Spring framework

웹 어플리케이션 틀 안에서 필요한 기능을 구현가능

(DI, AOP ,MVC JDBC 제공함)

메이븐 : 자바프로젝트의 빌드를 자동으로 해주는 도구

개발자가 xml에 작성한 프로젝트 정보를 토대로 컴파일, 라이브러리를 연결할 수 있다.

1. Spring Bean의 정의

Spring에서 POJO(plain, old java object)를 ‘Beans’라고 부른다.

* Beans는 애플리케이션의 핵심을 이루는 객체이며, Spring IoC(Inversion of Control) 컨테이너에 의해 인스턴스화, 관리, 생성된다.
* Beans는 컨테이너에 공급하는 설정 메타 데이터(XML 파일)에 의해 생성된다.
* 컨테이너는 이 메타 데이터를 통해 Bean의 생성, Bean Life Cycle, Bean Dependency(종속성) 등을 알 수 있다.
* 애플리케이션의 객체가 지정되면, 해당 객체는 getBean() 메서드를 통해 가져올 수 있다.
* 일반적으로 XML 파일에 정의한다.

스프링은 기본적으로 모든 bean을 **singleton**으로 생성하여 관리한다. (구체적으로는 애플리케이션 구동 시 JVM 안에서 스프링이 bean마다 하나의 객체를 생성하는 것을 의미한다.)

* 주요 속성
  + class(필수): 정규화된 자바 클래스 이름
  + id: bean의 고유 식별자
  + scope: 객체의 범위 (sigleton, prototype)
  + constructor-arg: 생성 시 생성자에 전달할 인수
  + property: 생성 시 bean setter에 전달할 인수
  + init method와 destroy method-method, lazy-init, scope속성

1. XML파일 : 데이터들을 쉽게 구조화해서 전달하기 위해 만들어진 문서 형식(데이터의 종류를 정확하게 알수 있다) -> 확장기능

Ex) 음식에 대한 정보를 저장하고 공유한다고 생각해라.

<?xml version=”1.0”?>

<Food>

<NAME>Pizza</NAME>

<Price>30000</Price>

</Food>

1. Maven Project

* pom.xml : 프로젝트 정보가 표시되며 스프링에서 사용되는 여러 가지 라이브러리를 설정해 다운로드 할 수 있다.
* src/main/java : 자바 소스 파일 위치
* src/main/resources : 프로퍼티 파일이나 XML파일 등 리소스 파일 위치