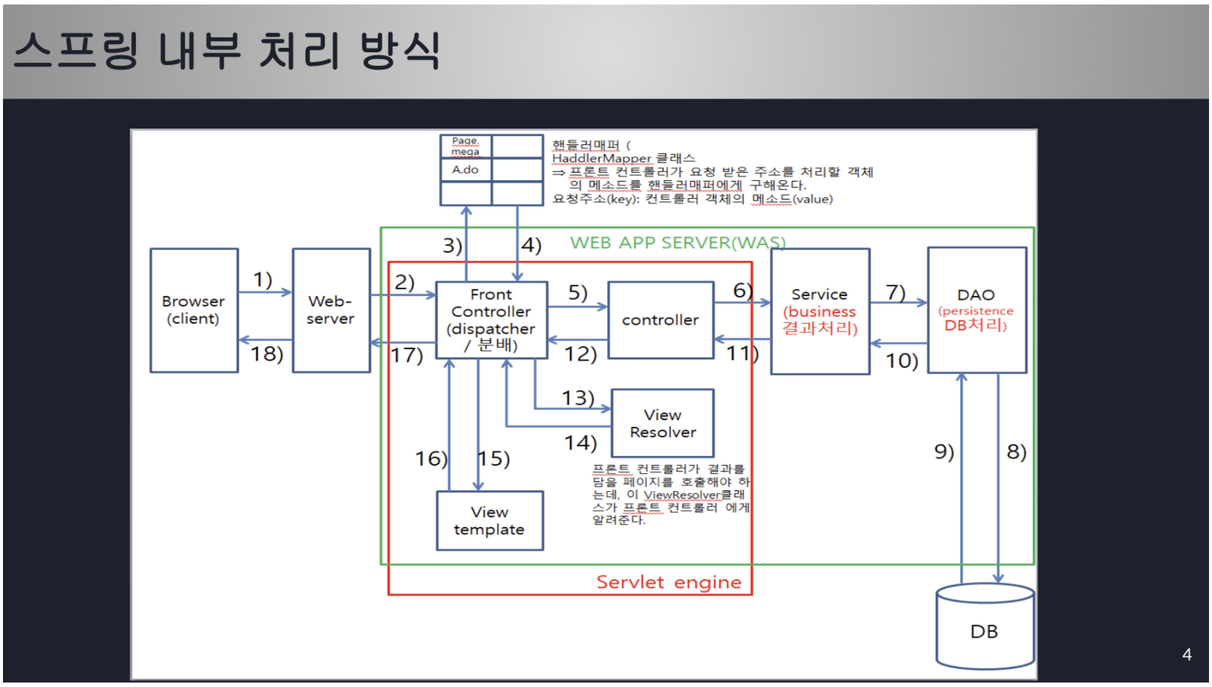
Spring 5일차 - 1114

1. 스프링 내부 처리 방식

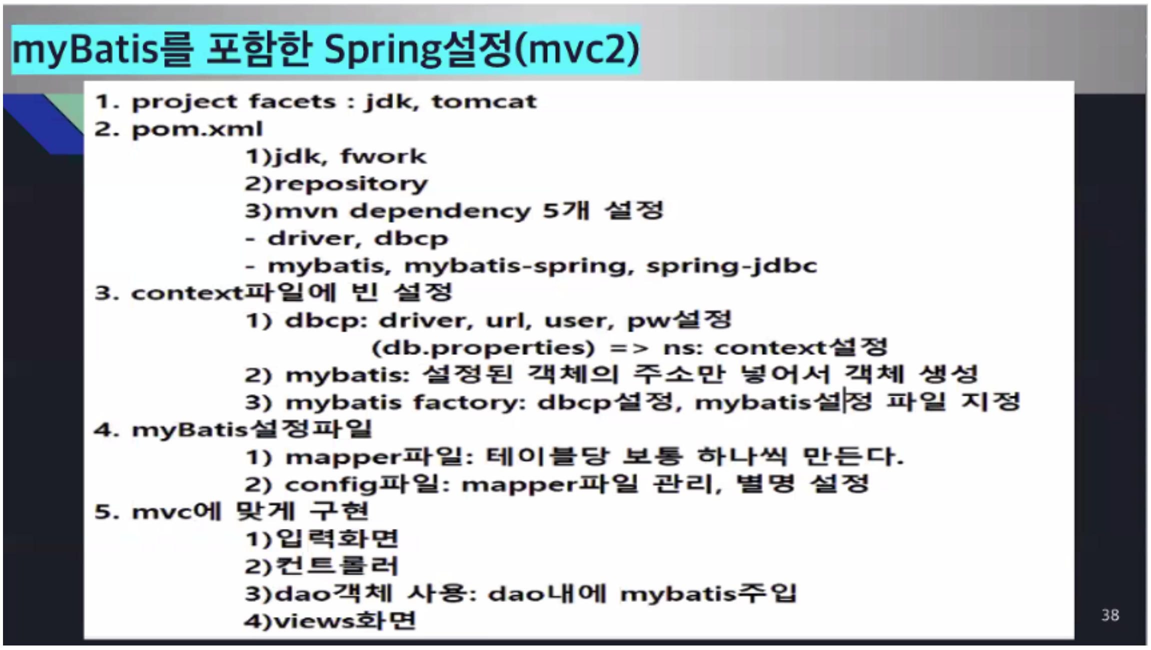


스프링 내부 처리 방식

핸들러매퍼 (
HaddlerMapper 클래스
⇒ 프론트 컨트롤러가 요청 받은 주소를 처리할 객체의 메소드를 핸들러매퍼에게 구해온다.
요청주소(key): 컨트롤러 객체의 메소드(value)

프론트 컨트롤러가 결과를 담을 페이지를 호출해야 하는데, 이 ViewResolver클래스가 프론트 컨트롤러 에게 알려준다.

2. Spring(mvc2) -myBatis포함



myBatis를 포함한 Spring설정(mvc2)

1. project facets : jdk, tomcat

2. pom.xml

1)jdk, fwork

2)repository

3)mvn dependency 5개 설정

- driver, dbcp

- mybatis, mybatis-spring, spring-jdbc

3. context파일에 빈 설정

1) dbcp: driver, url, user, pw설정

(db.properties) => ns: context설정

2) mybatis: 설정된 객체의 주소만 넣어서 객체 생성

3) mybatis factory: dbcp설정, mybatis설정 파일 지정

4. myBatis설정파일

1) mapper파일: 테이블당 보통 하나씩 만든다.

2) config파일: mapper파일 관리, 별명 설정

5. mvc에 맞게 구현

1)입력화면

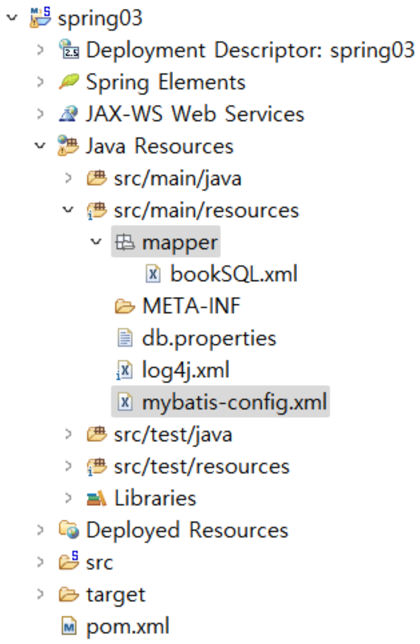
2)컨트롤러

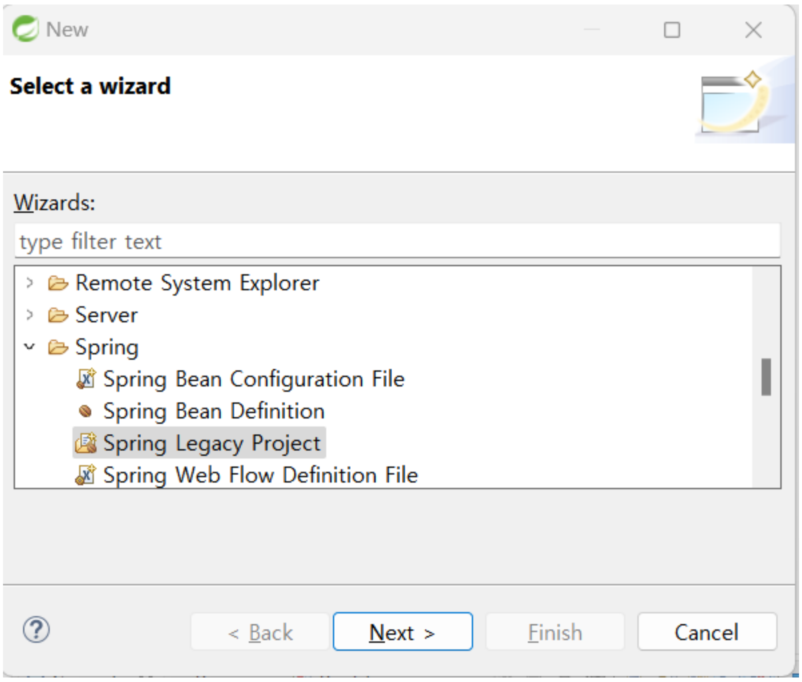
3)dao객체 사용: dao내에 mybatis주입

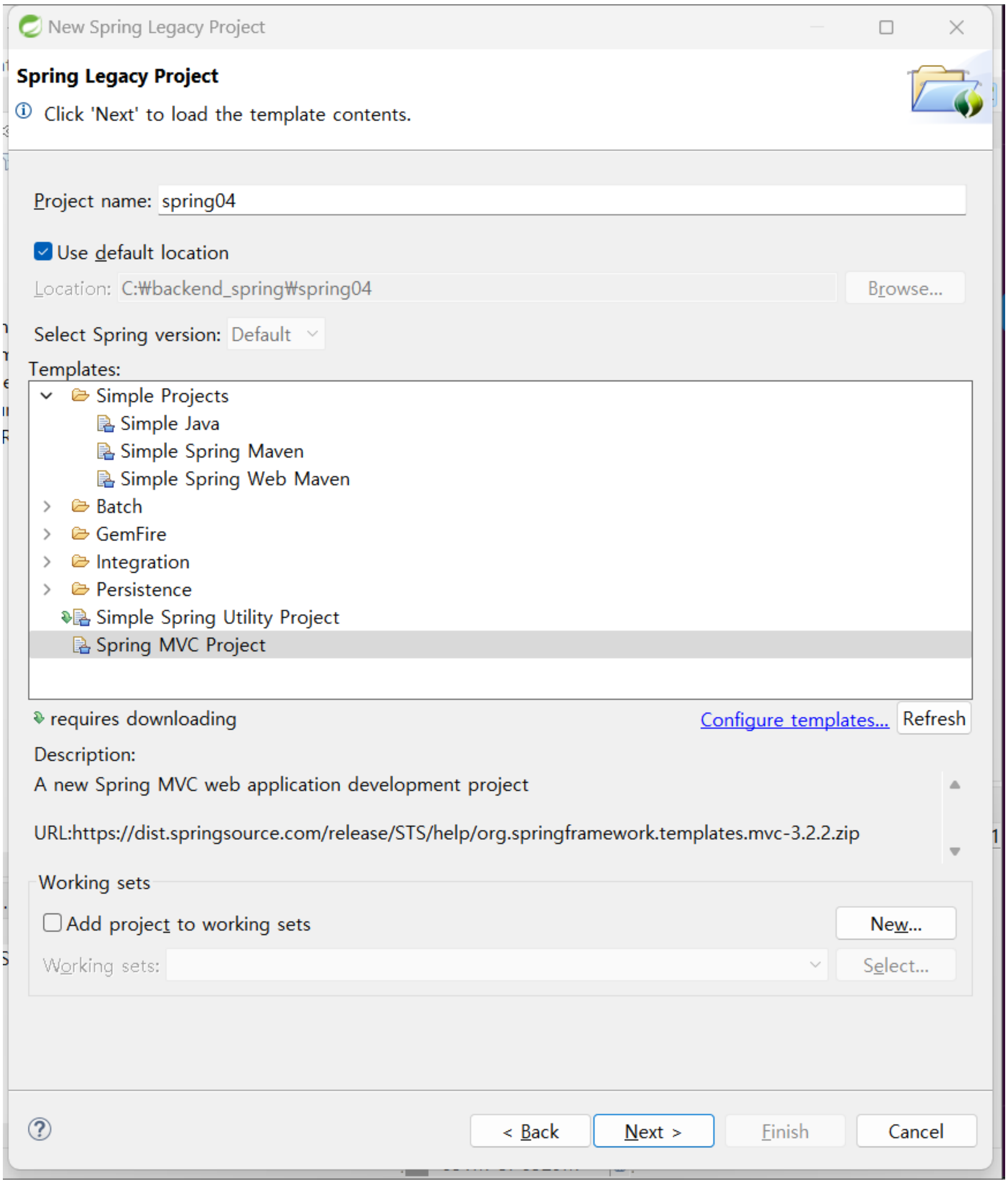
4)views화면

3. 실습

1. Spring mvc 프로젝트 만들기


spring03
Deployment Descriptor: spring03
Spring Elements
JAX-WS Web Services
Java Resources
src/main/java
src/main/resources
mapper
bookSQL.xml
META-INF
db.properties
log4j.xml
mybatis-config.xml
src/test/java
src/test/resources
Libraries
Deployed Resources
src
target
pom.xml


New
Select a wizard
Wizards:
type filter text
Remote System Explorer
Server
Spring
Spring Bean Configuration File
Spring Bean Definition
Spring Legacy Project
Spring Web Flow Definition File
< Back
Next >
Finish
Cancel



New Spring Legacy Project

Spring Legacy Project

ⓘ Click 'Next' to load the template contents.

Project name: spring04

☑ Use default location

Location: C:₩backend_spring₩spring04

Select Spring version: Default

Templates:

```
v Simple Projects
    Simple Java
    Simple Spring Maven
    Simple Spring Web Maven
> Batch
> GemFire
> Integration
> Persistence
  Simple Spring Utility Project
  Spring MVC Project
```

requires downloading

Description:

A new Spring MVC web application development project

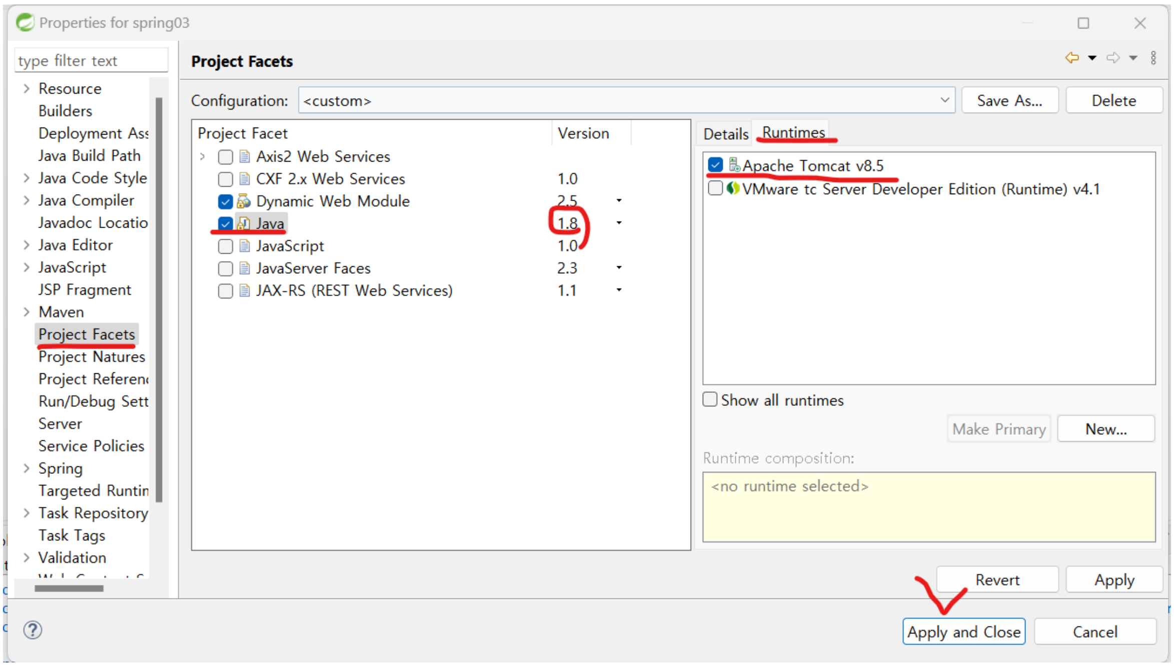
URL:https://dist.springsource.com/release/STS/help/org.springframework.templates.mvc-3.2.2.zip

Working sets

□ Add project to working sets

Working sets:

- project facets(jdk, tomcat)



Configuration: <custom>

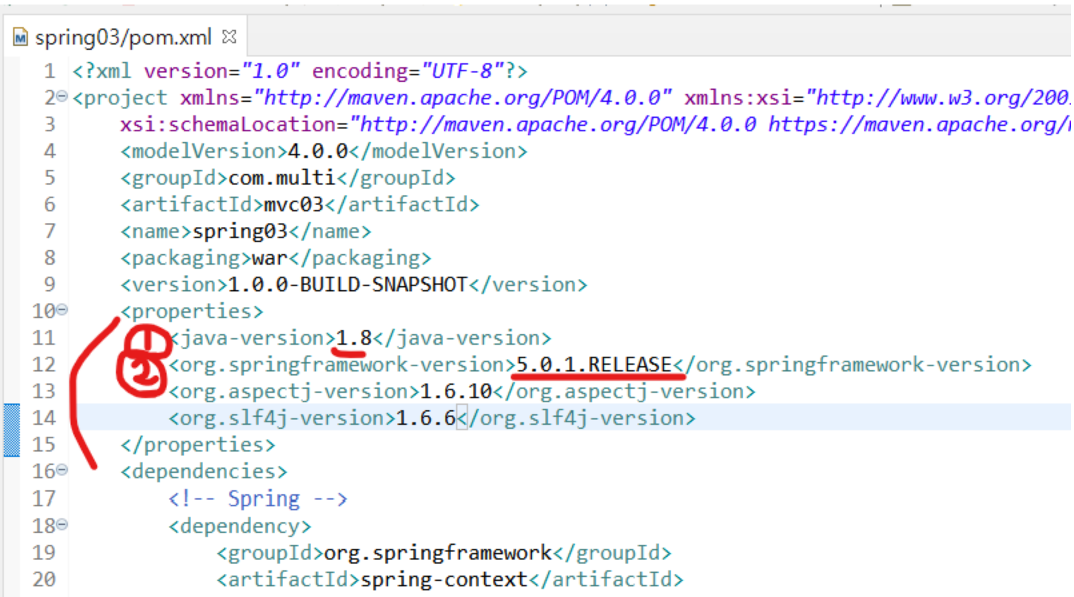
Project Facet	Version
□ Axis2 Web Services	
□ CXF 2.x Web Services	1.0
☑ Dynamic Web Module	2.5
☑ Java	1.8
□ JavaScript	1.0
□ JavaServer Faces	2.3
□ JAX-RS (REST Web Services)	1.1

☑ Apache Tomcat v8.5

□ VMware tc Server Developer Edition (Runtime) v4.1

2. pom.xml

1) jdk, springframework 설정



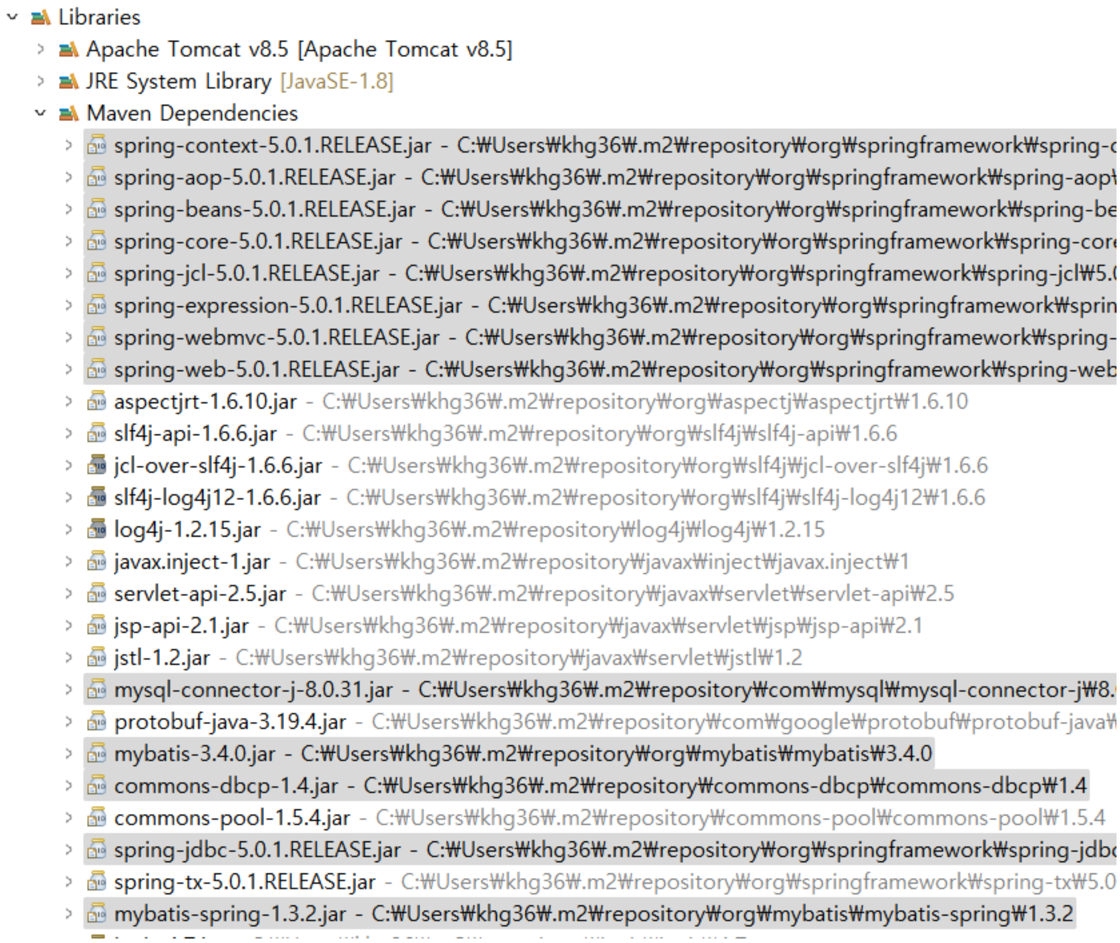
spring03/pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/200
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/
    <modelVersion>4.0.0</modelVersion>
    <groupId>com.multi</groupId>
    <artifactId>mvc03</artifactId>
    <name>spring03</name>
    <packaging>war</packaging>
    <version>1.0.0-BUILD-SNAPSHOT</version>
    <properties>
        <java-version>1.8</java-version>
        <org.springframework-version>5.0.1.RELEASE</org.springframework-version>
        <org.aspectj-version>1.6.10</org.aspectj-version>
        <org.slf4j-version>1.6.6</org.slf4j-version>
    </properties>
    <dependencies>
        <!-- Spring -->
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
```

2) repository 설정

3) mvn dependency 5개 설정

```
<!-- https://mvnrepository.com/artifact/com.mysql/mysql-connector-j -->
    <dependency>
      <groupId>com.mysql</groupId>
      <artifactId>mysql-connector-j</artifactId>
      <version>8.0.31</version>
    </dependency>


    <!-- https://mvnrepository.com/artifact/org.mybatis/mybatis -->
    <dependency>
      <groupId>org.mybatis</groupId>
      <artifactId>mybatis</artifactId>
      <version>3.4.0</version>
    </dependency>

    <!-- https://mvnrepository.com/artifact/commons-dbcp/commons-dbcp -->
    <dependency>
      <groupId>commons-dbcp</groupId>
      <artifactId>commons-dbcp</artifactId>
      <version>1.4</version>
    </dependency>

    <!-- https://mvnrepository.com/artifact/org.springframework/spring-jdbc -->
    <!-- spring jdbc기능 확장 -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-jdbc</artifactId>
      <version>5.0.1.RELEASE</version>
    </dependency>

    <!-- https://mvnrepository.com/artifact/org.mybatis/mybatis-spring -->
    <!-- mybatis + spring -->
    <dependency>
      <groupId>org.mybatis</groupId>
      <artifactId>mybatis-spring</artifactId>
      <version>1.3.2</version>
    </dependency>
```

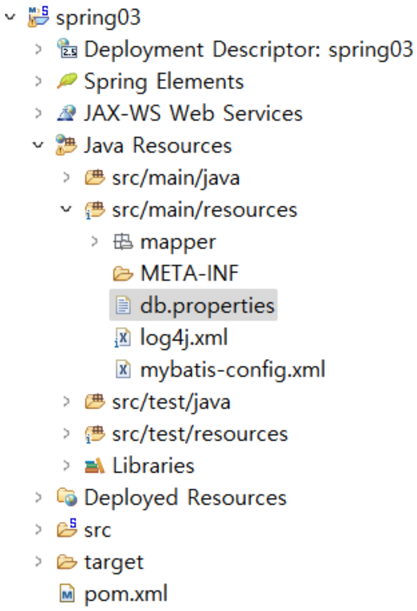


```
Libraries
  Apache Tomcat v8.5 [Apache Tomcat v8.5]
  JRE System Library [JavaSE-1.8]
  Maven Dependencies
    spring-context-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-c
    spring-aop-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-aop₩
    spring-beans-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-be
    spring-core-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-core
    spring-jcl-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-jcl₩5.0
    spring-expression-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩sprin
    spring-webmvc-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-
    spring-web-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-web
    aspectjrt-1.6.10.jar - C:₩Users₩khg36₩.m2₩repository₩org₩aspectj₩aspectjrt₩1.6.10
    slf4j-api-1.6.6.jar - C:₩Users₩khg36₩.m2₩repository₩org₩slf4j₩slf4j-api₩1.6.6
    jcl-over-slf4j-1.6.6.jar - C:₩Users₩khg36₩.m2₩repository₩org₩slf4j₩jcl-over-slf4j₩1.6.6
    slf4j-log4j12-1.6.6.jar - C:₩Users₩khg36₩.m2₩repository₩org₩slf4j₩slf4j-log4j12₩1.6.6
    log4j-1.2.15.jar - C:₩Users₩khg36₩.m2₩repository₩log4j₩log4j₩1.2.15
    javax.inject-1.jar - C:₩Users₩khg36₩.m2₩repository₩javax₩inject₩javax.inject₩1
    servlet-api-2.5.jar - C:₩Users₩khg36₩.m2₩repository₩javax₩servlet₩servlet-api₩2.5
    jsp-api-2.1.jar - C:₩Users₩khg36₩.m2₩repository₩javax₩servlet₩jsp₩jsp-api₩2.1
    jstl-1.2.jar - C:₩Users₩khg36₩.m2₩repository₩javax₩servlet₩jstl₩1.2
    mysql-connector-j-8.0.31.jar - C:₩Users₩khg36₩.m2₩repository₩com₩mysql₩mysql-connector-j₩8.
    protobuf-java-3.19.4.jar - C:₩Users₩khg36₩.m2₩repository₩com₩google₩protobuf₩protobuf-java₩
    mybatis-3.4.0.jar - C:₩Users₩khg36₩.m2₩repository₩org₩mybatis₩mybatis₩3.4.0
    commons-dbcp-1.4.jar - C:₩Users₩khg36₩.m2₩repository₩commons-dbcp₩commons-dbcp₩1.4
    commons-pool-1.5.4.jar - C:₩Users₩khg36₩.m2₩repository₩commons-pool₩commons-pool₩1.5.4
    spring-jdbc-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-jdbc
    spring-tx-5.0.1.RELEASE.jar - C:₩Users₩khg36₩.m2₩repository₩org₩springframework₩spring-tx₩5.0
    mybatis-spring-1.3.2.jar - C:₩Users₩khg36₩.m2₩repository₩org₩mybatis₩mybatis-spring₩1.3.2
```

3. context파일에 빈 설정

1) dbcp

- db.properties 경로



```
spring03
  Deployment Descriptor: spring03
  Spring Elements
  JAX-WS Web Services
  Java Resources
    src/main/java
    src/main/resources
      mapper
      META-INF
      db.properties
      log4j.xml
      mybatis-config.xml
    src/test/java
    src/test/resources
    Libraries
  Deployed Resources
  src
  target
  pom.xml
```

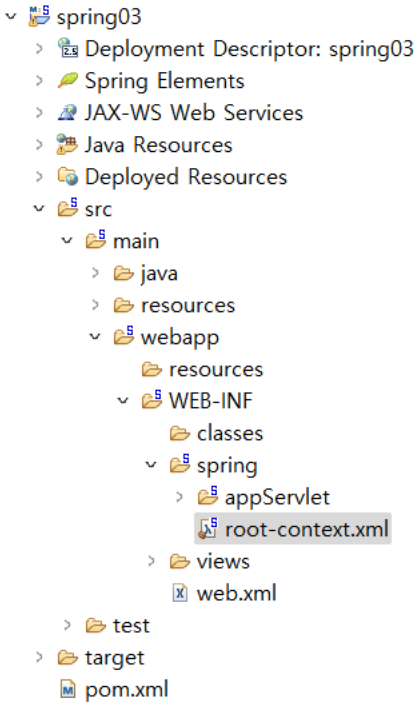
- db.properties

```
jdbc.driver=com.mysql.cj.jdbc.Driver
jdbc.url=jdbc:mysql://localhost:3306/shop5?useUnicode=true&serverTimezone=Asia/Seoul
jdbc.username=root
jdbc.password=1234
```

2) mybatis

3) mybatis factory : dbcp설정, mybatis설정 파일 지정

- root- context.xml 경로



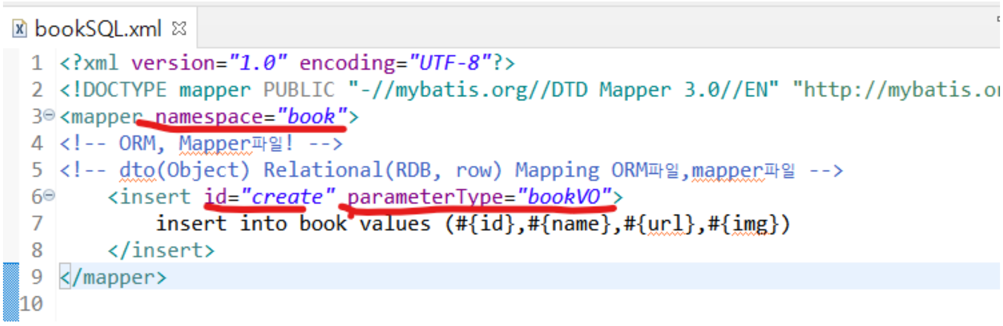
```
spring03
  Deployment Descriptor: spring03
  Spring Elements
  JAX-WS Web Services
  Java Resources
  Deployed Resources
  src
    main
      java
      resources
      webapp
        resources
        WEB-INF
          classes
          spring
            appServlet
            root-context.xml
          views
          web.xml
    test
  target
  pom.xml
```

- 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.xs
    http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.3.xsd">
  <context:property-placeholder
    location="classpath:db.properties" />
  <!-- Root Context: defines shared resources visible to all other web components -->
  <!-- myBatis의 싱글톤 객체 생성 -->
  <bean id="my" class="org.mybatis.spring.SqlSessionTemplate">
    <constructor-arg ref="config"></constructor-arg>
  </bean>

  <!-- myBatis의 설정 객체 생성 -->
  <bean id="config" class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dbcp"></property>
    <property name="configLocation"
      value="classpath:mybatis-config.xml"></property>
  </bean>
  <!-- classpath: src/main/resources아래를 말함. -->

  <!-- dbcp의 싱글톤 객체 생성 -->
  <!-- db프로그램의 2단계까지 담당. driver설정, db연결 -->
  <!-- BasicDataSource dbcp = new BasicDataSource -->
  <bean id="dbcp" class="org.apache.commons.dbcp.BasicDataSource">
    <property name="driverClassName" value="${jdbc.driver}"></property>
    <property name="url" value="${jdbc.url}"></property>
    <property name="username" value="${jdbc.username}"></property>
    <property name="password" value="${jdbc.password}"></property>
  </bean>
</beans>
```

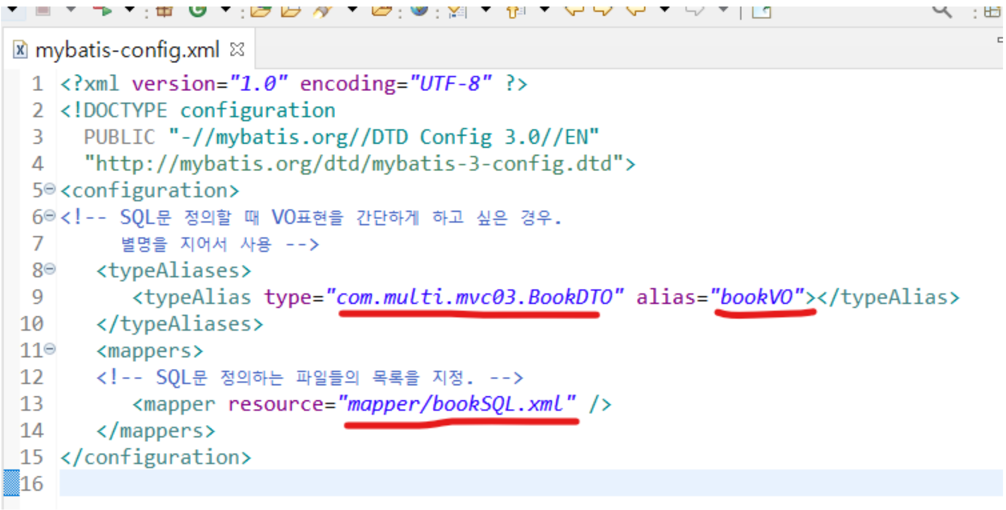
4. myBatis설정파일

1) mapper파일



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.o
<mapper namespace="book">
<!-- ORM, Mapper파일! -->
<!-- dto(Object) Relational(RDB, row) Mapping ORM파일,mapper파일 -->
    <insert id="create" parameterType="bookVO">
        insert into book values (#{id},#{name},#{url},#{img})
    </insert>
</mapper>
```

2) config파일

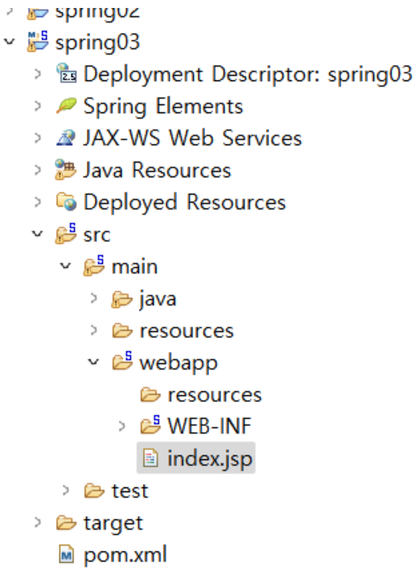


```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration
  PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
  "http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
<!-- SQL문 정의할 때 VO표현을 간단하게 하고 싶은 경우.
     별명을 지어서 사용 -->
    <typeAliases>
        <typeAlias type="com.multi.mvc03.BookDTO" alias="bookVO"></typeAlias>
    </typeAliases>
    <mappers>
    <!-- SQL문 정의하는 파일들의 목록을 지정. -->
        <mapper resource="mapper/bookSQL.xml" />
    </mappers>
</configuration>
```

5. mvc 맞게 구현

1) 입력화면 : index.jsp

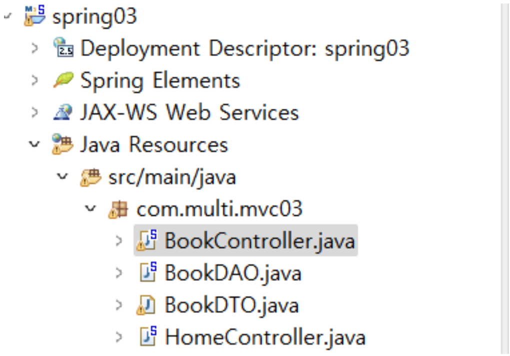
- 경로



```
> spring02
v spring03
  > Deployment Descriptor: spring03
  > Spring Elements
  > JAX-WS Web Services
  > Java Resources
  > Deployed Resources
  v src
    v main
      > java
      > resources
      v webapp
          resources
        > WEB-INF
          index.jsp
    > test
  > target
    pom.xml
```

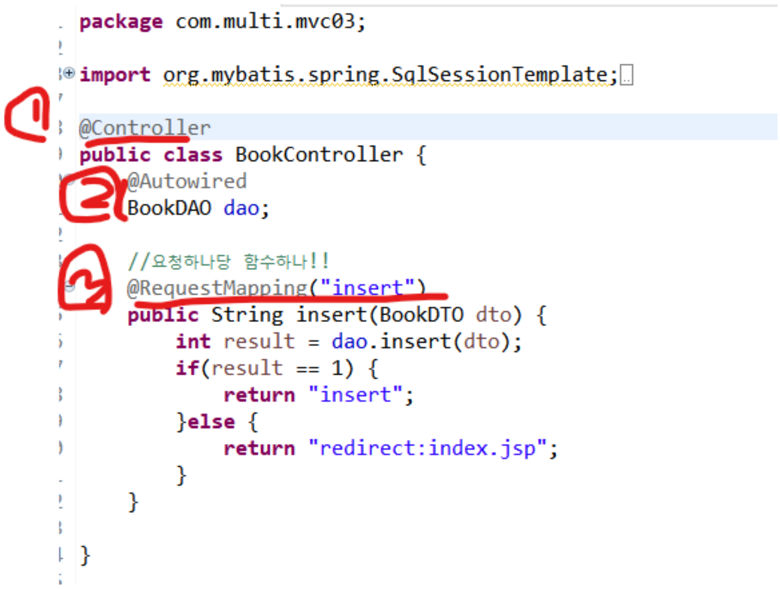
2) 컨트롤러

- 경로



```
v spring03
  > Deployment Descriptor: spring03
  > Spring Elements
  > JAX-WS Web Services
  v Java Resources
    v src/main/java
      v com.multi.mvc03
        > BookController.java
        > BookDAO.java
        > BookDTO.java
        > HomeController.java
```

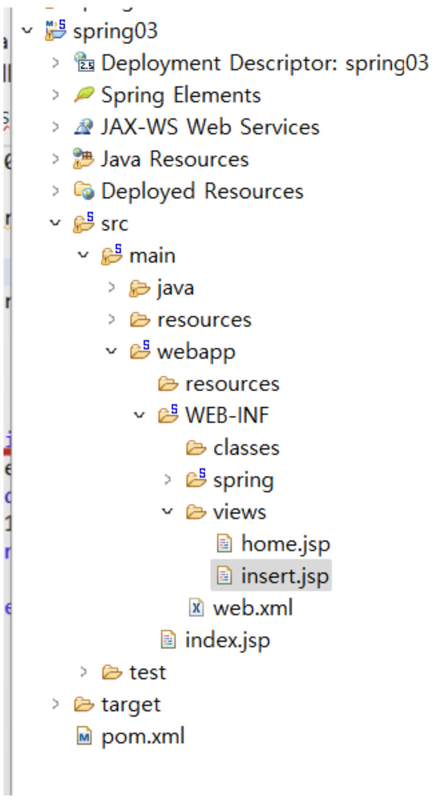
3) dao 객체 사용 : dao내에 mybatis 주입



```
package com.multi.mvc03;

import org.mybatis.spring.SqlSessionTemplate;

@Controller
public class BookController {
    @Autowired
    BookDAO dao;

    //요청하나당 함수하나!!
    @RequestMapping("insert")
    public String insert(BookDTO dto) {
        int result = dao.insert(dto);
        if(result == 1) {
            return "insert";
        }else {
            return "redirect:index.jsp";
        }
    }

}
```

4) views화면

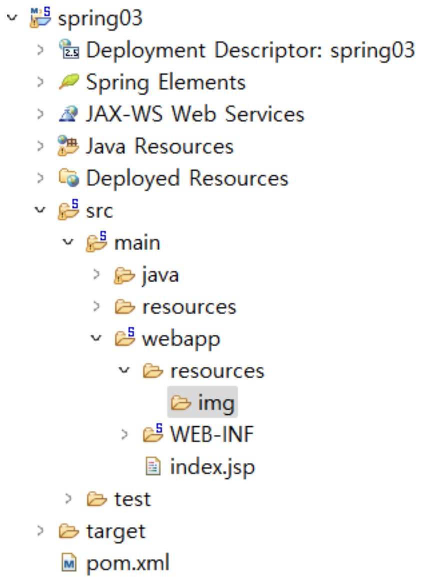
- 경로



```
v spring03
    > Deployment Descriptor: spring03
    > Spring Elements
    > JAX-WS Web Services
    > Java Resources
    > Deployed Resources
    v src
        v main
            > java
            > resources
            v webapp
                resources
                v WEB-INF
                    classes
                    > spring
                    v views
                        home.jsp
                        insert.jsp
                    web.xml
                index.jsp
        > test
    > target
    pom.xml
```

5) img, css, js 파일 설정

- 경로



```
spring03
  Deployment Descriptor: spring03
  Spring Elements
  JAX-WS Web Services
  Java Resources
  Deployed Resources
  src
    main
      java
      resources
      webapp
        resources
          img
        WEB-INF
        index.jsp
    test
  target
  pom.xml
```