**<의존주입의 여러 가지 방법>**

**생성자 호출 방법**

 : 생성자는 딱 한 번 호출되기 때문에, 다른 곳에서 세팅해주거나 할 수 없음. 한 번 정해지면 정해지지 않는다.

좋은 아키텍쳐는 제약과 한계점이 있어야한다. 다 열려있으면 어디서 수정했는지 알 수 없음.

생성자를 통한 주입 방법은 처음 생성할 때에만 주입해줄 수 있고, 해당 클래스에서 사용하는 필드의 값을 변경해주는 작업은 외부에서 처리할 수 없다.

스프링 컨테이너에 빌드될 때 사용되는 객체들의 변경가능성이 없다면, 외부에서 수정하는 걸 막아야한다. 애플리케이션 빌드 이후에 객체를 수정하고 싶은 생각이 없을 때 생성자를 호출하여 의존 주입하는 방식을 사용하면 좋다. -> 이런 경우에 set메소드와 같은 값을 변경해주는 메소드를 퍼블릭으로 열어두면 안된다. set메소드가 없는 이상 주입하는 객체가 변경될 가능성이 없다.

set메소드가 존재한다면 누군가는 set메소드를 사용해서 변경한다.

생성자를 통해서 값을 세팅을 하는 것 : 생성자의 매개변수로 들어오는 타입은 무조건 값이 존재한다는 의미. 매개변수로 들어오는 타입을 세팅해주지 않으면 컴파일 오류가 생김.

생성자에는 값을 다 채워넣는다.

생성자를 사용하여 의존 주입 시 -> 값이 불변이면서 무조건 들어가야하는 경우에 사용한다.

생성자가 딱 한 개만 존재할 때 생성자에 Autowired가 생략되어있어도 자동으로 생성해준다.

**수정자 주입(setter 주입)**



setMethod에 Autowired 추가하면 자동으로 세팅을 해준다.

스프링 빈을 컨테이너에 등록해주는 작업 -> 연관관계를 자동으로 세팅해주는 작업 진행.

스프링 라이프사이클이 컨테이너 생성단계와 의존주입하는 단계 두가지로 나뉘어져있다.

수정자 주입을 통해 의존관계 세팅이 일어남.

물론 생성자를 사용하여 객체를 생성할 때에도 의존관계 주입이 일어나긴 한다.

생성자 주입은 생성자를 부르면서 객체를 생성할 때 어쩔 수 없이 동시에 의존관계 주입이 일어난다.

하지만 setter같은 것들은 컨테이너 빈에 객체 생성이 먼저 일어난 후에, 빈에 생성되어있는 객체들의 주입이 나중에 일어나게 된다.

생성자에도 Autowired가 있고, 수정자에도 Autowired 가 존재하면, 생성자 호출이 먼저 일어나고, 수정자 호출이 먼저 일어난다. 사실 수정자에 Autowired가 존재한다면 생성자를 통한 주입이 필요없음.

언제사용하나면, 선택적으로 의존 관계 주입을 해야할 때 사용한다. 있어도 되고, 없어도 되는 값의 경우 @Autowird(required = false)를 사용해서 의존 주입 관계를 생략할 수도 있다.

중간에 변경하고자 할 때도 수정자를 통한 의존 주입을 사용할 수 있다. Ex) A db를 이용하다 b db를 이용하고자 할 때 사용

Autowired의 기본 동작은 주입할 대상이 없으면 오류가 발생한다. 주입할 대상이 없어도 동작하게 하려면 @Autowired(required = false)로 지정하면 된다.

객체 안에 있는 필드에 접근 및 수정할 때의 프로퍼티 규약 : 필드의 첫 문자를 대문자로 바꾸고 get이나 set을 앞에 붙임.

**필드주입**



코드가 굉장히 심플함.

필드 주입은 권장되지 않는다.

외부에서 변경이 불가능해서 테스트하기가 굉장히 어렵다.

예를 들어, db에 접근하는 객체를 사용할 수 없을 때, db에 접근하여 테스트하기 어려울 때 더미 데이터를 만들어주는 레포지토리로 테스트를 하고 싶을 때가 있음.

이때에 필드를 통한 주입을 하게 되면, 레포지토리를 변경하여 테스트하기가 매우 어렵다.

필드를 통한 주입을 사용하면, 스프링 컨테이너와 같은 DI 프레임워크가 없이, 순수한 자바로 테스트할 수 있는 방법이 없음. 값을 넣어주고 싶은데 넣어주는 방법이 없음. - > setter를 결국 만들어야 함.

그럴바엔 수정자에 Autowired를 사용하는 게 나음.

스프링 컨테이너를 사용한 테스트는 필드 주입을 하는 것이 편하다.

**일반 메서드 주입**

아무 메소드에도 Autowired를 쓸 수 있음.



일반적으로 사용하지 않는다.

스프링 빈이아닌 클래스에서 @Autowired를 쓰더라도 아무기능이 동작하지 않는다.