



Spring Boot Basic

스프링 부트 기초

스프링 부트 개념 및 특징



한국기술교육대학교
온라인평생교육원

학습내용

- 프레임워크의 개념
- 스프링 프레임워크의 개념 및 특징
- 스프링 부트의 개념 및 특징

학습목표

- 프레임워크의 개념을 설명할 수 있다.
- 스프링 프레임워크의 개념 및 특징을 설명할 수 있다.
- 스프링 부트의 개념 및 특징을 설명할 수 있다.



프레임워크의 개념

프레임워크의 개념

1 프레임워크(Framework) 이해

① 프레임워크란?

- 사전적 의미는 '뼈대' 혹은 '틀'을 의미

framework 미국식 [-wɜ:rk] 영국식 ['freɪmwɜ:k]

1. (건물 등의) 뼈대
2. (판단·결정 등을 위한) 틀
3. 체제, 체계

(건물 등의) 뼈대
(판단·결정 등을 위한) 틀

[출처 : 네이버 사전]

프레임워크의 개념

1 프레임워크(Framework) 이해

① 프레임워크란?

- 사전적 의미는 '뼈대' 혹은 '틀'을 의미

소프트웨어 관점

- 애플리케이션의 골격에 해당하는 아키텍처를 제공

- 프레임워크 기반의 애플리케이션을 개발하면 개발자는 비즈니스 로직에 집중할 수 있음

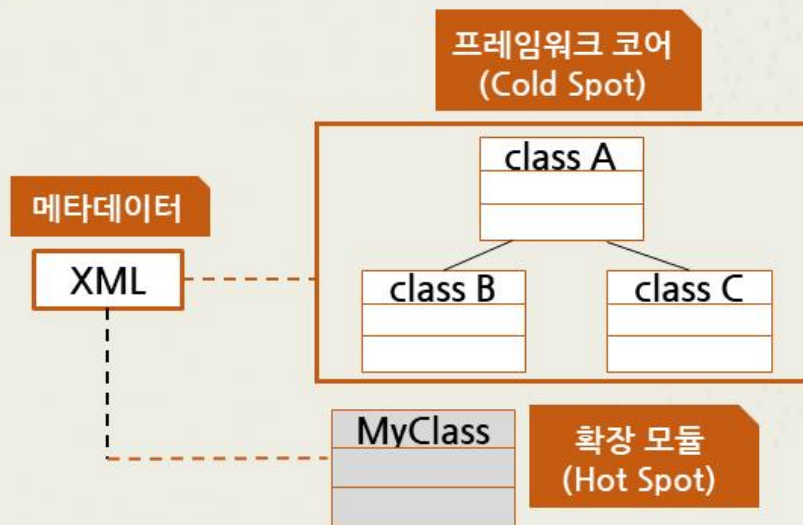


프레임워크의 개념

프레임워크의 개념

1 프레임워크(Framework) 이해

② 프레임워크 기반의 애플리케이션 구조



프레임워크의 개념

2 오픈 소스 프레임워크(Open Source Framework) 이해

① 오픈 소스 자바(Java) 프레임워크 종류

레이어별 가장 많이 사용하는 자바 기반 프레임워크

처리 영역	프레임워크	설명
Presentation	Struts	MVC(Model View Controller) 아키텍처를 제공하는 프레임워크
	Spring (MVC)	스트러츠(Struts) 같은 MVC 아키텍처를 제공하지만 독립적으로 존재하지 않고 스프링 프레임워크에 포함되어 있음



프레임워크의 개념

프레임워크의 개념

2 오픈 소스 프레임워크(Open Source Framework) 이해

① 오픈 소스 자바(Java) 프레임워크 종류

레이어별 가장 많이 사용하는 자바 기반 프레임워크

처리 영역	프레임워크	설명
Business	Spring (IoC, AOP)	스프링의 IoC와 AOP 모듈을 이용하여 스프링 컨테이너에서 동작하는 비즈니스 컴포넌트를 개발함

프레임워크의 개념

2 오픈 소스 프레임워크(Open Source Framework) 이해

① 오픈 소스 자바(Java) 프레임워크 종류

레이어별 가장 많이 사용하는 자바 기반 프레임워크

처리 영역	프레임워크	설명
Persistence	JPA (Java Persistent API)	하이버네이트(Hibernate)를 비롯한 모든 ORM(Object Relation Mapping) 프레임워크의 표준이다.
	MyBatis	XML 파일에 작성한 SQL을 자바 객체 (VO)와 매핑해주는 데이터 매퍼 프레임 워크



프레임워크의 개념

프레임워크의 개념

2 오픈 소스 프레임워크(Open Source Framework) 이해

② 오픈 소스 프레임워크의 장점

- 1 빠른 구현 시간
- 2 관리의 용이성 증가
- 3 개발자들의 역량 획일화
- 4 검증된 아키텍처의 재사용
- 5 아키텍처의 일관성 유지



스프링 프레임워크의 개념 및 특징

스프링 프레임워크의 개념 및 특징

1 스프링 프레임워크

① 스프링의 탄생

- 로드 존슨(Rod Johnson)이 집필한 『J2EE Development without EJB』에서 소개됨
- POJO(Plain Old Java Object)를 기반으로 하는 경량의 환경을 제공함
- 애플리케이션 운용에 필요한 객체를 생성하고, 객체들 사이에 의존성(Dependency)을 관리하는 컨테이너를 제공함

스프링 프레임워크의 개념 및 특징

1 스프링 프레임워크

② 스프링의 변화

초기

- 단순히 객체를 관리하는 컨테이너 기능만을 제공

현재

- 엔터프라이즈 시스템 개발에 필요한 모든 분야를 지원하는 하나의 플랫폼으로 사용됨



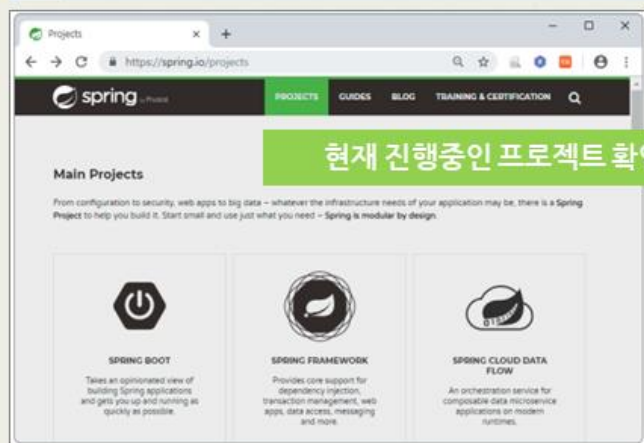
스프링 프레임워크의 개념 및 특징

스프링 프레임워크의 개념 및 특징

1 스프링 프레임워크

2 스프링의 변화

스프링 홈페이지 (<https://spring.io>)



현재 진행중인 프로젝트 확인 가능

스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

1 스프링 부트의 등장

① 스프링의 복잡도 증가

- 본래 웹 애플리케이션만을 목적으로 만든 것은 아님
- 하지만 대부분 웹 애플리케이션 개발 시 스프링을 사용

- ↓
- 웹 애플리케이션이 복잡해지면서 원래 목적인 비즈니스 로직에 집중하지 못하게 함 (라이브러리 관리 및 XML 환경 설정)

스프링 부트의 개념 및 특징

1 스프링 부트의 등장

② 스프링 부트의 등장



스프링과 달리 웹 애플리케이션을 목적으로 함

! 스프링처럼 많은 설정이 필요하지 않음

개발자들이 쉽게 접근 가능

스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

1 스프링 부트의 등장

② 스프링 부트의 등장



- 스프링에서는 시장의 요구와 변화에 대응하기 위해 스프링 부트 프로젝트를 진행하게 됨

스프링 부트의 개념 및 특징

1 스프링 부트의 등장

② 스프링 부트의 등장

- 커맨드 도구를 제공하고 톰캣(Tomcat) 같은 내장 서버를 통해 서버와 관련된 복잡한 설정들을 제거함



- 스프링을 처음 사용하는 개발자도 애플리케이션 관련 설정을 쉽게 처리, 관리함으로써 개발 자체에 집중할 수 있도록 함

스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

2 스프링 부트의 특징

① 라이브러리 관리 자동화

- 메이븐(Maven)을 기반으로 하는 스타터(Starter)를 통해 애플리케이션에 필요한 라이브러리 의존성 문제를 해결함

스프링 부트의 개념 및 특징

2 스프링 부트의 특징

② 설정의 자동화

- 추가된 라이브러리와 환경 설정을 자동으로 처리함

- 예** JPA를 사용하기 위해 JPA 스타터 추가 시
스프링 부트가 라이브러리에 포함된 자동설정 클래스를
인지해 JPA 사용에 필요한 객체들을 자동으로 생성

스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

2 스프링 부트의 특징

③ 라이브러리 버전 자동 관리

- 스프링 프로젝트에서는 스프링 라이브러리 뿐만 아니라 서드파티 라이브러리도 사용
- 스프링 부트는 스프링 라이브러리 뿐만 아니라 관련된 서드파티 라이브러리도 함께 다운로드

스프링 부트의 개념 및 특징

2 스프링 부트의 특징

④ 테스트 환경과 내장 서버

- 기본적으로 JUnit을 비롯한 테스트 관련 라이브러리들을 포함
- 다양한 계층의 테스트 케이스를 쉽게 작성 가능
- 톰캣 같은 웹 서버 내장
- 일반 자바 애플리케이션을 실행하는 것과 동일한 방식으로 웹서버를 구동하기 때문에 빠르게 실행 결과를 확인할 수 있음

스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

2 스프링 부트의 특징

⑤ 독립적으로 실행 가능한 JAR

- 애플리케이션을 실제 운영 서버에 배포하기 위해서는 패키징(Packaging)을 해야 함

웹 애플리케이션

WAR(Web Archive)
파일로 패키징

스프링 부트

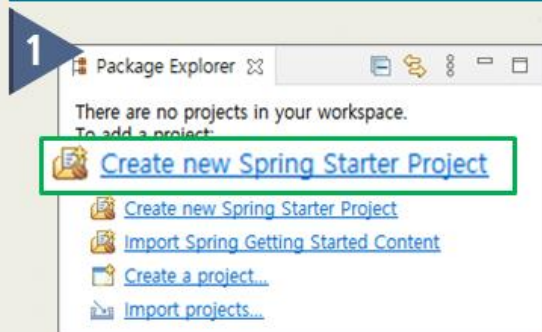
웹 애플리케이션도
JAR 파일로 패키징 가능
(빠른 개발과 빠른 배포)

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

① 스프링 부트 프로젝트 생성 메뉴

- 이클립스에서 “Create new Spring Starter Project”를 선택하거나



스프링 부트의 개념 및 특징

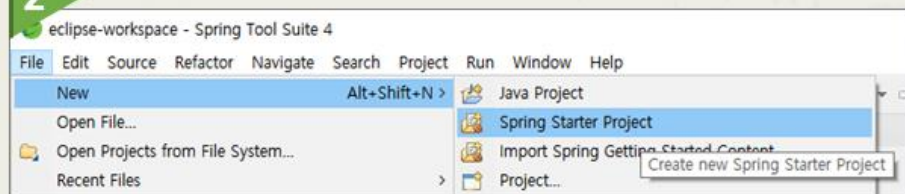
스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

① 스프링 부트 프로젝트 생성 메뉴

- 이클립스에서 “Create new Spring Starter Project”를 선택하거나
File → New → Spring Starter Project 메뉴를
선택하여 스프링 부트 프로젝트를 생성함

2

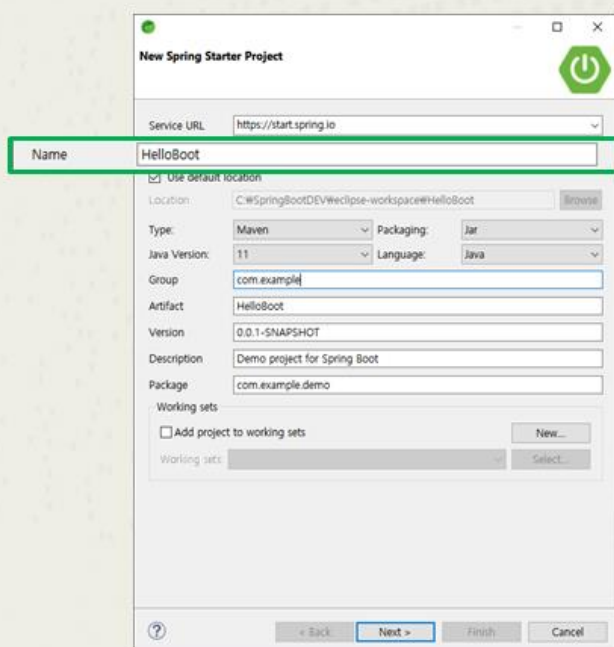


스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo



스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo

스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo

The screenshot shows the 'New Spring Starter Project' dialog. The 'Java Version' dropdown is highlighted with a green box and set to '11'. Other fields include Service URL (https://start.spring.io), Name (HelloBoot), Use default location (checked), Location (C:\SpringBootDEV\workspace\HelloBoot), Type (Maven), Packaging (Jar), Language (Java), Artifact (HelloBoot), Version (0.0.1-SNAPSHOT), Description (Demo project for Spring Boot), and Package (com.example.demo).

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo

The screenshot shows the 'New Spring Starter Project' dialog. The 'Language' dropdown is highlighted with a green box and set to 'Java'. Other fields include Service URL (https://start.spring.io), Name (HelloBoot), Use default location (checked), Location (C:\SpringBootDEV\workspace\HelloBoot), Type (Maven), Packaging (Jar), Language (Java), Artifact (HelloBoot), Version (0.0.1-SNAPSHOT), Description (Demo project for Spring Boot), and Package (com.example.demo).

스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo

The screenshot shows the 'New Spring Starter Project' dialog. The 'Group' field is highlighted with a green box and contains the text 'com.example'. Other fields include 'Service URL' (https://start.spring.io), 'Name' (HelloBoot), 'Type' (Maven), 'Packaging' (Jar), 'Java Version' (11), 'Language' (Java), 'Artifact' (HelloBoot), 'Version' (0.0.1-SNAPSHOT), 'Description' (Demo project for Spring Boot), and 'Package' (com.example.demo).

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo

The screenshot shows the 'New Spring Starter Project' dialog. The 'Artifact' field is highlighted with a green box and contains the text 'HelloBoot'. Other fields include 'Service URL' (https://start.spring.io), 'Name' (HelloBoot), 'Type' (Maven), 'Packaging' (Jar), 'Java Version' (11), 'Language' (Java), 'Group' (com.example), 'Version' (0.0.1-SNAPSHOT), 'Description' (Demo project for Spring Boot), and 'Package' (com.example.demo).

스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo

The screenshot shows the 'New Spring Starter Project' dialog box. The 'Version' field is highlighted with a green box and contains the text '0.0.1-SNAPSHOT'. Other fields include Service URL (https://start.spring.io), Name (HelloBoot), Location (C:\springbootDEV\workspace\HelloBoot), Type (Maven), Packaging (Jar), Java Version (11), Language (Java), Group (com.example), and Artifact (HelloBoot). The Description is 'Demo project for Spring Boot' and the Package is 'com.example.demo'.

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

② 프로젝트 기본 설정

- Name : HelloBoot
- Type : Maven
- Packaging : Jar
- Java Version : 11
- Language : Java
- Group : com.example
- Artifact : HelloBoot
- Version : 0.0.1-SNAPSHOT
- Package : com.example.demo

The screenshot shows the 'New Spring Starter Project' dialog box. The 'Package' field is highlighted with a green box and contains the text 'com.example.demo'. Other fields include Service URL (https://start.spring.io), Name (HelloBoot), Location (C:\springbootDEV\workspace\HelloBoot), Type (Maven), Packaging (Jar), Java Version (11), Language (Java), Group (com.example), and Artifact (HelloBoot). The Description is 'Demo project for Spring Boot' and the Version is '0.0.1-SNAPSHOT'.

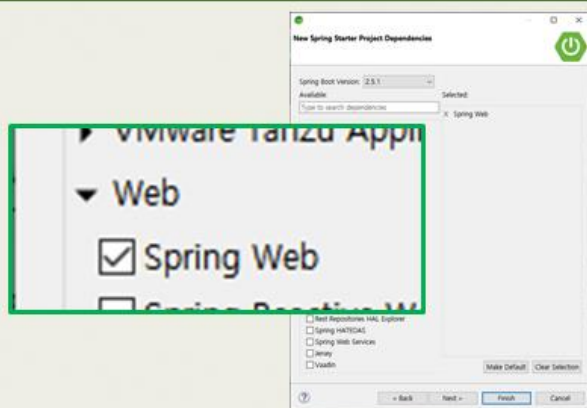
스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

③ 라이브러리 의존성 설정

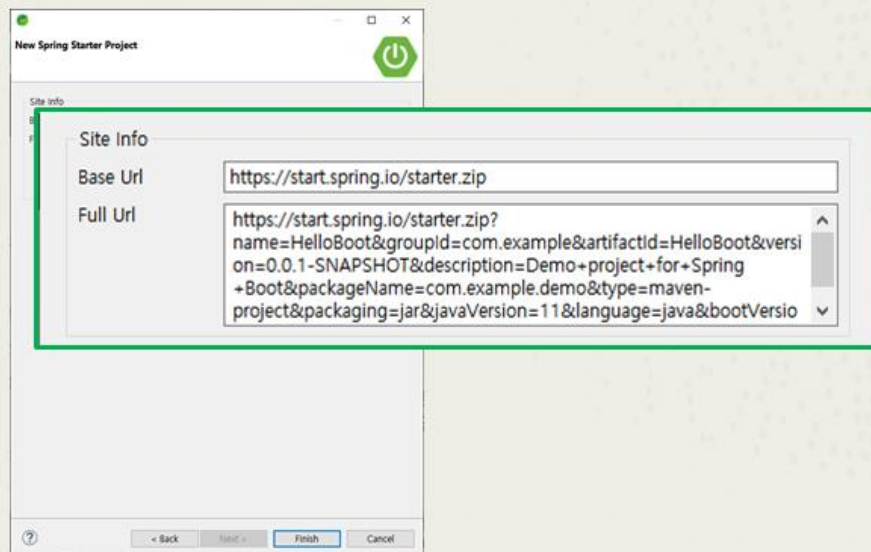
- 스타터를 이용해 라이브러리 의존성 설정
- Web → Spring Web 하나만 선택



스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

④ 프로젝트 생성 URL



스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

5 스프링 부트 프로젝트 실행

- 기본적으로 웹 애플리케이션으로 실행됨

```
package com.example.demo;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class HelloBootApplication {

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

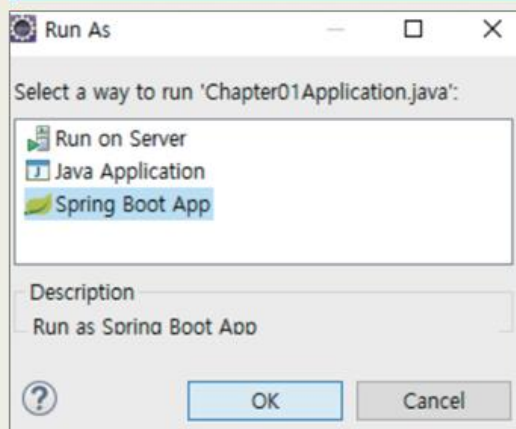
}
```

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

5 스프링 부트 프로젝트 실행

- 기본적으로 웹 애플리케이션으로 실행됨



스프링 부트의 개념 및 특징

스프링 부트의 개념 및 특징

3 스프링 부트 프로젝트 생성

5 스프링 부트 프로젝트 실행

- 일반 자바 애플리케이션으로도 실행 가능

```
package com.example.demo;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.WebApplicationType;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class HelloBootApplication {

    public static void main(String[] args) {
        SpringApplication application =
            new SpringApplication(HelloBootApplication.class);
        application.setWebApplicationType(WebApplicationType.NONE);
        application.run(args);
    }
}
```



실전 스프링 부트

스프링 부트 개념 및 특징

실전 스프링 부트

Spring Boot Basic

There are no projects in your workspace. To add a project:

- Create a Maven project
- Create new Spring Starter Project
- Import Spring Getting Started Content
- Create a Java EE EAR project
- Create a Dynamic Web project
- Create an EJB project
- Create a Connector project
- Create a Java EE application client project
- Create a generic deployable web project
- Create a project...
- Import projects...

1. 스프링 프레임워크 기반의 웹 애플리케이션 프로젝트 구조 분석하기

- Zulu(OpenJDK) 16 사용
- Spring Tools 4 for Eclipse 4.0.10 사용

스프링 프레임워크 기반의 웹 애플리케이션 프로젝트 구조 분석

실습단계
스프링 프레임워크 기반의 웹 애플리케이션 프로젝트 구조 분석
Eclipse에 스프링 부트 프로젝트 생성 후 스프링 부트 제공 애플리케이션 실행
Project Explorer 이용
Create new Spring Starter Project 선택
Project Explorer에 템플릿이 보이고, Project 생성 이후에는 화면이 제공되지 않음
프로젝트 생성 가능 : File → New → 템플릿 이용(Spring Starter Project)
프로젝트 이름 지정 : HelloBoot
라이브러리 관리를 어떤 방식으로 할 것인지 선택해야함
Maven : 아파치에서 제공하는 빌드 자동화 도구, JAVA 쪽 빌드 도구의 표준
Gradle : 언어와 무관하게 라이브러리 관리를 할 수 있음
Java Version 설정
Packaging 설정 : JAVA 프로젝트 → Jar 파일, 웹 프로젝트 → War
스프링 부트의 경우 일반 JAVA 프로젝트든, 웹 프로젝트든 Jar 파일로 패키징 후 배포해서 운영 서버에 기능 제공



실전 스프링 부트

실습단계
Language : Java
Group ID는 어느 회사에서 관리, 개발 하는지에 대한 정보
유니크한 문자를 사용해야 함, 도메인 기반으로 작성
Artifact : 프로젝트 이름에 해당
Version : 0.0.1-SNAPSHOT
프로젝트에 대한 간단한 설명 지정
Package : Group ID와 Package를 일치시키는 것이 일반적
Available에서 선택하여 사용, 한 번 선택하면 이력이 남게 됨
가장 최신 버전
Web → Spring Web(웹과 관련된 라이브러리를 스프링 부트 프로젝트에 추가할 수 있음)
프로젝트 생성 시 필요한 라이브러리 다운로드에 시간이 걸림
다운로드 진행 상황 확인은 우측 하단의 버튼을 눌러서 확인 가능
중요! Maven이라는 도구가 프로젝트에 필요한 라이브러리를 가지게 됨
중요! 루트 패키지에 기본적으로 애플리케이션 하나가 제공됨
글씨 크기가 작을 경우 <Ctrl> + <+>
실행 : <Ctrl> + <F11> 또는 오른쪽 마우스 → Run As → Spring Boot App
스프링 부트 애플리케이션 실행 시 '내장 톰캣'이 동작하여 기본적으로 '8080' 포트로 구동됨
즉, 웹 애플리케이션으로 동작한 것



실전 스프링 부트

2. 스프링 부트 기반의 웹 애플리케이션 프로젝트 생성 및 실행하기

```

1 package com.mycompany;
2
3 import org.springframework.boot.SpringApplication;
4
5
6 @SpringBootApplication
7 public class HelloBootApplication {
8
9     public static void main(String[] args) {
10         SpringApplication.run(HelloBootApplication.class, args);
11     }
12 }
13
14

```

톰캣 서버를 구동하지 않고 JAVA 애플리케이션으로 실행 가능

```

HelloBootApplication : Starting HelloBootApplication using Java 16.0.1 on DESKTOP-N8EIBLT with PID 1601
HelloBootApplication : No active profile set, falling back to default profiles: default
ed.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)

```

실습단계

톰캣 서버를 구동하지 않고 JAVA 애플리케이션으로 실행 가능

현재 실행 중인 애플리케이션 중지

스프링 애플리케이션 객체를 생성함

웹 애플리케이션 타입을 세팅함

애플리케이션 실행

일반 JAVA 애플리케이션으로 동작