개요

- JUnit Platform : 테스트를 실행해주는 런처 제공, TestEngine API 제공
- Jupiter: TestEngine API 구현체로 JUnit 5를 제공
- Vintage: JUnit4와 3을 지원하는 TestEngine 구현체

기초

- @Test
- @BeforeAll
- @AfterAll
- @BeforeEach
- @AfterEach
- @Disabled

테스트 이름 표기하기

- 일반적으로 테스트 메소드명은 camel case가 아닌 '_'로 한다.
- @DisplayNameGeneration(DisplayNameGenerator.ReplaceUnderscores.class)
  - 클래스명에 붙는 애노테이션
  - '_'를 공백으로 치환
- @DisplayName(“스터디 만들기 \uD83D\uDE31”) - 권장
  - 메소드명에 붙는 애노테이션
  - 한글, 이모지 사용 가능

Assertion

테스트 케이스	테스트 방법
실제 값이 기대한 값과 같은지 확인	assertEquals(expected, actual)
값이 null이 아닌지 확인	assertNotNull(actual)
다음 조건이 참(true)인지 확인	assertTure(boolean)
모든 확인 구문 확인	assertAll(executables...)
예외 발생 확인	assertThrows(expectedType, executable
특정 시간 안에 실행이 완료되는지 확인	assertTimeout(duration, executable)

대안으로 AssertJ, Hemcrest, TGruth라이브러리도 사용 가능

Ref

- 더 자바, "코드를 테스트 하는 다양한 방법" 강의 노트
- JUNIT5 사용자 가이드(공홈)

주레피, junit5

From:
http://rwiki.repia.com/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://rwiki.repia.com/doku.php?id=wiki:java:junit:junit5&rev=1651027910

Last update: **2022/04/27 11:51**