

SpringIdol2 예제를 다음과 같이 수정하시오.

1. **spring-idol-autoload.xml** 에 나타난 bean 들의 DI 설정을 최대한 자동 설정(auto-wiring)으로 변경하시오.
 - 각 bean에 대해 어떤 방식을 사용하는 것이 가장 적합한지 고려한다.
 - **SpringIdolMain_autowire.java** 프로그램을 실행하여 테스트 함
2. 기존 XML 기반 bean 설정을 모두 Java Code 기반 설정으로 변경하시오.
 - **spring-idol.xml**의 설정과 동일한 설정이 되도록 **com.example.springidol.javaconf1** 패키지의 **SpringConfig** 클래스의 코드를 수정 및 추가한다.
 - ✓ **SpringIdolMain_javaconf1.java** 프로그램을 실행하여 테스트 함
 - 위 결과와 동일한 설정이 되도록 **com.example.springidol.javaconf3** 패키지의 **SpringConf1** 및 **SpringConf2** 클래스의 코드를 수정 및 추가한다.
 - ✓ **SpringIdolMain_javaconf3.java** 프로그램을 실행하여 테스트 함
3. 기존 XML 기반 bean 설정을 모두 annotation 기반 설정 및 bean scan 방식으로 변경하시오.
 - **spring-idol-annotation.xml**의 설정과 동일한 설정이 되도록 한다.
 - **com.example.springidol.annotation** 패키지 내의 클래스들에 대해 아래와 같은 annotation들을 한 번 이상 사용한다.
 - ✓ @Autowired를 이용한 DI 설정
 - 예: PoeticJuggler, SpringIdol bean에 대한 의존 객체 주입
 - ✓ @Autowired 및 @Qualifier를 이용한 DI 설정
 - 예: OneManBand, Singer, Instrumentalist bean에 대한 의존 객체 주입
 - ✓ @Value를 이용한 DI 설정
 - 예: Instrumentalist, Singer, Song, PoeticJuggler, Encore에 대한 String/int/SpEL식 주입
 - ✓ @Resource를 이용한 DI 설정
 - 예: Instrumentalist, Singer bean에 대한 의존 객체 주입
 - ✓ @Component를 통한 자동 bean 검색 및 등록 설정
 - Annotation 기반 설정을 위해 **spring-idol-annotation.xml**을 수정한다.
 - ✓ 명시적인 bean 설정들을 모두 삭제하고 <context:component-scan />만 포함
 - **SpringIdolMain_annotation.java** 프로그램을 실행하여 테스트 함

- Due: 4.6/8

- 제출방법: 실습#1의 제출방법과 동일 (주의사항 참조)