@RepeatedTest example

This lesson demonstrates how to use @RepeatedTest annotation to run tests multiple times.

@Repeated Tests

JUnit 5 Jupiter provides the ability to repeat a test a specified number of times. It is done by simply annotating a method with `@RepeatedTest` . To this annotation, we specify a number which is the total number of repetitions desired. For each and every invocation of a repeated test, it behaves like the execution of a regular `@Test` method. Also, each test is executed with support for the same lifecycle callbacks and extensions. Instead of using `@Test` annotation, we use `@RepeatedTest` annotation.

Let's take a look at a demo.

```java
package io.educative.junit5;

import static org.junit.jupiter.api.Assertions.assertTrue;

import org.junit.jupiter.api.RepeatedTest;

public class RepeatedTestDemo {

        @RepeatedTest(5)
        public void simpleRepeatedTest() {
                assertTrue(0 < 5);
        }

}
```
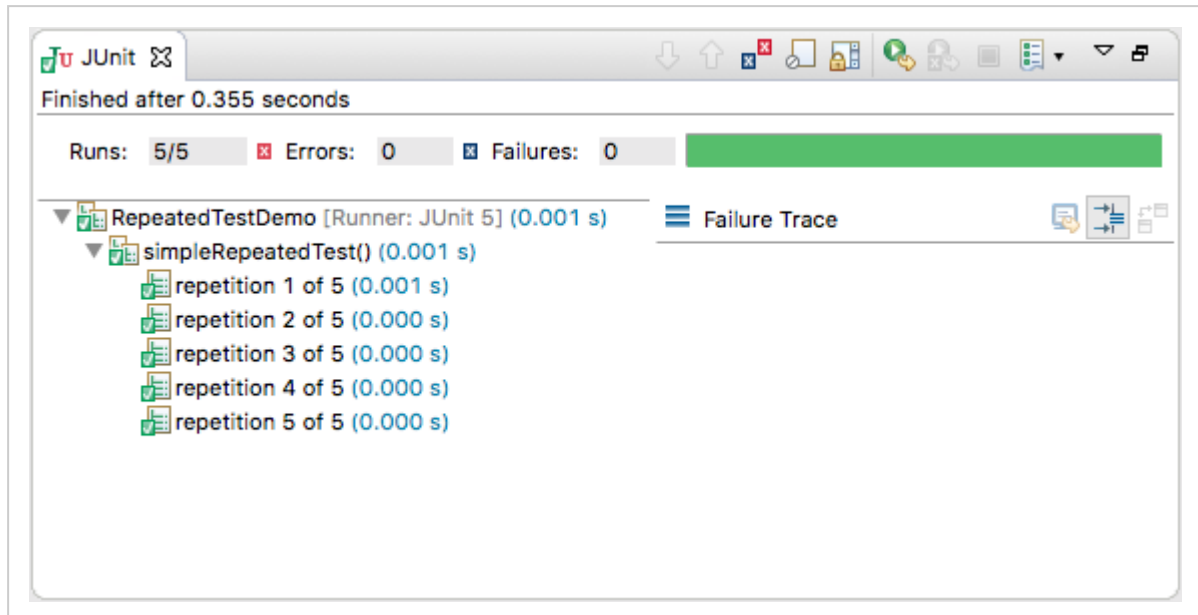
Explanation

On running `RepeatedTestDemo.java` as JUnit Test case, the output generated is demonstrated in the above image.

`@RepeatedTest` annotation takes in a value of type integer. This integer value is the count of the number of times the test case will be executed. The output image above tells us the format in which it is run -

`"repetition "` + `{currentRepetition}` + " of " + `{totalRepetitions}`;

where `currentRepetition` is the current repetition count and `totalRepetitions` is the total number of times the test case will repeat its execution.

In the next lesson, we will discuss about `@RepeatedTest` with `DisplayName`.