Java Unit Testing Assignments - JUnit

# Exercise 1: Setting Up JUnit

Scenario:  
You need to set up JUnit in your Java project to start writing unit tests.  
  
Steps:  
1. Create a new Java project in your IDE (e.g., IntelliJ IDEA, Eclipse, or VS Code).  
2. Add JUnit dependency to your project.  
 - If using Maven, add the following to your pom.xml:  
  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>4.13.2</version>  
 <scope>test</scope>  
 </dependency>  
  
3. Create a new test class and write your first test.

✔️ Answer:

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

import org.junit.Test;  
import static org.junit.Assert.\*;  
  
public class CalculatorTest {  
 @Test  
 public void testAdd() {  
 Calculator calc = new Calculator();  
 assertEquals(5, calc.add(2, 3));  
 }  
}

# Exercise 3: Assertions in JUnit

Scenario:  
You need to use different assertions in JUnit to validate your test results.  
  
Steps:  
1. Use various assertions like assertEquals, assertTrue, assertFalse, assertNull, assertNotNull.

✔️ Answer:

import org.junit.Test;  
import static org.junit.Assert.\*;  
  
public class AssertionsTest {  
  
 @Test  
 public void testAssertions() {  
 assertEquals(5, 2 + 3);  
 assertTrue(5 > 3);  
 assertFalse(5 < 3);  
 assertNull(null);  
 assertNotNull(new Object());  
 }  
}

# Exercise 4: Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and Teardown Methods in JUnit

Scenario:  
You need to organize your tests using the Arrange-Act-Assert (AAA) pattern and use setup and teardown methods.  
  
Steps:  
1. Use the @Before method for setup.  
2. Use the @After method for cleanup.  
3. Apply AAA (Arrange, Act, Assert) in test methods.

✔️ Answer:

import org.junit.Before;  
import org.junit.After;  
import org.junit.Test;  
import static org.junit.Assert.\*;  
  
public class CalculatorTest {  
  
 private Calculator calculator;  
  
 @Before  
 public void setUp() {  
 calculator = new Calculator(); // Arrange  
 }  
  
 @Test  
 public void testAddition() {  
 int result = calculator.add(2, 3); // Act  
 assertEquals(5, result); // Assert  
 }  
  
 @After  
 public void tearDown() {  
 calculator = null;  
 }  
}