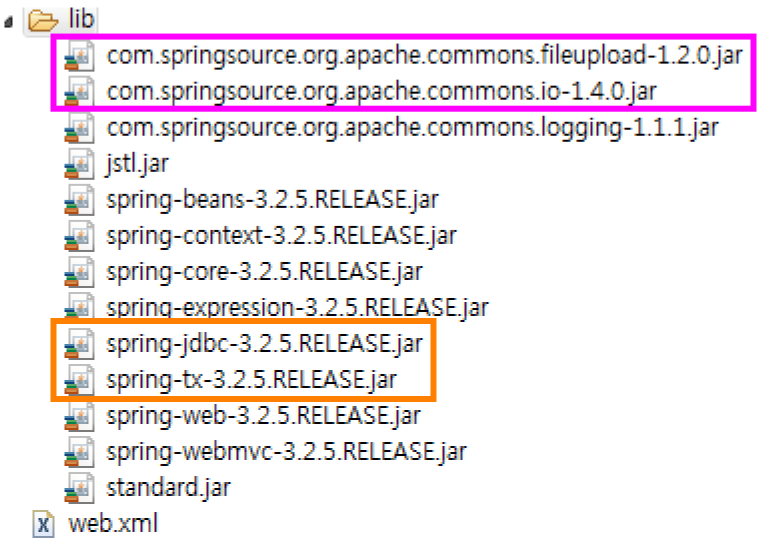


lib
com.springsource.org.apache.commons.fileupload-1.2.0.jar
com.springsource.org.apache.commons.io-1.4.0.jar
com.springsource.org.apache.commons.logging-1.1.1.jar
jstl.jar
spring-beans-3.2.5.RELEASE.jar
spring-context-3.2.5.RELEASE.jar
spring-core-3.2.5.RELEASE.jar
spring-expression-3.2.5.RELEASE.jar
spring-jdbc-3.2.5.RELEASE.jar
spring-tx-3.2.5.RELEASE.jar
spring-web-3.2.5.RELEASE.jar
spring-webmvc-3.2.5.RELEASE.jar
standard.jar
web.xml

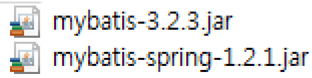
- **파일업로드 관련 라이브러리**
- **JDBC관련 라이브러리**

mybatis

http://blog.mybatis.org/p/products.html
http://mybatis.github.io/mybatis-3/ko/index.html : DOCS

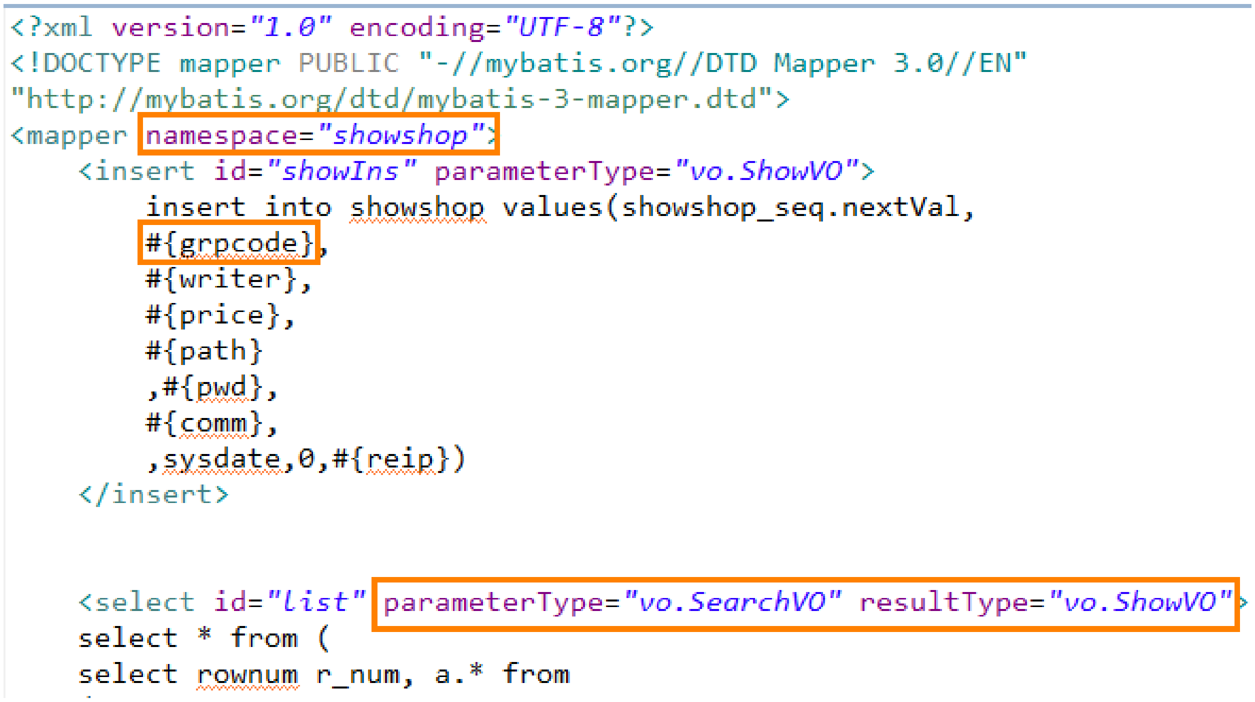
spring,mybatis3 를 다운받는다.

마이바티스 관련 라이브러리



mybatis-3.2.3.jar
mybatis-spring-1.2.1.jar

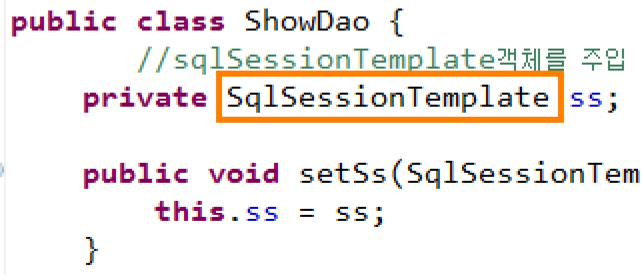
SQL map 생성



```
<?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="showshop">
    <insert id="showIns" parameterType="vo.ShowVO">
        insert into showshop values(showshop_seq.nextVal,
        #{grpcode},
        #{writer},
        #{price},
        #{path}
        ,#{pwd},
        #{comm},
        ,sysdate,0,#{reip})
    </insert>


    <select id="list" parameterType="vo.SearchVO" resultType="vo.ShowVO">
    select * from (
    select rownum r_num, a.* from
```

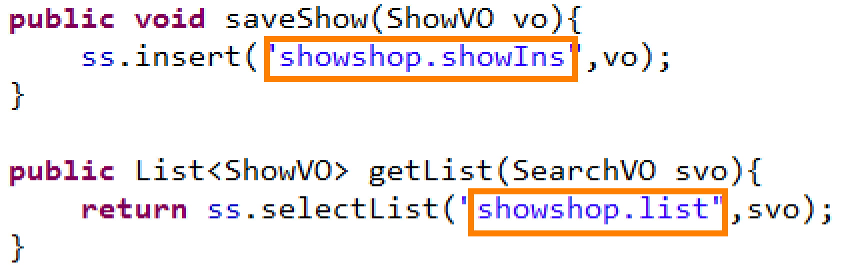
- 네임스페이스
- parameterType, resultType
- #{}

DAO



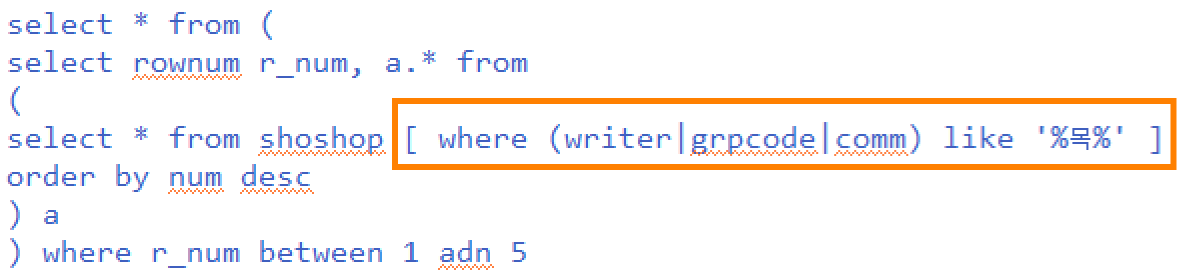
```
public class ShowDao {
        //sqlSessionTemplate객체를 주입
    private SqlSessionTemplate ss;

    public void setSs(SqlSessionTemplate ss) {
        this.ss = ss;
    }
```

- 스프링에서는 SqlSessionTemplate를 사용하여 sql맵을 관리한다.



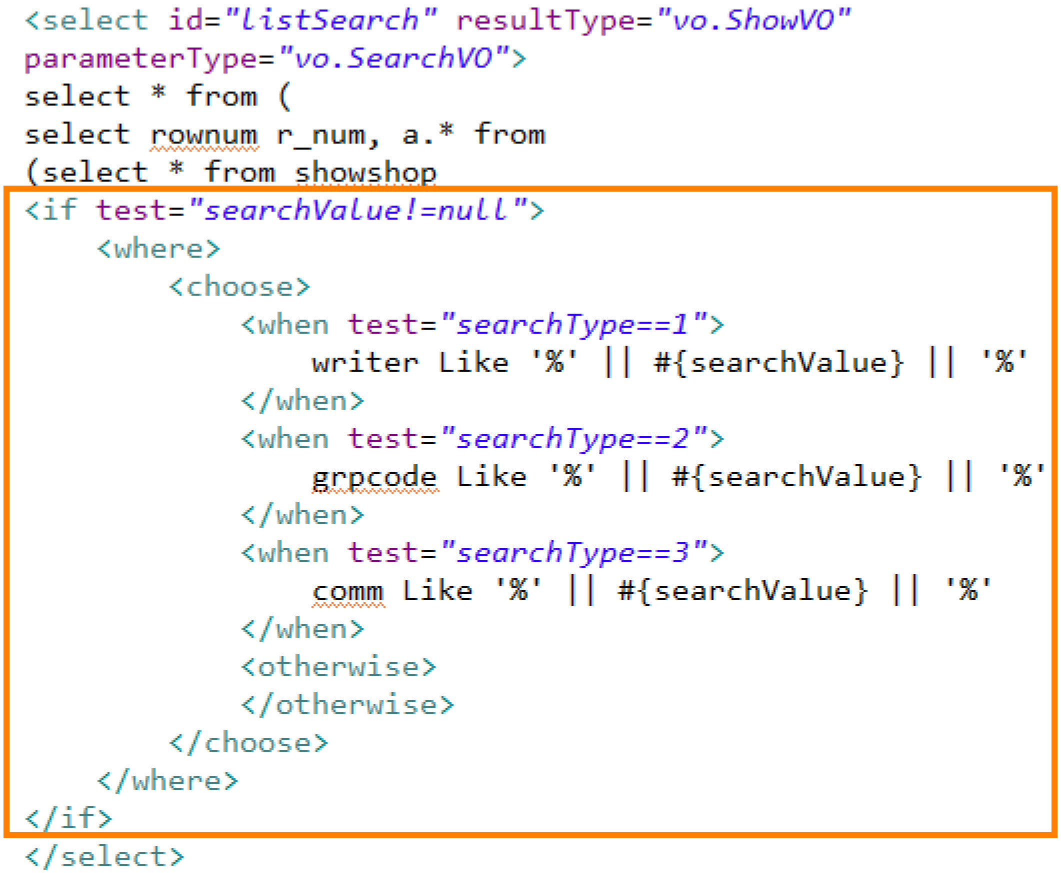
```
public void saveShow(ShowVO vo){
    ss.insert("showshop.showIns",vo);
}

public List<ShowVO> getList(SearchVO svo){
    return ss.selectList("showshop.list",svo);
}
```

동적쿼리



```
select * from (
select rownum r_num, a.* from
(
select * from shoshop [ where (writer|grpcode|comm) like '%묵%' ]
order by num desc
) a
) where r_num between 1 adn 5
```

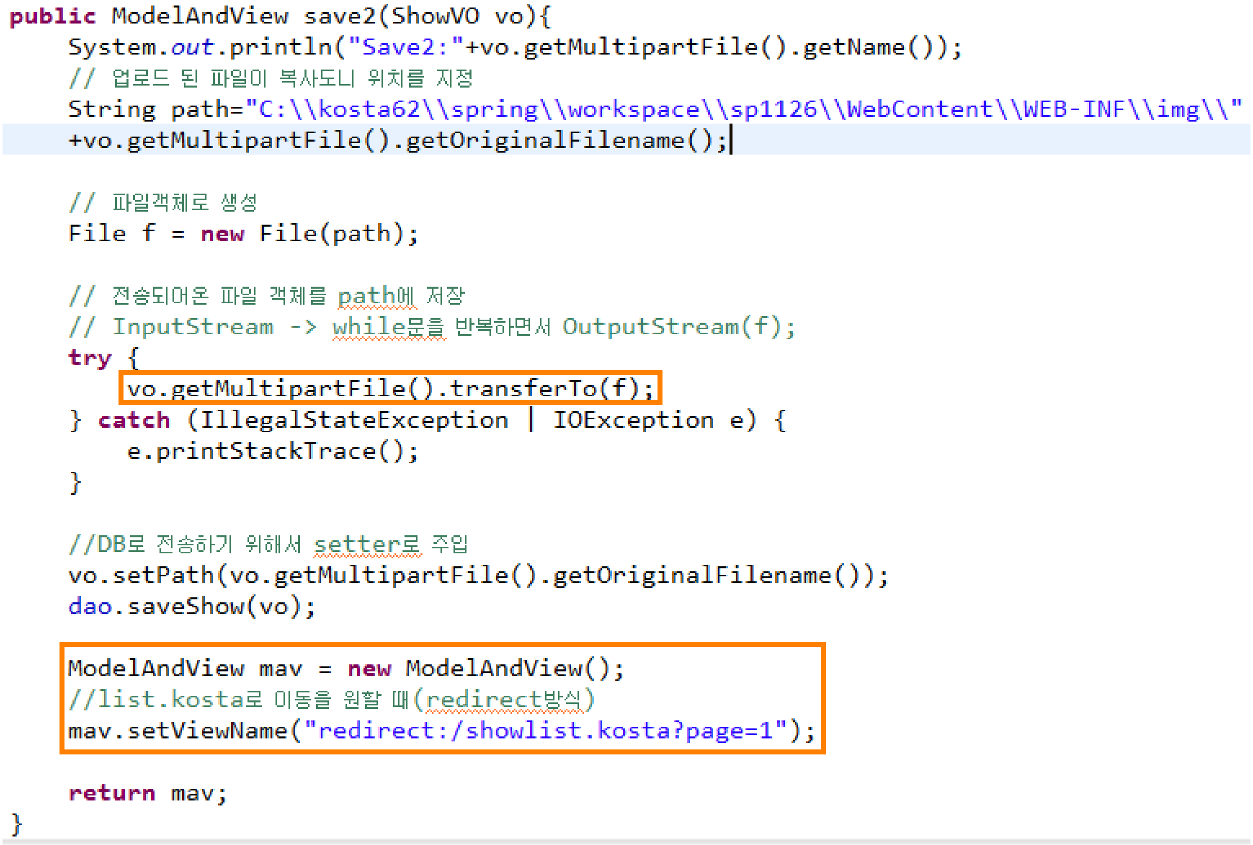
- 네모 부분에 대한 mybatis 동적 쿼리



```
<select id="listSearch" resultType="vo.ShowVO"
parameterType="vo.SearchVO">
select * from (
select rownum r_num, a.* from
(select * from showshop
<if test="searchValue!=null">
    <where>
        <choose>
            <when test="searchType==1">
                writer Like '%' || #{searchValue} || '%'
            </when>
            <when test="searchType==2">
                grpcode Like '%' || #{searchValue} || '%'
            </when>
            <when test="searchType==3">
                comm Like '%' || #{searchValue} || '%'
            </when>
            <otherwise>
            </otherwise>
        </choose>
    </where>
</if>
</select>
```

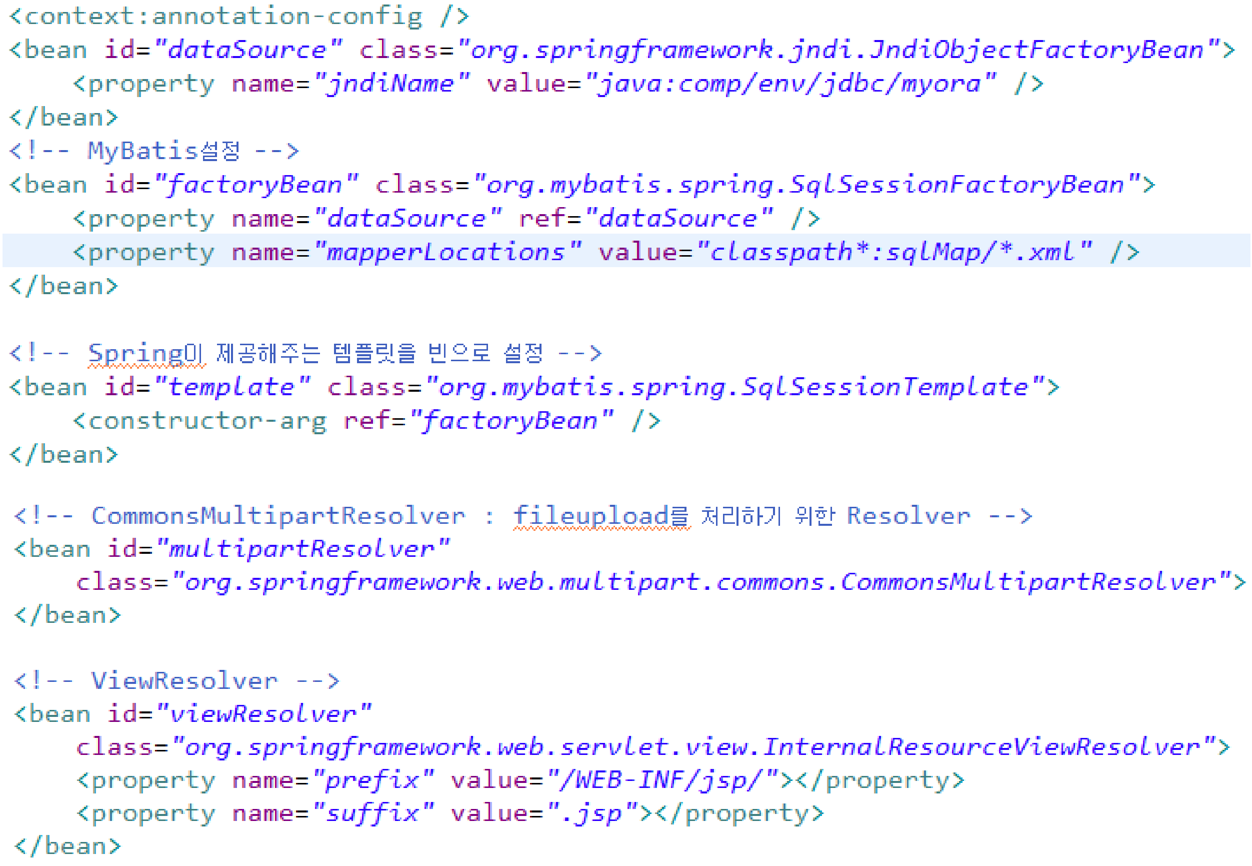
- <if><where><choose><when><otherwise>

@RequestMapping(value="/upsave2.kosta",method=RequestMethod.POST)



```
public ModelAndView save2(ShowVO vo){
    System.out.println("Save2:"+vo.getMultipartFile().getName());
    // 업로드 된 파일이 복사도니 위치를 지정
    String path="C:\\kosta62\\spring\\workspace\\sp1126\\WebContent\\WEB-INF\\img\\"
    +vo.getMultipartFile().getOriginalFilename();

    // 파일객체로 생성
    File f = new File(path);

    // 전송되어온 파일 객체를 path에 저장
    // InputStream -> while문을 반복하면서 OutputStream(f);
    try {
        vo.getMultipartFile().transferTo(f);
    } catch (IllegalStateException | IOException e) {
        e.printStackTrace();
    }

    //DB로 전송하기 위해서 setter로 주입
    vo.setPath(vo.getMultipartFile().getOriginalFilename());
    dao.saveShow(vo);

    ModelAndView mav = new ModelAndView();
    //list.kosta로 이동을 원할 때(redirect방식)
    mav.setViewName("redirect:/showlist.kosta?page=1");

    return mav;
}
```

servlet.xml



```
<context:annotation-config />
<bean id="dataSource" class="org.springframework.jndi.JndiObjectFactoryBean">
    <property name="jndiName" value="java:comp/env/jdbc/myora" />
</bean>
<!-- MyBatis설정 -->
<bean id="factoryBean" class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dataSource" />
    <property name="mapperLocations" value="classpath*:sqlMap/*.xml" />
</bean>

<!-- Spring이 제공해주는 템플릿을 빈으로 설정 -->
<bean id="template" class="org.mybatis.spring.SqlSessionTemplate">
    <constructor-arg ref="factoryBean" />
</bean>

<!-- CommonsMultipartResolver : fileupload를 처리하기 위한 Resolver -->
<bean id="multipartResolver"
    class="org.springframework.web.multipart.commons.CommonsMultipartResolver">
</bean>

<!-- ViewResolver -->
<bean id="viewResolver"
    class="org.springframework.web.servlet.view.InternalResourceViewResolver">
    <property name="prefix" value="/WEB-INF/jsp/"></property>
    <property name="suffix" value=".jsp"></property>
</bean>
```