Exercise 1: Control Structures

Code:

Pom.xml

<dependencies>  
 <!-- JUnit 5 -->  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter</artifactId>  
 <version>5.10.2</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>jakarta.persistence</groupId>  
 <artifactId>jakarta.persistence-api</artifactId>  
 <version>3.1.0</version>  
 </dependency>  
 <!-- Mockito -->  
 <dependency>  
 <groupId>org.mockito</groupId>  
 <artifactId>mockito-core</artifactId>  
 <version>5.12.0</version>  
 <scope>test</scope>  
 </dependency>  
  
 <!-- For Parameterized Tests -->  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter-params</artifactId>  
 <version>5.10.2</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>org.testng</groupId>  
 <artifactId>testng</artifactId>  
 <version>RELEASE</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>RELEASE</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>org.junit.platform</groupId>  
 <artifactId>junit-platform-suite</artifactId>  
 <version>1.10.2</version> <!-- Use a version that matches your JUnit 5 setup -->  
 <scope>test</scope>  
 </dependency>  
  
</dependencies>

Calculator.java

package org.example;  
  
public class Calculator {  
 public int add(int a, int b) {  
 return a + b;  
 }  
  
 public int subtract(int a, int b) {  
 return a - b;  
 }  
  
 public int multiply(int a, int b) {  
 return a \* b;  
 }  
  
 public int divide(int a, int b) {  
 if (b == 0) throw new ArithmeticException("Cannot divide by zero");  
 return a / b;  
 }  
}

CalculatorTest.java

package com.example;  
  
import org.example.Calculator;  
import org.junit.Test;  
import static org.junit.Assert.\*;  
import static org.junit.jupiter.api.Assertions.*assertEquals*;  
  
public class CalculatorTest {  
  
 Calculator calc = new Calculator();  
  
 @Test  
 public void testAdd() {  
 *assertEquals*(10, calc.add(6, 4));  
 }  
  
 @Test  
 public void testSubtract() {  
 *assertEquals*(2, calc.subtract(5, 3));  
 }  
  
 @Test  
 public void testMultiply() {  
 *assertEquals*(20, calc.multiply(4, 5));  
 }  
  
 @Test  
 public void testDivide() {  
 *assertEquals*(3, calc.divide(9, 3));  
 }  
  
 @Test(expected = ArithmeticException.class)  
 public void testDivideByZero() {  
 calc.divide(10, 0);  
 }  
}

Output:

