



02 스프링 부트 개발 시작하기

스프링 부트 개발환경

SpringBoot 2.7.6 기준

JDK

JDK 8,11,17 설치

- 공개 JDK 다운로드 후 설치
 - Eclipse Temurin
 - Microsoft OpenJDK
 - Amazon Corretto
 - Azul JDK
 - Oracle JDK
- <https://sdkman.io/>
- <https://github.com/shyiko/jabba>

IDE

- IntelliJ IDEA : <https://www.jetbrains.com/idea/download>
 - Ultimate
 - Community
- STS : <https://spring.io/tools>
- Visual Studio Code : <https://code.visualstudio.com/>

SpringBoot

Spring Boot CLI : <https://docs.spring.io/spring-boot/docs/2.7.x/reference/htmlsingle/#getting-started.installing.cli>

HelloBoot 웹 프로젝트 생성

스프링 부트 프로젝트 생성

- 웹 Spring Initializr - <https://start.spring.io/>
- IDE의 Spring Initializr 프로젝트 생성 메뉴
- Spring Boot CLI

생성 옵션

- Project: Gradle
- Language: Java
- SpringBoot Version: 2.7.6
- Group Id: tobyspring
- Name: hello
- Packaging: Jar
- Java Version: 11
- Dependency: Web



Project

☒ Gradle - Groovy ☐ Gradle - Kotlin ☐ Maven

Language

☒ Java ☐ Kotlin ☐ Groovy

Spring Boot

☐ 3.0.1 (SNAPSHOT) ☐ 3.0.0 ☐ 2.7.7 (SNAPSHOT) ☒ 2.7.6

Project Metadata

Group

Artifact

Name

Description

Package name

Packaging ☒ Jar ☐ War

Java ☐ 19 ☐ 17 ☒ 11 ☐ 8

Dependencies

ADD ...

Spring Web **WEB**

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

생성된 myboot 프로젝트 빌드와 실행

zip 파일로 생성된 경우엔 압축을 풀어준다

프로젝트 루트에서 다음 명령을 실행

```
./gradlew bootRun
```

(윈도우에서는 gradlew.bat)

웹 브라우저에서 다음과 같이 404 메시지가 뜨면 생성된 스프링 부트 애플리케이션이 정상적으로 빌드와 실행된 것임을 알 수 있다



Whitelabel Error Page

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

Tue Nov 15 08:41:10 AEST 2022

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



Gradle과 IDE의 빌드와 실행 옵션이 미묘하게 다르게 설정되는 경우가 있으니 주의. IDE에서 프로젝트를 오픈한 뒤에 프로젝트 자바 버전을 확인하자. Gradle의 자바 버전도 확인한다.

Hello 컨트롤러

```
package tobyspring.hello;

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

@RestController
public class HelloController {
    @GetMapping("/hello")
    public String hello(String name) {
        return "Hello " + name;
    }
}
```

Hello API 테스트

hello() 컨트롤러 메소드를 호출할 수 있는 HTTP Request를 생성하고 리턴되는 HTTP Response를 확인한다.

HTTP 요청을 만들고 응답을 확인하는데 사용되는 도구

- 웹 브라우저 개발자 도구
- curl
- HTTPie
- IntelliJ IDEA Ultimate- http request
- Postman API Platform
- JUnit Test
- 각종 API 테스트 도구

HTTP

웹 Request와 Response의 기본 구조를 이해하고 내용을 확인할 수 있어야 한다

Request

- Request Line: Method, Path, HTTP Version
- Headers
- Message Body

Response

- Status Line: HTTP Version, Status Code, Status Text
- Headers
- Message Body

```
> http -v ":8080/hello?name=Spring"
GET /hello?name=Spring HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
Connection: keep-alive
Host: localhost:8080
User-Agent: HTTPie/3.2.1
```

```
HTTP/1.1 200
```

```
Connection: keep-alive
Content-Length: 12
Content-Type: text/plain; charset=UTF-8
Date: Thu, 01 Dec 2022 01:45:15 GMT
Keep-Alive: timeout=60
```

```
Hello Spring
```