JUNIT ADVANCED TESTING:

Exercise 1: Parameterized Tests

CODE:

EvenChecker.java:

package example;

public class EvenChecker {

public boolean isEven(int number) {

return number % 2 == 0;

}

}

EvenCheckerTest.java:

package example;

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

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

import org.junit.jupiter.params.ParameterizedTest;

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

public class EvenCheckerTest {

EvenChecker evenChecker = new EvenChecker();

@ParameterizedTest

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

void testIsEven\_ShouldReturnTrueForEvenNumbers(int number) {

*assertTrue*(evenChecker.isEven(number), number + " should be even");

}

@ParameterizedTest

@ValueSource(ints = {1, 3, 5, 7, 9})

void testIsEven\_ShouldReturnFalseForOddNumbers(int number) {

*assertFalse*(evenChecker.isEven(number), number + " should be odd");

}

}

OUTPUT:

