Exercise 4: Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and Teardown methods in Junit

MathUtil.java:

**public** **class** MathUtil {

**public** **int** add(**int** a, **int** b) {

**return** a + b;

}

**public** **boolean** isPositive(**int** num) {

**return** num > 0;

}

**public** String getNullValue() {

**return** **null**;

}

}

MathUtilTest.java

**import** org.junit.jupiter.api.Test;

**import** **static** org.junit.jupiter.api.Assertions.\*;

**import** org.junit.jupiter.api.BeforeEach;

**import** org.junit.jupiter.api.AfterEach;

**public** **class** MathUtilTest {

**private** MathUtil util;

@BeforeEach

**public** **void** setUp() {

System.***out***.println("Setting up...");

util = **new** MathUtil();

}

@AfterEach

**public** **void** tearDown() {

System.***out***.println("Cleaning up...");

util = **null**;

}

@Test

**public** **void** testAddition() {

**int** res =util.add(2, 3);

*assertEquals*(5, res );

}

@Test

**public** **void** testIsPositive() {

*assertTrue*(util.isPositive(5));

*assertFalse*(util.isPositive(-2));

}

@Test

**public** **void** testNullChecks() {

*assertNull*(util.getNullValue());

*assertNotNull*(**new** Object());

}

}

Output:

