스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명

Console을 이름으로 불러서 쓸꺼면 위에 캐스팅 방식을 사용하고 -> @component(“console”)

이름으로 부르지 않고 ExamConsole.class라는 객체로 불러올거면 component에 이름을 넣어줄 필요가 없다. -> @component



여러 패키지를 추가하는 방법은 , 쉼표 찍고 이어서 쓰면 된다.

16강 정리

스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷, 모니터, 녹색, 화면이(가) 표시된 사진

자동 생성된 설명

Component로 붙이나 controller나 service나 repository를 붙여도 상관은 없으나

이름을 붙여줘서 명시하는 것이 어노테이션 처리기의 목적이다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

17강 정리 XML configuration을 JAVA configuration으로 변경하기

스크린샷이(가) 표시된 사진

자동 생성된 설명

Xml을 java파일로 변경한 캡처

컴포넌트스캔 할 때 여러 개 하고싶으면 -> @componentscan({“xx”,”yy”}) 중괄호로 묶어주면됨.

스크린샷이(가) 표시된 사진

자동 생성된 설명

어노테이션으로 메인 작성방법이 2가지가 있고 편한걸로 하면됨.

18강 정리 AOP란?

스프링으로도 구현이 가능하지만 JAVA로도 구현가능한 방법론임, 관점

사용자 -> 주 업무 로직

개발자 운영자 -> 주 업무 로직이 아닌 것 (로그처리, 보안처리, 트랜잭션처리)

EX) 빵또아 같은 느낌 빵은 부 업무 속은 주 업무

부 업무는 뺏다 꼈다 할 수 있어야 한다.

모니터, 시계, 검은색, 측정기이(가) 표시된 사진

자동 생성된 설명스크린샷, 모니터, 화면, 디스플레이이(가) 표시된 사진

자동 생성된 설명스크린샷, 시계, 모니터이(가) 표시된 사진

자동 생성된 설명

19강 AOP 자바 코드 이해하기

스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명그리기이(가) 표시된 사진

자동 생성된 설명

20강 정리 – 순수 자바로 AOP 구현해보기

스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

곁다리 업무 사용시 proxy와invoke메소드 사용

21강 – 스프링으로 AOP 구현해보기

스크린샷, 모니터, 검은색이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명

22강 AOP – beforeadvice 구현하기

스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명

Setting.xml에 빈으로 추가해주고 spring.aop.advice에 클래스로 추가해서 사용하면 됨

사용방법은 위에 사진처럼 하면된다.

Aroundadvice도 같은 맥락이다.

23강 AOP – After Returning / Throwing 어드바이스 구현하기

스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명

처음에 before가 실행되고 다음으로 afterreturning이나 throwing으로 가는데

예외발생이 된다면 throwing으로 아니면 afterreturning으로 간다.

스크린샷이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명

코드 작성방법은 기존것들과 비슷함.