



Day2; 20220902

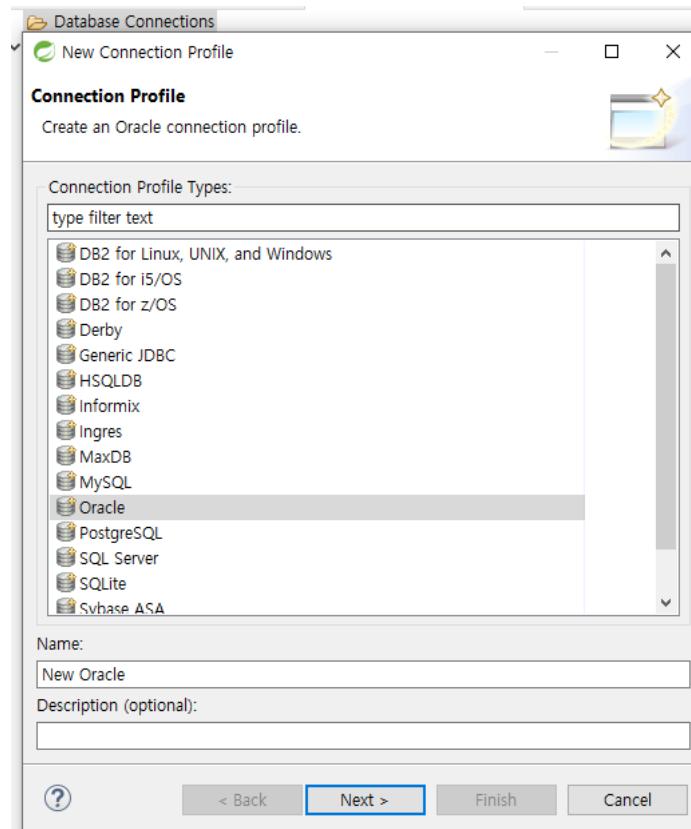
날짜	
유형	@2022년 9월 2일
태그	

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/ef12db58-5560-4c56-be99-45aa38dc0306/03%EC%9E%A5._Spring_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_%EC%83%9D%EC%84%B1.pdf

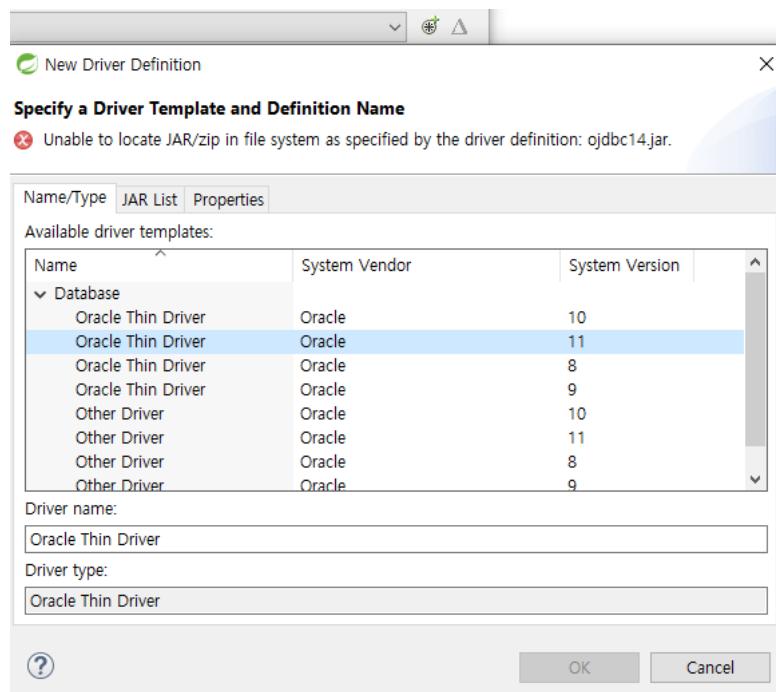
[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/a6ddd14e-a928-4dae-ac01-a9cf27f4e181/04%EC%9E%A5._Dependency_Injection\(%EC%9D%98%EC%A1%B4%EC%84%B1_%EC%A3%BC%EC%9E%85\).pdf](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/a6ddd14e-a928-4dae-ac01-a9cf27f4e181/04%EC%9E%A5._Dependency_Injection(%EC%9D%98%EC%A1%B4%EC%84%B1_%EC%A3%BC%EC%9E%85).pdf)

STS3과 오라클 11 연결

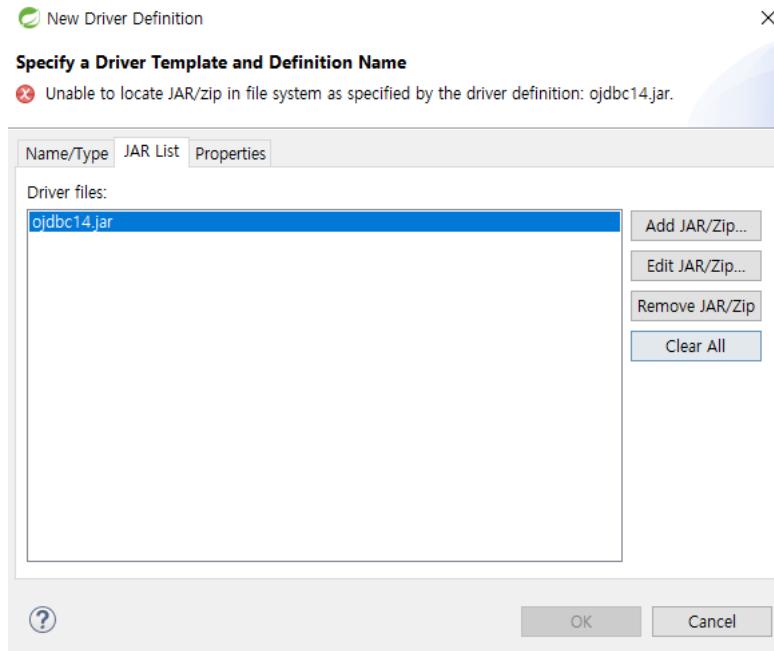
Data Source Explorer - New



별 모양을 클릭

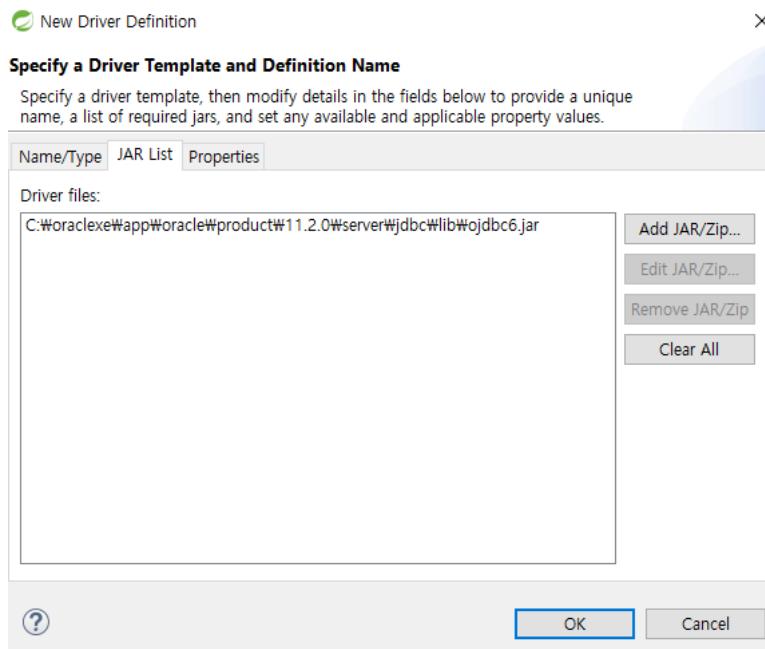


Clear All(Driver files 내용을 지워주자)

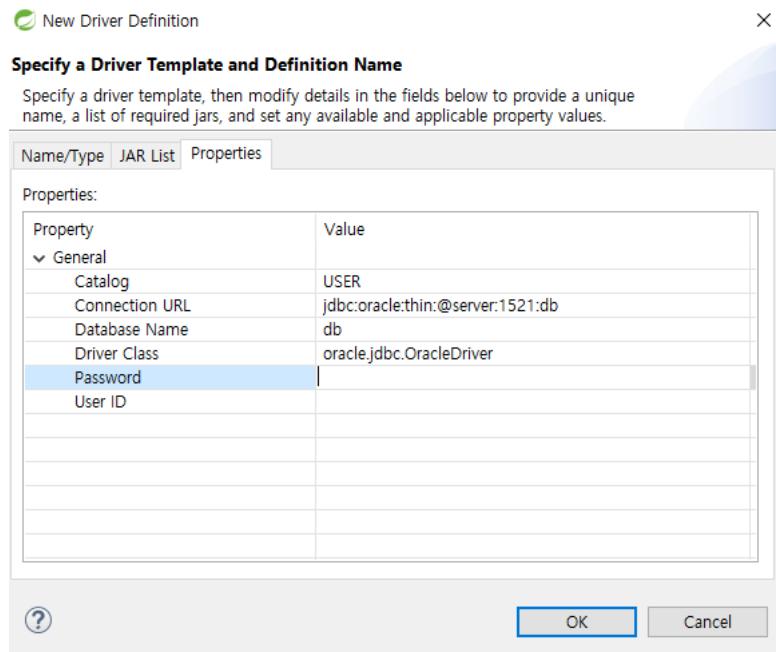


Add jaR/Zip 클릭

C:\oracle\product\11.2.0\server\jdbc\lib\ojdbc6.jar



Properties 탭 클릭



jdbc:oracle:thin:@localhost:1521:xe로 변경해주자.

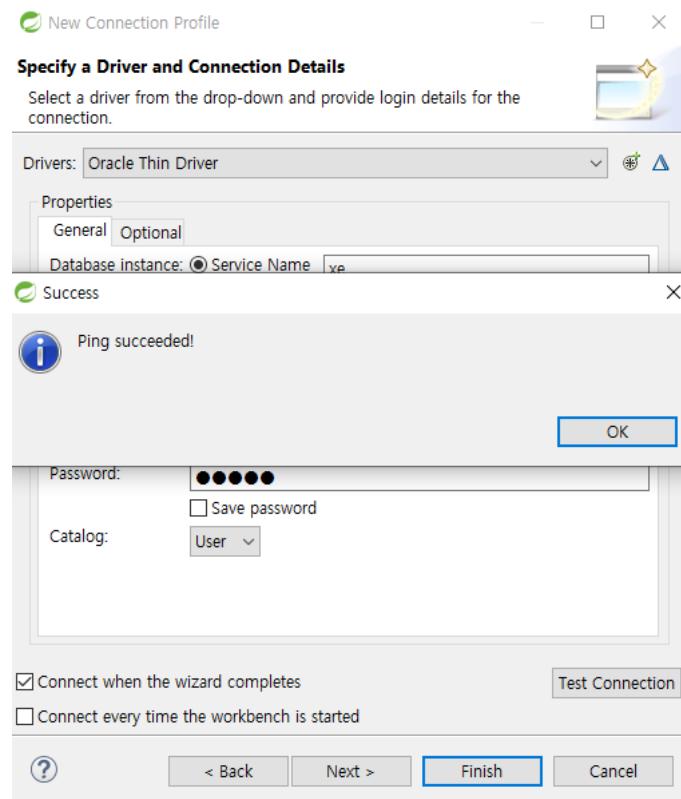
user ID: scott

Password: tiger

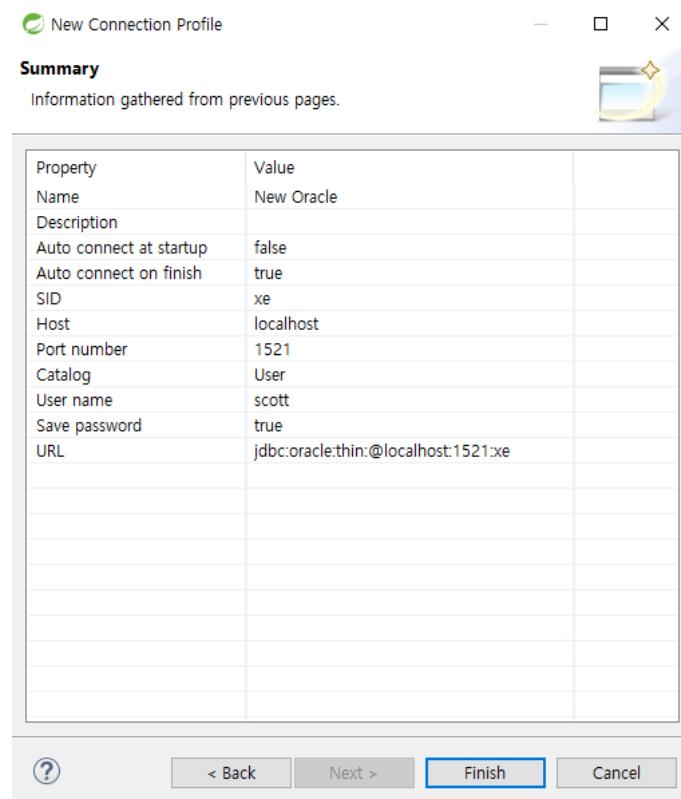
로컬이고 오라클은 버전이 xe이기 때문에 변경

1521은 오라클이 만든 번호이다.

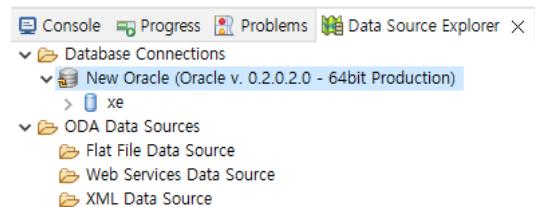
Ping succeeded! 가 뜨게끔 Test Connection을 눌러주자.



Finish

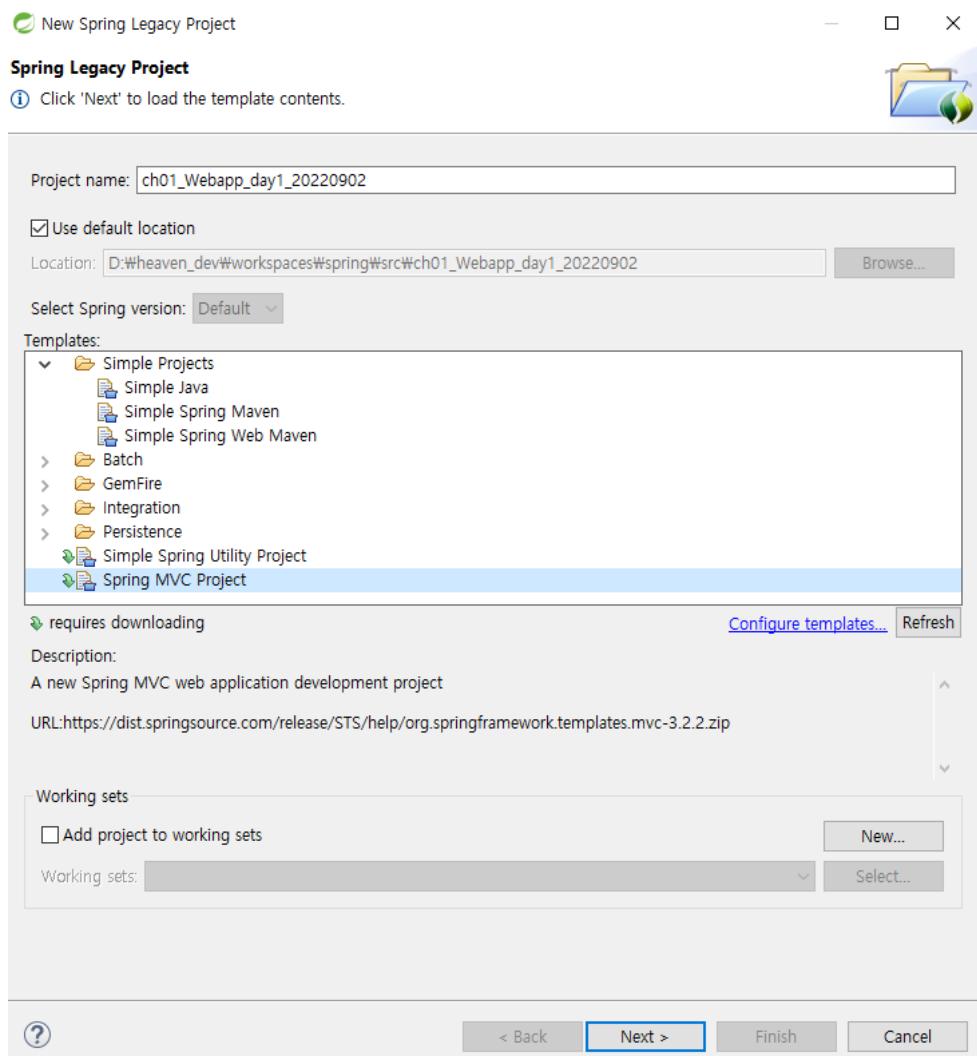


이제 만들어진 것을 눈으로 확인.

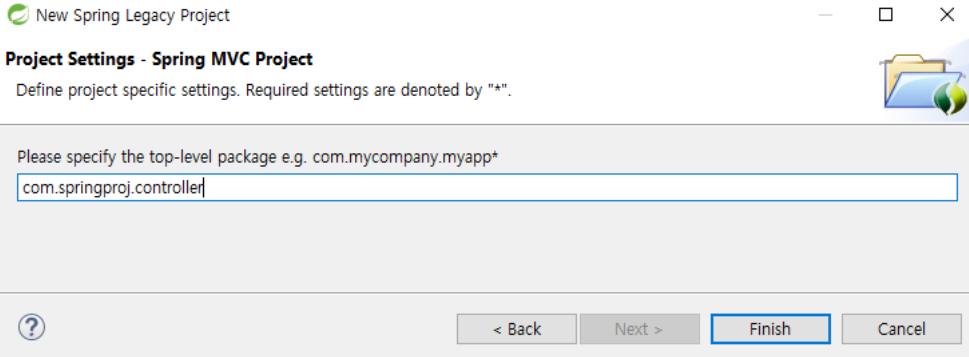


이제 프로젝트를 만들어보자.

Spring MVC project - Next



com.springproj.controller 로 MVC project 입력해준다.



pom.xml에서 자바 버전을 지금 깔려있는 11로 변경해준다.

```

<?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>com.springproj</groupId>
    <artifactId>controller</artifactId>
    <name>ch01_Webapp_day1_20220902</name>
    <packaging>war</packaging>
    <version>1.0.0-BUILD-SNAPSHOT</version>
    <properties>
        <java-version>11</java-version>
        <org.springframework-version>3.1.1.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>
            <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>
        <!-- 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>
        </dependency>
    </dependencies>

```

spring core도 이름만 release 기준으로 변경해줌.

<https://mvnrepository.com/artifact/org.springframework/spring-core>

Maven Repository: org.springframework

mvnrepository.com/artifact/org.springframework/spring-core

	Version	Vulnerabilities	Repository	Downloads	Last Update	
	5.3.6	8 vulnerabilities	Central	592	Mar 13, 2021	View
	5.3.5	8 vulnerabilities	Central	549	Mar 16, 2021	View
	5.3.4	8 vulnerabilities	Central	539	Feb 16, 2021	View
	5.3.3	8 vulnerabilities	Central	572	Jan 12, 2021	View
	5.3.2	8 vulnerabilities	Central	111	Dec 09, 2020	View
	5.3.1	8 vulnerabilities	Central	108	Nov 10, 2020	View
	5.3.0	8 vulnerabilities	Central	92	Oct 27, 2020	View
	5.2.22.RELEASE		Central	76	May 11, 2022	View
	5.2.21.RELEASE	2 vulnerabilities	Central	31	Apr 13, 2022	View
	5.2.20.RELEASE	3 vulnerabilities	Central	106	Mar 31, 2022	View
	5.2.19.RELEASE	5 vulnerabilities	Central	101	Dec 16, 2021	View
	5.2.18.RELEASE	6 vulnerabilities	Central	43	Oct 14, 2021	View
	5.2.17.RELEASE	6 vulnerabilities	Central	73	Sep 15, 2021	View
	5.2.16.RELEASE	7 vulnerabilities	Central	34	Jul 14, 2021	View
				...	May	

```

8 <packaging>war</packaging>
9 <version>1.0.0-BUILD-SNAPSHOT</version>
0<properties>
1   <java-version>11</java-version>
2   <org.springframework-version>5.2.22.RELEASE</org.springpri
3   <org.aspectj-version>1.6.10</org.aspectj-version>
4   <org.slf4j-version>1.6.6</org.slf4j-version>
5 </properties>
6<dependencies>
7   <!-- Spring -->
8<dependency>
9   <groupId>org.springframework</groupId>
0   <artifactId>spring-context</artifactId>

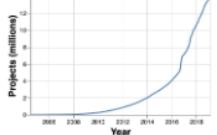
```

Servlet을 Tomcat9버전으로 맞춰서 바꿔주자 (2.5 → 4.0.1 으로)


mvnrepository.com/artifact/javax.servlet/javax.servlet-api

Search for groups, artifacts, categories Search

Indexed Artifacts (29.5M)



Popular Categories

- Aspect Oriented
- Actor Frameworks
- Application Metrics
- Build Tools
- Bytecode Libraries
- Command Line Parsers
- Cache Implementations
- Cloud Computing
- Code Analyzers
- Collections
- Configuration Libraries
- Core Utilities
- Date and Time Utilities
- Dependency Injection
- Embedded SQL Databases
- HTML Parsers
- HTTP Clients
- I/O Utilities
- JDBC Extensions
- JDBC Pools
- JPA Implementations
- JSON Libraries
- JVM Languages
- Logging Frameworks
- Logging Bridges
- Mail Clients
- Maven Plugins
- Mocking

Java Servlet API

Java Servlet is the foundation web specification in the Java Enterprise Platform. Developers can build web applications using the Servlet API to interact with the request/response workflow.

License	CDDL GPL 2.0
Categories	Java Specifications
Tags	standard servlet javax api specs
Ranking	#17 in MvnRepository (See Top Artifacts) #1 in Java Specifications
Used By	16,859 artifacts

[Central \(20\)](#) [Redhat GA \(1\)](#) [ICM \(5\)](#)

	Version	Vulnerabilities	Repository	Usages	Date
4.0.x	4.0.1		Central	3,970	Apr 20, 2018
	4.0.0		Central	449	Aug 15, 2017
	4.0.0-b07		Central	22	Jun 02, 2017
	4.0.0-b06		Central	0	May 24, 2017
	4.0.0-b05		Central	4	Mar 29, 2017
	4.0.0-b04		Central	0	Mar 16, 2017
	4.0.0-b03		Central	2	Mar 02, 2017
	4.0.0-b02		Central	3	Feb 03, 2017
	4.0.0-b01		Central	26	Oct 09, 2015
	3.1.0		Central	9,680	Apr 25, 2013
	3.1-b09		Central	1	Apr 24, 2013
	3.1-b08		Central	3	Apr 05, 2013

MVNREPOSITORY

Indexed Artifacts (29.5M)

Popular Categories

- Aspect Oriented
- Actor Frameworks
- Application Metrics
- Build Tools
- Bytecode Libraries
- Command Line Parsers
- Cache Implementations
- Cloud Computing
- Code Analyzers
- Collections
- Configuration Libraries
- Core Utilities
- Date and Time Utilities
- Dependency Injection
- Embedded SQL Databases
- HTML Parsers
- HTTP Clients
- I/O Utilities
- JDBC Extensions
- JDBC Pools
- JPA Implementations
- JSON Libraries

Search for groups, artifacts, categories

Home » javax.servlet » javax.servlet-api » 4.0.1

Java Servlet API » 4.0.1

Java Servlet is the foundation web specification in the Java Enterprise Platform. Developers can build web applications using the Servlet API to interact with the request/response workflow.

License	CDDL GPL 2.0
Categories	Java Specifications
Tags	standard servlet javax api specs
Organization	GlassFish Community
HomePage	https://javaee.github.io/servlet-spec/
Date	Apr 20, 2018
Files	pom (15 KB) jar (93 KB) View All
Repositories	Central
Ranking	#17 in MvnRepository (See Top Artifacts) #1 in Java Specifications
Used By	16,859 artifacts

[Maven](#) | [Gradle](#) | [Gradle \(Short\)](#) | [Gradle \(Kotlin\)](#) | [SBT](#) | [Ivy](#) | [Grape](#)
[Leiningen](#) | [Buildr](#)

```

<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>4.0.1</version>
    <scope>provided</scope>
</dependency>

```

Include comment with link to declaration
Copied to clipboard!

<https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api>

```
<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>4.0.1</version>
    <scope>provided</scope>
</dependency>
```

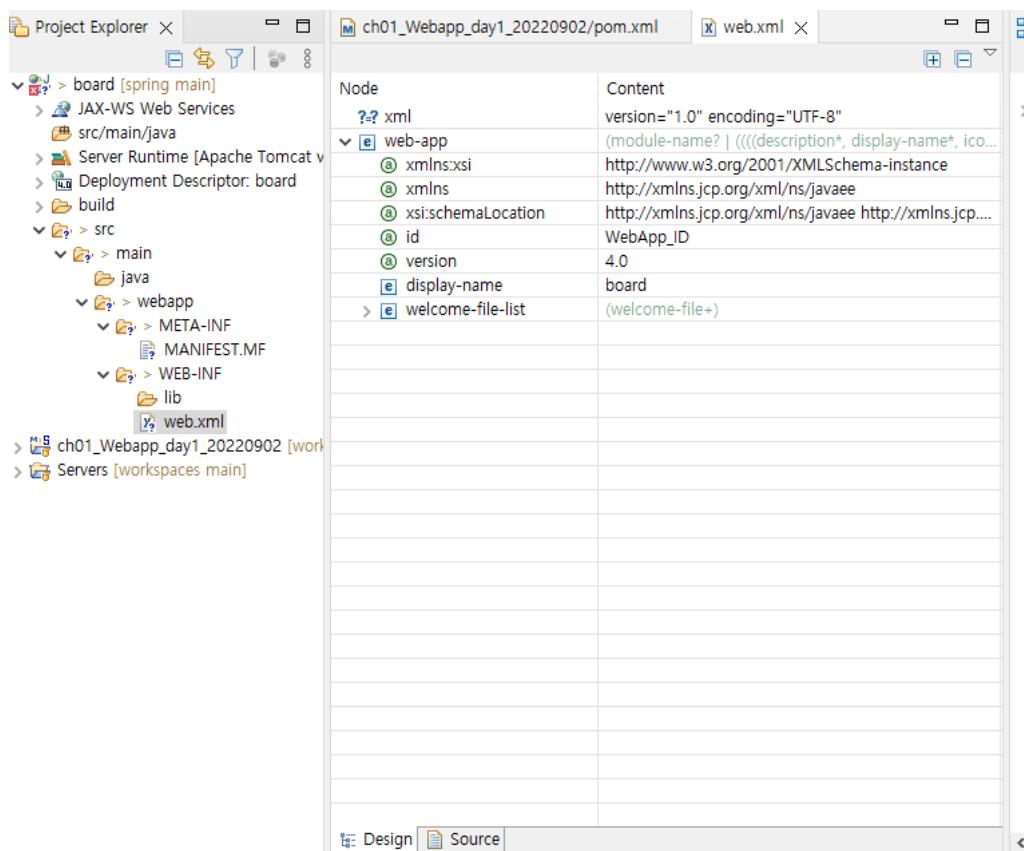
```
<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>4.0.1</version>
    <scope>provided</scope>
</dependency>
<dependency>
```

자바스크립트로 프로젝트를 만들 경우에

tomcat9에게 알려줄 내용이 있다면 web.xml로 알려주면 된다.

서블릿 버전(4.0)과 자바환경으로 변경해준 것이다.

만약에 서블릿 버전이 2.5라면 4.0dml web-app 환경으로 변경해준다.



```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/ns/javaee" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://www.w3.org/2001/XMLSchema-instance" version="3.1">
3   <display-name>board</display-name>
4   <welcome-file-list>
5     <welcome-file>index.html</welcome-file>
6     <welcome-file>index.htm</welcome-file>
7     <welcome-file>index.jsp</welcome-file>
8     <welcome-file>default.html</welcome-file>
9     <welcome-file>default.htm</welcome-file>
10    <welcome-file>default.jsp</welcome-file>
11  </welcome-file-list>
12 </web-app>

```

자바 프로젝트 서블릿 4.0 을 Spring에 있는 web.xml에 넣어준다.

```

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/ns/javaee" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://www.w3.org/2001/XMLSchema-instance" version="3.1">
  <!-- The definition of the Root Spring Container shared by all Servlets and Filters -->
  <context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>/WEB-INF/spring/root-context.xml</param-value>
  </context-param>
  <!-- Creates the Spring Container shared by all Servlets and Filters -->
  <listener>
    <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
  </listener>
  <!-- Processes application requests -->
  <servlet>
    <servlet-name>appServlet</servlet-name>
    <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
    <init-param>
      <param-name>contextConfigLocation</param-name>
      <param-value>/WEB-INF/spring/appServlet/servlet-context.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>

```

pom.xml 에서 plugin 내용을 1.6을 11로 변경해준다.

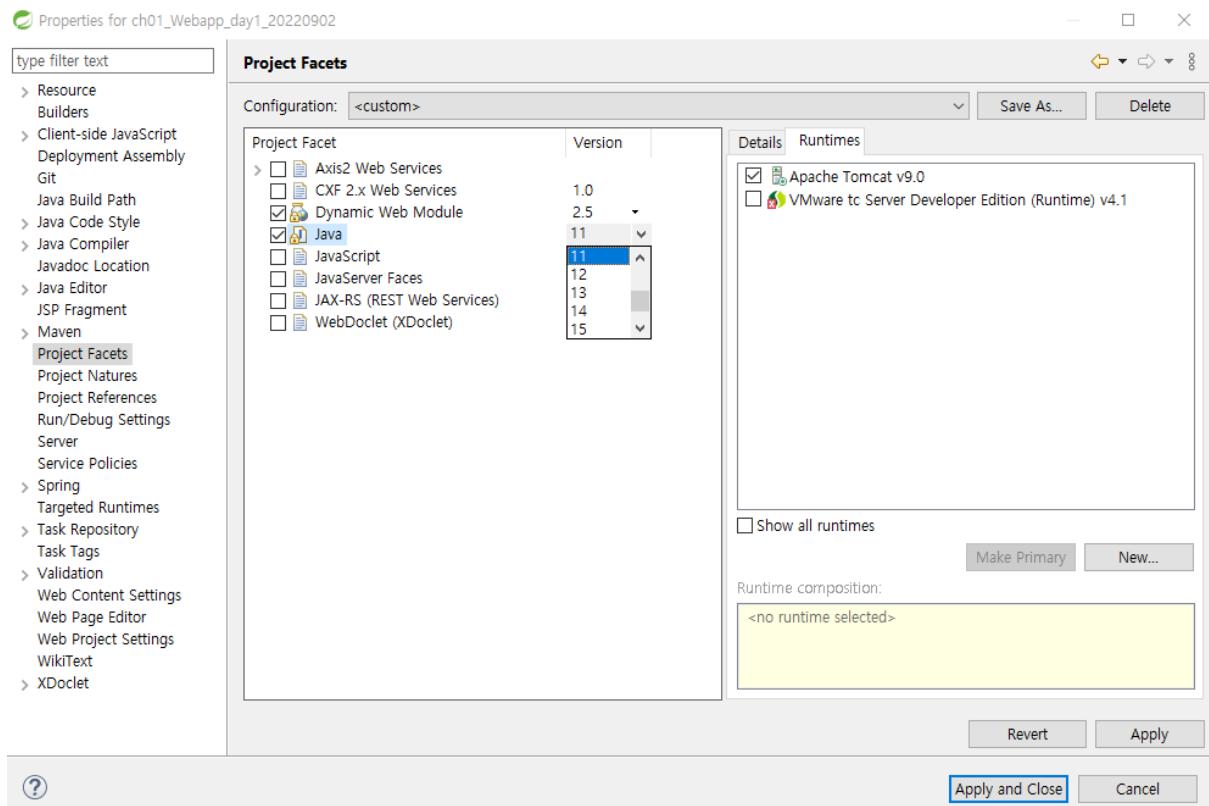
```
</configuration>
</plugin>
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-compiler-plugin</artifactId>
    <version>2.5.1</version>
    <configuration>
        <source>1.6</source>
        <target>1.6</target>
        <compilerArgument>-Xlint:all</compilerArgument>
        <showWarnings>true</showWarnings>
        <showDeprecation>true</showDeprecation>
    </configuration>
</plugin>
<plugin>
    <groupId>org.codehaus.mojo</groupId>
```

11로 변경.

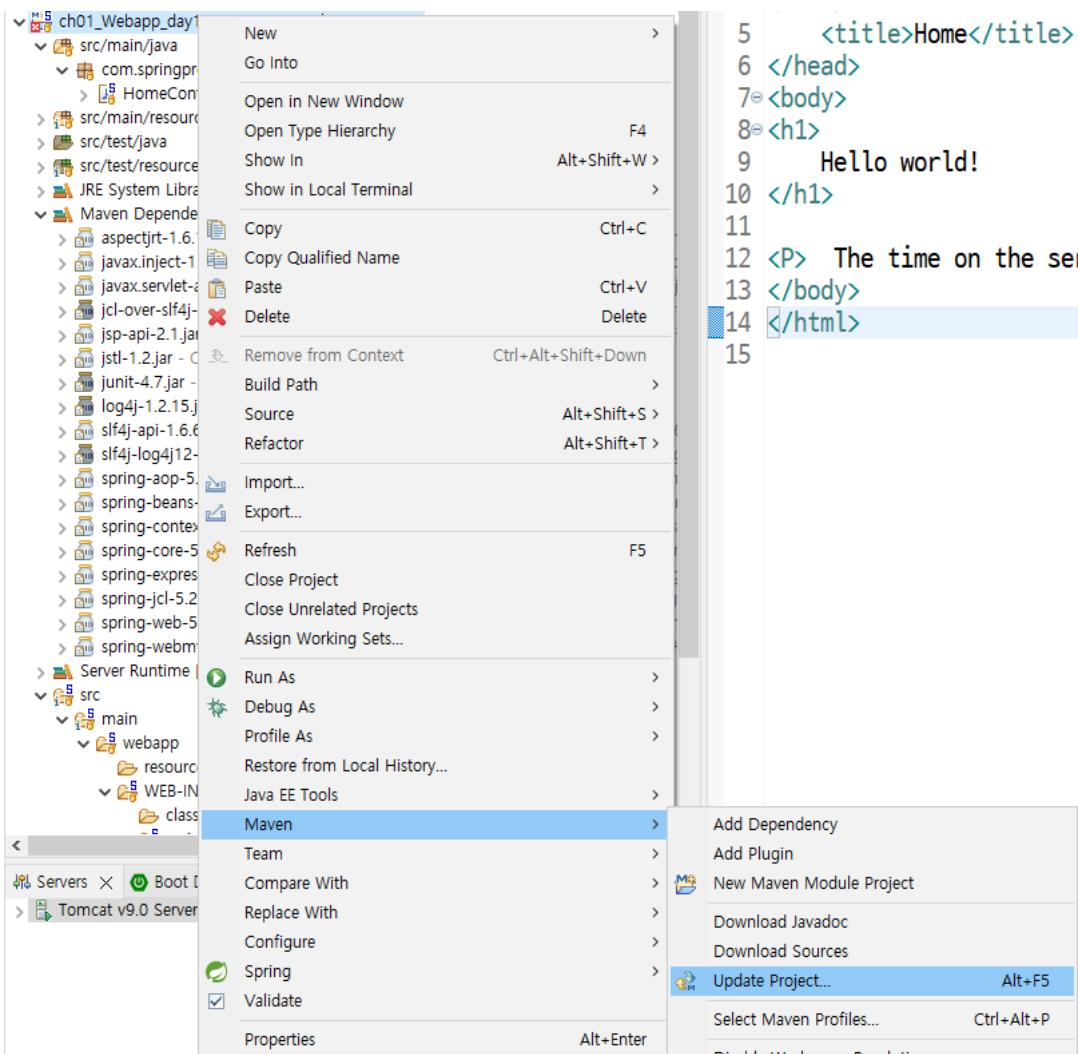
```
<artifactId>maven-compiler-plugin</artifactId>
<version>2.5.1</version>
<configuration>
    <source>11</source>
    <target>11</target>
    <compilerArgument>-Xlint:all</compilerArgument>
    <showWarnings>true</showWarnings>
```

spring에서 properties 들어가서 자바 버전을 11로 변경해주고, tomcat9 체크박스를 클릭해준다.

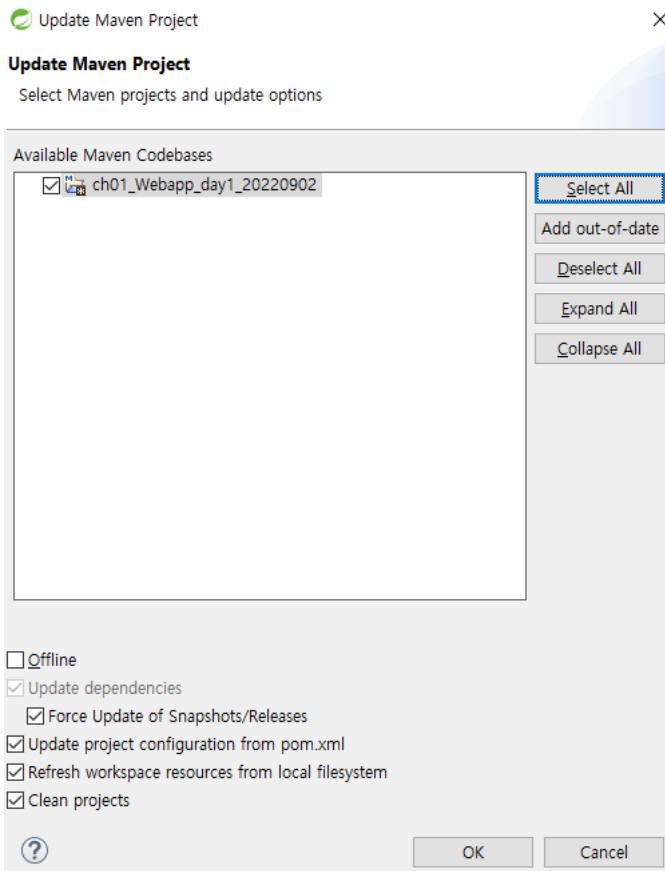
Dynamic Web Module은 4로 변경.



업데이트 해준다.



Force Update of Snapshots/Releases 클릭해준다.



Model2 버전으로 화면으로 구현해보자.

src - main - webapp - WEB-INF - views - home.jsp

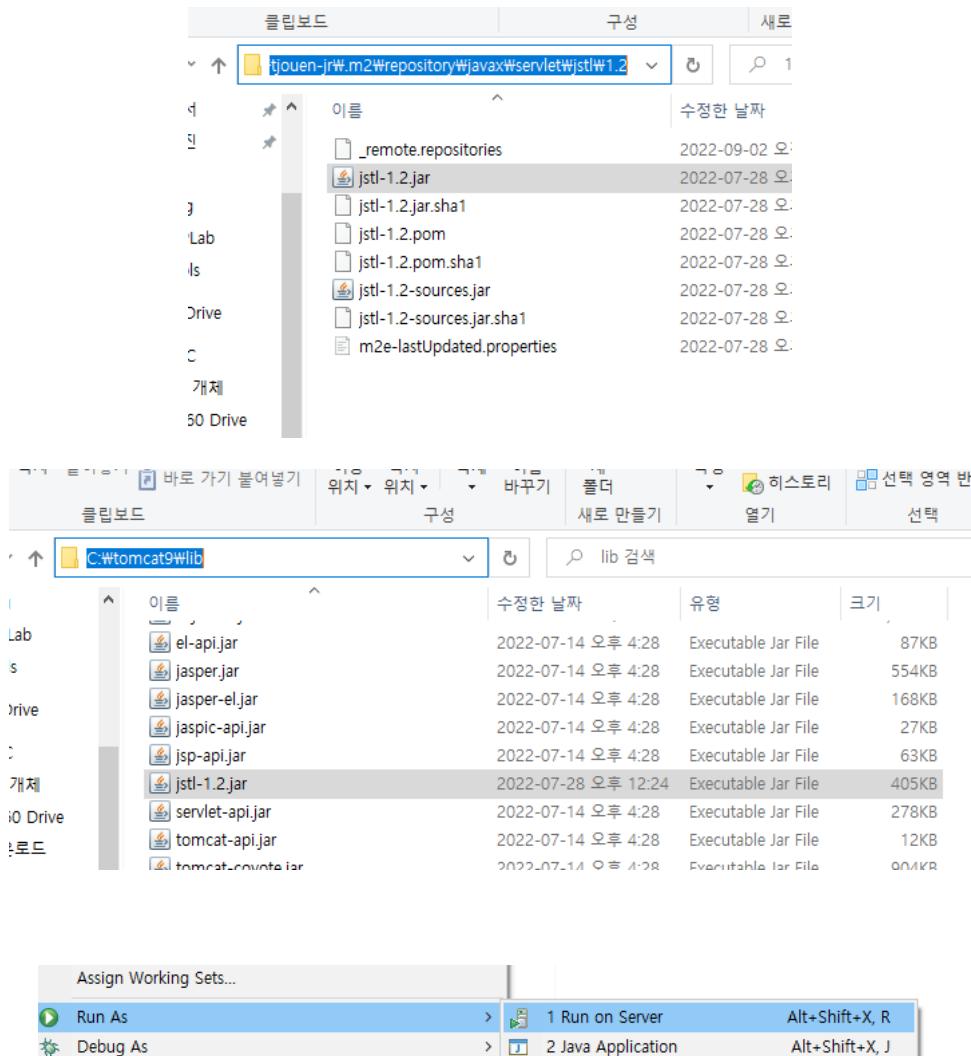
```

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
<title>Home</title>
</head>
<body>
<h1>
Hello world!
</h1>
11
12 <P> The time on the server is ${serverTime}. </P>
13 </body>
14 </html>
15

```

tomcat9 - lib 폴더에 jstl-1.2.jar(화면을 구현해주는 lib) 폴더를 넣어준다.

C:\Users\tjouen-jr\m2\repository\javax\servlet\jstl\1.2



Hello world!

The time on the server is 2022? 9? 2? ?? 12? 16? 27? KST.

완성!!

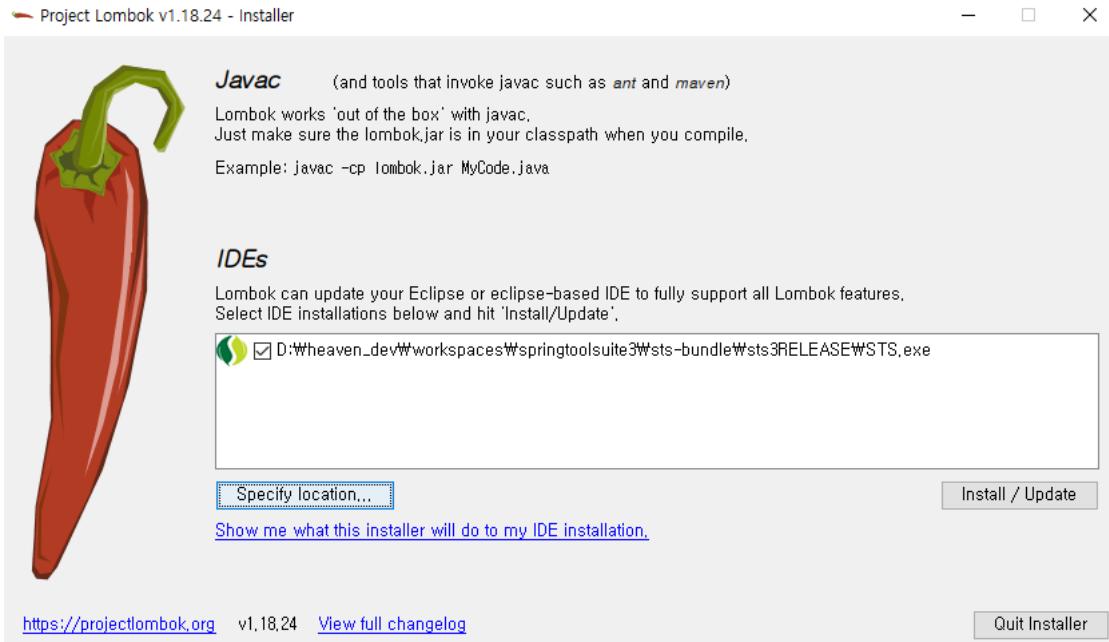
Lombok 라이브러리 설치

여기에서 PowerShell 창 열기



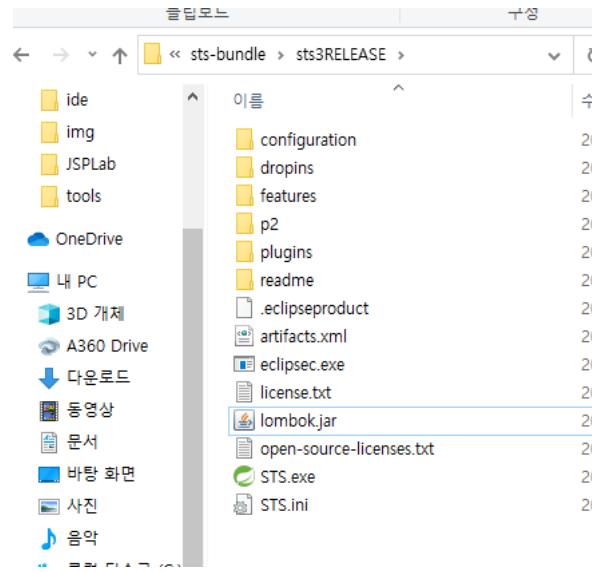
Java -jar lombok.jar 엔터

```
Windows PowerShell
PS D:\heaven_dev\workspaces\tools> java -jar lombok.jar
```



Install 눌러주면 Success!!

lombok.jar 이 포함되어져 있음.



DI(Dependency Injection) 의존성 주입

클래스 안에서 기본적으로는 필드, 생성자, 메서드가 들어간다.

기능에 해당되면 메서드로 표현하라는 것이다.

삼성이라는 속성은 변수로 선언한다.

색상을 관리해야 한다면 속성이니깐 String으로 변수선언해서 관리하면 될 것이다.

찬바람 나오라는 것은 기능이니 메서드로.

필드는 생성자 안에서 초기화하라고 적용하라고 하는 참조자료형이다.

int나 double처럼 키워드를 선언해주는 것이다.

```
package com.springproj.ex01;

public class LgTV {

    public void turnOn() {
        System.out.println("LgTV 전원을 켜다.");
    }

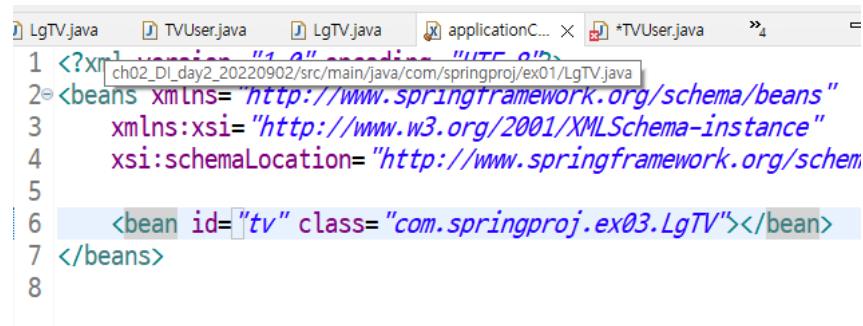
    public void turnOff() {
        System.out.println("LgTV 전원을 끈다.");
    }

    public void soundUp() {
        System.out.println("LgTV 소리를 커다.");
    }

    public void soundoff() {
    }
}
```

```
        System.out.println("LgTV 소리를 끄다.");
    }
}
```

bean



The screenshot shows a code editor with several tabs at the top: LgTV.java, TVUser.java, LgTV.java, applicationC..., and *TVUser.java. The main content area displays an XML configuration file:

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
                           http://www.springframework.org/schema/beans/spring-beans.xsd">
    <bean id="tv" class="com.springproj.ex03.LgTV"></bean>

```

Spring Framework

Framework 란?

- 빠대나 근간을 이루는 코드들의 묶음.
 - “소프트웨어의 구체적인 부분에 해당하는 설계화 구현을 재사용이 가능하게끔 일련의 협업화 된 형태로 클래스들을 제공하는 것”.
 - 설계의 기반이 되는 부분을 기술한 확장 가능한 기반 코드와 사용자가 이 코드를 자기 입맛대로 확장하는데 필요한 라이브러리 이 두 가지 요소가 통합되어 제공되는 형태.
 - 사용자가 이를 이용해 일정 수준 이상의 품질을 보장받는 코드를, 비교적 빠른 시간에 완성 및 유지 보수할 수 있는 환경을 제공.
 - “기본적인 설계나 필요한 라이브러리는 알아서 제공해 주므로 기능 구현에만 전념!”이라는 취지에서 만들어진 것.
-

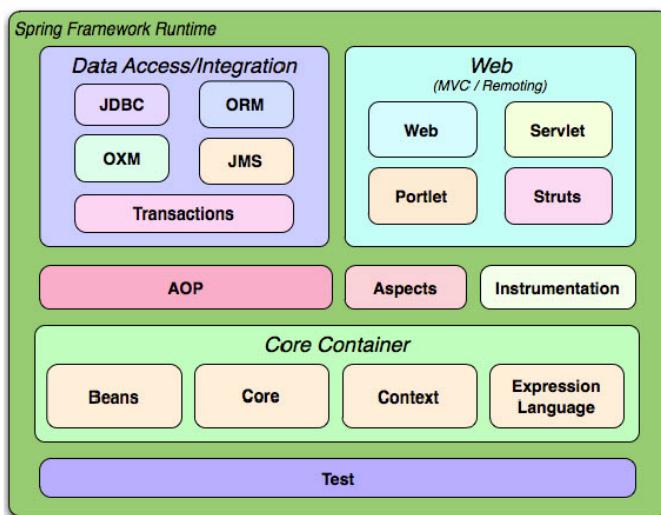
Spring 이란?

- 오픈 소스 프레임워크
 - Rod Johnson 창시
 - Expert one-on-one J2EE Design - Development, 2002, Wrox
 - Expert one-on-one J2EE Development without EJB, 2004, Wrox
 - 엔터프라이즈 어플리케이션 개발의 복잡성을 줄여주기 위한 목적
 - EJB 사용으로 수행되었던 모든 기능을 일반 POJO(Plain Old Java Object) 를 사용해서 가능하게 함.
 - 경량 컨테이너(light weight container)
 - 주요 개념
 - 의존성 주입(lightweight dependency injection)
 - 관점 지향 컨테이너(aspect-oriented container)
 - 포터블 서비스 추상화(Portable Service Abstract)
-

Spring 장점

- 전자 정부 프레임워크
- 객체의 라이프 사이클 관리, JEE 구현을 위한 다양한 API제공
- DI (Dependency Injection) 지원
 - 각 계층이나 서비스들 사이 또는 객체들 사이에 의존성이 존재할 경우 스프링 프레임워크가 서로를 연결 시켜 줌. 이는 클래스들 사이에 약한 결합을 가능케 함.
- AOP (Aspect Oriented Programming) 지원
 - 트랜잭션, 로깅, 보안등 여러 모듈에서 공통적으로 지원하는 기능을 분리하여 사용할 수 있음.
- POJO (Plain Old Java Object) 지원
 - 자바 객체의 라이프사이클을 스프링 컨테이너가 직접 관리하며, 스프링 컨테이너로부터 필요한 객체를 얻어올 수 있음.
- JDBC를 위한 다양한 API 지원
- Transaction 처리를 위한 일관된 방법 제공
- 확장성이 높다.
 - 스프링 프레임워크의 소스는 모두 라이브러리로 분리시켜 놓음으로써 필요한 라이브러리만 가져다 쓸 수 있음. 그리고 많은 외부 라이브러리들도 이미 스프링 프레임워크와 연동되고 있음
- Model2 방식의MVC Framework를 지원

Spring의 구성 요소

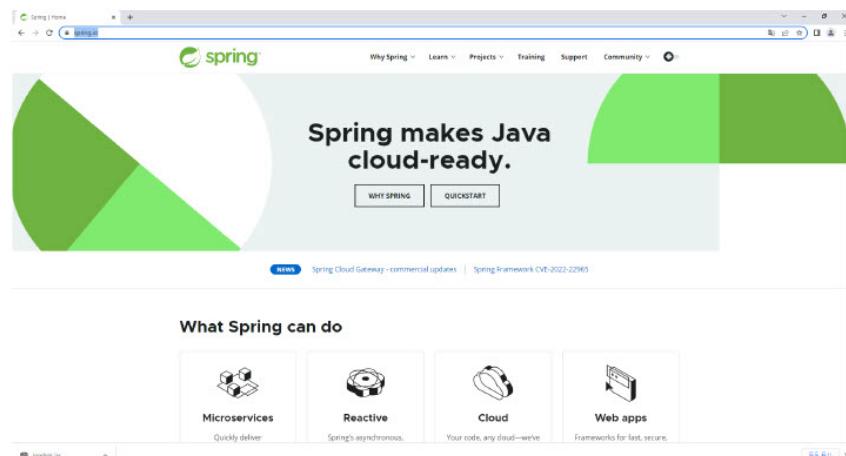


Spring Framework 모듈

모듈	설명
spring-beans	스프링 컨테이너를 이용해서 객체를 생성하는 기본 기능을 제공합니다.
spring-context	객체 생성, 라이프 사이클 관리, 스키마 확장 등의 기능을 제공합니다.
spring-aop	프락시 기반 AOP 기능을 제공합니다.
spring-web	REST 클라이언트, 데이터 변환 처리, 서블릿 필터, 파일 업로드 지원 등 웹 개발에 필요한 기반 기능을 제공합니다.
spring-webmvc	스프링 기반의 웹 MVC 프레임워크입니다. 웹 애플리케이션을 개발하는데 필요한 컨트롤러와 뷰 구현을 제공합니다.
spring-websocket	스프링 MVC에서 웹소켓을 사용하기 위한 기능을 지원합니다.
spring-oxm	XML과 자바 객체 간의 매핑을 처리하기 위한 API를 제공합니다.
spring-tx	트랜잭션 처리를 위한 추상 레이어를 제공합니다.
spring-jdbc	JDBC 프로그래밍을 보다 쉽게 할 수 있는 템플릿을 제공합니다. 이를 이용하면 JDBC 프로그래밍에서 반복적으로 입력해야 하는 코드를 줄일 수 있습니다.
spring-orm	하이버네이트, JPA, MyBatis 등과의 연동을 지원합니다.
spring-jms	JMS 서비스와 메시지를 쉽게 주고받을 수 있도록 템플릿과 아노테이션 등을 제공합니다.
spring-context-support	스케줄링, 메일 발송, 캐시 연동, 벨로시티 등 부가기능을 제공합니다.

Spring Container 설치(1/2)

- 스프링 사이트 : <https://spring.io/>
 - Reference / API 참조 사이트



Spring Container 설치(2/2)

- 스프링 다운로드 사이트
 - <https://repo.spring.io/ui/native/release/org/springframework/spring/>
 - 스프링 버전별 다운로드 가능

Name	Last Modified	Size	Download Link
...			
1.0.1	00-03-22 11:45:51 +0900		
1.0.1-SNAPSHOT	00-03-22 12:20:57 +0900		
1.0.2	00-03-22 23:18:24 +0900		
1.0.3	00-03-22 23:18:29 +0900		
1.1.0	00-03-22 16:11:32 +0900		
1.1.0-SNAPSHOT	00-03-22 16:11:32 +0900		
1.1.1	00-03-22 16:20:31 +0900		
1.1.1-SNAPSHOT	00-03-22 21:18:34 +0900		
1.1.2	00-03-22 21:18:38 +0900		
1.1.3	00-03-22 21:18:40 +0900		
1.1.4	00-03-22 21:18:40 +0900		
1.1.5	00-03-22 21:18:45 +0900		
1.1.6	00-03-22 21:18:49 +0900		
1.1.7	00-03-22 21:18:51 +0900		
1.1.8	00-03-22 21:18:53 +0900		
1.1.9	00-03-22 21:18:56 +0900		
1.2.0	00-03-22 15:01:17 +0900		
1.2.1	00-03-22 21:19:00 +0900		
1.2.2	00-03-22 21:19:03 +0900		
1.2.3	00-03-22 21:19:07 +0900		
1.2.4	00-03-22 21:19:18 +0900		
1.2.5	00-03-22 21:19:20 +0900		
1.2.6	00-03-22 21:19:18 +0900		
1.2.7	00-03-22 21:19:18 +0900		
1.2.8	00-03-22 21:19:18 +0900		
1.3.0	00-03-22 21:19:21 +0900		
1.3.1	00-03-22 21:19:25 +0900		
1.3.2	00-03-22 21:19:27 +0900		

Spring 개발 환경 구축

STS 다운로드

- <https://github.com/spring-attic/toolsuite-distribution/wiki/Spring-Tool-Suite-3#spring-toolsuite-3918>

The screenshot shows the 'Latest STS3 Downloads' section of the GitHub page. It includes the title 'Spring Tool Suite 3.9.18' and a note 'full distribution on Eclipse 4.21'. Below is a list of download links:

- https://download.springsource.com/release/STS/3.9.18.RELEASE/dist/e4.21/spring-tool-suite-3.9.18.RELEASE-e4.21.0-win32-x86_64.zip (highlighted with a red box)
- https://download.springsource.com/release/STS/3.9.18.RELEASE/dist/e4.21/spring-tool-suite-3.9.18.RELEASE-e4.21.0-macosx-cocoa-x86_64.dmg
- https://download.springsource.com/release/STS/3.9.18.RELEASE/dist/e4.21/spring-tool-suite-3.9.18.RELEASE-e4.21.0-linux-gtk-x86_64.tar.gz

- 압축 해제 후 STS.ini 파일에 아래 내용 추가.

```
-vm  
C:\Program Files\Java\jdk-설치버전\bin\javaw.exe
```

Tomcat 설치

- <https://tomcat.apache.org/>

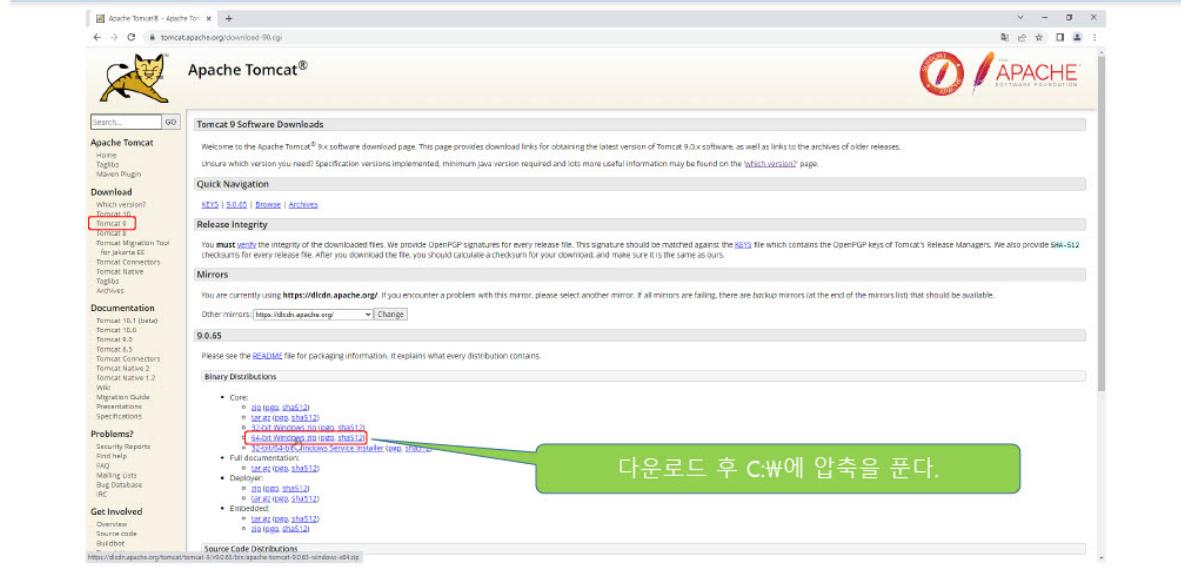
The screenshot shows the Apache Tomcat website's release history. It lists several releases:

- Tomcat 8.5.82 Released** (2022-08-13):
The Apache Tomcat Project is proud to announce the release of version 8.5.82 of Apache Tomcat. This release implements specifications that are part of the Java EE 7 platform. The notable changes compared to 8.5.81 include:
 - Updated the packaged version of the Tomcat Native Library to 1.2.35 to pick up Windows builds built with OpenSSL 1.1.1g.
 - Enable the use of the RPS provider in TELT enabled Connectors when using Tomcat Native 1.2.34 onwards built with OpenSSL 5.0.x onwards.
 - Fix CVE-2022-34936, a low severity XSS vulnerability in the Form authentication example.Full details of these changes, and all the other changes, are available in the [Tomcat 8.5 changelog](#).
- Tomcat 10.0.23 Released** (2022-07-26):
The Apache Tomcat Project is proud to announce the release of version 10.0.23 of Apache Tomcat. This release implements specifications that are part of the Jakarta EE 8 platform. Applications that run on Tomcat 9 and earlier will not run on Tomcat 10 without changes; Java EE based applications designed for Tomcat 9 and earlier may be placed in the `WEBALIVE_8555/webapps/jarname` directory and Tomcat will automatically convert them to Jakarta EE and copy them to the webapps directory. This conversion is performed using the [Apache Tomcat migration tool for Jakarta EE](#) which is also available as a separate [download](#) for off-line use.
The notable changes in this release are:
 - Implement support for repeatable builds.
 - Update the packaged version of the Tomcat Native Library to 1.2.35. This includes Windows binaries built with OpenSSL 1.1.1g.
 - Fix CVE-2022-34936, a low severity XSS vulnerability in the Form authentication example.Full details of these changes, and all the other changes, are available in the [Tomcat 10 changelog](#).
- Tomcat 9.0.65 Released** (2022-07-20):
The Apache Tomcat Project is proud to announce the release of version 9.0.65 of Apache Tomcat. This release implements specifications that are part of the Java EE 8 platform. The notable changes compared to 9.0.64 include:

Tomcat Versions

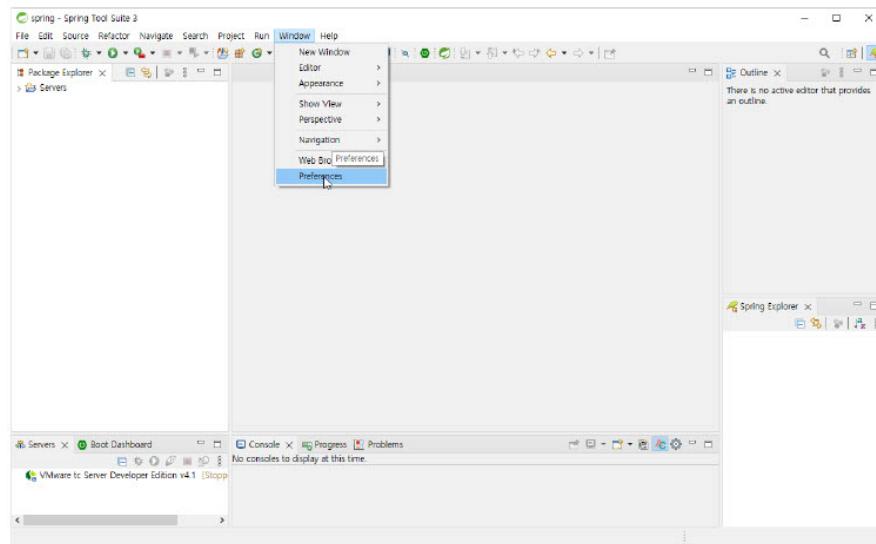
Servlet Spec	JSP Spec	EL Spec	WebSocket Spec	Authentication (JASPI) Spec	Apache Tomcat Version	Latest Released Version	Supported Java Versions
6.0	3.1	5.0	2.1	3.0	10.1.x	10.1.0-M17 (beta)	11 and later
5.0	3.0	4.0	2.0	2.0	10.0.x	10.0.23	8 and later
4.0	2.3	3.0	1.1	1.1	9.0.x	9.0.65	8 and later
3.1	2.3	3.0	1.1	1.1	8.5.x	8.5.82	7 and later
3.1	2.3	3.0	1.1	N/A	8.0.x (superseded)	8.0.53 (superseded)	7 and later
3.0	2.2	2.2	1.1	N/A	7.0.x (archived)	7.0.109 (archived)	6 and later (7 and later for WebSocket)
2.5	2.1	2.1	N/A	N/A	6.0.x (archived)	6.0.53 (archived)	5 and later
2.4	2.0	N/A	N/A	N/A	5.5.x (archived)	5.5.36 (archived)	1.4 and later
2.3	1.2	N/A	N/A	N/A	4.1.x (archived)	4.1.40 (archived)	1.3 and later
2.2	1.1	N/A	N/A	N/A	3.3.x (archived)	3.3.2 (archived)	1.1 and later

Tomcat Versions

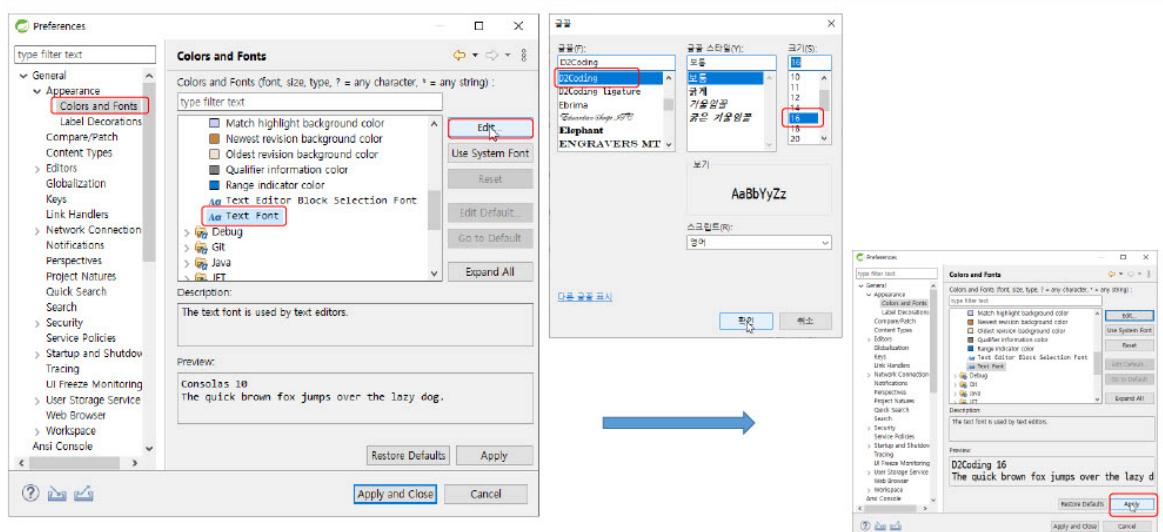


The screenshot shows the Apache Tomcat 9 Software Downloads page. The URL is <https://tomcat.apache.org/download-90.cgi>. The page features the Apache Tomcat logo and navigation links for search, quick navigation, release integrity, mirrors, and documentation. The main content area displays download links for various distributions, with a green callout bubble highlighting the '64-bit Windows zip (java_shas120)' link under the 'Binary Distributions' section.

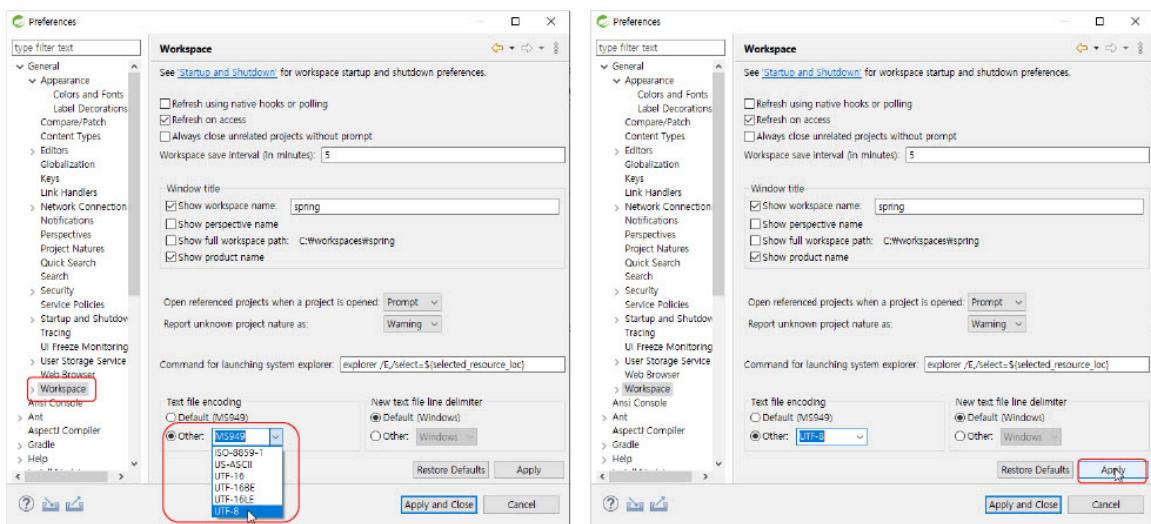
개발 환경 설정



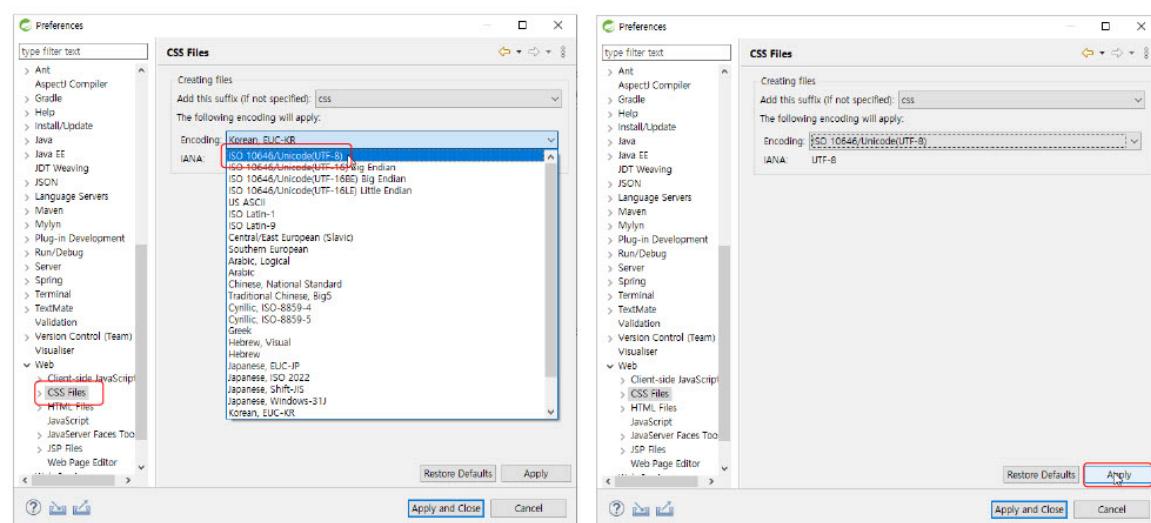
개발 환경 설정



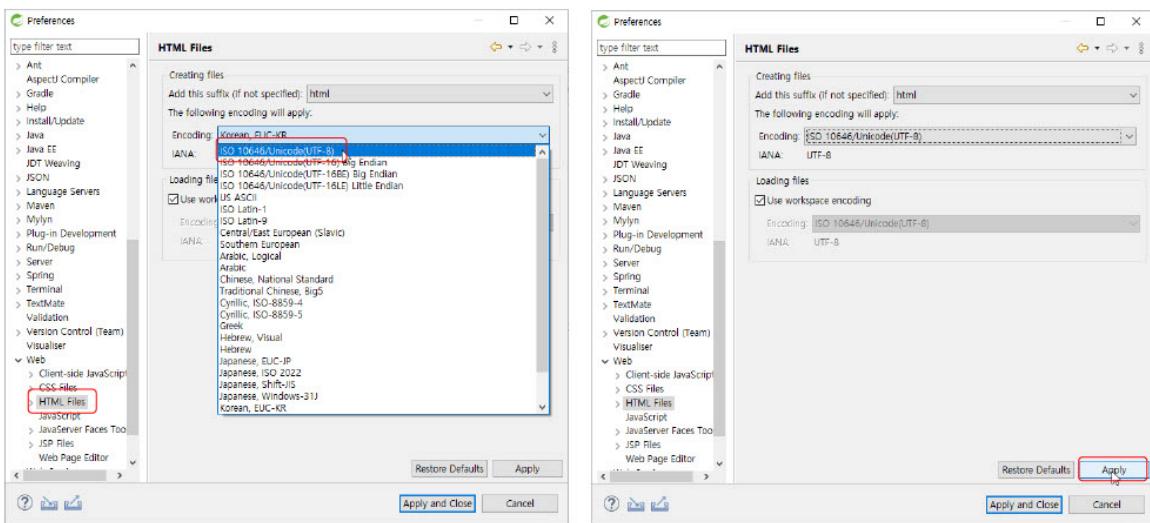
개발 환경 설정



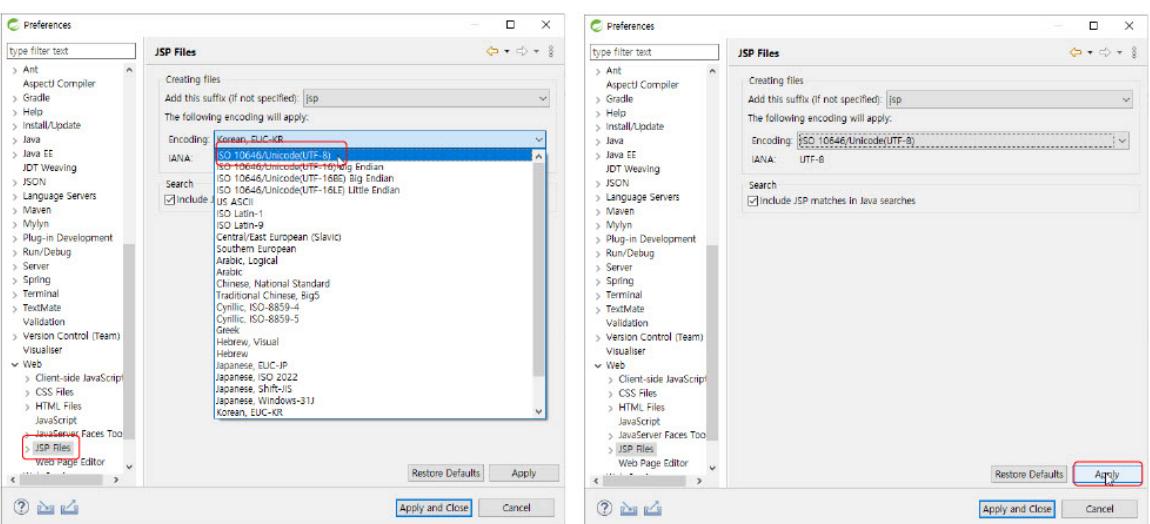
개발 환경 설정



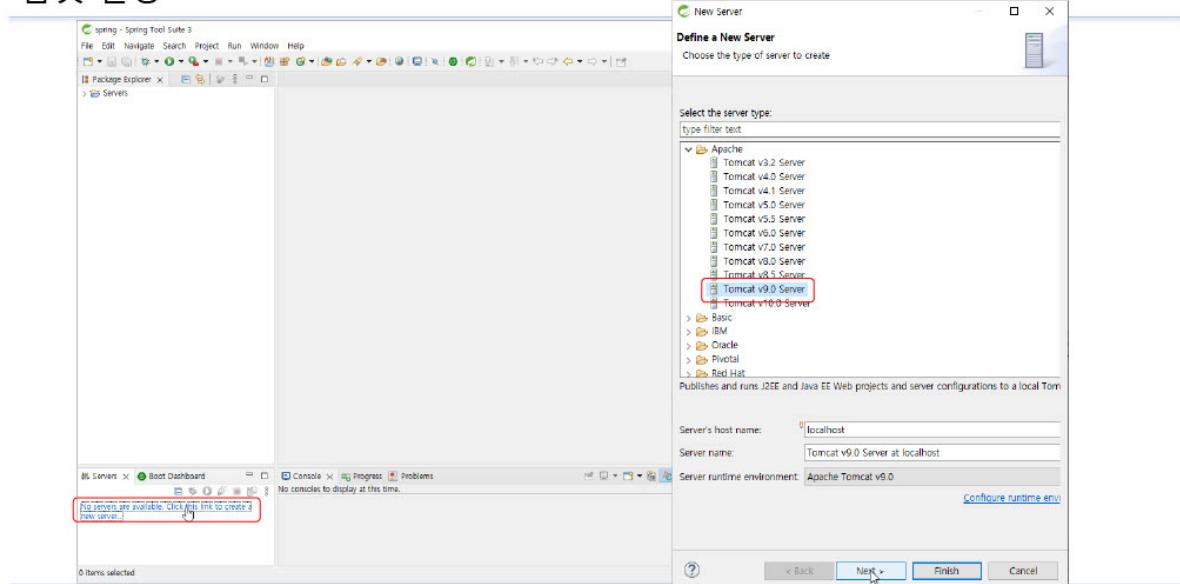
개발 환경 설정



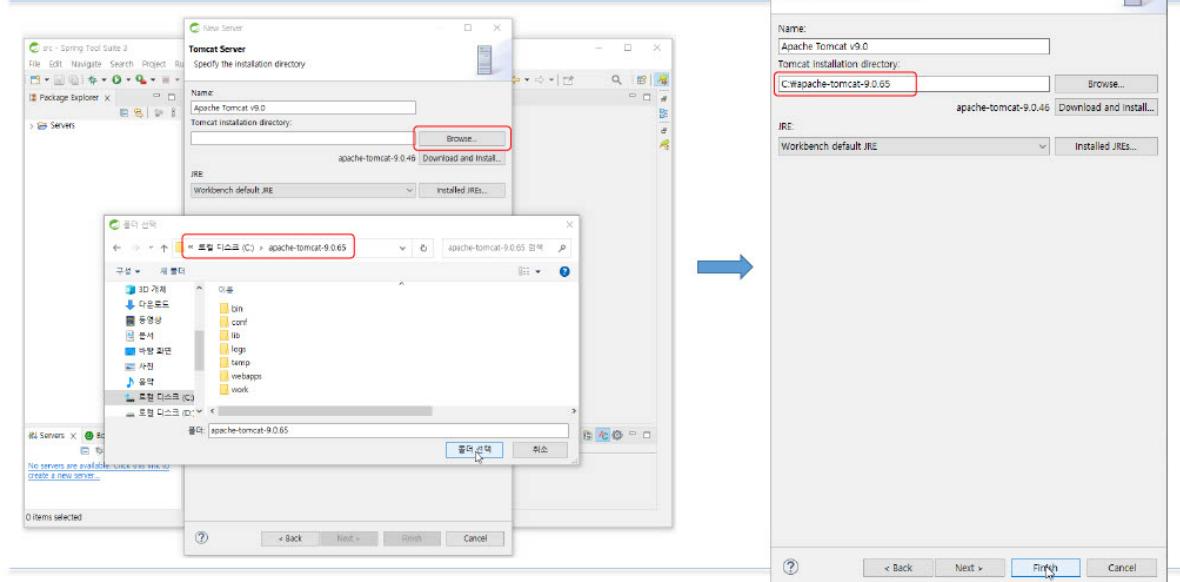
개발 환경 설정



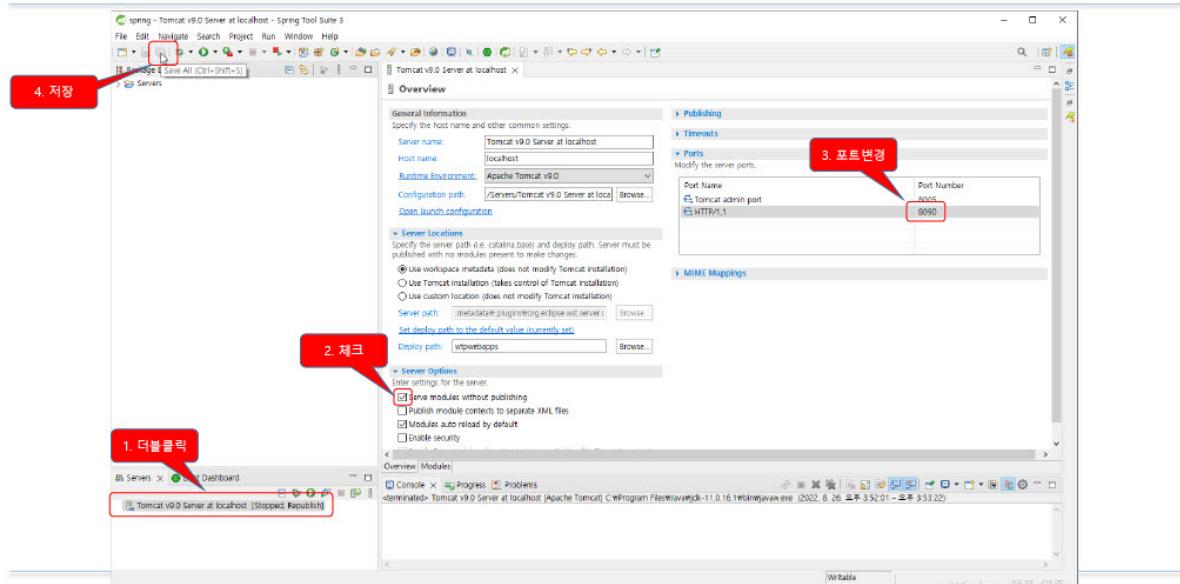
톰캣 설정



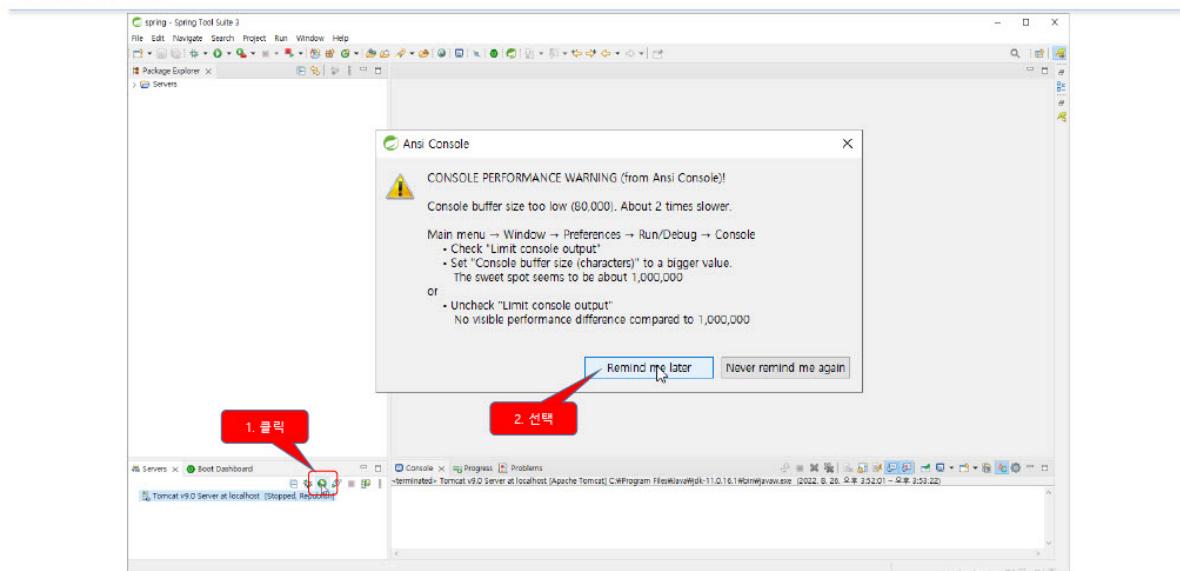
톰캣 설정



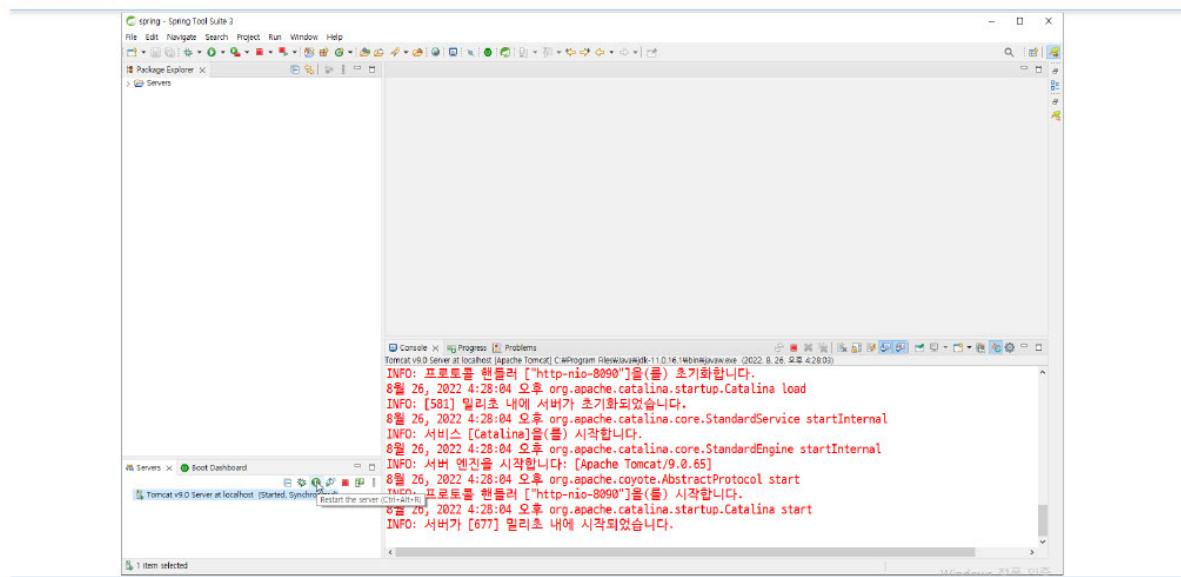
톰캣 설정



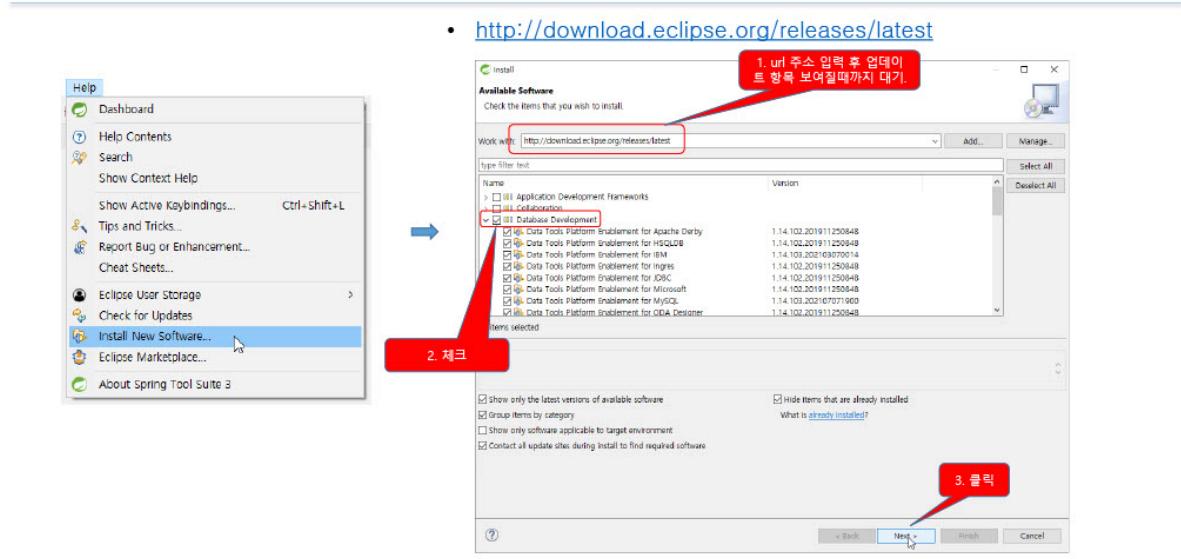
톰캣 설정



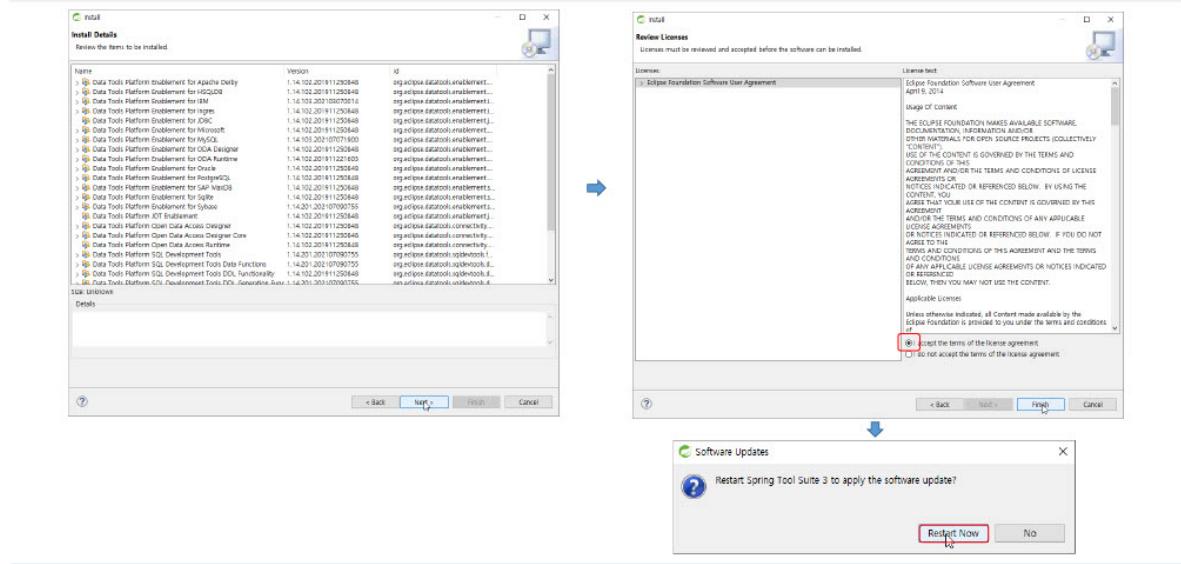
톰캣 설정



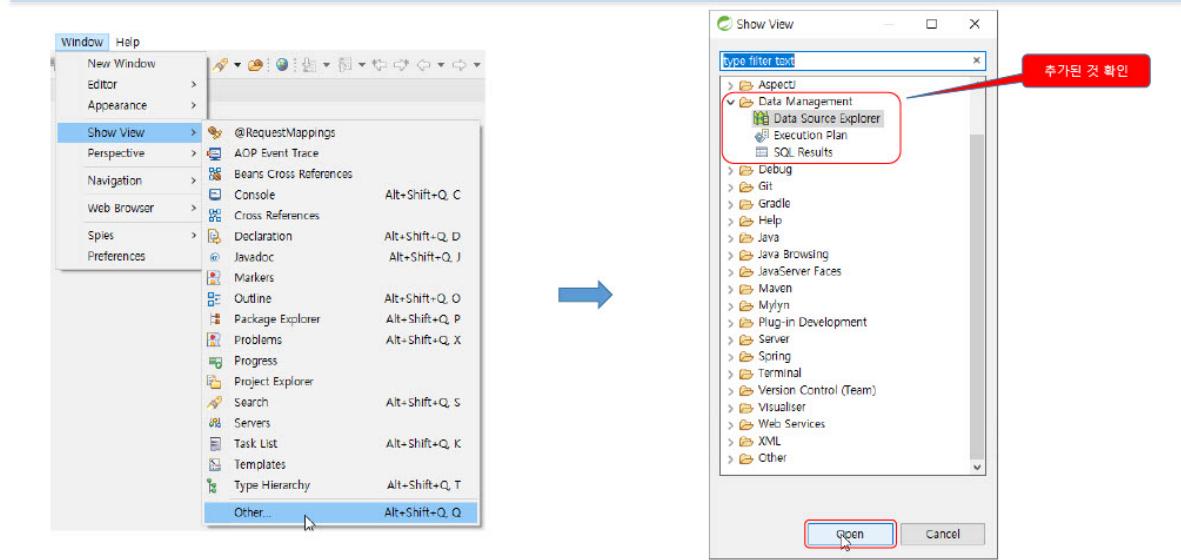
STS 항목 추가(Data Source Explorer)



STS 항목 추가(Data Source Explorer)



STS 항목 추가(Data Source Explorer)



STS 항목 추가(Data Source Explorer)

