

Chap07

스프링 MVC 프로젝트의 기본구성

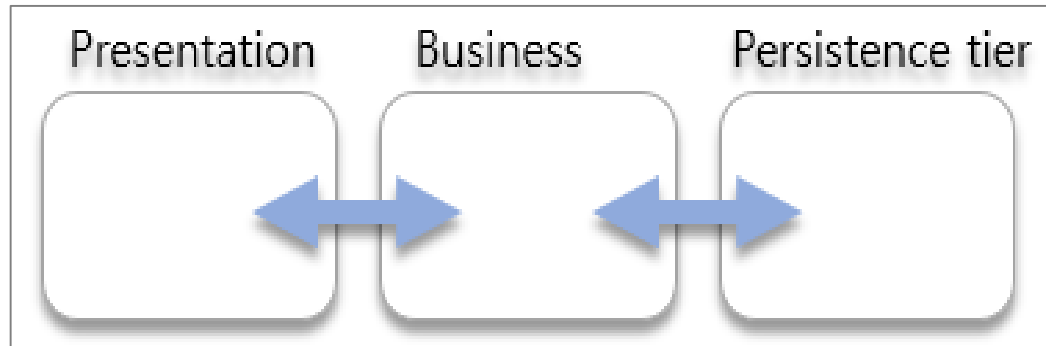
동의과학대학교
컴퓨터정보과
김진숙

Objectives

- 스프링 MVC를 이용하는 웹 프로젝트 전체 구조에 대한 이해
- 개발의 각 단계에 필요한 설정 및 테스트 환경
- 기본적인 조회, 등록, 수정, 삭제, 구현

프로젝트의 구성

- 일반적인 웹 프로젝트의 구조는 **3-Tier**의 구조를 활용

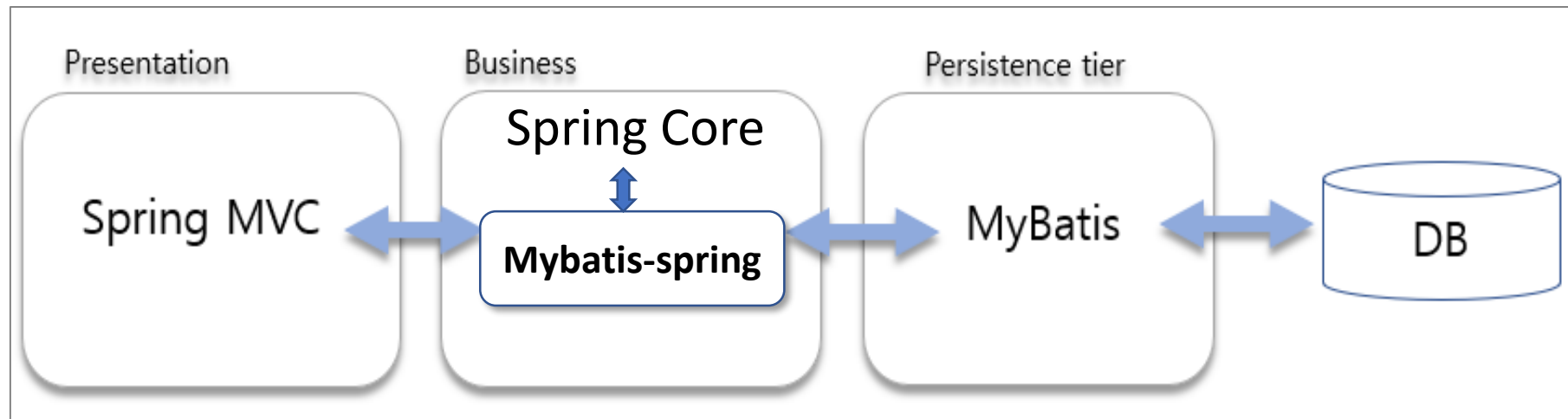


이렇게 구성하는 이유?

- 각 영역을 독립적으로 설계하여 특정 기술이 변하더라도 필요한 부분을 부품처리 쉽게 교환할 수 있게 하는 방식

- 스프링 MVC를 이용하는 예제의 구성

➔ 유지보수의 필요성 때문



각 영역의 네이밍 규칙

일반적으로 프로젝트를 시작할 때
팀 내 또는 회사차원의 네이밍
규칙을 정한다.

- **xxxController**

- 스프링 MVC에서 동작하는 Controller 클래스

- **xxxService, xxxServiceImpl**

- 비즈니스 영역을 담당하는 인터페이스는 'xxxService'라는 방식을 사용
- 인터페이스를 구현한 클래스는 'xxxServiceImpl'이라는 이름을 사용

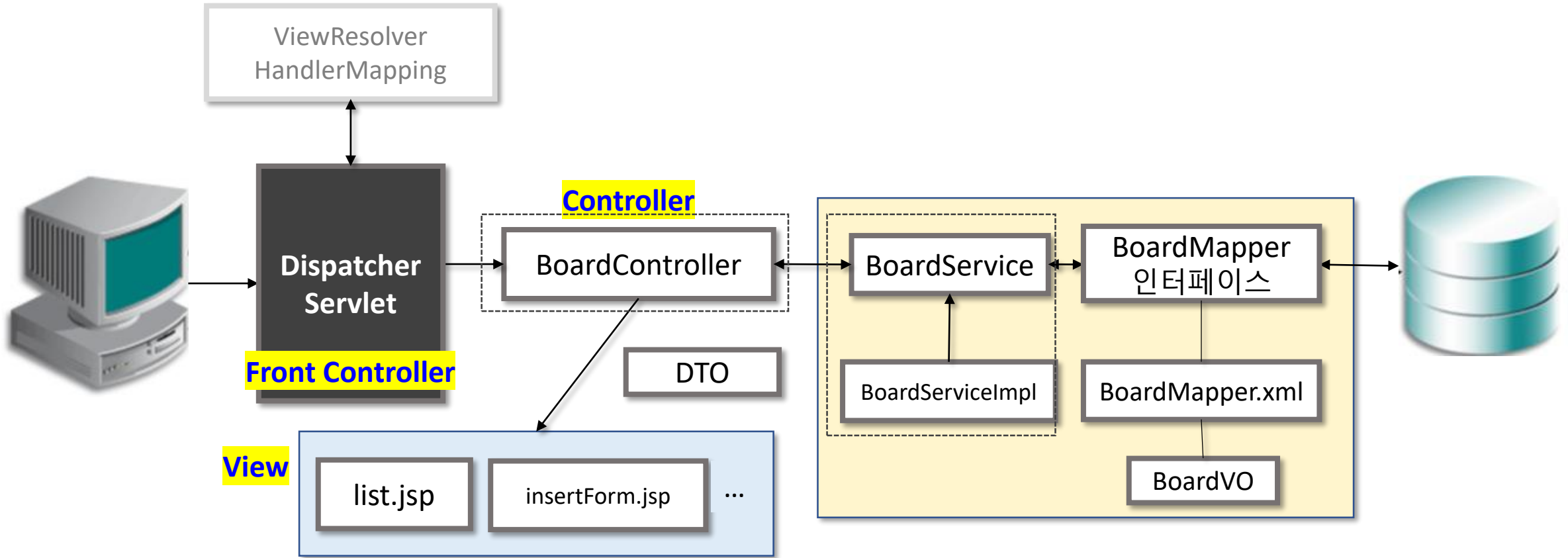
- **xxxDAO 또는 xxxRepository**

- DAO(Data-Access-Object)나 Repository(저장소)라는 이름으로 영역을 따로 구성하는 것이 보편적
- 예제에서는 별도의 DAO를 구성하는 대신에 MyBatis의 Mapper 인터페이스를 활용(**xxxMapper**)

- **VO, DTO**

- VO의 경우는 주로 Read Only의 목적이 강하고, 데이터도 Immutable(불변)하게 설계
- DTO 주로 데이터 수집의 용도
- 둘 다 스프링에서 빈으로 등록하지 않음

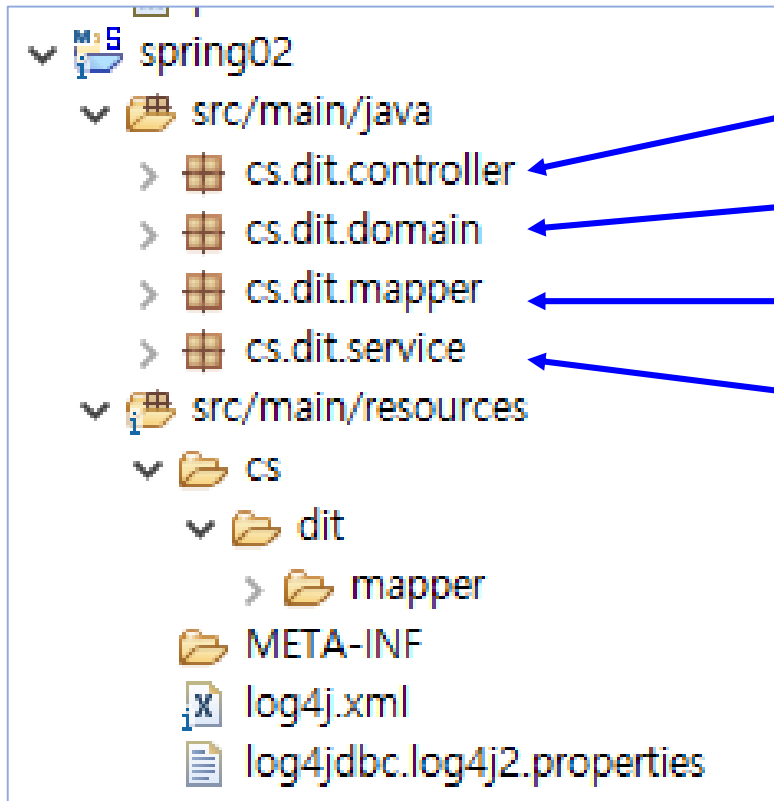
구성도



구현 순서

4. 화면 구현 3. 컨트롤러 계층 2. 서비스 계층 1. 영속 계층

프로젝트 패키지의 구성



스프링 MVC의 Controller 보관 패키지

VO, DTO 클래스들의 패키지

Mybatis Mapper 인터페이스 패키지

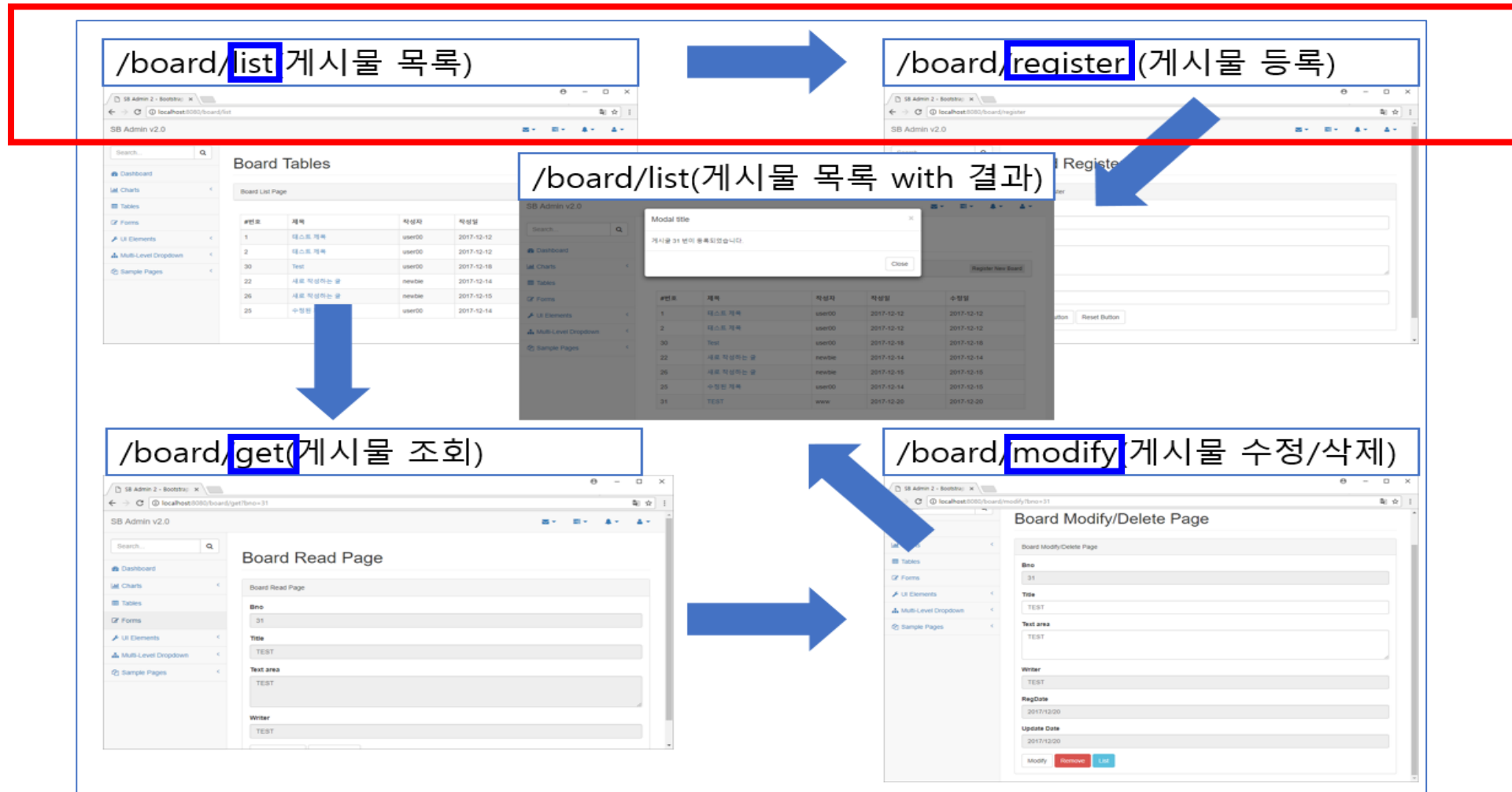
스프링 Service 인터페이스와 구현 클래스 패키지

패키지 구성은 프로젝트의 크기나 구성원의
성향으로 결정됨

기본적인 게시물의 화면 설계

- 코딩 전에 흐름을 설계하는 것은 아주 중요
 - 화면설계를 위한 SW로 정리하는 것 필요(Story board)

[Wireframe tool]
<https://ovenapp.io/>



프로젝트의 생성 및 준비

1. Spring Legacy Project의 생성

2. pom.xml

- spring 버전 변경(5.2.7)
- Jdk 버전 변경(14)
- Spring-test, spring-jdbc, spring-tx 추가(DB연동에 필요)
- mariadb 드라이버 추가(2.7.5)
- HikariCP(3.4.5), mybatis(3.5.10), mybatis-spring(2.0.6) 추가
- Log4j 버전변경 (1.2.17, <exclusions>와 <scope> 삭제)
- Log4jdbc-log4j2 추가(1.16) : SQL로그를 보기위해 설치
- junit버전 변경(4.12)
- Servlet 버전 변경 (4.0.1)
- Jsp 버전 변경(2.3.3)
- Lombok(1.18.24)의 설정 등

프로젝트의 생성 및 준비(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/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>dit.cs</groupId>
  <artifactId>controller</artifactId>
  <name>ex02</name>
  <packaging>war</packaging>
  <version>1.0.0-BUILD-SNAPSHOT</version>
  <properties>
    <java-version>14</java-version>
    <org.springframework-version>5.2.7.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>
```

```
    <version>${org.springframework-version}</version>
```

```
    <exclusions>
```

```
      <!-- Exclude Commons Logging in favor of SLF4j -->
```

```
      <exclusion>
```

```
        <groupId>commons-logging</groupId>
```

```
        <artifactId>commons-logging</artifactId>
```

```
      </exclusion>
```

```
    </exclusions>
```

```
  </dependency>
```

```
  <dependency>
```

```
    <groupId>org.springframework</groupId>
```

```
    <artifactId>spring-webmvc</artifactId>
```

```
    <version>${org.springframework-version}</version>
```

```
  </dependency>
```

```
  <dependency>
```

```
    <groupId>org.springframework</groupId>
```

```
    <artifactId>spring-test</artifactId>
```

```
    <version>${org.springframework-version}</version>
```

```
  </dependency>
```

```
  <dependency>
```

```
    <groupId>org.springframework</groupId>
```

```
    <artifactId>spring-jdbc</artifactId>
```

```
    <version>${org.springframework-version}</version>
```

```
  </dependency>
```

```
<dependency>
```

```
  <groupId>org.springframework</groupId>
```

```
  <artifactId>spring-tx</artifactId>
```

```
  <version>${org.springframework-version}</version>
```

```
</dependency>
```

```
  <!-- AspectJ -->
```

```
  <dependency>
```

```
    <groupId>org.aspectj</groupId>
```

```
    <artifactId>aspectjrt</artifactId>
```

```
    <version>${org.aspectj-version}</version>
```

```
</dependency>
```

```
  <!-- Logging -->
```

```
  <dependency>
```

```
    <groupId>org.slf4j</groupId>
```

```
    <artifactId>slf4j-api</artifactId>
```

```
    <version>${org.slf4j-version}</version>
```

```
</dependency>
```

```
  <dependency>
```

```
    <groupId>org.slf4j</groupId>
```

```
    <artifactId>jcl-over-slf4j</artifactId>
```

```
    <version>${org.slf4j-version}</version>
```

```
    <scope>runtime</scope>
```

```
</dependency>
```

```
  <dependency>
```

```
    <groupId>org.slf4j</groupId>
```

```
    <artifactId>slf4j-log4j12</artifactId>
```

```
    <version>${org.slf4j-version}</version>
```

```
    <scope>runtime</scope>
```

```
</dependency>
```

```

<dependency>
  <groupId>log4j</groupId>
  <artifactId>log4j</artifactId>
  <version>1.2.17</version>
</dependency>
<dependency>
  <groupId>org.bgee.log4jdbc-log4j2</groupId>
  <artifactId>log4jdbc-log4j2-jdbc4.1</artifactId>
  <version>1.16</version>
</dependency>
<!-- @Inject -->
<dependency>
  <groupId>javax.inject</groupId>
  <artifactId>javax.inject</artifactId>
  <version>1</version>
</dependency>
<!-- Servlet-->
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>javax.servlet-api</artifactId>
  <version>4.0.1</version>
  <scope>provided</scope>
</dependency>
<dependency>
  <groupId>javax.servlet.jsp</groupId>
  <artifactId>jsp-api</artifactId>
  <version>2.1</version>
  <scope>provided</scope>
</dependency>

```

```

<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>jstl</artifactId>
  <version>1.2</version>
</dependency>
<!-- Test -->
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.12</version>
  <scope>test</scope>
</dependency>

<!-- DB 연동 관련 -->
<dependency>
  <groupId>org.mariadb.jdbc</groupId>
  <artifactId>mariadb-java-client</artifactId>
  <version>2.7.5</version>
</dependency>
<!-- DBCP -->
<dependency>
  <groupId>com.zaxxer</groupId>
  <artifactId>HikariCP</artifactId>
  <version>3.4.5</version>
</dependency>

```

```

<dependency>
  <groupId>org.mybatis</groupId>
  <artifactId>mybatis</artifactId>
  <version>3.5.6</version>
</dependency>
<dependency>
  <groupId>org.mybatis</groupId>
  <artifactId>mybatis-spring</artifactId>
  <version>2.0.6</version>
</dependency>

<!-- lombok-->
<dependency>
  <groupId>org.projectlombok</groupId>
  <artifactId>lombok</artifactId>
  <version>1.18.22</version>
  <scope>provided</scope>
</dependency>
</dependencies>

```

```

<build>
  <plugins>
    <plugin>
      <artifactId>maven-eclipse-plugin</artifactId>
      <version>2.9</version>
      <configuration>
        <additionalProjectnatures>
          <projectnature>org.springframework.ide.eclipse.core.springnature</projectnature>
        </additionalProjectnatures>
        <additionalBuildcommands>
          <buildcommand>org.springframework.ide.eclipse.core.springbuilder</buildcommand>
        </additionalBuildcommands>
        <downloadSources>true</downloadSources>
        <downloadJavadocs>true</downloadJavadocs>
      </configuration>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>2.5.1</version>
      <configuration>
        <source>14</source>
        <target>14</target>
        <compilerArgument>-Xlint:all</compilerArgument>
        <showWarnings>true</showWarnings>
        <showDeprecation>true</showDeprecation>
      </configuration>
    </plugin>
    <plugin>
      <groupId>org.codehaus.mojo</groupId>
      <artifactId>exec-maven-plugin</artifactId>
      <version>1.2.1</version>
      <configuration>
        <mainClass>org.test.int1.Main</mainClass>
      </configuration>
    </plugin>
  </plugins>
</build>
</project>

```

데이터베이스 테이블 생성

```
USE jinsookdb;
```

```
CREATE TABLE tbl_board(  
    bno INT AUTO_INCREMENT PRIMARY KEY,  
    title VARCHAR(200),  
    content TEXT,  
    writer VARCHAR(50),  
    regdate DATE DEFAULT NOW(),  
    update date DATE DEFAULT NOW()  
);
```

```
INSERT INTO tbl_board(title, content, writer) VALUES('테스트1','테스트내용1','홍길동');  
INSERT INTO tbl_board(title, content, writer) VALUES('테스트2','테스트내용2','홍길동');  
INSERT INTO tbl_board(title, content, writer) VALUES('테스트3','테스트내용3','홍길동');
```

```
SELECT * FROM tbl_board;
```

tbl_board (3r x 6c)					
BNO	TITLE	CONTENT	WRITER	REGDATE	UPDATEDATE
1	테스트1	테스트내용1	홍길동	2022-10-26	2022-10-26
2	테스트2	테스트내용2	홍길동	2022-10-26	2022-10-26
3	테스트3	테스트내용3	홍길동	2022-10-26	2022-10-26

데이터베이스 설정 및 테스트(3~4장에서 진행)

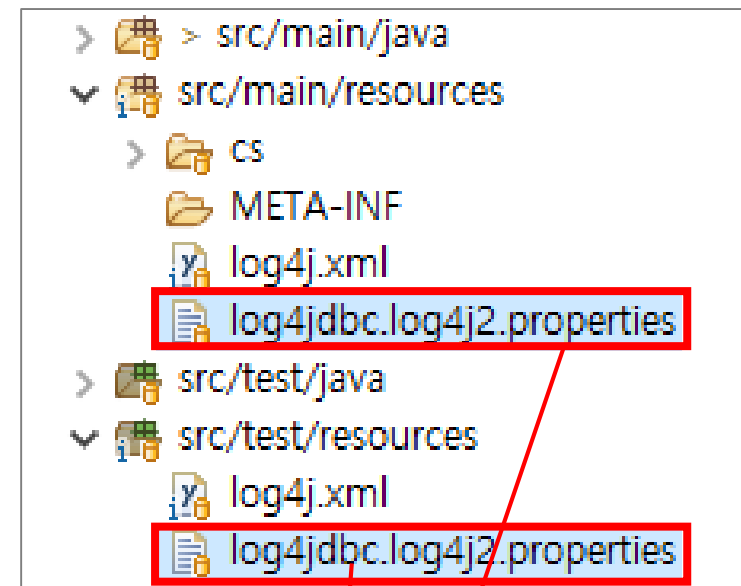
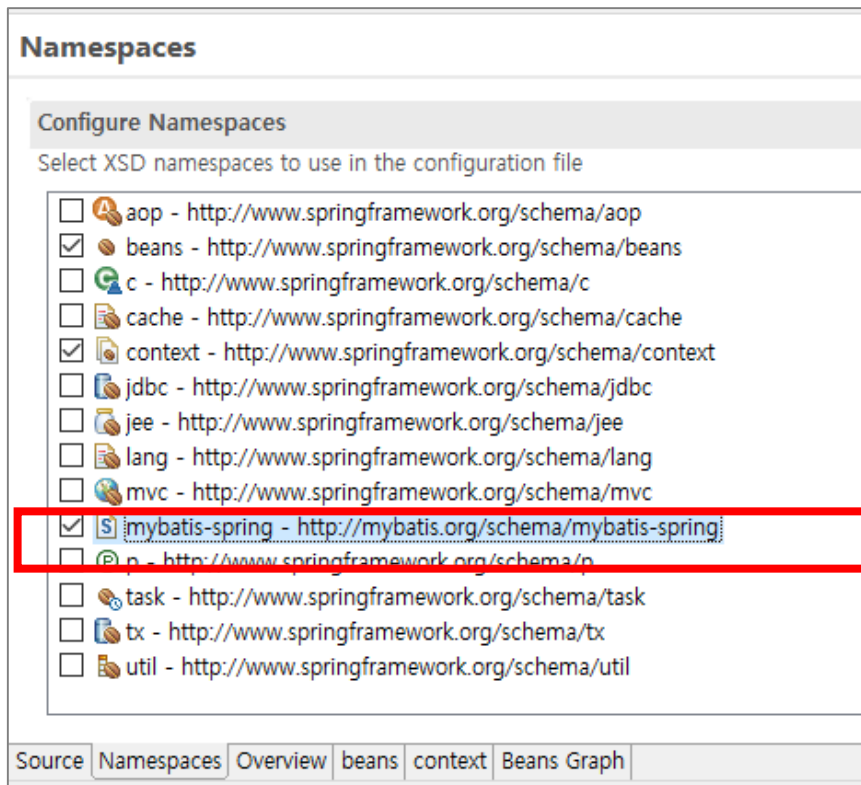
- root-context.xml
 - DataSource의 설정
 - SqlSessionFactory 설정

```
<bean id="hikariConfig" class="com.zaxxer.hikari.HikariConfig">
    <property name="driverClassName" value="net.sf.log4jdbc.sql.jdbcapi.DriverSpy"></property>
    <property name="jdbcUrl" value="jdbc:log4jdbc:oracle:thin:@localhost:1521/xepdb1"></property>
    <property name="username" value="book_ex"></property>
    <property name="password" value="book_ex"></property>
</bean>

<!-- HikariCP configuration -->
<bean id="dataSource" class="com.zaxxer.hikari.HikariDataSource" destroy-method="close">
    <constructor-arg ref="hikariConfig" />
</bean>

<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dataSource"></property>
</bean>
```

root-context.xml



```
*log4jdbc.log4j2.properties  ⌕  
1 log4jdbc.spylogdelegator.name = net.sf.log4jdbc.log.slf4j.Slf4jSpyLogDelegator  
2
```

```
log4jdbc.spylogdelegator.name = net.sf.log4jdbc.log.slf4j.Slf4jSpyLogDelegator
```

복사해서 붙여넣으세요 ^^