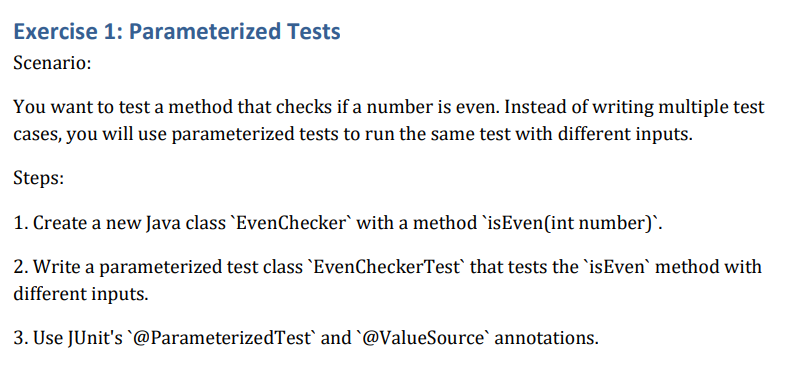
**Advanced JUnit Testing Exercises**

**Solution:**

**EvenChecker.java**

package com.utils;

public class EvenChecker {

public boolean isEven(int number) {

return number % 2 == 0;

}

}

**EvenCheckerTest.java**

package com.utils;

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

import org.junit.jupiter.params.ParameterizedTest;

import org.junit.jupiter.params.provider.ValueSource;

public class EvenCheckerTest {

@ParameterizedTest

@ValueSource(ints = {2, 4, 6, 8, 10})

void testEven(int number) {

EvenChecker checker = new EvenChecker();

assertTrue(checker.isEven(number));

}

@ParameterizedTest

@ValueSource(ints = {1, 3, 5, 7})

void testOdd(int number) {

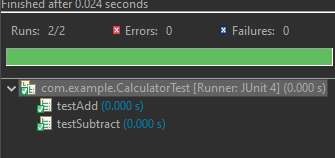
EvenChecker checker = new EvenChecker();

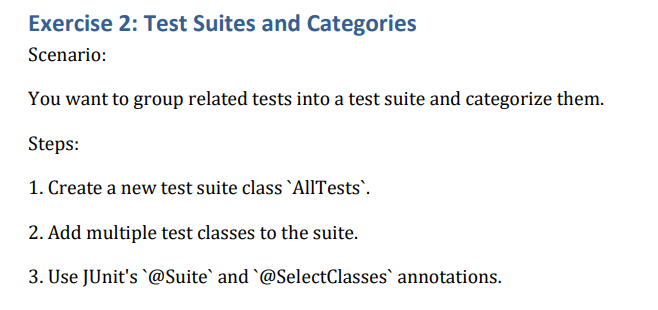
assertFalse(checker.isEven(number));

}

}

**Output:**





**Solution:**

package com.utils;

import org.junit.platform.suite.api.SelectClasses;

import org.junit.platform.suite.api.Suite;

@Suite

@SelectClasses({

EvenCheckerTest.class,

OrderedTests.class,

ExceptionThrowerTest.class,

PerformanceTesterTest.class

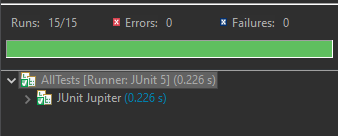
})

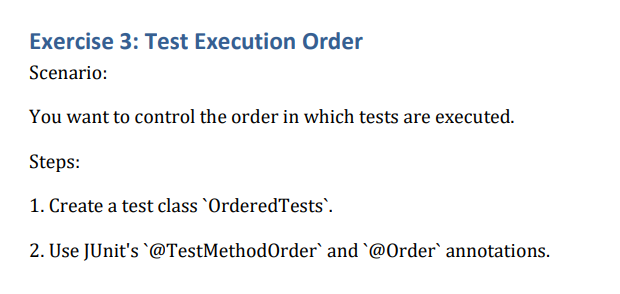
public class AllTests {

// This class runs all listed test classes together.

}

**Output:**





**Solution:**

package com.utils;

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

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

@TestMethodOrder(MethodOrderer.OrderAnnotation.class)

public class OrderedTests {

static StringBuilder tracker = new StringBuilder();

@Test

@Order(1)

void stepOne() {

tracker.append("A");

assertTrue(true);

}

@Test

@Order(2)

void stepTwo() {

tracker.append("B");

assertEquals("AB", tracker.toString());

}

@Test

@Order(3)

void stepThree() {

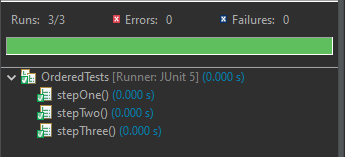
tracker.append("C");

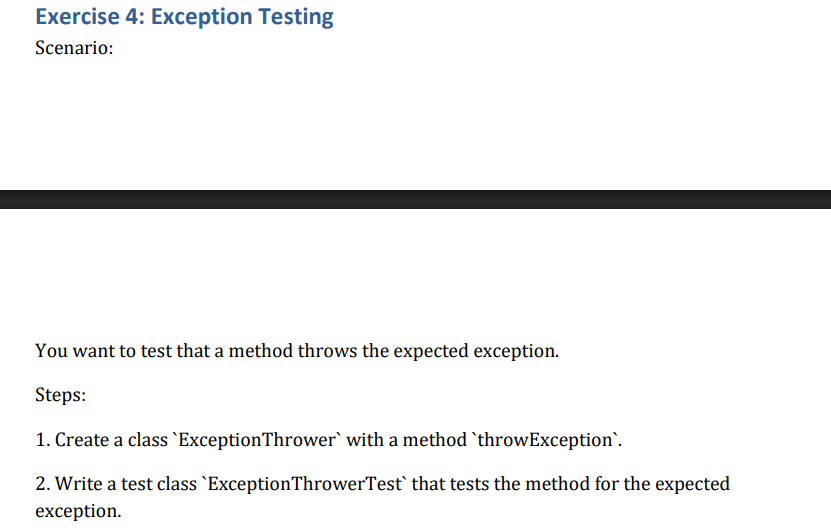
assertEquals("ABC", tracker.toString());

}

}

**Solution:**





**Solution:**

**ExceptionThrower.java**

package com.utils;

public class ExceptionThrower {

public void throwIfNegative(int number) {

if (number < 0) {

throw new IllegalArgumentException("Negative not allowed");

}

}

}

**ExceptionThrowerTest.java**

package com.utils;

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

import org.junit.jupiter.api.Test;

public class ExceptionThrowerTest {

@Test

void testThrowsForNegative() {

ExceptionThrower et = new ExceptionThrower();

assertThrows(IllegalArgumentException.class, () -> et.throwIfNegative(-1));

}

@Test

void testDoesNotThrowForPositive() {

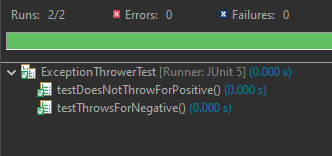
ExceptionThrower et = new ExceptionThrower();

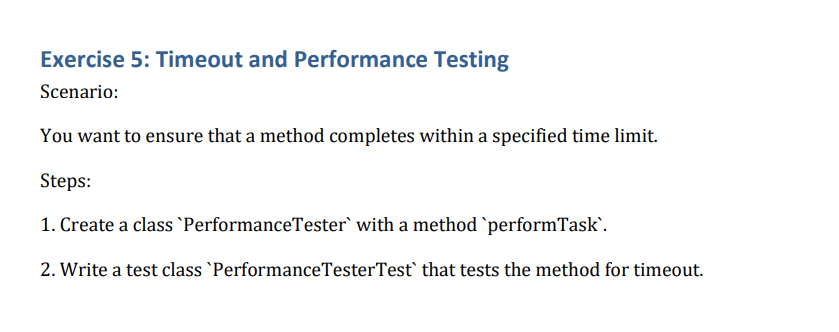
assertDoesNotThrow(() -> et.throwIfNegative(10));

}

}

**Output:**





**Solution:**

**PerformanceTester.java**

package com.utils;

public class PerformanceTester {

public void performTask() throws InterruptedException {

Thread.sleep(100); // Simulates delay

}

}

**PerformanceTesterThrower.java**

package com.utils;

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

import org.junit.jupiter.api.Test;

import java.time.Duration;

public class PerformanceTesterTest {

@Test

void testTaskCompletesOnTime() {

PerformanceTester pt = new PerformanceTester();

assertTimeout(Duration.ofMillis(200), () -> pt.performTask());

}

}

**Output:**

