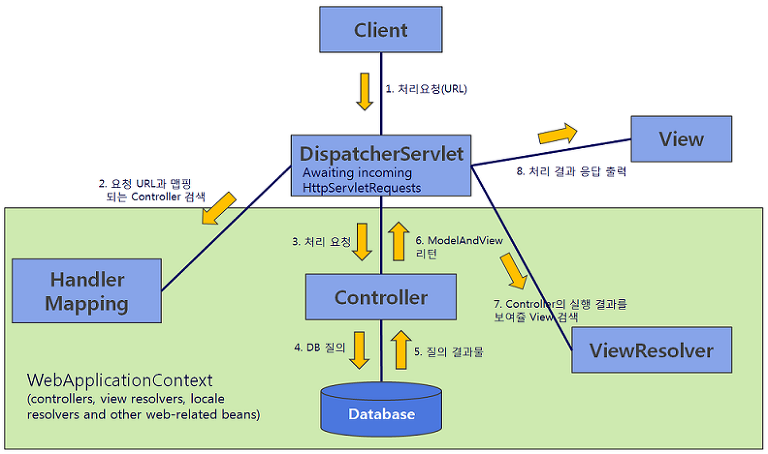
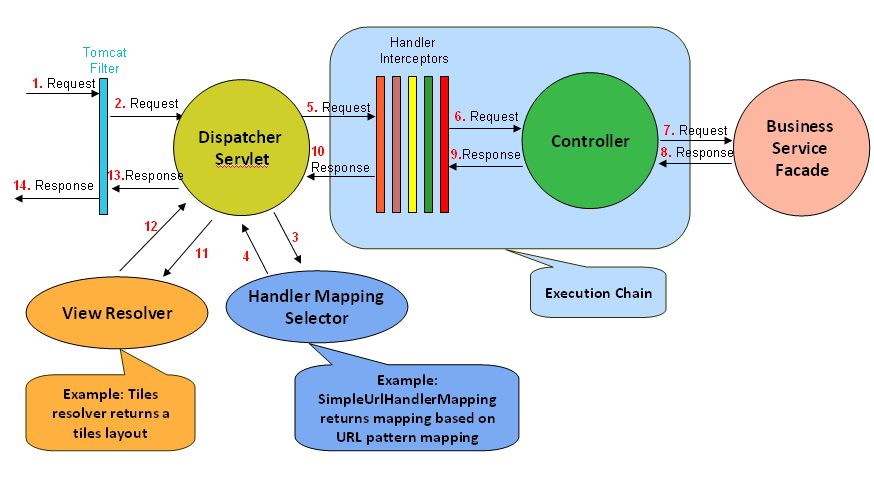
[스프링 프로젝트 종류]

|  |  |  |
| --- | --- | --- |
| **영어명** | **한국어명** | **설명** |
| Spring Boot | [스프링 부트](https://zetawiki.com/wiki/%EC%8A%A4%ED%94%84%EB%A7%81_%EB%B6%80%ED%8A%B8) | 초기 구성 및 기동 간소화 |
| Spring MVC | [스프링 MVC](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_MVC&action=edit&redlink=1) | 웹애플리케이션 MVC |
| Spring Integration | [스프링 연동](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%97%B0%EB%8F%99&action=edit&redlink=1) | 시스템 연동 |
| Spring Batch | [스프링 배치](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EB%B0%B0%EC%B9%98&action=edit&redlink=1) | 배치 처리 |
| Spring Security | [스프링 시큐리티](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8B%9C%ED%81%90%EB%A6%AC%ED%8B%B0&action=edit&redlink=1) | 인증 |
| Spring Data | [스프링 데이터](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EB%8D%B0%EC%9D%B4%ED%84%B0&action=edit&redlink=1) | 데이터 접근 추상화 |
|  |  |  |

[스프링 MVC구조]





**\* 스프링 MVC의 주요 구성 요소**1) DispatcherServlet : 클라이언트의 요청을 전달 받는 역할. 컨트롤러(Controller)에게 클라이언트의 요청을 전달하고, 컨트롤러가 리턴한 결과 값을 뷰(View)에 전달하여 알맞은 응답을 생성하도록 함. (스프링 제공)  
2) HandlerMapping : 클라이언트의 요청 URL을 어떤 컨트롤러가 처리할지를 결정. (스프링 제공)  
3) Controller : 클라이언트의 요청을 처리한 뒤, 그 결과를 DispatcherServle에 알려준다. (실제 로직을 담당)  
4) SQL 쿼리를 통한 데이터 질의문 전달.  
5) 데이터베이스가 반환(Return)한 데이터 획득.  
6) ModelAndView : 컨트롤러가 처리한 결과 정보 및 뷰 선택에 필요한 정보를 담음.  
7) ViewResolver : 컨트롤러의 처리 결과를 생성할 뷰를 결정. (스프링 제공)  
8) View : 컨트롤러의 처리 결과 화면을 생성 (Freemarker 등)

**\*  스프링 MVC 구조 및 처리 순서**1) 클라이언트의 요청이 DispatcherServlet에 전달됨.  
2) DispatcherServlet은 HandleMapping을 사용하여 클라이언트의 요청을 처리할 컨트롤러(Controller)를 검색.  
3) 매치되는 컨트롤러가 있다면 요청을 해당 컨트롤러에게 전달.  
    DispatcherServlet은 컨트롤러 객체의 handleRequest() 메서드를 호출하여 클라이언트 요청을 처리 .  
4) DB 관련 처리 (DB 질의 요청)

5) DB 관련 처리 (DB 질의 결과)  
6) 컨트롤러의 handlerRequest()메서드는 처리 결과 정보를 담은 ModelAndView 객체를 리턴.  
7) DispatcherServlet은 ViewResolver로부터 응답 결과를 생성할 뷰 객체를 구함.  
    JSP, Velocity, Freemarker 등 다양한 뷰를 지원.  
    - AbstractCachingViewResolver

    - XmlViewResolver

    - ResourceBundleViewResolver

    - UrlBasedViewResolver

    - InternalResourceViewResolver

    - FreeMakerViewResolver

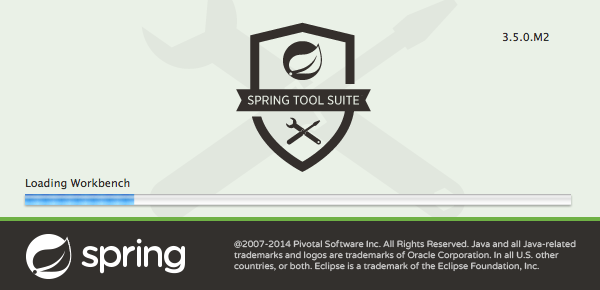
    -  ViewResolvers Chaining  
8) 뷰는 클라이언트에 전송할 응답을 생성.

1 사전작업[[편집](https://zetawiki.com/w/index.php?title=%EC%9C%88%EB%8F%84%EC%9A%B0_%EC%8A%A4%ED%94%84%EB%A7%81%EA%B0%9C%EB%B0%9C%ED%99%98%EA%B2%BD_STS_%EC%84%A4%EC%B9%98&action=edit&section=1)]

* [윈도우 JDK 설치](https://zetawiki.com/wiki/%EC%9C%88%EB%8F%84%EC%9A%B0_JDK_%EC%84%A4%EC%B9%98)
* <https://spring.io/tools> 접속
* DOWNLOAD STS (3.9.1.RELEASE for Windows) 클릭하여 spring-tool-suite-3.x.x.RELEASE-e4.x.x-win32.zip 다운로드 ( 400MB )
* zip 파일 압축 해제
* sts-bundle 폴더를 [D:\](https://zetawiki.com/w/index.php?title=D:%5C&action=edit&redlink=1)로 이동

2. STS 설치 및 실행 [[편집](https://zetawiki.com/w/index.php?title=%EC%9C%88%EB%8F%84%EC%9A%B0_%EC%8A%A4%ED%94%84%EB%A7%81%EA%B0%9C%EB%B0%9C%ED%99%98%EA%B2%BD_STS_%EC%84%A4%EC%B9%98&action=edit&section=3)]

* D:\sts-bundle\sts-3.x.x.RELEASE 폴더에 있는 STS.exe 실행



* "Select a directory as workspace" --- Workspace:[[1]](https://zetawiki.com/wiki/%EC%9C%88%EB%8F%84%EC%9A%B0_%EC%8A%A4%ED%94%84%EB%A7%81%EA%B0%9C%EB%B0%9C%ED%99%98%EA%B2%BD_STS_%EC%84%A4%EC%B9%98#cite_note-1) D:\workspace-sts --- ☑ Use this as the default and do not ask again --- [Launch]
* Dashboard 탭 닫기

3 새 프로젝트[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=3)]

* Package Explorer의 빈공간에서 우클릭 --- New --- Spring Starter Project
* "New Spring Starter Project" --- Name: bootweb1 --- Package: com.example.bootweb1 --- [Next >]
* Template Engines --- ☑ Mustache
* Web --- ☑ Web
* [Finish]

4 프로젝트 실행[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=4)]

* Package Explorer --- bootweb1 우클릭 --- Run As --- Spring Boot App
* 하단 Console 창 확인

. \_\_\_\_ \_ \_\_ \_ \_

/\\ / \_\_\_'\_ \_\_ \_ \_(\_)\_ \_\_ \_\_ \_ \ \ \ \

( ( )\\_\_\_ | '\_ | '\_| | '\_ \/ \_` | \ \ \ \

\\/ \_\_\_)| |\_)| | | | | || (\_| | ) ) ) )

' |\_\_\_\_| .\_\_|\_| |\_|\_| |\_\\_\_, | / / / /

=========|\_|==============|\_\_\_/=/\_/\_/\_/

:: Spring Boot :: (v1.3.3.RELEASE)

2016-03-10 12:37:43.509 INFO 56 --- [ main] c.example.bootweb1.Bootweb1Application : Starting Bootweb1Application on WIN-DBOPKTBU0KB with PID 56 (D:\workspace-sts\bootweb1\target\classes started by jmnote in D:\workspace-sts\bootweb1)

2016-03-10 12:37:43.514 INFO 56 --- [ main] c.example.bootweb1.Bootweb1Application : No active profile set, falling back to default profiles: default

2016-03-10 12:37:43.576 INFO 56 --- [ main] ationConfigEmbeddedWebApplicationContext : Refreshing org.springframework.boot.context.embedded.AnnotationConfigEmbeddedWebApplicationContext@4ba2ca36: startup date [Thu Mar 10 12:37:43 KST 2016]; root of context hierarchy

2016-03-10 12:37:44.229 INFO 56 --- [ main] o.s.b.f.s.DefaultListableBeanFactory : Overriding bean definition for bean 'beanNameViewResolver' with a different definition: replacing [Root bean: class [null]; scope=; abstract=false; lazyInit=false; autowireMode=3; dependencyCheck=0; autowireCandidate=true; primary=false; factoryBeanName=org.springframework.boot.autoconfigure.web.ErrorMvcAutoConfiguration$WhitelabelErrorViewConfiguration; factoryMethodName=beanNameViewResolver; initMethodName=null; destroyMethodName=(inferred); defined in class path resource [org/springframework/boot/autoconfigure/web/ErrorMvcAutoConfiguration$WhitelabelErrorViewConfiguration.class]] with [Root bean: class [null]; scope=; abstract=false; lazyInit=false; autowireMode=3; dependencyCheck=0; autowireCandidate=true; primary=false; factoryBeanName=org.springframework.boot.autoconfigure.web.WebMvcAutoConfiguration$WebMvcAutoConfigurationAdapter; factoryMethodName=beanNameViewResolver; initMethodName=null; destroyMethodName=(inferred); defined in class path resource [org/springframework/boot/autoconfigure/web/WebMvcAutoConfiguration$WebMvcAutoConfigurationAdapter.class]]

2016-03-10 12:37:44.930 INFO 56 --- [ main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat initialized with port(s): 99 (http)

2016-03-10 12:37:44.946 INFO 56 --- [ main] o.apache.catalina.core.StandardService : Starting service Tomcat

2016-03-10 12:37:44.947 INFO 56 --- [ main] org.apache.catalina.core.StandardEngine : Starting Servlet Engine: Apache Tomcat/8.0.32

2016-03-10 12:37:45.053 INFO 56 --- [ost-startStop-1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext

2016-03-10 12:37:45.053 INFO 56 --- [ost-startStop-1] o.s.web.context.ContextLoader : Root WebApplicationContext: initialization completed in 1481 ms

2016-03-10 12:37:45.352 INFO 56 --- [ost-startStop-1] o.s.b.c.e.ServletRegistrationBean : Mapping servlet: 'dispatcherServlet' to [/]

2016-03-10 12:37:45.357 INFO 56 --- [ost-startStop-1] o.s.b.c.embedded.FilterRegistrationBean : Mapping filter: 'characterEncodingFilter' to: [/\*]

2016-03-10 12:37:45.358 INFO 56 --- [ost-startStop-1] o.s.b.c.embedded.FilterRegistrationBean : Mapping filter: 'hiddenHttpMethodFilter' to: [/\*]

2016-03-10 12:37:45.358 INFO 56 --- [ost-startStop-1] o.s.b.c.embedded.FilterRegistrationBean : Mapping filter: 'httpPutFormContentFilter' to: [/\*]

2016-03-10 12:37:45.358 INFO 56 --- [ost-startStop-1] o.s.b.c.embedded.FilterRegistrationBean : Mapping filter: 'requestContextFilter' to: [/\*]

2016-03-10 12:37:45.671 INFO 56 --- [ main] s.w.s.m.m.a.RequestMappingHandlerAdapter : Looking for @ControllerAdvice: org.springframework.boot.context.embedded.AnnotationConfigEmbeddedWebApplicationContext@4ba2ca36: startup date [Thu Mar 10 12:37:43 KST 2016]; root of context hierarchy

2016-03-10 12:37:45.753 INFO 56 --- [ main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/error]}" onto public org.springframework.http.ResponseEntity<java.util.Map<java.lang.String, java.lang.Object>> org.springframework.boot.autoconfigure.web.BasicErrorController.error(javax.servlet.http.HttpServletRequest)

2016-03-10 12:37:45.754 INFO 56 --- [ main] s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/error],produces=[text/html]}" onto public org.springframework.web.servlet.ModelAndView org.springframework.boot.autoconfigure.web.BasicErrorController.errorHtml(javax.servlet.http.HttpServletRequest,javax.servlet.http.HttpServletResponse)

2016-03-10 12:37:45.786 INFO 56 --- [ main] o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/webjars/\*\*] onto handler of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]

2016-03-10 12:37:45.786 INFO 56 --- [ main] o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/\*\*] onto handler of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]

2016-03-10 12:37:45.825 INFO 56 --- [ main] o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/\*\*/favicon.ico] onto handler of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]

2016-03-10 12:37:45.954 INFO 56 --- [ main] o.s.j.e.a.AnnotationMBeanExporter : Registering beans for JMX exposure on startup

2016-03-10 12:37:46.032 INFO 56 --- [ main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port(s): 8080 (http)

2016-03-10 12:37:46.038 INFO 56 --- [ main] c.example.bootweb1.Bootweb1Application : Started Bootweb1Application in 2.861 seconds (JVM running for 3.867)

5 실행확인[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=5)]

* 브라우저에서 [http://localhost:8080](http://localhost:8080/) 접속

Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Tue Mar 15 10:40:58 KST 2016

There was an unexpected error (type=Not Found, status=404).

No message available

→ URL을 풀어주는 곳이 없음. 구현하지 않았기 때문에.

→ 어쨌든 Web 모듈은 정상 작동하고 있음

6 HomeController 작성[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=6)]

* Package Explorer 영역 --- bootweb1 [boot] --- src/main/java 우클릭 --- New --- Package
* "Java Package" --- Name: com.example.bootweb1.controllers --- [Finish]
* com.example.bootweb1.controllers 우클릭 --- New --- Class
* "Java Class" --- Name: HomeController --- [Finish]

( HomeController.java 파일이 아래와 같이 자동 생성됨 )

package com.example.bootweb1.controllers;

public class HomeController {

}

* 파일 내용 수정

package com.example.bootweb1.controllers;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

public class HomeController {

@RequestMapping("/")

public String home() {

return "안녕 Spring Boot!";

}

}

→ 일반 Controller가 아닌 RestController로 하였음

→ 별도의 View 템플릿 없이 return 값이 브라우저에 그대로 출력될 것임

7 실행확인 2[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=7)]

* 상단의 빨간 정지 버튼(■)을 눌러 앱 중지
* 상단의 빨간 시작 버튼(▶)을 눌러 앱 시작
* 브라우저에서 [http://localhost:8080](http://localhost:8080/) 접속하여 확인

안녕 Spring Boot!

8 Greet 컨트롤러 작성[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=8)]

* com.example.bootweb1.controllers 우클릭 --- New --- Class
* "Java Class" --- Name: GreetController --- [Finish]
* GreetController.java 파일 내용 수정

package com.example.bootweb1.controllers;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.RequestMapping;

@Controller

public class GreetController {

@RequestMapping("/greet")

public String greet() {

return "greet";

}

}

→ 이대로 실행하면 <http://localhost:8080/greet> 접근시 템플릿 "greet"를 찾지 못하는 오류 발생

Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Tue May 31 11:23:13 KST 2016

There was an unexpected error (type=Internal Server Error, status=500).

Circular view path [greet]: would dispatch back to the current handler URL [/greet] again. Check your ViewResolver setup! (Hint: This may be the result of an unspecified view, due to default view name generation.)

9 Greet 뷰 작성[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=9)]

* Package Explorer --- src/main/resources --- templates 우클릭 --- New --- Other...
* "New" --- Wizards: html --- [Next >]
* File name: greet.html --- [Finish]
* greet.html 내용 수정

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>Greet</title>

</head>

<body>

<p>안녕하세요, Greet!</p>

</body>

</html>

10 실행확인 3[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=10)]

* 상단의 재시작 버튼(■▶)을 눌러 앱 재시작
* 브라우저에서 <http://localhost:8080/greet> 접속하여 확인

안녕하세요, Greet!

11 Greet 컨트롤러 수정[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=11)]

* GreetController.java 파일 수정

package com.example.bootweb1.controllers;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.RequestMapping;

@Controller

public class GreetController {

@RequestMapping("/greet/{name}")

public String greet(@PathVariable("name") String name) {

return "greet";

}

}

12 Greet 뷰 수정[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=12)]

* greet.html 파일 수정

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>Greet</title>

</head>

<body>

<p>안녕하세요, {{name}}!</p>

</body>

</html>

13 실행확인 4[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EC%8A%A4%ED%83%80%ED%84%B0_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_bootweb1&action=edit&section=13)]

* 상단의 재시작 버튼(■▶)을 눌러 앱 재시작
* 브라우저에서 <http://localhost:8080/greet/홍길동> 접속하여 확인

안녕하세요, 홍길동!

pom.xml에 독립실행형 jar로 선언

1 사전작업[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EB%A0%88%EA%B1%B0%EC%8B%9C_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_%EC%8B%A4%EC%8A%B5&action=edit&section=1)]

* [윈도우 STS 설치](https://zetawiki.com/wiki/%EC%9C%88%EB%8F%84%EC%9A%B0_STS_%EC%84%A4%EC%B9%98)

2 새 프로젝트[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EB%A0%88%EA%B1%B0%EC%8B%9C_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_%EC%8B%A4%EC%8A%B5&action=edit&section=2)]

* Package Explorer의 빈공간에서 우클릭 --- New --- Spring Legacy Project
* "Spring Legacy Project" --- Project name: MyFirstSpring
* Templates: Spring MVC Project 선택 --- [Next >]
* "Spring MVC Project requires a download of 16608 bytes." --- [Yes]
* Please specify the top-level package e.g. com.mycompany.myapp\*: com.example.myfirstspring --- [Finish]

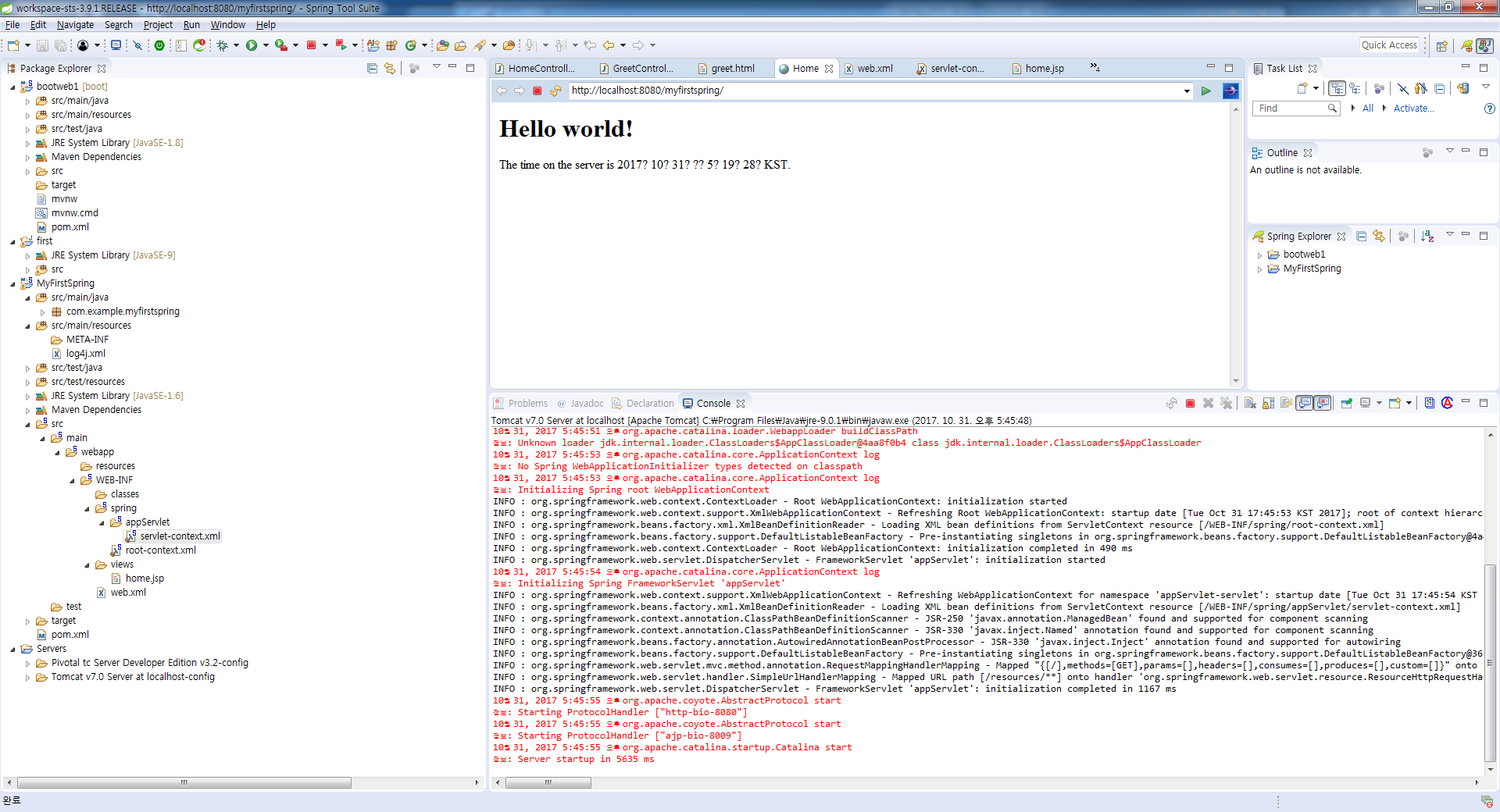
3 프로젝트 실행[[편집](https://zetawiki.com/w/index.php?title=%EC%8A%A4%ED%94%84%EB%A7%81_%EB%A0%88%EA%B1%B0%EC%8B%9C_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_%EC%8B%A4%EC%8A%B5&action=edit&section=3)]

* Package Explorer --- MyFirstSpring 우클릭 --- Run As --- Run on Server
* "Run On Server" ---
* Apache Server 버젼 tomcat v7.0 지정
* 톰캣설치할 디렉토리 생성후(tomcat)
* Download and Install에서 해당 디렉토리 지정 – finish
* "Windows 보안 경고" --- [액세스 허용(A)]

Hello world!

The time on the server is 2015? 1? 26? (?) ?? 1? 27? 06?.

pom.xml에 톰캣에 앱 실행형 war로 선언

**[WEB.XML]**

  \* XML 형식(XML Schema)의 파일로써, WEB-INF 폴더에 위치해 있습니다.  
 \* 파일 내용은 <web-app> 태그로 시작하고 </web-app> 태그로 종료되는 문서로써 web.xml 이 정의 된 Web Application의 동작과 관련된 다양한 환경 정보를 태그 기반으로 설정하는 파일입니다.  
 \* 2.4 부터는 XML Schema 파일 형태로 변경되었습니다.

 \* WEB.XML 파일은 다음과 같이 구성되어 있습니다.

1) ServletContext의 초기 파라미터  
2) Session의 유효시간 설정  
3) Servlet/JSP에 대한 정의  
4) Servlet/JSP 매핑  
5) Mime Type 매핑  
6) Welcome File list  
7) Error Pages 처리  
8) Listen/Filter 설정  
9) 보안

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app version=*"2.5"* xmlns=*"http://java.sun.com/xml/ns/javaee"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"*>

<!-- 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>

<servlet-mapping>

<servlet-name>appServlet</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

</web-app>

**[root-context.xml]**

**[servlet-context.xml]**