

# Open Framework Spring



# 전체 학습 내용

- 스프링 개요 및 환경 설정
- 스프링 프로젝트 만들기
- DI(Dependency Injection)
- AOP
- Spring @MVC
- Spring MVC 게시판
- Security
- Mybatis

# 제1 장 스프링 개요 및 환경설정

스프링이란?

Maven 환경설정



# 1. Spring이란?

## □ 프레임워크

- ▣ 어떤일에 대한 틀 구조를 정의하여 그 틀에 맞추어 그 일을 쉽게 완정할 수 있도록 하는 것
- ▣ 특정 목적에 맞게 프로그램을 쉽게 하기 위한 약속
- ▣ 소프트웨어의 특정한 클래스에 대하여 재사용할 수 있는 설계로 구성된 관련된 클래스의 집합

## □ 프레임워크의 이점

- ▣ 모듈화, 재사용성, 확장성

## □ 프레임워크

- ▣ AnyFrame, DEVON 프레임워크, NEXCORE 프레임워크, KESA, eGovFrame(정부 표준 프레임워크: Spring) 등

# 1. Spring이란?

## 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

## Enterprise Application 개발의 복잡성을 줄여 주기 위한 목적

EJB 사용으로 수행되었  
던 모든 기능을 일반  
POJO 를 사용해서 가능  
하게 함

경량 컨테이너(light  
weight container)

## 주요 개념

의존성 주입(lightweight  
dependency injection)

관점 지향 컨테이너  
(aspect-oriented  
container)

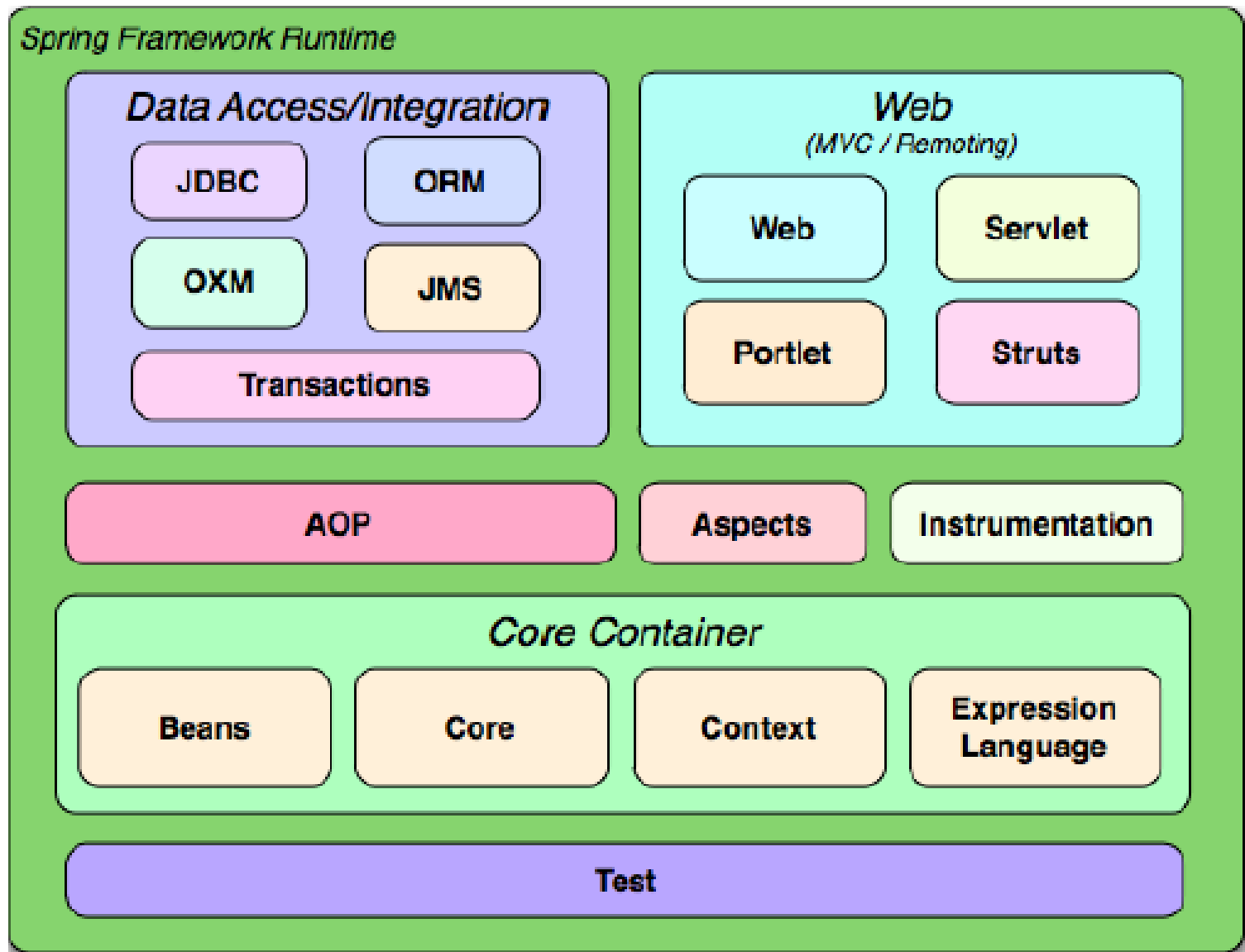
포터블 서비스 추상화  
(Portable Service  
Abstract)

# 1. Spring 이란?

## □ String Framework의 장점

- ▣ 개발자들이 애플리케이션 로직 개발에만 집중
  - 기술에 대한 접근 방식 일관성이 있고 특정 환경에 종속적이지 않음
  - 실행 로직의 기능이 변경되지 않고 서버 등의 실행 환경, 제공되는 조건이 변경될 때 코드 수정 필요 없음
- ▣ 개발이 단순해짐
  - Spring의 의존관계 트랜잭션들의 설정 방법에 대한 지식을 습득한 후에 설정 적용 기술만으로 Enterprise 개발의 기술적인 복잡함과 그에 따른 수고제거 가능
- ▣ POJO(Plain Old Java Object)방식의 기술 사용이 가능
  - 특정 규약 및 환경에 종속되지 않은 일반 자바 클래스를 지원, 컨테이너에 의존적인 코드를 추가하지 않아도 애플리케이션을 개발가능
  - 개발 후 테스트도 쉽고 빠르게 할 수 있음
- ▣ 다른 Framework와 연동해서 쓸 수 있는 확장성 용이

# Spring Framework rnwh



## 2. Maven을 알자

### □ 프로젝트 관리 도구

### □ 목표

- ▣ 빌드 절차를 간소화
- ▣ 통합된 빌드 시스템 제공
- ▣ Quality Project Information 제공
- ▣ 개발에 있어서 Best Practices를 위한 지침 제공
- ▣ 투명하게 새로운 기능으로 이전하도록 도와 준다

### □ 주요 기능

- ▣ 프로젝트 관리
- ▣ 모듈간 의존성을 관리하는 메커니즘 제공
- ▣ 프로젝트에서 사용하는 라이브러리를 관리하는 방법 제공
- ▣ 빌드의 결과를 관리하는 메커니즘
- ▣ 소스코드의 구조를 표준화



## 2. STS(Spring Tool Suite) 설치

### □ STS

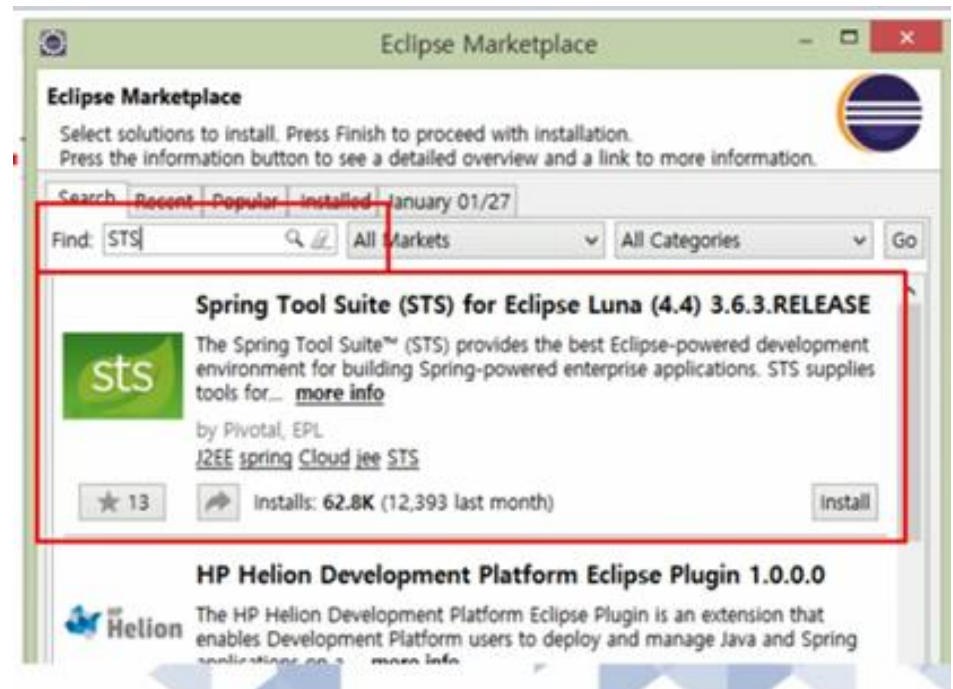
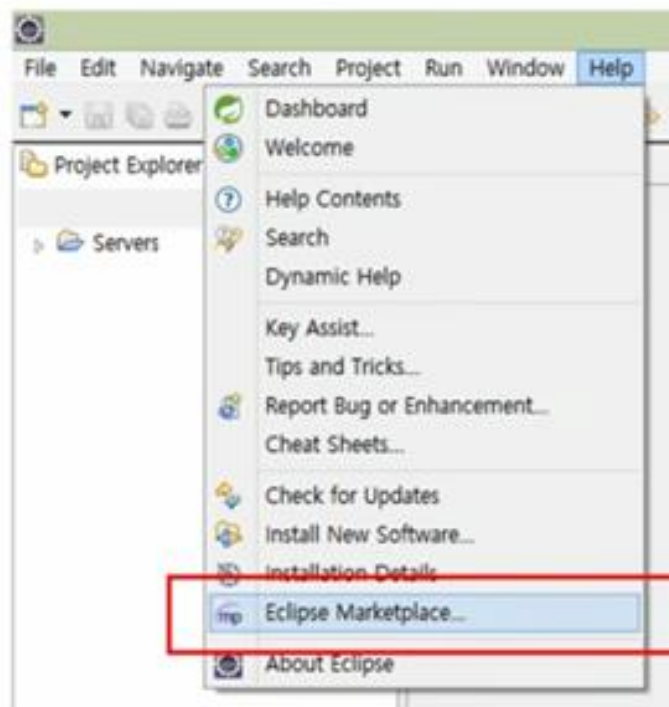
- 이클립스에서 스프링을 개발을 편리하게 하는 개발도구.
- 스프링에서는 STS를 다음과 같이 정의한다.(<http://spring.io/tools>)

Whether developing with Spring and/or Groovy/Grails we provide a customized all-in-one Eclipse based distribution that makes application development easy. The tool suites provide ready-to-use combinations of language support, framework support, and runtime support, and combine them with the existing Java, Web and Java EE tooling from Eclipse.

## 2. STS(Spring Tool Suite) 설치

- 1. 상단의 Help > Eclipse Marketplace를 선택
- 2. Marketplace에서 STS를 검색.

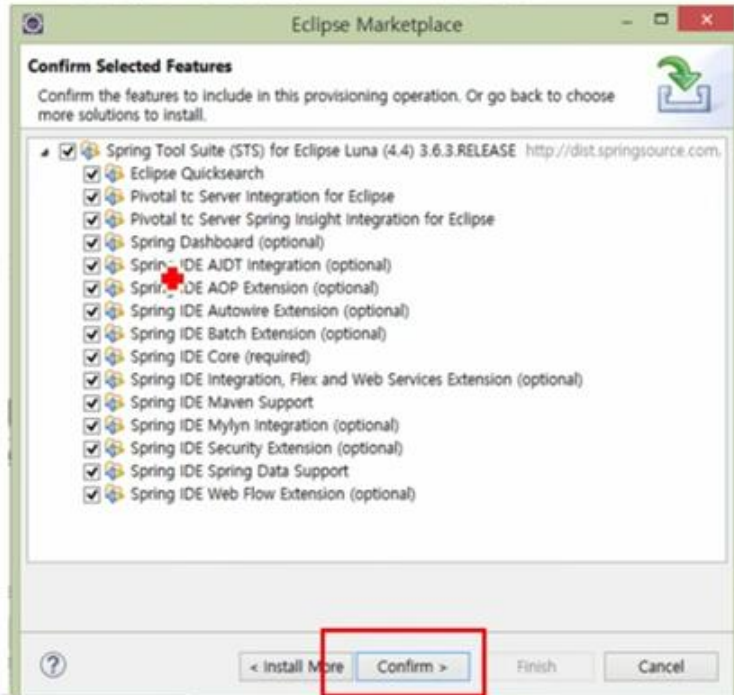
이클립스 실행 및 Marketplace... 진입



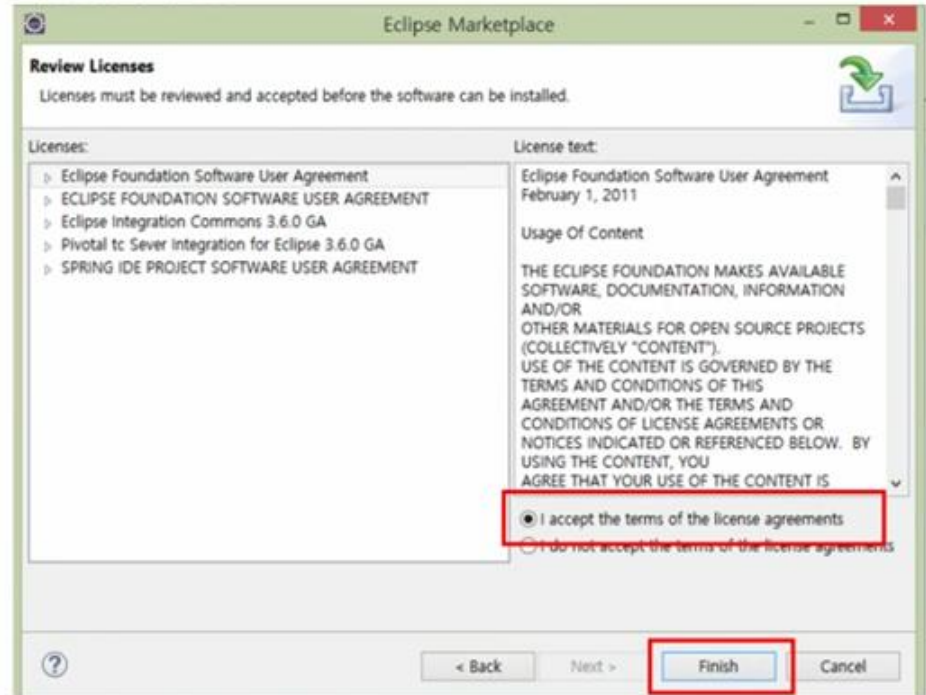
## 2. STS(Spring Tool Suite) 설치

### □ 이클립스의 버전에 맞는걸 설치.

프로그램 항목들 모두 선택 하고 Confirm 버튼 클릭

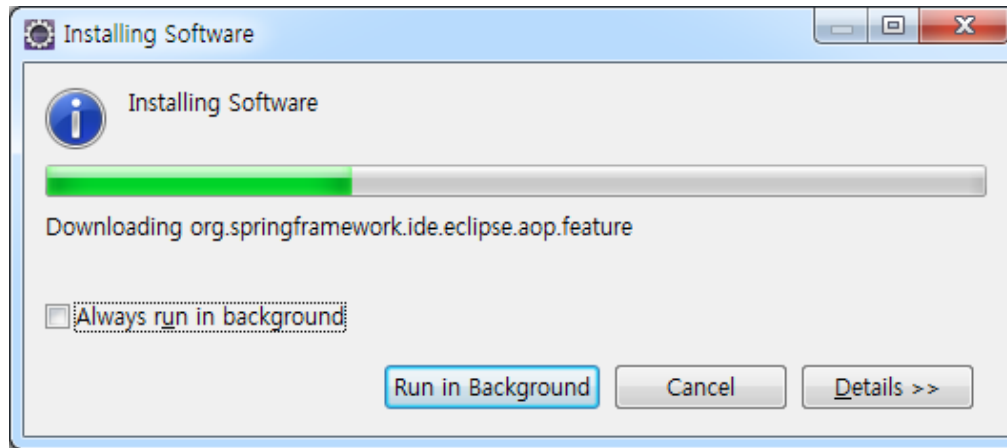


약관에 동의 후 진행

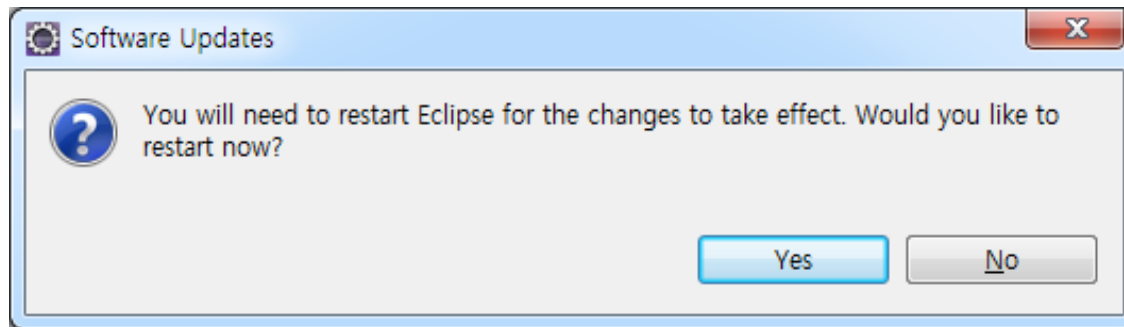


## 2. STS(Spring Tool Suite) 설치

- 3. 라이선스에 동의하고나면, 설치가 진행된다.

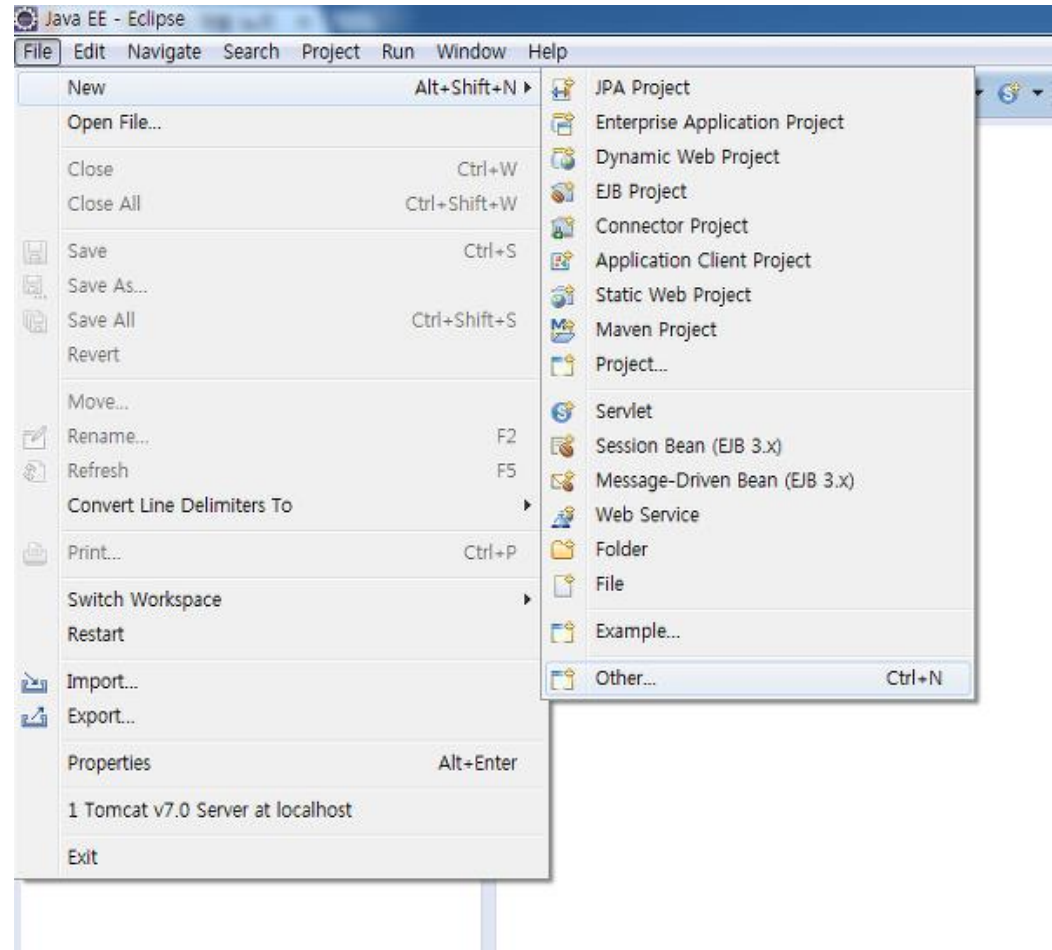


- 4. 설치가 완료되고 나면, 이클립스를 재시작 할거냐고 물어본다. Yes를 눌러주자.

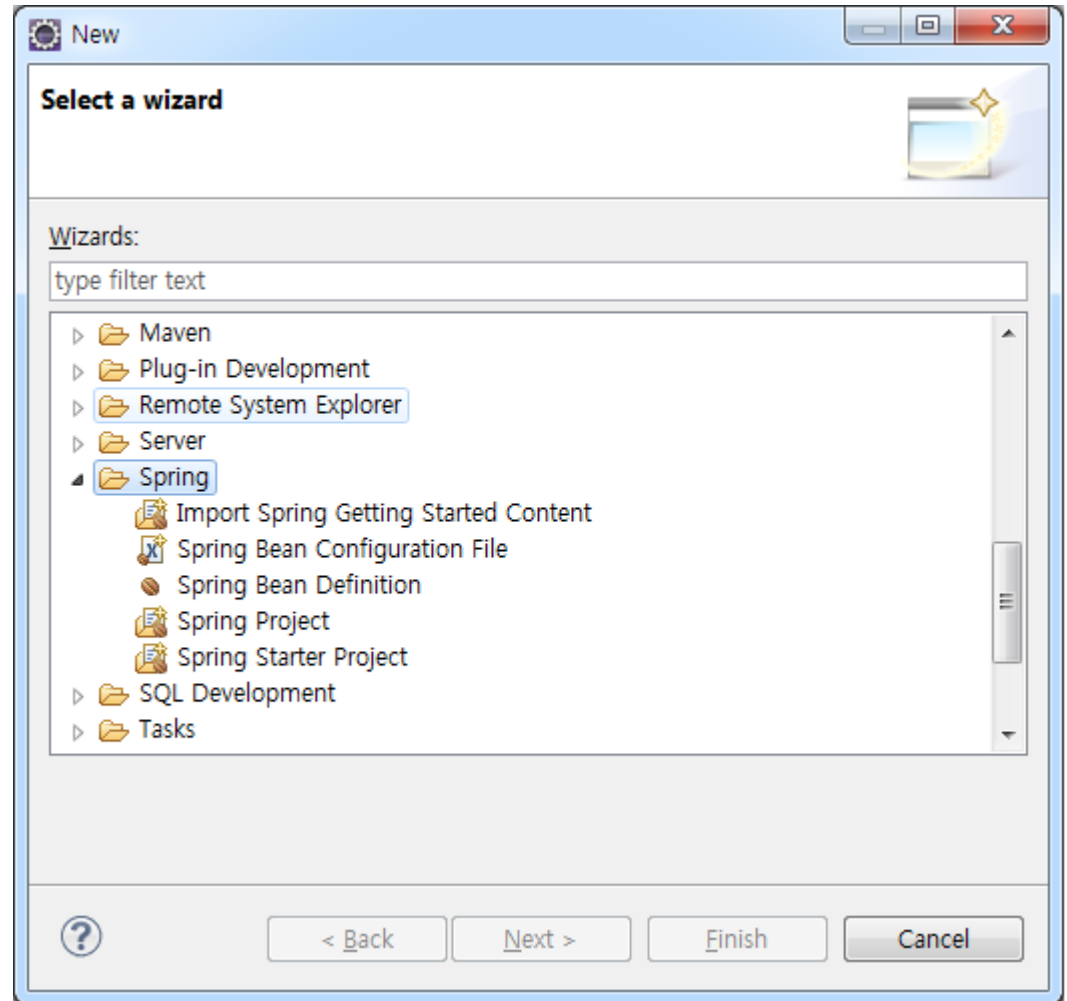


## 2. STS(Spring Tool Suite) 설치

- 5. STS가 잘 설치되었는지 확인.
- 1) File > New > Others를 선택.



- 2) 아래쪽에 Spring이 보이는지 확인한다



## 2. STS(Spring Tool Suite) 설치

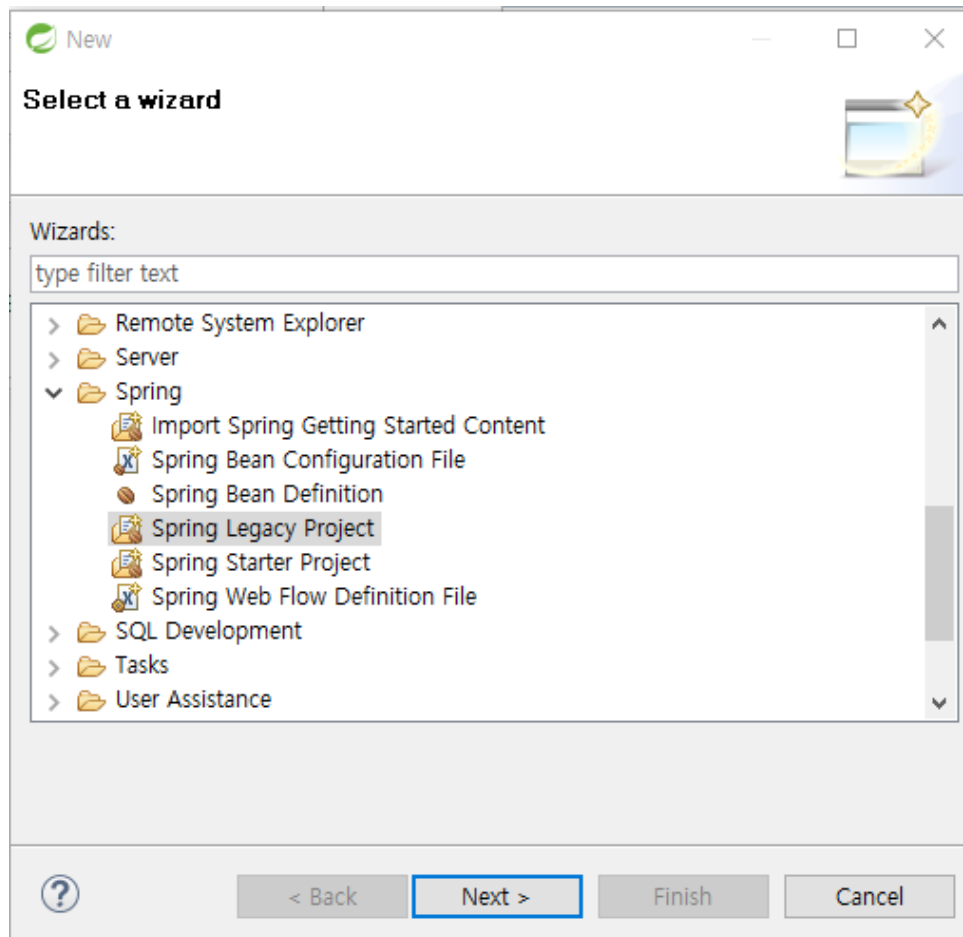
- STS 별도 설치

- ▣ <http://spring.io/tools>에서 다운로드

# 3. 스프링 프로젝트 생성

## □ step1

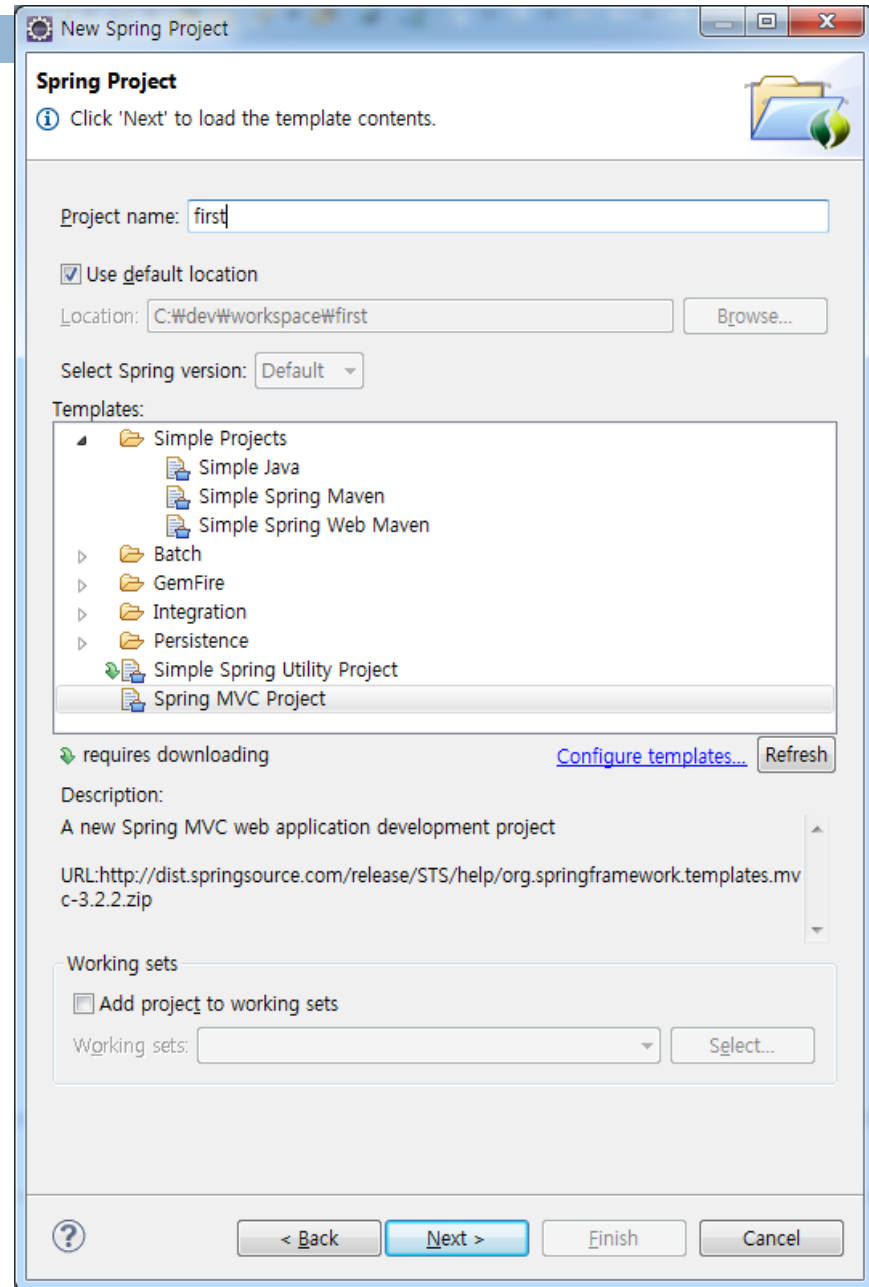
- File > New > Other 를 선택, Spring > Spring Legacy Project를 선택 -> [next]버튼 클릭





## □ step2

- 프로젝트의 이름을 first로 입력, Spring MVC Project를 선택, [next]버튼 클릭



## □ Step3

- package를 입력.
- package는 최소 3레벨 이상 ( [1레벨].[2레벨].[3레벨] )로 구성.
- 이는 자바 코딩 규칙 (Coding Convention을 찾아 보면 확인할 수 있다.)
- 여기서는, "com.pgm.first"라는 package를 사용

New Spring Legacy Project

Project Settings - Spring MVC Project

Define project specific settings. Required settings are denoted by "\*".

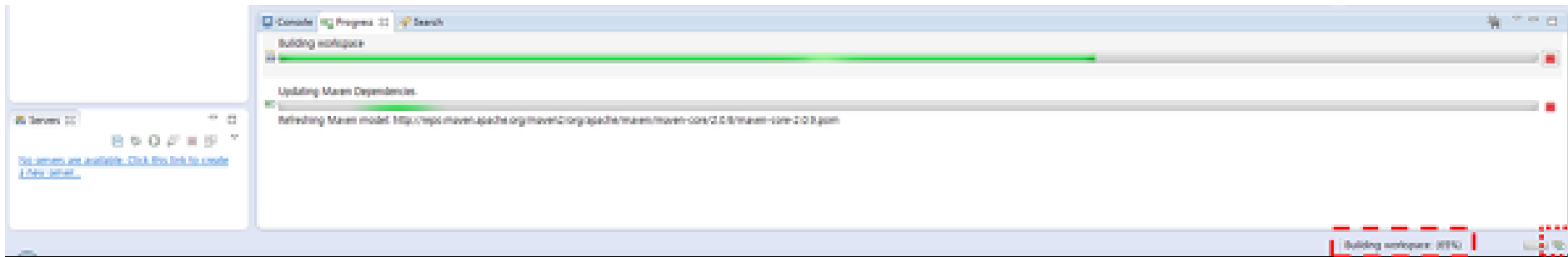
Please specify the top-level package e.g. com.mycompany.myapp\*

com.pgm.first

? < Back Next > Finish Cancel

## □ step4

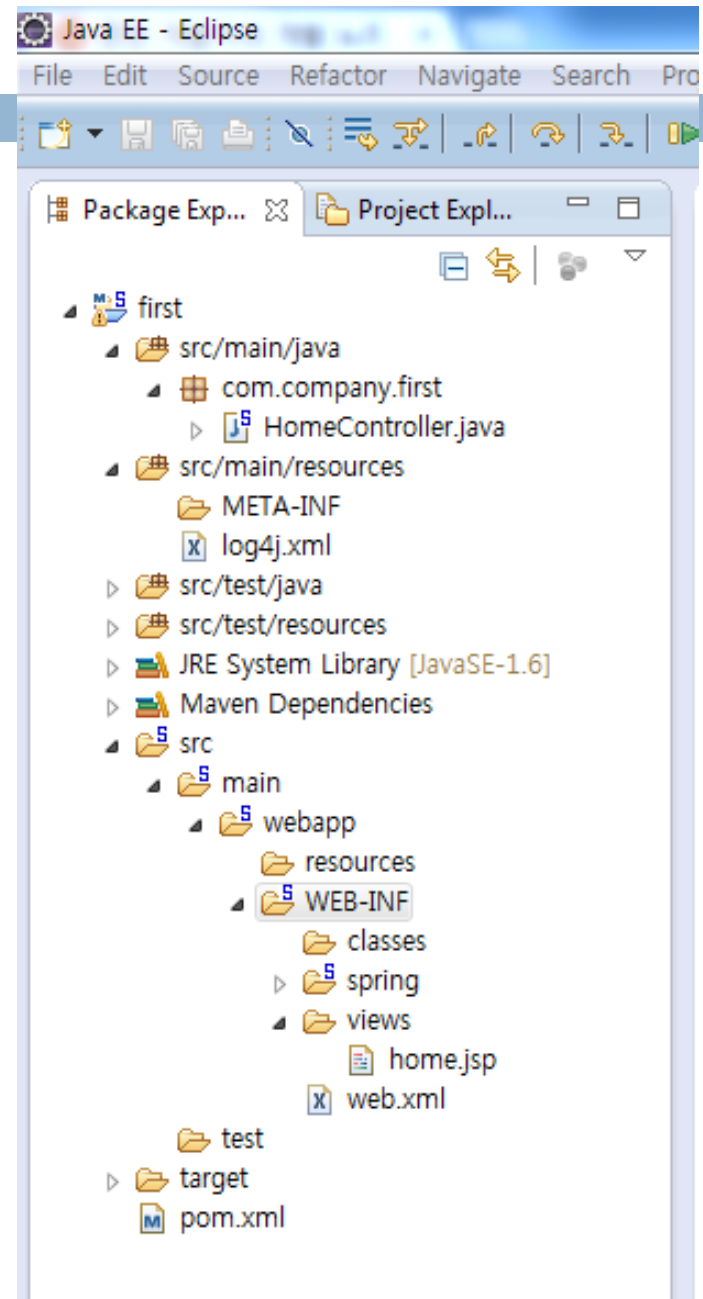
- Finish를 누르면 프로젝트가 생성이 되고, 인터넷에서 스프링 프로젝트에 필요한 라이브러리를 자동으로 다운받기 시작



- 생성한 Spring MVC Project에는 여러가지 라이브러리들이 필요한데, 프로젝트의 생성과 동시에 메이븐이 자동적으로 인터넷에서 필요한 라이브러리를 다운받는 과정이다.

## □ Step5

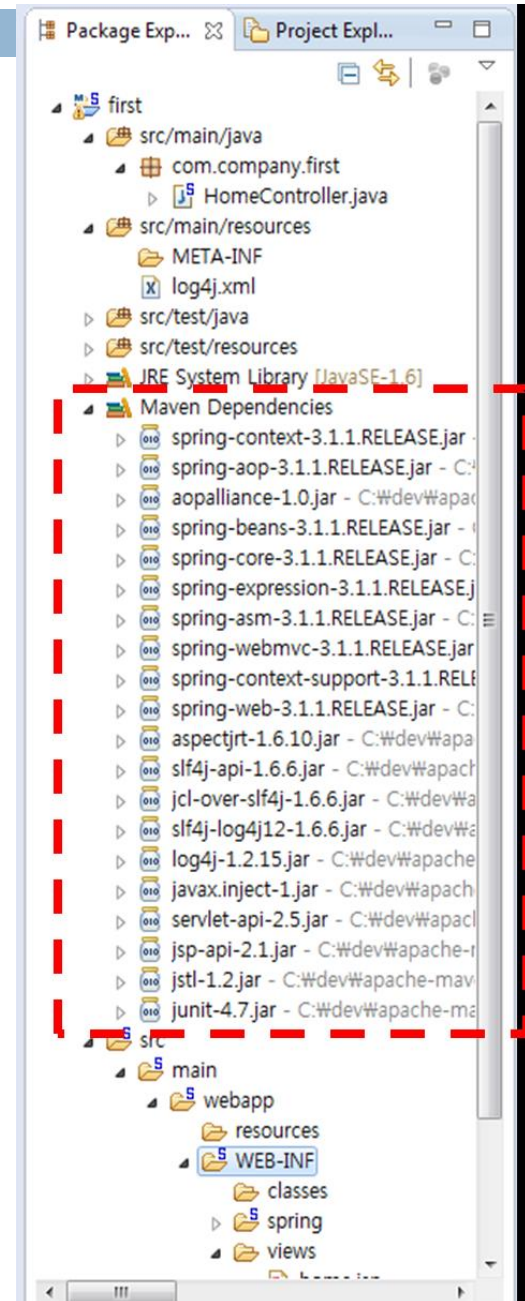
- 다운로드가 완료된 것을 확인하고, 생성된 프로젝트에러 확인.



# 전체적인 구조와 각 폴더의 역할을 설명

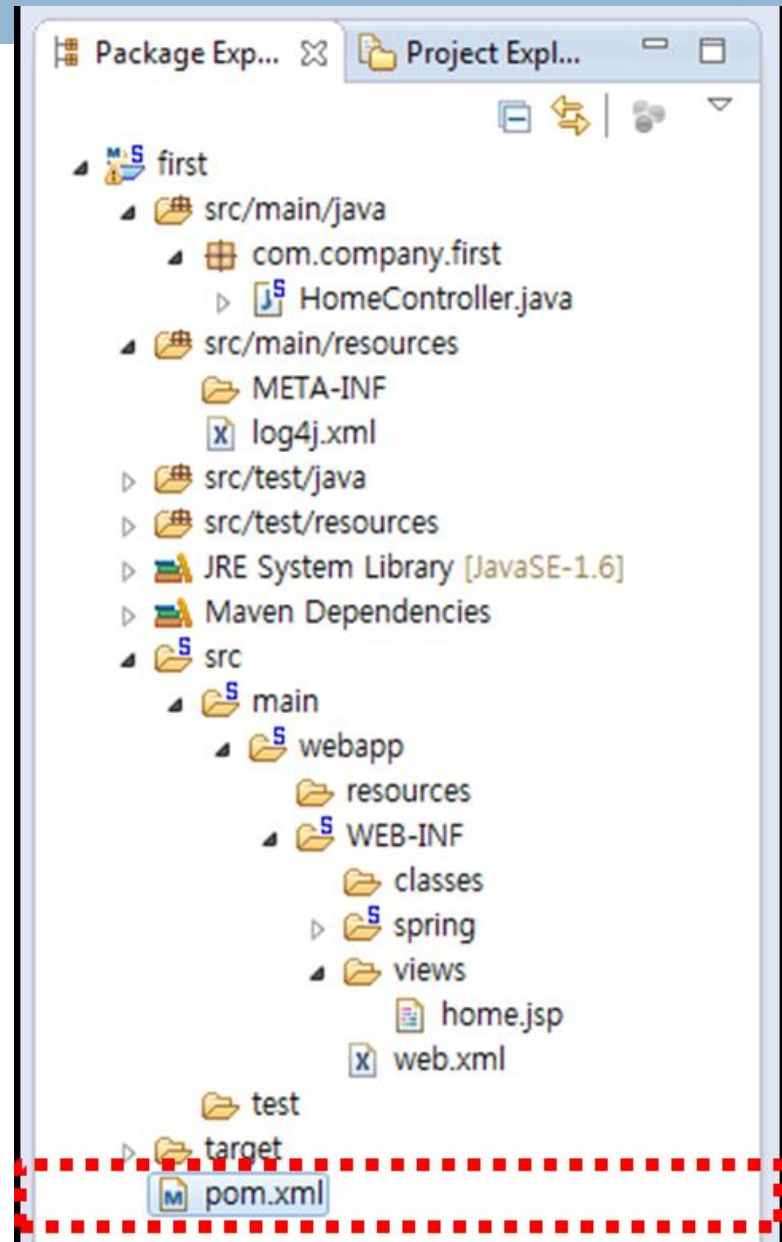
## □ Maven Dependencies

- Maven이 자동적으로 필요한 라이브러리를
- 받아온 라이브러리들은 Maven Dependencies라는 곳에서 확인.

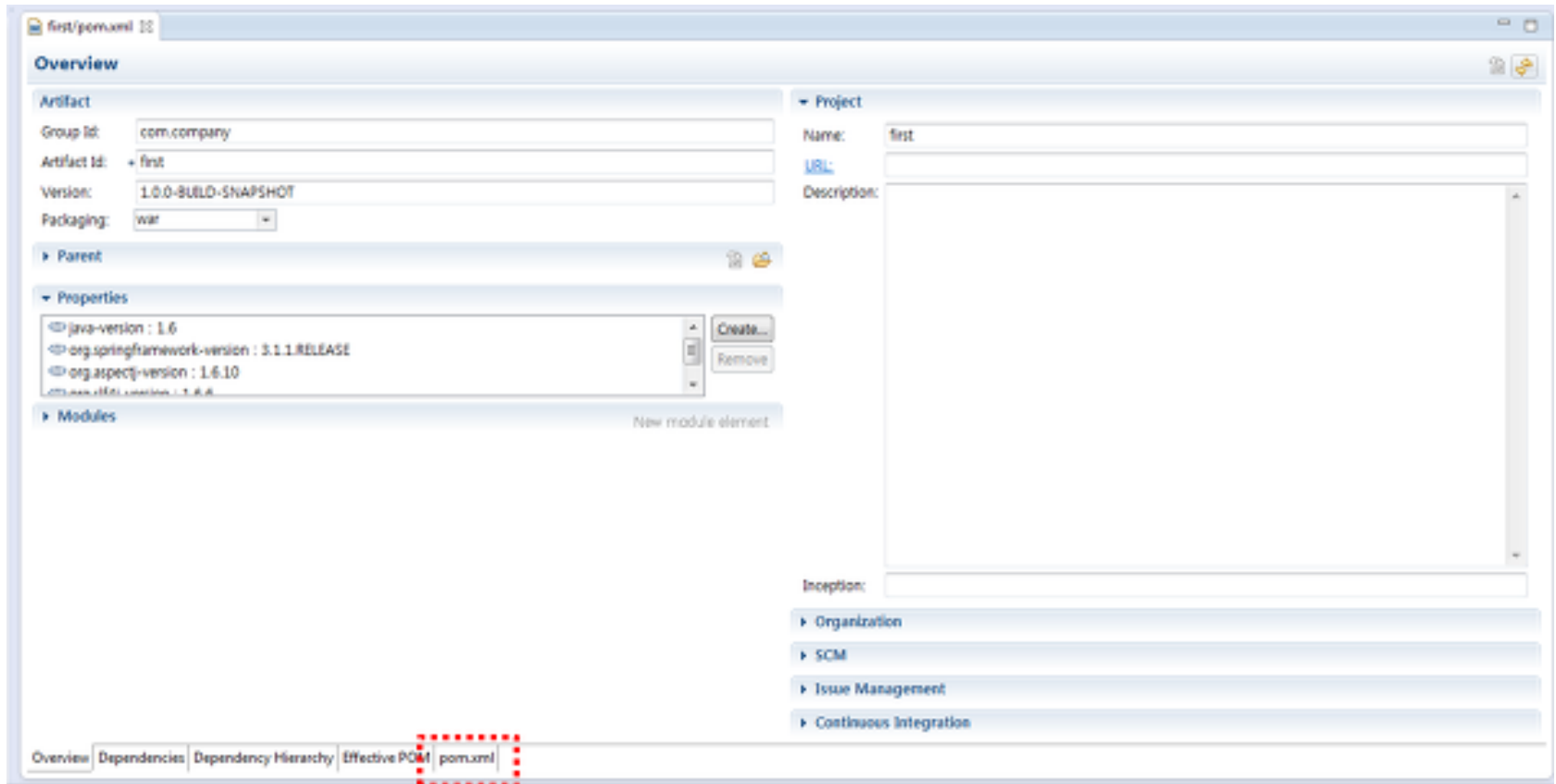


## □ POM.xml


- ▣ pom.xml 파일을 더블클릭.



- 다음과 같은 화면이 보이고, 중간쯤에 있는 pom.xml 탭을 선택.



## □ pom.xml을 확인.



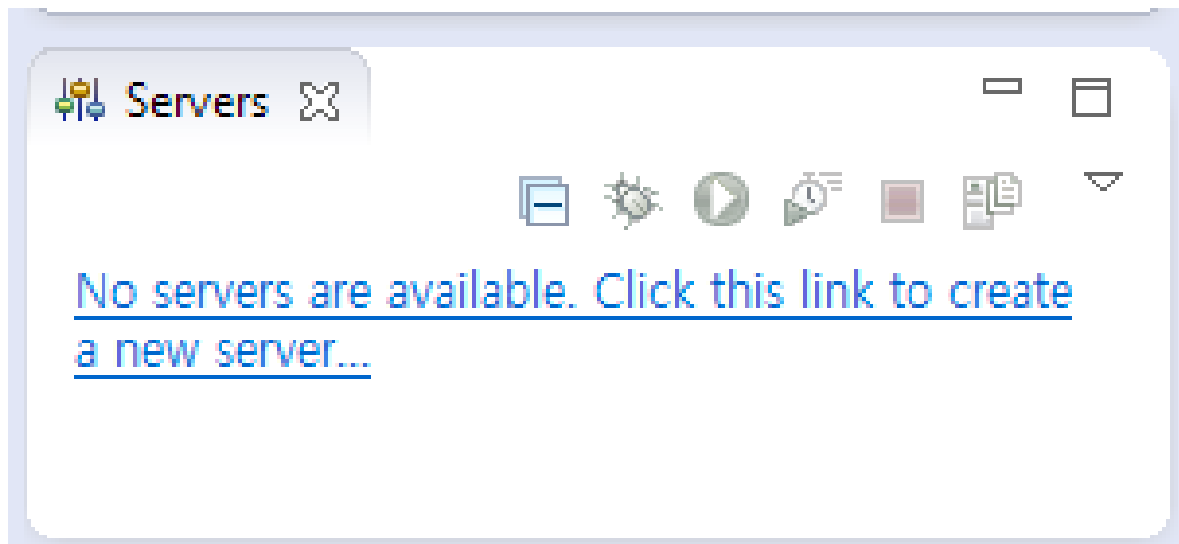
```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.company</groupId>
  <artifactId>first</artifactId>
  <name>first</name>
  <packaging>war</packaging>
  <version>1.0.0-BUILD-SNAPSHOT</version>
  <properties>
    <java-version>1.6</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>
```

Overview | Dependencies | Dependency Hierarchy | Effective POM | pom.xml



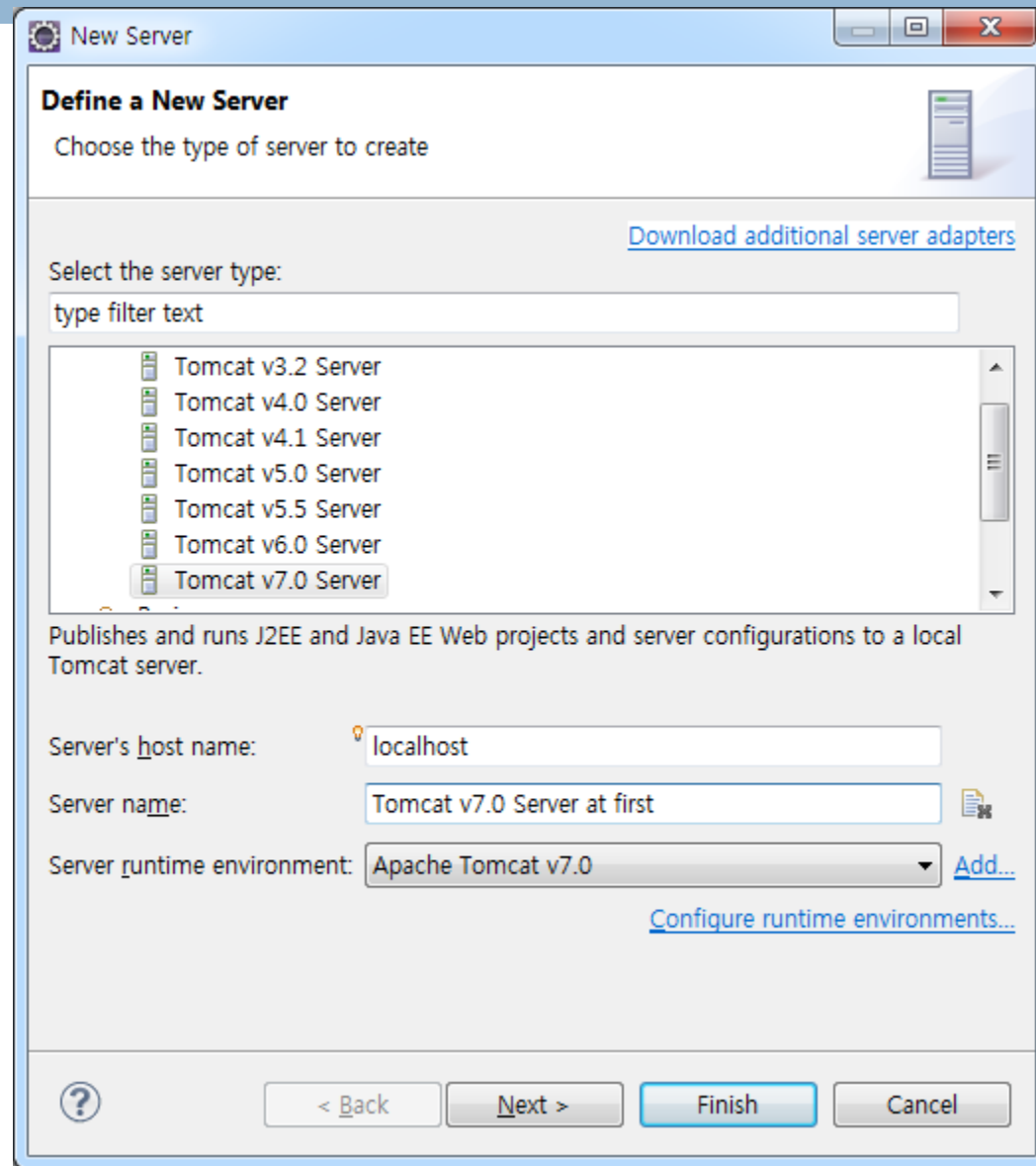
## □ first 프로젝트 실행

- ▣ Servers view에서 새로운 서버를 만든다.
- ▣ 앞에서 톰캣이 제대로 설치되었는지 확인하기 위해서 만든 서버가 남아있다면 지우자.
- ▣ Servers 뷰에서 마우스 우클릭 > New > Server를 선택해도 된다.

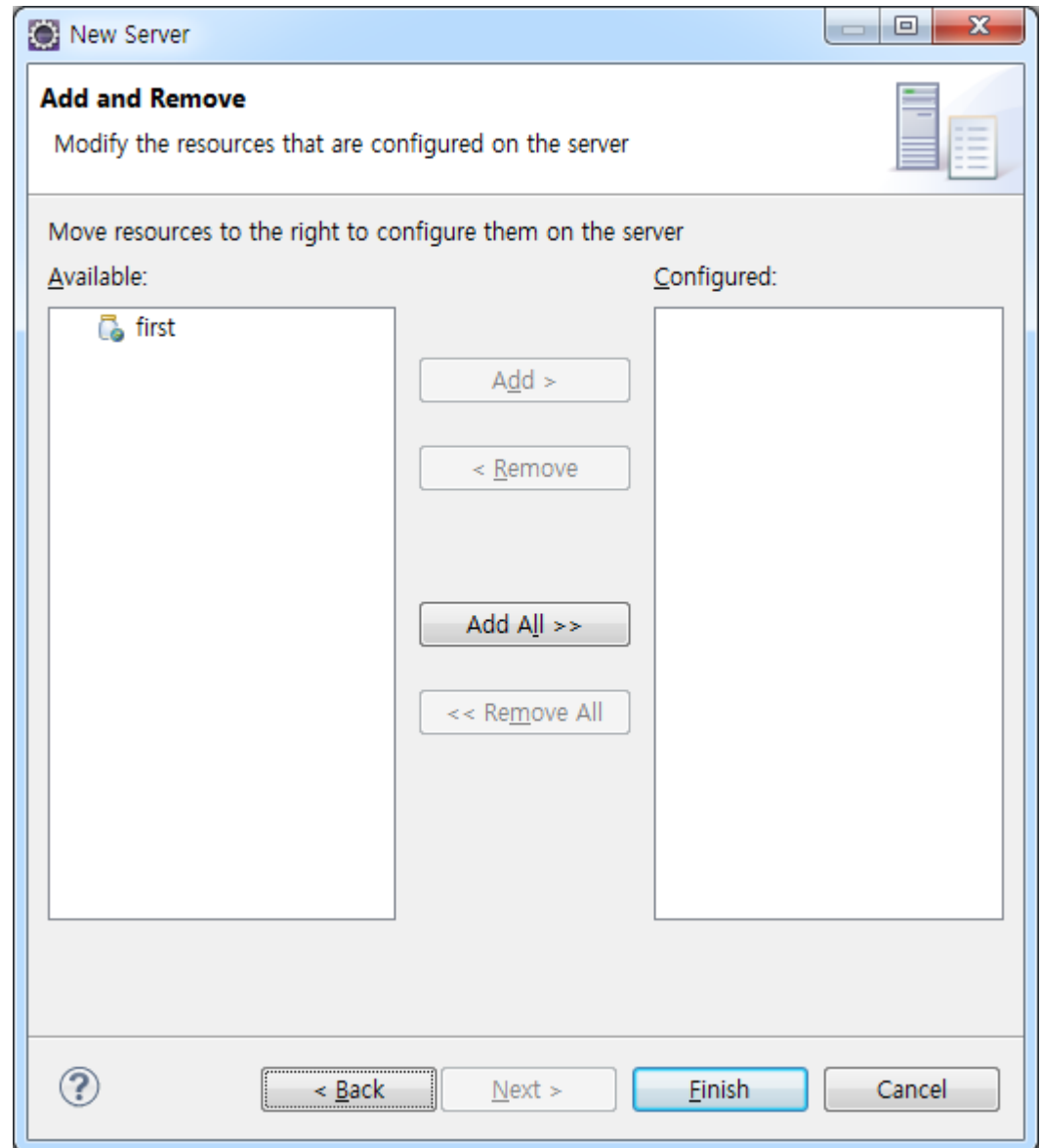


## 서버생성

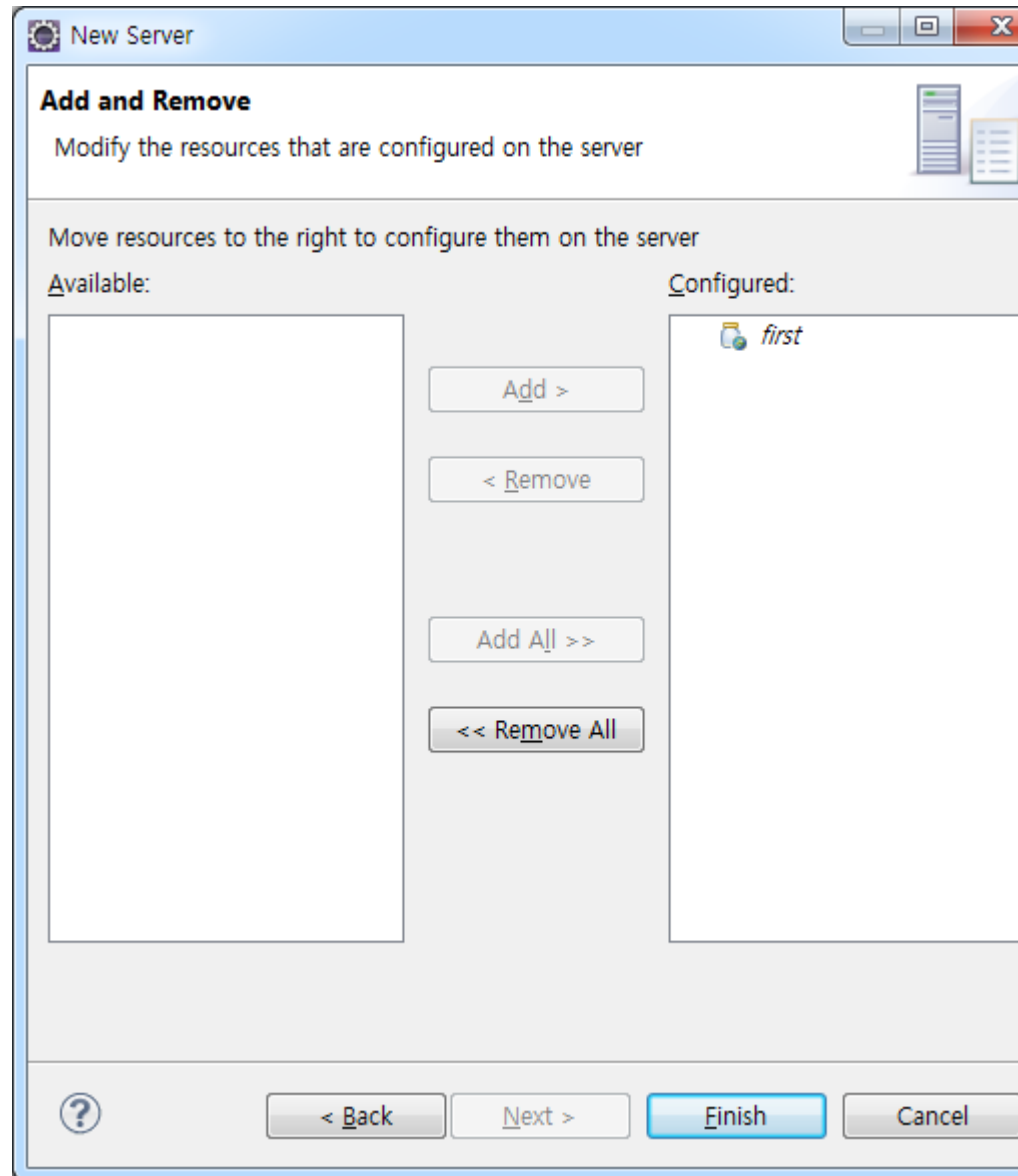
- Tomcat 8.0을 선택하고, 서버 이름은 first로 바꾼다.
- 여러 개의 프로젝트가 있을 경우, 하나의 프로젝트에는 하나의 서버만 할당.
- 여러개의 서버를 생성해야 할 경우, 같은 이름으로는 서버를 여러 개 생성할 수도 없고
- 어떤 서버가 어떤 프로젝트인지를 알 수 없기 때문에, 본인은 서버의 이름과 프로젝트의 이름을 동일하게 생성.



□ Next > 를 누르자.



- first를 더블클릭하거나 Add > 버튼을 눌러서 Configured 쪽으로 옮기고 Finish를 누른다.



#### □ 4. 서버가 생성되었을테니, 서버를 실행시키자.

- 1) 서버 선택후 Ctrl + Alt + R을 누르거나,
- 2) 서버 우클릭 > Start를 선택하거나,
- 3) 녹색 동그라미안에 ▷가 있는 아이콘을 눌러서 서버를 실행시키  
자.

```
Tomcat v7.0 Server at localhost Tomcat C:\Windows\System32\cmd.exe [2016-7-8 下午 2:09:11]
T@06, 2016-7-08-11 10 org.apache.catalina.core.AprLifecycleListener init
INFO : The APR based Apache Tomcat Native library which allows optimal performance in production environments was not found on the java.library.path: C:/Users/TOMIKU/Desktop/Catland/Servers/Jakarta/Catlands/system32/C/windows
T@06, 2016-7-08-11 10 org.apache.tomcat.util.digester.SetPropertyRule.begin
INFO : ([SetPropertyRule]{[Server/SetEngine/onContext]} Setting property "source" to "org.eclipse.jst.jee.server.first" did not find a matching property.
T@06, 2016-7-08-11 10 org.apache.coyote.AbstractProtocol init
WARN : Initializing ProtocolHandler ["http-tcp-nio"]
T@06, 2016-7-08-11 10 org.apache.coyote.AbstractProtocol init
WARN : Initializing ProtocolHandler ["ajp-shm-nio"]
T@06, 2016-7-08-11 10 org.apache.catalina.startup.Catalina load
WARN : Initialization process failed as follows
T@06, 2016-7-08-11 10 org.apache.catalina.core.StandardService startInternal
WARN : Starting service Catalina
T@06, 2016-7-08-11 10 org.apache.catalina.core.StandardEngine startInternal
WARN : Starting Servlet Engine: apache/tomcat/7.0.54
T@06, 2016-7-08-11 10 org.apache.catalina.core.ApplicationContext log
WARN : No Spring webapplicationInitializer types detected on classpath
T@06, 2016-7-08-11 10 org.springframework.web.servlet.mvc.annotation.AnnotationMethodMappingImpl.
INFO : Initializing Spring root WebApplicationContext
INFO : org.springframework.context.support.GenericWebApplicationContext: startup date [Sun Jul 06 14:09:11 EDT 2014]; root of context hierarchy
INFO : org.springframework.beans.factory.xml.XmlBeanDefinitionReader - Loading XML bean definitions from ServletContext resource [/WEB-INF/classes/org.springframework.root-context.xml]
INFO : org.springframework.beans.factory.support.DefaultListableBeanFactory - Pre-instantiating singletons in org.springframework.beans.factory.support.DefaultListableBeanFactory@bb3c2d7: defining beans [], root of factory &
INFO : org.springframework.web.servlet.mvc.annotation.AnnotationMethodMappingImpl.
INFO : Initializing Spring FrameworkServlet 'appServlet'
INFO : org.springframework.web.servlet.DispatcherServlet - FrameworkServlet 'appServlet': initialization started
INFO : org.springframework.context.support.GenericWebApplicationContext - Refreshing WebApplicationContext for namespace 'appServlet-servlet'; startup date [Sun Jul 06 14:09:11 EDT 2014]; parent: Root WebApplicationContext
INFO : org.springframework.beans.factory.xml.XmlBeanDefinitionReader - Loading XML bean definitions from ServletContext resource [/WEB-INF/classes/org.springframework.appServlet-servlet-context.xml]
INFO : org.springframework.context.annotation.ClassPathScanningCandidateComponentProvider - 354-359 [java.lang.annotation.AnnotationManager] found and supported for component scanning
INFO : org.springframework.context.annotation.ClassPathScanningCandidateComponentProvider - 354-359 [java.lang.reflect.Method] annotation found and supported for component scanning
INFO : org.springframework.beans.factory.xml.XmlBeanDefinitionReader - Loading XML bean definitions from ServletContext resource [/WEB-INF/classes/org.springframework.root-context.xml]
INFO : org.springframework.beans.factory.support.DefaultListableBeanFactory - Pre-instantiating singletons in org.springframework.beans.factory.support.DefaultListableBeanFactory@bb3c2d7: defining beans [org.springframework....]
INFO : org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerMapping - Mapping [{"url":"/api/{param:[\"headers\",\"cookies\"]}","consumes":["text/html"],"customizer["]] onto public java.lang.String com.company.FirstNo...
INFO : org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerMapping - Mapping [{"url":"/api/{param:[\"headers\",\"cookies\"]}","consumes":["text/html"],"customizer["]] onto handler [org.springframework.web.servlet.resource.ResourceController$HeaderW...
T@06, 2016-7-08-11 10 org.apache.coyote.AbstractProtocol start
WARN : Starting protocol handler ["http-tcp-nio"]
T@06, 2016-7-08-11 10 org.apache.coyote.AbstractProtocol start
WARN : Starting protocol handler ["ajp-shm-nio"]
T@06, 2016-7-08-11 10 org.apache.catalina.startup.Catalina start
WARN : Server startup in 2149 ms
```

- 브라우저 실행시키고, 주소창에 **http://localhost:8080/first/** 를 입력.

