource.DriverManagerDataSource"*>

<property name=*"driverClassName"* value=*"${db.driverClassName}"* />

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

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

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

</bean>

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

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

<property name=*"configLocation"* value=*"classpath:META-INF/sqlmap/config/mybatis-config.xml"* />

</bean>

<bean id=*"sqlSessionTemplate"* class=*"org.mybatis.spring.SqlSessionTemplate"*>

<constructor-arg index=*"0"* ref=*"sqlSessionFactory"*/>

</bean>

</beans>

1. 여기까지 하고 index.jsp 나 프로젝트를 실행한다. 실행되면 다음을 진행
2. BookDao.java

**public** **interface** BookDao {

**public** List<Book> mainList();

**public** List<Book> bookList(Book book);

**public** Book getBook(**int** bnum);

**public** **int** registerBook(Book book);

**public** **int** modifyBook(Book book);

**public** **int** cntBook();

}

1. BookDaoImpl.java

@Repository

**public** **class** BookDaoImpl **implements** BookDao{

@Autowired

**private** SqlSession sessionTemplate;

@Override

**public** List<Book> mainList() {

**return** sessionTemplate.selectList("mainList");

}

@Override

**public** List<Book> bookList(Book book) {

**return** sessionTemplate.selectList("bookList", book);

}

@Override

**public** Book getBook(**int** bnum) {

**return** sessionTemplate.selectOne("getBook", bnum);

}

@Override

**public** **int** registerBook(Book book) {

**return** sessionTemplate.insert("registerBook", book);

}

@Override

**public** **int** modifyBook(Book book) {

**return** sessionTemplate.update("modifyBook", book);

}

@Override

**public** **int** cntBook() {

**return** sessionTemplate.selectOne("cntBook");

}

}

1. MemberDao.java

**public** **interface** MemberDao {

**public** **int** idConfirm(String mid);

**public** **int** joinMember(Member member);

**public** Member getMember(String mid);

**public** **int** modifyMember(Member member);

}

1. MemberDaoImpl.java

@Repository

**public** **class** MemberDaoImpl **implements** MemberDao {

@Autowired

**private** SqlSession sessionTemplate;

@Override

**public** **int** idConfirm(String mid) {

**return** sessionTemplate.selectOne("idConfirm", mid);

}

@Override

**public** **int** joinMember(Member member) {

**return** sessionTemplate.insert("joinMember", member);

}

@Override

**public** Member getMember(String mid) {

**return** sessionTemplate.selectOne("getMember", mid);

}

@Override

**public** **int** modifyMember(Member member) {

// **TODO** Auto-generated method stub

**return** sessionTemplate.update("modifyMember", member);

}

}

1. MemberService.java

**public** **interface** MemberService {

**public** Member getMember(String mid);

**public** **int** idConfirm(String mid);

**public** **int** joinMember(Member member, HttpSession httpSession);

**public** String loginCheck(String mid, String mpw, HttpSession httpSession);

**public** **int** modifyMember(Member member);

}

1. MemberServiceImpl.java

**import** javax.mail.Message;

**import** javax.mail.internet.InternetAddress;

**import** javax.mail.internet.MimeMessage;

**import** javax.servlet.http.HttpSession;

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

**import** org.springframework.mail.javamail.JavaMailSenderImpl;

**import** org.springframework.mail.javamail.MimeMessagePreparator;

**import** org.springframework.stereotype.Service;

**import** com.tj.ch18.dao.MemberDao;

**import** com.tj.ch18.dto.Member;

@Service

**public** **class** MemberServiceImpl **implements** MemberService{

@Autowired

**private** MemberDao memberDao;

@Autowired

**private** JavaMailSenderImpl mailSender;

@Override

**public** Member getMember(String mid) {

**return** memberDao.getMember(mid);

}

@Override

**public** **int** idConfirm(String mid) {

**return** memberDao.idConfirm(mid);

}

@Override

**public** **int** joinMember(**final** Member member, HttpSession httpSession) {

// member.getmmail()에게 가입감사메일 보내기

MimeMessagePreparator message = **new** MimeMessagePreparator() {

String content =

" <h1>"+member.getMname()+"님의 회원가입을 축하드립니다</h1>"+

" <div>" +

" <div>" +

" 아무개 싸이트에서만 쓰실 수 있는 쿠폰을 드립니다\n\n" +

" <hr color=\"red\"/>" +

" </div>" +

" <did>" +

" <p style=\"color:blue;\" >파란색 감사</p>" +

" <p style=\"color:red;\">빨간색 감사</p>" +

" </did>" +

" </div>" +

" <div>" +

" <img src=\"https://t1.daumcdn.net/daumtop\_chanel/op/20170315064553027.png\" />\n" +

" </div>" +

" <div style=\"font-weight: blod; font-size: 1.2em;\">아무개 사이트 일동</div>\n" +

" <div style=\"color:red;\">서울시 종로구 수표로 육의전BD 803호</div>";

@Override

**public** **void** prepare(MimeMessage mimeMessage) **throws** Exception {

mimeMessage.setRecipient(Message.RecipientType.***TO***,

**new** InternetAddress(member.getMmail()));

mimeMessage.setFrom(**new** InternetAddress("hybrid0506@gmail.com "));

mimeMessage.setSubject(member.getMname()+"님 회원가입감사합니다(HTML)");

mimeMessage.setText(content, "utf-8", "html");

}

};

mailSender.send(message);

httpSession.setAttribute("mid", member.getMid());

**return** memberDao.joinMember(member);

}

@Override

**public** String loginCheck(String mid, String mpw, HttpSession httpSession) {

String result = "로그인 성공";

Member member = memberDao.getMember(mid);

**if**(member == **null**) {

result = "유효하지 않은 아이디입니다";

}**else** **if**(!member.getMpw().equals(mpw)) {

result = "비밀번호가 맞지 않습니다";

}**else** {

// 로그인 성공

httpSession.setAttribute("member", member);

httpSession.setAttribute("mid", mid);

}

**return** result;

}

@Override

**public** **int** modifyMember(Member member) {

**return** memberDao.modifyMember(member);

}

}

1. BookService.java

**public** **interface** BookService {

**public** List<Book> mainList();

**public** List<Book> bookList(Book book);

**public** Book getBook(**int** bnum);

**public** **int** registerBook(MultipartHttpServletRequest mRequest, Book book);

//public int registerBook(MultipartHttpServletRequest mRequest); 위나 아래 둘중에 하나

//public int modifyBook(MultipartHttpServletRequest mRequest, Book book);

**public** **int** modifyBook(MultipartHttpServletRequest mRequest);

**public** **int** cntBook();

}

1. BookServiceImpl.java

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**import** java.sql.Date;

**import** java.util.Iterator;

**import** java.util.List;

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

**import** org.springframework.stereotype.Service;

**import** org.springframework.web.multipart.MultipartFile;

**import** org.springframework.web.multipart.MultipartHttpServletRequest;

**import** com.tj.ch18.dao.BookDao;

**import** com.tj.ch18.dto.Book;

@Service

**public** **class** BookServiceImpl **implements** BookService {

@Autowired

**private** BookDao bookDao;

String backupPath = "D:/mega\_IT/source/9\_Spring/ch18/src/main/webapp/bookImgFileUpload/";

**private** **int** fileCopy(String serverFile, String backupFile) {

**int** isCopy = 0;

FileInputStream is = **null**;

FileOutputStream os = **null**;

**try** {

is = **new** FileInputStream(serverFile);

os = **new** FileOutputStream(backupFile);

File sFile = **new** File(serverFile);

**byte**[] buff = **new** **byte**[(**int**)sFile.length()];

**while**(**true**) {

**int** nRead = is.read(buff);

**if**(nRead== -1) **break**;

os.write(buff, 0, nRead);

}

isCopy = 1;

} **catch** (IOException e) {

System.***out***.println(e.getMessage());

} **finally** {

**try** {

**if**(os!=**null**) os.close();

**if**(is!=**null**) is.close();

} **catch** (Exception e) { }

}

**return** isCopy;

}

}

1. Paging.java
2. MainController.java
3. MemberController.java