Superset ID: 6416838

Week 2

# Advanced JUnit Testing Exercises

**Exercise 1: Parameterized Tests**

**Code:**

**EvenChecker.java**

public class EvenChecker {  
 public boolean isEven(int number) {  
 return number % 2 == 0;  
 }  
}

**EvenCheckerTest.java**

import static org.junit.jupiter.api.Assertions.\*;  
import org.junit.jupiter.params.ParameterizedTest;  
import org.junit.jupiter.params.provider.ValueSource;  
  
public class EvenCheckerTest {  
  
 EvenChecker checker = new EvenChecker();  
  
 @ParameterizedTest  
 @ValueSource(ints = {2, 4, 6, 8, 10})  
 void testEvenNumbers(int number) {  
 assertTrue(checker.isEven(number));  
 }  
  
 @ParameterizedTest  
 @ValueSource(ints = {1, 3, 5, 7, 9})  
 void testOddNumbers(int number) {  
 assertFalse(checker.isEven(number));  
 }  
}

**Output:**

testEvenNumbers(2)  
 testEvenNumbers(4)  
 testEvenNumbers(6)  
 testEvenNumbers(8)  
 testEvenNumbers(10)  
 testOddNumbers(1)  
 testOddNumbers(3)  
 testOddNumbers(5)  
 testOddNumbers(7)  
 testOddNumbers(9)

**Exercise 2: Test Suites and Categories**

**Code:**

**MathTest.java**

import static org.junit.jupiter.api.Assertions.\*;  
import org.junit.jupiter.api.Test;  
  
public class MathTest {  
 @Test  
 void testAdd() {  
 assertEquals(5, 2 + 3);  
 }  
}

**StringTest.java**

import static org.junit.jupiter.api.Assertions.\*;  
import org.junit.jupiter.api.Test;  
  
public class StringTest {  
 @Test  
 void testLength() {  
 assertEquals(4, "test".length());  
 }  
}

**AllTests.java**

import org.junit.platform.suite.api.SelectClasses;  
import org.junit.platform.suite.api.Suite;  
  
@Suite  
@SelectClasses({ MathTest.class, StringTest.class })  
public class AllTests {  
}

**Output:**

testAdd()  
 testLength()

**Exercise 3: Test Execution Order**

**Code:**

**OrderedTests.java**

import static org.junit.jupiter.api.Assertions.\*;  
import org.junit.jupiter.api.\*;  
  
@TestMethodOrder(MethodOrderer.OrderAnnotation.class)  
public class OrderedTests {  
  
 @Test  
 @Order(2)  
 void secondTest() {  
 System.out.println("Second test executed");  
 }  
  
 @Test  
 @Order(1)  
 void firstTest() {  
 System.out.println("First test executed");  
 }  
  
 @Test  
 @Order(3)  
 void thirdTest() {  
 System.out.println("Third test executed");  
 }  
}

**Output:**

First test executed  
Second test executed  
Third test executed

**Exercise 4: Exception Testing**

**Code:**

**ExceptionThrower.java**

public class ExceptionThrower {  
 public void throwException() {  
 throw new IllegalArgumentException("Invalid input");  
 }  
}

**ExceptionThrowerTest.java**

import org.junit.jupiter.api.Test;  
import static org.junit.jupiter.api.Assertions.\*;  
  
public class ExceptionThrowerTest {  
  
 ExceptionThrower thrower = new ExceptionThrower();  
  
 @Test  
 void testException() {  
 assertThrows(IllegalArgumentException.class, () -> {  
 thrower.throwException();  
 });  
 }  
}

**Output:**

testException()

**Exercise 5: Timeout and Performance Testing**

**Code:**

**PerformanceTester.java**

public class PerformanceTester {  
 public void performTask() throws InterruptedException {  
 Thread.sleep(500); // 0.5 second delay  
 }  
}

**PerformanceTesterTest.java**

import org.junit.jupiter.api.Test;  
import static org.junit.jupiter.api.Assertions.\*;  
  
import java.time.Duration;  
  
public class PerformanceTesterTest {  
  
 PerformanceTester tester = new PerformanceTester();  
  
 @Test  
 void testTimeout() {  
 assertTimeout(Duration.ofSeconds(1), () -> {  
 tester.performTask();  
 });  
 }  
}

**Output:**

testTimeout() (completed in < 1 second)