

스프링 부트란 무엇인가?

스프링 부트는 **자바로 웹 애플리케이션을 쉽게 만들 수 있도록 도와주는 프레임워크**이다.

원래 스프링(Spring)만으로 웹 프로그램을 만들려면 설정이 복잡했지만,
스프링 부트는 **설정을 자동화**해주고 **톰캣(Tomcat) 서버를 내장**하고 있어서
따로 서버를 설치하거나 복잡한 설정을 할 필요가 없다.

웹 프레임워크란 쉽게 말해서 **웹 개발용 스타터 키트**라고 할 수 있다.

로그인, 세션, DB 연결, 보안 등 웹 프로그램에 꼭 필요한 기능들이 이미 만들어져 있기
때문에

개발자는 핵심 기능에만 집중할 수 있다.

스프링 부트는 **보안에도 강하다**.

예를 들어 SQL 인젝션, XSS, CSRF, 클릭재킹 같은 공격을 자동으로 막아준다.

이 덕분에 개발자가 직접 복잡한 보안 코드를 짤 필요가 없다.

정리하자면, 스프링 부트는

- 설정이 간단하고,
- 톰캣 서버 내장으로 실행이 쉬우며,
- 보안 기능이 강하고,
- 개발 속도가 빠른 프레임워크이다.

2. 개발 환경 준비하기

JDK 설치

자바 개발에는 ****JDK(Java Development Kit)****가 필수이다.

JDK 는 자바 코드를 **컴파일**하고 **실행**하는 도구가 들어 있다.

Oracle 공식 사이트에서 'JDK Download'를 검색해 운영체제에 맞게 설치한다.
설치 후 `java -version` 명령으로 확인할 수 있다.

STS(Spring Tool Suite) 설치

스프링 부트 프로젝트를 작성하기 위한 IDE 로 **STS** 를 설치한다.

STS 는 이클립스 기반이며, 스프링 개발에 최적화되어 있다.

설치 후 workspace(작업 공간)를 지정하면 된다.

프로젝트 생성

STS 실행 후 **Create new Spring Starter Project** 를 클릭해 새 프로젝트를 만든다.

Group, Artifact, Package 이름 등을 설정하고, 버전을 선택한다.

만약 오류가 생기면 JDK 경로를 수정하고 인코딩(UTF-8)도 설정해야 한다.

이후 'Hello World'를 출력하는 간단한 웹 앱을 만들어 스프링 부트를 맛볼 수 있다.

3. 스프링 부트 맛보기 (Hello World 출력)

1. `HelloController.java` 라는 클래스를 만들고 `@Controller` 를 붙인다.
2. `@GetMapping("/hello")`을 추가하면 `/hello` 요청이 들어올 때 이 메서드가 실행된다.
3. `@ResponseBody` 를 붙이면 문자열 그대로 응답된다.

즉, 브라우저에서 `http://localhost:8080/hello` 에 접속하면

****“Hello World”****가 화면에 출력된다.

이 과정을 통해 스프링 부트는 로컬 서버를 실행하고, 컨트롤러를 통해 요청을 처리한다는 구조를 이해했다.

4. 개발 편의 도구 설치

◆ Spring Boot DevTools

프로그램 수정 후 매번 서버를 꺾다 켜야 하는 불편함을 해결해준다.

DevTools 를 설치하면 **파일을 수정하면 자동으로 반영**되어 개발 속도가 훨씬 빨라진다.

build.gradle 파일에 DevTools 의존성을 추가하고, Gradle 프로젝트를 새로고침하면 된다.

◆ Lombok

Getter, Setter, 생성자 등을 자동으로 만들어주는 편리한 라이브러리이다.

@Getter, @Setter, @RequiredArgsConstructor 등의 애너테이션을 사용하면 코드가 훨씬 간결해진다.

예를 들어 HelloLombok 클래스에 속성을 정의하고 애너테이션만 붙이면 별도의 메서드를 작성하지 않아도 된다.

제가 느낀 점

스프링 부트는 단순한 웹 프레임워크가 아니라,

****“개발자가 빠르고 안전하게 웹 서비스를 만들 수 있는 통합 도구”****라는 걸 알게 되었다.

특히 톰캣 내장과 자동 설정 덕분에 초보자도 쉽게 웹 서버를 실행할 수 있었다.

앞으로는 DevTools 와 Lombok 을 적극 활용해서

개발 시간을 단축하고 코드를 깔끔하게 유지하고 싶다.