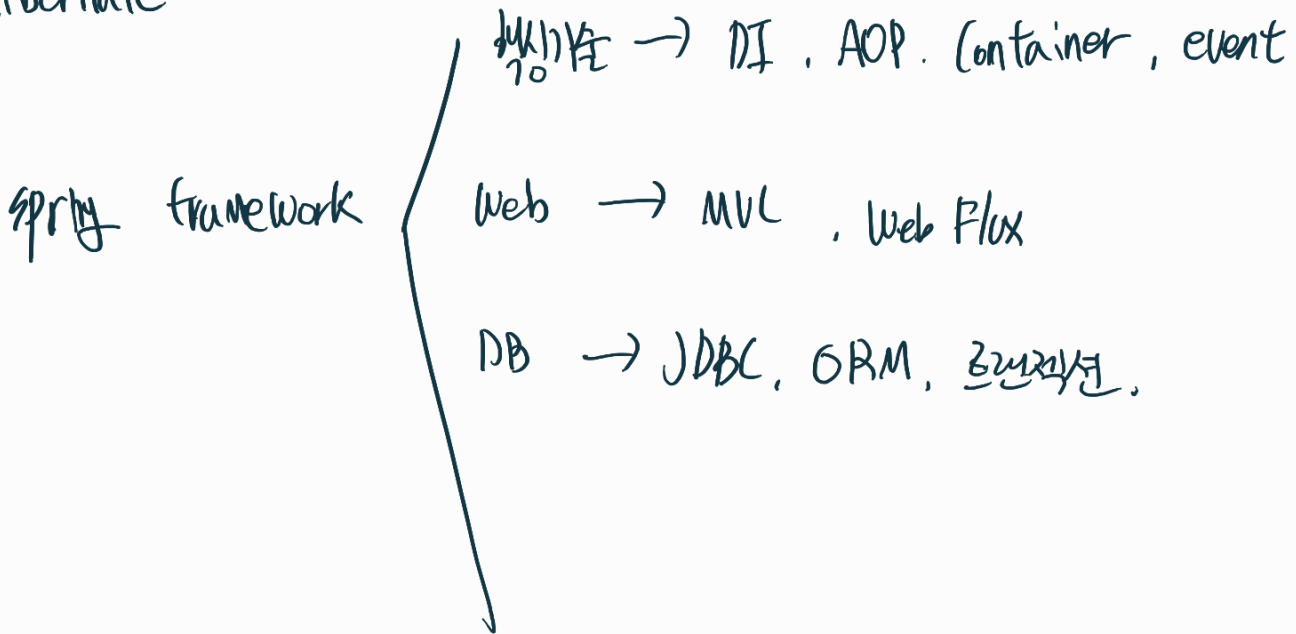


JPA == ORM

..

Hibernate



spring boot → tomcat server 내장 외부서버 필요x
starter 종속성제공
스프링과 외부라이버러리 자동구성.

스프링?

Java 기반의 프레임워크

↓
적체지향 애플리케이션을 개발할 수 있도록 만든 언어.

좋은 적체지향의 특징? → 프로그래밍의 변형, 유연.

예상과 유연성.

→ 강타입을 쉽고 유연하게 변경가능하다.

⇒ Polymorphism (다형성)

구현체가 바뀌더라도

객체는 사용자에게 영향이 X. 제반 객체 → interface

→ 사용자는 객체의 특성에 의존한다. ⇒ 객체 → class, object

→ 객체 내부의 내부 환경도 X

가성성의 원칙

클라이언트를 변경하지 않고, 서버의 기능을 유연하게 변경가능.

소프트웨어 설계 원칙 SOLID

SRP (single responsibility principle) 한의 책임 원칙

→ 한 클래스에 있는 과제가 적다면 Good.

OC (open / closed principle) 개방-폐쇄
기능은 여려있지만 형태는 잘변하지 않는다.

→ 새로운 작업이 들어올 때 인터페이스와 클래스가 필요.

LSP (Liskov substitution principle) 리스코프 치환
정확성을 해치지 않고 하위인스턴스로 변경가능

ISP 인터페이스 분리 원칙

DIP (Dependency inversion principle) 의존관계 역전.

추상화에 의존하지 구체화에 의존 X

인터페이스에 의존하면 생각이변함.

역삼역 앞