

Phase-3 Practice Project: Assisted Practice

4. Write a program to demonstrate nested and repeated tests

pom.xml:

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
<modelVersion>4.0.0</modelVersion>
```

```
<groupId>UsingJUnit</groupId>
```

```
<artifactId>UsingJUnit</artifactId>
```

```
<version>0.0.1-SNAPSHOT</version>
```

```
<dependencies>
```

```
<dependency>
```

```
<groupId>org.junit.jupiter</groupId>
```

```
<artifactId>junit-jupiter-api</artifactId>
```

```
<version>5.7.2</version>
```

```
<scope>test</scope>
```

```
</dependency>
```

```
<dependency>
```

```
<groupId>org.junit.jupiter</groupId>
```

```
<artifactId>junit-jupiter-engine</artifactId>
```

```
<version>5.7.2</version>
```

```
</dependency>
```

```
<dependency>
```

```
<groupId>org.junit.platform</groupId>
```

```
<artifactId>junit-platform-launcher</artifactId>
```

```
<version>1.2.0</version>
```

```
</dependency>
```

```
</dependencies>
```

```
<build>
```

```
<plugins>
```

```
<plugin>
```

```
<groupId>org.apache.maven.plugins</groupId>
```

```
<artifactId>maven-compiler-plugin</artifactId>
```

```
<version>3.8.1</version>
```

```
<configuration>
```

```
<source>8</source>
```

```
<target>8</target>
```

```
</configuration>
```

```
</plugin>
```

```
</plugins>
</build>
```

```
</project>
```

RepeatedTests.java:

```
package com.ecommerce.tests;
```

```
import org.junit.jupiter.api.AfterAll;
import org.junit.jupiter.api.AfterEach; import org.junit.jupiter.api.Assertions; import
org.junit.jupiter.api.BeforeAll;
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.DisplayName; import org.junit.jupiter.api.RepeatedTest; import
org.junit.jupiter.api.Test;
import org.junit.jupiter.api.TestInfo;
```

```
@DisplayName("JUnit 5 Repeated Tests Example") public class RepeatedTests {
@BeforeAll
public static void init(){
System.out.println("Before All init() method called");
}
```

```
@BeforeEach
public void initEach(){
System.out.println("Before Each initEach() method called");
}
```

```
@Test
@DisplayName("Add operation test") @RepeatedTest(5)
void addNumber(TestInfo testInfo) { Calculator calculator = new Calculator();
Assertions.assertEquals(2, calculator.add(1, 1), "1 + 1 should equal 2");
System.out.println("===addNumber testcase executed===");
}
```

```
@AfterEach
public void cleanUpEach(){
System.out.println("After Each cleanUpEach() method called");
}
```

```
@AfterAll
public static void cleanUp(){

    System.out.println("After All cleanUp() method called");
}
}
```

Calculator.java:
package com.ecommerce.tests;

```
public class Calculator
{

    public int add(int a, int b) { return a + b;
    }
}
```

NestedCases.java:
package com.ecommerce.tests;

```
import org.junit.jupiter.api.AfterAll;
import org.junit.jupiter.api.AfterEach; import org.junit.jupiter.api.BeforeAll;
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.DisplayName; import org.junit.jupiter.api.Nested;
import org.junit.jupiter.api.Test;
```

```
@DisplayName("JUnit 5 Nested Example") public class NestedCases {
```

```
@BeforeAll
static void beforeAll() {
    System.out.println("Before all test methods");

}
```

```
@BeforeEach
void beforeEach() {
    System.out.println("Before each test method");
}
```

```
@AfterEach
void afterEach() {
    System.out.println("After each test method");
}
```

```
@AfterAll
static void afterAll() {
    System.out.println("After all test methods");
}
```

```
@Nested
@DisplayName("Tests for the method A") class A {
```

```
@BeforeEach

void beforeEach() {
    System.out.println("Before each test method of the A class");
}
```

```
@AfterEach
void afterEach() {
    System.out.println("After each test method of the A class");
}
```

```
@Test
@DisplayName("Example test for method A") void sampleTestMethodA() {
    System.out.println("Example test for method A");
}
```

```
@Nested
@DisplayName("When X is true") class WhenX {

    @BeforeEach
    void beforeEach() {
        System.out.println("Before each test method of the WhenX class");
    }

    @AfterEach
    void afterEach() {
        System.out.println("After each test method of the WhenX class");
    }

    @Test
    @DisplayName("Example test for method A when X is true") void
    sampleTestForMethodAWhenX() {
        System.out.println("Example test for method A when X is true");
    }
}
}
```

Output:

<div><div>JUnit 5 Repeated Tests Example [Runner: JUnit 5]</div><div><div>Add operation test (0.003 s)</div><div>> Add operation test (0.000 s)</div></div><div>JUnit 5 Nested Example [Runner: JUnit 5] (0.010 s)</div><div><div>Tests for the method A (0.010 s)</div></div></div>	<div>Jun 15, 2023 7:24:23 PM org.junit.jupiter.engine.discovery.Met</div> <div>WARNING: Possible configuration error: method [void com.ecomme</div> <div>Before All init() method called</div> <div>Before Each initEach() method called</div> <div>===addNumber testcase executed===</div> <div>After Each cleanUpEach() method called</div> <div>Before Each initEach() method called</div> <div>===addNumber testcase executed===</div> <div>After Each cleanUpEach() method called</div> <div>Before Each initEach() method called</div> <div>===addNumber testcase executed===</div> <div>After Each cleanUpEach() method called</div> <div>Before Each initEach() method called</div> <div>===addNumber testcase executed===</div> <div>After Each cleanUpEach() method called</div> <div>Before Each initEach() method called</div> <div>===addNumber testcase executed===</div> <div>After Each cleanUpEach() method called</div> <div>Before Each initEach() method called</div> <div>===addNumber testcase executed===</div> <div>After Each cleanUpEach() method called</div> <div>After All cleanUp() method called</div> <div>Before all test methods</div> <div>Before each test method</div> <div>Before each test method of the A class</div> <div>Example test for method A</div> <div>After each test method of the A class</div> <div>After each test method</div> <div>Before each test method</div> <div>Before each test method of the A class</div> <div>Before each test method of the WhenX class</div> <div>Example test for method A when X is true</div> <div>After each test method of the WhenX class</div> <div>After each test method of the A class</div> <div>After each test method</div> <div>After all test methods</div>
<div>Failure Trace</div>	