

Spring Boot

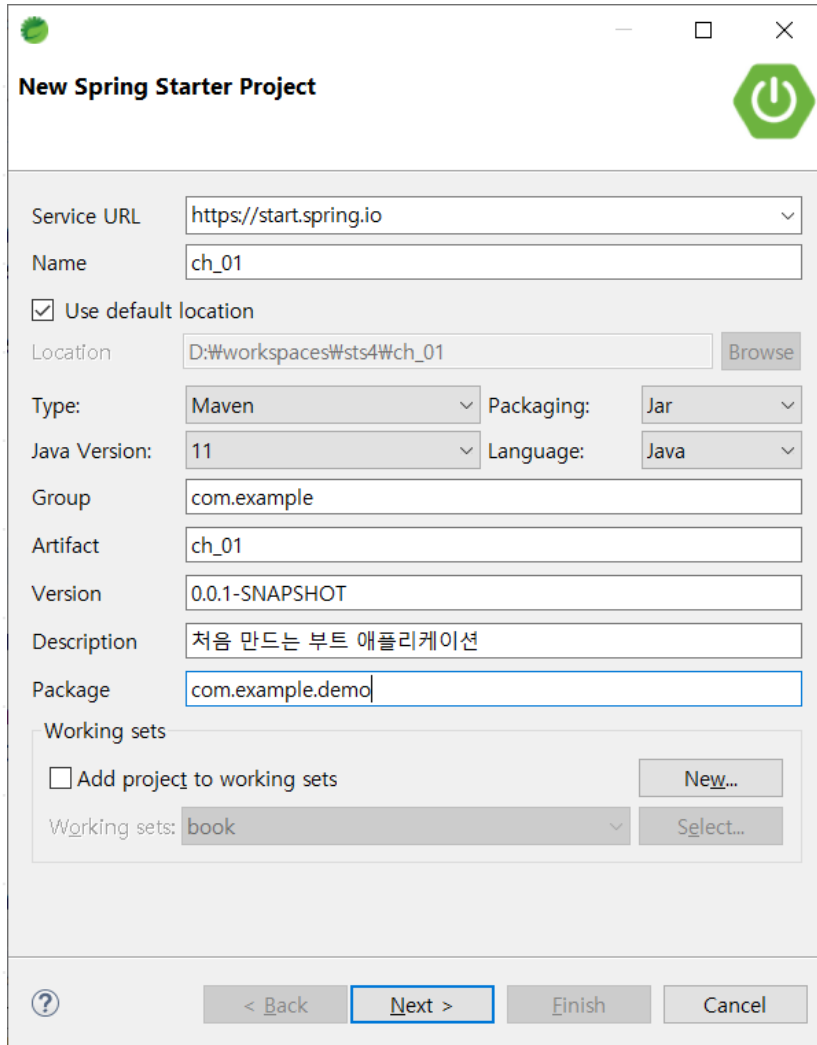
- 환경 설정

조용준

(stgray22@gmail.com)

Hello Boot

❖ 프로젝트 생성



The image shows a 'New Spring Starter Project' dialog box from an IDE. It contains various input fields for project configuration. The 'Service URL' is set to 'https://start.spring.io'. The 'Name' is 'ch_01'. The 'Location' is 'D:\workspaces\Wsts4\ch_01'. The 'Type' is 'Maven', 'Packaging' is 'Jar', 'Java Version' is '11', and 'Language' is 'Java'. The 'Group' is 'com.example', 'Artifact' is 'ch_01', 'Version' is '0.0.1-SNAPSHOT', and 'Description' is '처음 만드는 부트 애플리케이션'. The 'Package' is 'com.example.demo'. There are also options for 'Working sets' and a 'Next >' button.

New Spring Starter Project

Service URL:

Name:

☒ Use default location

Location:

Type: Packaging:

Java Version: Language:

Group:

Artifact:

Version:

Description:

Package:

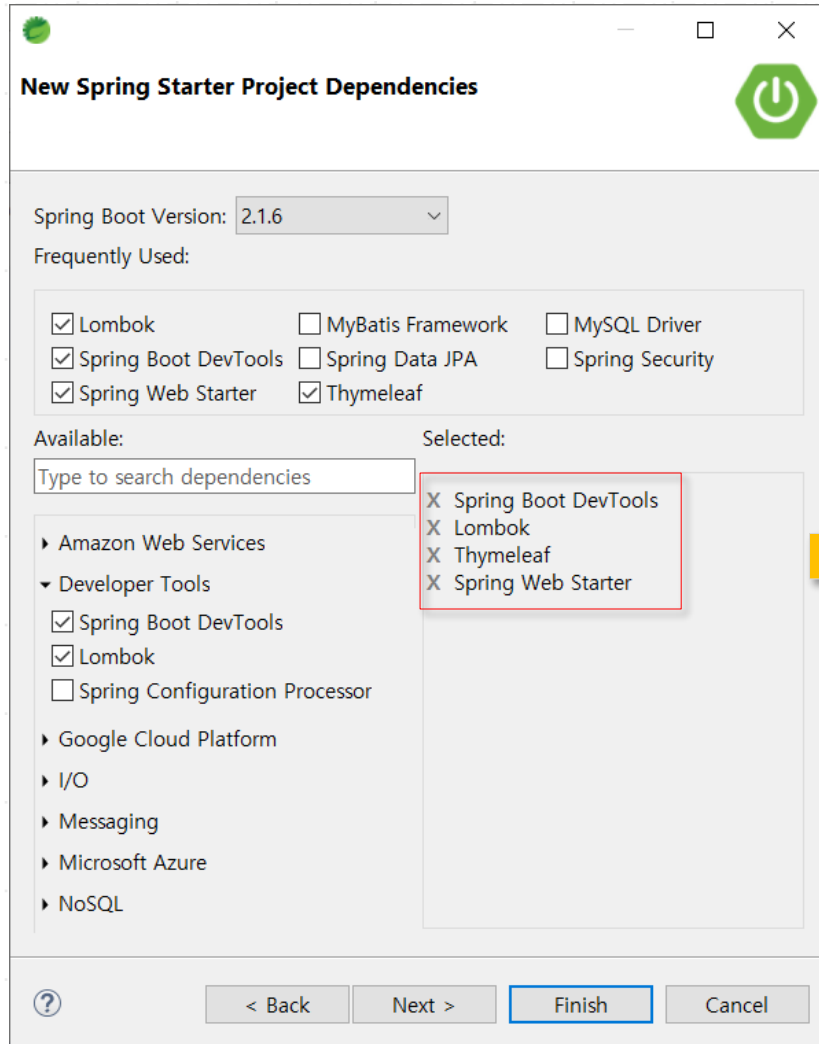
Working sets

☐ Add project to working sets

Working sets:

Hello Boot

❖ 의존성 설정



The dialog box titled "New Spring Starter Project Dependencies" is shown. It features a green power button icon in the top right corner. The "Spring Boot Version" is set to "2.1.6". Under "Frequently Used:", several dependencies are listed with checkboxes: Lombok (checked), Spring Boot DevTools (checked), Spring Web Starter (checked), MyBatis Framework (unchecked), Spring Data JPA (unchecked), Thymeleaf (checked), and MySQL Driver (unchecked). The "Available:" section on the left lists categories like Amazon Web Services, Developer Tools, Google Cloud Platform, I/O, Messaging, Microsoft Azure, and NoSQL. The "Selected:" section on the right, highlighted with a red border, lists the chosen dependencies: Spring Boot DevTools, Lombok, Thymeleaf, and Spring Web Starter. At the bottom, there are buttons for "?", "< Back", "Next >", "Finish" (highlighted with a blue border), and "Cancel".

New Spring Starter Project Dependencies

Spring Boot Version: 2.1.6

Frequently Used:

- ☒ Lombok
- ☒ Spring Boot DevTools
- ☒ Spring Web Starter
- ☐ MyBatis Framework
- ☐ Spring Data JPA
- ☒ Thymeleaf
- ☐ MySQL Driver
- ☐ Spring Security

Available:

Type to search dependencies

- ▶ Amazon Web Services
- ▶ Developer Tools
 - ☒ Spring Boot DevTools
 - ☒ Lombok
 - ☐ Spring Configuration Processor
- ▶ Google Cloud Platform
- ▶ I/O
- ▶ Messaging
- ▶ Microsoft Azure
- ▶ NoSQL

Selected:

- X Spring Boot DevTools
- X Lombok
- X Thymeleaf
- X Spring Web Starter

? < Back Next > Finish Cancel

필요한 모듈 선택

Hello Boot

❖ 프로젝트의 구조

```
ch_01 [boot] [devtools] [sts4 master]
├── src/main/java
│   └── com.example.demo
│       └── Ch01Application.java
├── src/main/resources
│   ├── static
│   └── templates
├── application.properties
├── src/test/java
├── JRE System Library [JavaSE-11]
├── Maven Dependencies
├── target/generated-sources/annotations
├── target/generated-test-sources/test-annotations
├── src
├── target
├── HELP.md
├── mvnw
├── mvnw.cmd
└── pom.xml
```

Hello Boot

❖ pom.xml

book_ch_01/pom.xml

Dependencies

간단한 설정

Filter: |

Dependencies	Dependency Management
<ul style="list-style-type: none">spring-boot-starter-thymeleaf (managed:2.1.6.RELEASE)spring-boot-starter-web (managed:2.1.6.RELEASE)spring-boot-devtools [runtime] (managed:2.1.6.RELEASE)lombok (managed:1.18.8)spring-boot-starter-test [test] (managed:2.1.6.RELEASE)	<ul style="list-style-type: none">

Overview Dependencies Dependency Hierarchy Effective POM pom.xml

book_ch_01/pom.xml

Dependency Hierarchy [test]

미리 잘 짜여진 의존성들

Filter: |

Dependency Hierarchy	Resolved Dependencies
<ul style="list-style-type: none">spring-boot-starter-thymeleaf : 2.1.6.RELEASE [compile]spring-boot-starter : 2.1.6.RELEASE [compile]<ul style="list-style-type: none">spring-boot : 2.1.6.RELEASE [compile]spring-boot-autoconfigure : 2.1.6.RELEASE [compile]spring-boot-starter-logging : 2.1.6.RELEASE [compile]<ul style="list-style-type: none">logback-classic : 1.2.3 [compile]<ul style="list-style-type: none">logback-core : 1.2.3 [compile]slf4j-api : 1.7.26 (managed from 1.7.25) [compile]log4j-to-slf4j : 2.11.2 [compile]<ul style="list-style-type: none">slf4j-api : 1.7.26 (managed from 1.7.25) [compile]log4j-api : 2.11.2 [compile]jul-to-slf4j : 1.7.26 [compile]<ul style="list-style-type: none">slf4j-api : 1.7.26 [compile]	<ul style="list-style-type: none">accessors-smart : 1.2 [test]android-json : 0.0.20131108.vaadin1 [test]asm : 5.0.4 [test]assertj-core : 3.11.1 [test]attoparser : 2.0.5.RELEASE [compile]byte-buddy : 1.9.13 [test]byte-buddy-agent : 1.9.13 [test]classmate : 1.4.0 [compile]hamcrest-core : 1.3 [test]hamcrest-library : 1.3 [test]hibernate-validator : 6.0.17.Final [compile]jackson-annotations : 2.9.0 [compile]jackson-core : 2.9.9 [compile]

Overview Dependencies Dependency Hierarchy Effective POM pom.xml

Hello Boot

❖ pom.xml

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>2.1.6.RELEASE</version>
  <relativePath/> <!-- lookup parent from repository -->
</parent>
```

spring boot 설정 정보 상속
상속받을 spring boot의 version 설정

```
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
  <java.version>11</java.version>
</properties>
```

기타 설정

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>
  <dependency> . . . </dependency>
</dependencies>
```

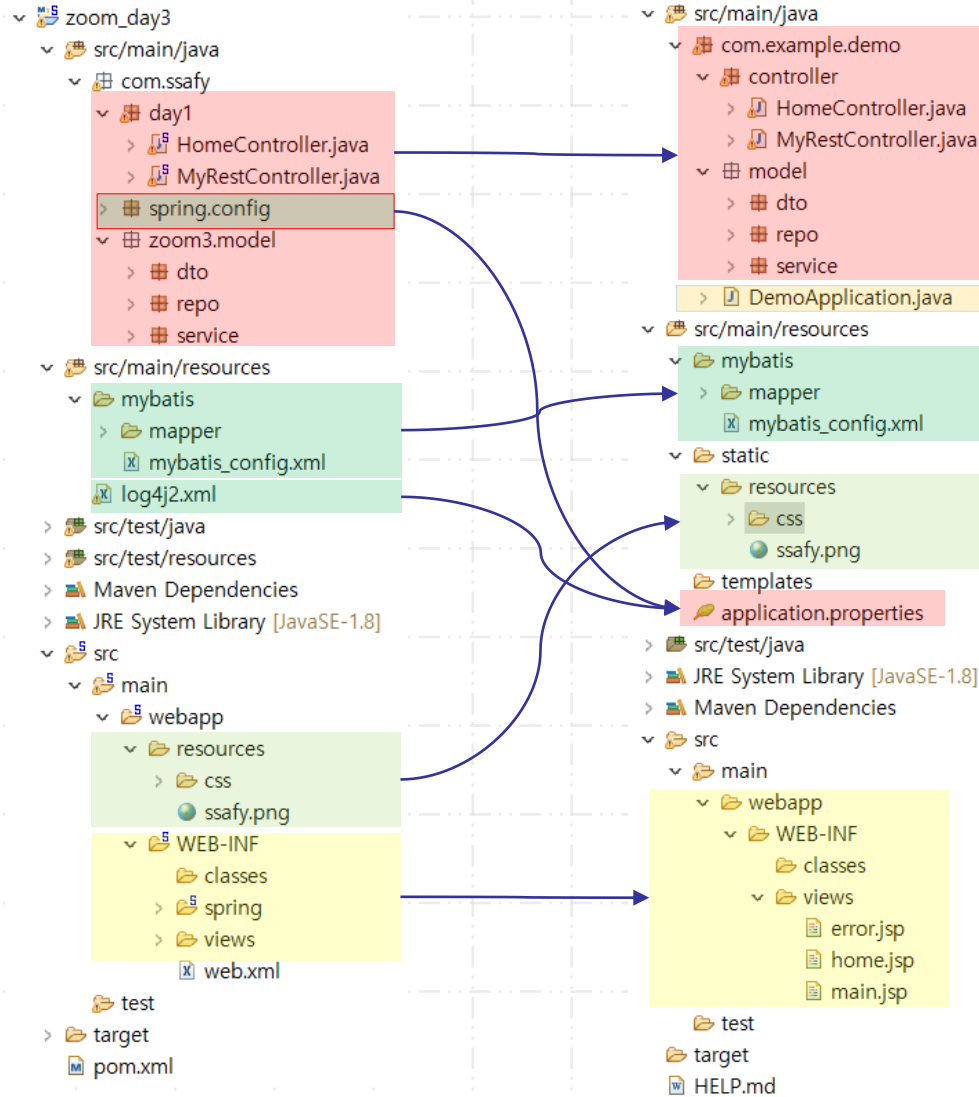
참조할 라이브러리 설정
버전은 모두 parent에서 관리

```
<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
```

애플리케이션 빌드 및 실행을 위한
maven plugin 설정

Legacy to boot

❖ 초기 파일 이동



자동 스캔

시작 파일

모든 설정들은
application.properties로 이동

Legacy to boot

❖ 자동으로 빈을 스캐닝하므로 Service관련 명시적 빈 등록 제거

- ◆ 기본 XXApplication.java의 하위 패키지는 자동으로 스캔
- ◆ Configuration에서 component scan 제거

Legacy to boot

❖ ApplicationConfig.java에서 PropertySource 설정 제거

- Config.properties의 내용은 application.properties로 이동
- DB 접속 관련 설정은 다음으로 변경

```
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://localhost:3306/ssafy_sf?serverTimezone=UTC~~
spring.datasource.username=root
spring.datasource.password=ssafy
spring.datasource.type=com.zaxxer.hikari.HikariDataSource
```

- 이에 따라 DataSource도 자동 생성되므로 아래 코드 삭제

```
@Bean
public DataSource ds(@Value("${db.driverClassName}") String driver,
                    @Value("${db.url}") String url,
                    @Value("${db.userName}") String user,
                    @Value("${db.password}") String pass) {
    DriverManagerDataSource ds = new DriverManagerDataSource();
    ds.setDriverClassName(driver);
    ds.setUrl(url);
    ds.setUsername(user);
    ds.setPassword(pass);
    return ds;
}
```

Legacy to boot

❖ Mybatis관련

● Application.properties에 아래 추가

- ◆ mybatis.config-location=classpath:/mybatis/mybatis_config.xml
- ◆ 이 설정으로 SqlSession 자동 생성

● Mybatis 관련 빈 삭제

```
@Bean
public SqlSessionFactoryBean sqlSessionFactoryBean(DataSource ds, ApplicationContext ctx) {
    SqlSessionFactoryBean bean = new SqlSessionFactoryBean();
    bean.setDataSource(ds);
    String location = "classpath:/mybatis/mybatis_config.xml";
    bean.setConfigLocation(ctx.getResource(location));
    return bean;
}
```

```
@Bean
public SqlSessionTemplate template(SqlSessionFactoryBean factory) throws Exception {
    SqlSessionTemplate template = new SqlSessionTemplate(factory.getObject());
    return template;
}
```

Legacy to boot

❖ JSP 사용 설정 --> JSP 자체가 비권장. ㅜㅜ

● 명시적 view resolver 삭제 및 @EnableWebMvc 제거

```
//@EnableWebMvc
public class MVCCConfig implements WebMvcConfigurer {
    /*@Bean
    public ViewResolver internalResourceViewResolver() {
        InternalResourceViewResolver resolver = new InternalResourceViewResolver();
        resolver.setPrefix("/WEB-INF/view/");
        resolver.setSuffix(".jsp");
        return resolver;
    }*/
}
```

● Application.properties에 아래 추가

- ◆ spring.mvc.view.prefix=/WEB-INF/view/
- ◆ spring.mvc.view.suffix=.jsp

● 의존성 추가

```
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>jstl</artifactId>
</dependency>
<dependency>
    <groupId>org.apache.tomcat.embed</groupId>
    <artifactId>tomcat-embed-jasper</artifactId>
</dependency>
```

내장 톰캣에는 jasper 가 없어서 JSP 못돌림

- 버전은 가져오지 않는다!! 자동 관리