스프링 DI

|  |  |
| --- | --- |
| 1 | 다음 중 맞는 설명을 고르면 ?  (1) 어플리케이션 개발 시 유지 보수성을 높이려면 객체간 의존도 및 결합도는 높아야 한다.x  (2) 스프링을 지배하는 두가지 핵심 원리는 DI와 AOP이다.o  (3) DI란 Dependency Injection의 약자로 객체의 의존도 및 결합도를 강화하기 위한 개발 기법을 의미한다.x  (4) DI란 스프링의 창시자인 로드 존슨이 제안한 개발 패턴이다x  (5) DI를 이용하면 코드안에서 객체를 직접 생성하는 과정을 피할 수 있으며, 다형성을 이용하여 객체간 의존 관계를 약화시킬 수 있다.o |
| 2 | 다음 설명 중 틀린 것은?  (1) 스프링 프레임워크를 사용하지 않더라도 어플리케이션에 DI를 적용할 수 있다o  (2) DI를 구현하기 위해서는 어플리케이션 내에서 사용되는 객체들을 대신 생성하거나 주입시켜주는 역할을 수행할 누군가가 필요하다o  (3) (2)번과 같은 역할을 수행하는 주체를 스프링에서는 컨테이너라 하며, 객체의 생명주기를 관리하고 객체와 객체를 엮어주는(Weaving) 역할까지 수행한다o  (4) 스프링에서 컨테이너의 종류는 크게 2가지로 분류되며, BeanFactory와 ApplicationContext 가 있다o  (5) BeanFactory는 DI에 대한 단순한 기능을 제공하는 컨테이너다  (6)ApplicationContext는 BeanFactory를 확장한 컨테이너로 외부 설정파일을 읽거나 이벤트 리스너를 지원한다  (7)실제 개발에서는 주로 BeanFactoryr를 사용하게 된다 |
| 3 | 알맞는 내용을 채워 넣으세요   |  |  | | --- | --- | | 클래스명 | 역할 | | ( ClassPathX )ApplicationContext | 클래스패스에 위치한 XML 파일의 컨텍스트 정의 내용을 로드한다 | | ( )ApplicationContext | 파일시스템 경로로 XML파일의 컨텍스트 정의 내용을 로드한다 | | ( )ApplicationContext | 웹어플리케이션에 포함된 XML 파일 경로로 컨텍스트 정의 내용을 로드한다 | |