

초격차 패키지 Online.

API 설계

PART1 | 애노테이션 기반 설계

@Controller, @RestController

PART2 | 함수 기반 설계

함수형 프로그래밍으로 만드는 API

PART3 | 요청, 응답의 설계

Handler Methods의 활용

PART4 | @ControllerAdvice

공통 에러 응답을 설계하는 법

PART5 | 컨트롤러 테스트

Spring Web API를 테스트하는 방법

PART6 | TDD 방식으로 복습하기

테스트 주도 개발 과정 맛보기

API 설계

5 컨트롤러 테스트

스프링 부트 테스트

5. 컨트롤러 테스트

```
package com.uno.getinline;

import org.junit.jupiter.api.Test;
import org.springframework.boot.test.context.SpringBootTest;

@SpringBootTest
class GetInlineApplicationTests {

    @Test
    void contextLoads() {
    }

}
```

스프링 부트 테스트

5. 컨트롤러 테스트

@SpringBootTest 기본 구성

일단 스프링 컨테이너와 스프링 부트 환경을 포함하는 테스트는 이것으로 다 작성 가능

- 통합 테스트를 할 때 적절한 선택
- 애플리케이션 컨텍스트를 로드하는데 시간이 걸리므로 테스트가 다소 무거운 편
- @SpringBootTest
 - 포함한 애노테이션: @BootstrapWith + @ExtendWith
 - 흔한 실수: JUnit 5 쓸 때 @ExtendWith(SpringExtension.class) 쓰지 않기
- @Test
 - JUnit 5 애노테이션
 - 단위 테스트 메소드를 지정

스프링 부트 테스트

5. 컨트롤러 테스트

@SpringBootTest

```
@SpringBootTest(  
    properties = {  
        "server.error.whitelabel.enabled=false",  
        "debug=false"  
    },  
    args = "--my-argument=true",  
    classes = GetInLineApplication.class,  
    webEnvironment = SpringBootTest.WebEnvironment.MOCK  
)
```

@SpringBootTest

- value, properties: 프로퍼티 설정
- args: 애플리케이션 실행 시 커맨드라인으로 입력하는 인자(옵션) 설정
- classes: ApplicationContext 로딩을 위한 설정 클래스를 직접 지정
- webEnvironment: ApplicationContext 의 웹 환경 설정
 - WebEnvironment.MOCK: mock servlet, embedded server 동작 x
 - @AutoConfigureMockMvc, @AutoConfigureWebTestClient 와 함께 써서 mock test 가능
 - WebEnvironment.RANDOM_PORT: 랜덤 포트, embedded server 동작
 - WebEnvironment.DEFINED_PORT: 포트 지정(server.port), embedded server 동작
 - WebEnvironment.NONE: 웹 환경 구성 안 함, embedded server 동작 x

스프링 부트 테스트: Slice Test

5. 컨트롤러 테스트

Auto-configured Test (Slice Test)

스프링 애플리케이션에서 내가 필요한 일부분(slice)의 자동 설정만 불러오는 방법

- @DataCassandraTest
- @DataJdbcTest
- @DataJpaTest
- @DataLdapTest
- @DataMongoTest
- @DataNeo4jTest
- @DataR2dbcTest
- @DataRedisTest
- @JdbcTest
- @JooqTest
- @JsonTest
- @RestClientTest
- @WebFluxTest
- @WebMvcTest
- @WebServiceClientTest

스프링 부트 테스트: @WebMvcTest

5. 컨트롤러 테스트

@WebMvcTest

Spring MVC 컨트롤러 레이어를 슬라이스 테스트할 때 사용

- MockMvc 빈을 자동 설정하고 테스트에 사용
- 로드할 컨트롤러 클래스를 지정 가능 (기본 동작: 전체 컨트롤러 로드)

Reference

5. 컨트롤러 테스트

- <https://docs.spring.io/spring-boot/docs/current/reference/html/features.html#features.testing.spring-boot-applications>
- <https://docs.spring.io/spring-boot/docs/current/reference/html/test-auto-configuration.html#test-auto-configuration>
- <https://github.com/json-path/JsonPath>