



스프링 부트

백기선

[whiteship2000@gmail.com](mailto:whiteship2000@gmail.com)

# 스프링 부트

- 제품 수준의 독립적으로 실행 가능한 스프링 애플리케이션을 빠르고 쉽게 개발할 수 있게 해준다.
- 주요 기능
  - 독립적으로 실행 가능한 스프링 애플리케이션 생성
  - 내장형 Tomcat, Jetty, Undertow 지원 (WAR 파일로 묶어서 배포할 필요 없음)
  - 메이븐 설정을 간편하게 해줄 견고한 'starter' POM 제공
  - 자동 설정
  - 각종 지표, 헬스 체크, 외부 설정 지원 등의 제품 수준에 필요한 기능 제공
  - 어떠한 코드 생성이나 XML 설정도 필요 없음

# 목차

- 스프링 부트 설치
- Hello World 스프링 부트
- 핵심 기능
  - SpringApplication
  - 외부 설정
  - 프로파일
  - 로깅
- 웹 관련 기능
  - MVC
  - 내장형 컨테이너
- 스프링 시큐리티
- 데이터 다루기
  - SQL
  - NO-SQL
- 테스트
- 스프링 웹소켓
- 모니터링
  - Actuator
  - JMX
  - SSH

스프링 부트 설치

# 스프링 부트 설치

- 필요한 환경 (스프링 부트 1.2.3.RELEASE 기준)

- 자바 7
- 메이븐 3.2+ 또는 그레들 1.12+

- 지원하는 서블릿 컨테이너

- Tomcat 8 (서블릿 3.1, 자바 7+)
- Tomcat 7 (서블릿 3.0, 자바 6+)
- Jetty 9 (서블릿 3.1, 자바 7+)
- Jetty 8 (서블릿 3.0, 자바 6+)
- Undertow 1.1 (서블릿 3.1, 자바 7+)

# 스프링 부트 설치 :: 메이븐

```
<!-- Inherit defaults from Spring Boot -->
<parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>1.2.3.RELEASE</version>
</parent>

<!-- Add typical dependencies for a web application -->
<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
</dependencies>

<!-- Package as an executable jar -->
<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
    </plugins>
</build>
```

sbt01-install-spring-boot

# 스프링 부트 설치 :: CLI

- 직접 설치
  - [spring-boot-cli-1.2.3.RELEASE-bin.zip](#)
  - 내려 받아서 압축 풀고 INSTALL.txt 참고
- GVM으로 설치
  - gvm install springboot
- Homebrew로 설치
  - brew tap pivotal/tap
  - brew install springboot
- Macports로 설치
  - sudo port install spring-boot-cli

# 스프링 부트 설치 :: CLI

app.groovy

```
@RestController
class HelloBootCLI {

    @RequestMapping("/")
    String home() {
        "Hello World!"
    }
}
```

> spring run app.groovy

sbt02-spring-boot-cli



Hello Spring Boot

# Hello World 스프링 부트

- parent pom 설정 :: spring-boot-starter-parent
- 의존성 추가 및 확인 :: spring-boot-starter-web
- hello world 코드 작성
- 예제 실행
- 실행 가능한 JAR 파일로 패키징
- (덤) 자바 버전 1.8로 바꾸기

# Hello World 스프링 부트 parent pom 설정

```
<parent>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-parent</artifactId>  
  <version>1.2.3.RELEASE</version>  
</parent>
```

- <java.version>1.6</java.version>
- UTF-8
- common-logging 제거
- resource filtering
- plugin management

# Hello World 스프링 부트 의존성 설정

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-web</artifactId>  
</dependency>
```

- mvn dependency:tree
- boot-starter-web
  - boot-starter
  - boot-starter-tomcat
  - jackson-databind
  - hibernate-validator
  - spring-core, spring-web, spring-webmvc (4.1.6.RELEASE)

# Hello World 스프링 부트

## Hello World 코드

```
@RestController
@EnableAutoConfiguration
public class Example {

    @RequestMapping("/")
    public String home() {
        return "Hello Spring Boot";
    }

    public static void main(String[] args) {
        SpringApplication.run(Example.class, args);
    }
}
```

- @RestController
  - 스프링 4.0부터 지원하는 애노테이션으로 @Controller와 @ResponseBody를 사용한 것과 같다.
- @EnableAutoConfiguration
  - 자동 설정 기능을 활성화 시켜준다. 그래야 어떤 빈이 있을 때, 없을 때, 어떤 클래스가 있을 때 필요한 빈을 등록해준다.
- SpringApplication
  - 자바 메인 메서드에서 스프링 애플리케이션을 실행할 때 사용한다.

# Hello World 스프링 부트 예제 실행

- IDE에서 실행
  - main 메서드를 실행한다.
- 콘솔에서 메이븐으로 실행
  - mvn spring-boot:run
- 애플리케이션 실행 종료 (gracefully)
  - ctrl + c
- 실행 가능한 JAR 파일 실행하기
  - java -jar x.jar

# Hello World 스프링 부트 실행 가능한 JAR 패키지 만들기

```
<plugin>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-maven-plugin</artifactId>  
</plugin>
```

- provided 의존성을 추가해서 repackaging 한다.
  - 독립적으로 실행 가능하려면 외부에서 제공받을 코드의 의존성도 같이 패키징 해야 하니까!
  - 원본 파일은 .original로 이름을 수정해둔다.
- manifest를 수정한다.
  - Main-Class 헤더를 관리한다.
- <http://docs.spring.io/spring-boot/docs/1.2.3.RELEASE/maven-plugin/usage.html>
- <https://docs.oracle.com/javase/tutorial/deployment/jar/appman.html>

# Hello World 스프링 부트 자바 버전 1.8로 변경

```
<properties>  
    <java.version>1.8</java.version>  
</properties>
```



스프링 부트 사용하기

# 스프링 부트 사용하기

- 빌드 시스템: 메이븐과 그레들 지원. 앤트는 딱히...
- 다양한 Starter POM 제공
- 코드 구조: 기본 패키지(package 선언문 없는 클래스) 사용 **지양**.
- 설정 파일: XML 도 사용할 수는 있지만 자바 설정 권장.
- 자동 설정
- 빈 등록과 의존성 주입
- @SpringBootApplication
- IDE 지원
- Hot Swapping: spring-loaded
- production deployment: spring-boot-actuator

SpringApplication

# SpringApplication

- 배너 수정하기
- SpringApplication 수정하기
- Fluent 빌드 API
- 애플리케이션 이벤트와 리스너
- 웹 환경
- CommandLineRunner
- 애플리케이션 종료

외부 환경 설정 파일

# 외부 설정 파일

- 랜덤 값 사용하기
- 명령행 프로퍼티 사용하기
- 애플리케이션 프로퍼티 파일
- 프로파일 프로퍼티 파일
- `${placeholder}`
- YAML
- `@ConfigurationProperties`

프로파일

# 프로파일

- `spring.profiles.active`
- `spring.profiles.include`
- `SpringApplication.setAdditionalProfiles(...)`
- `@ConfigurationProperties`



로깅

# 로깅

- 로그 포맷
- 콘솔 출력
- 파일 출력
- 로그 레벨
- 커스텀 로그 설정

웹 애플리케이션 개발

# 웹 애플리케이션 개발

- 스프링 웹 MVC 프레임워크
- 내장형 톰캣 컨테이너 지원

SQL 데이터베이스

# SQL 데이터베이스

- spring-boot-starter-jdbc
  - DataSource 설정하기
  - JdbcTemplate
- spring-boot-starter-jpa
  - Spring Data JPA

NoSQL 사용하기

시큐리티



테스트

웹소켓

# Spring Boot Actuator

# HTTP 사용해서 모니터링 및 관리

# JMX 사용해서 모니터링 및 관리

원격 쉘 사용해서  
모니터링 및 관리