JUNIT TESTING EXERCISES

EXERCISE:1 SETTING UP JUNIT

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>org.example</groupId>

<artifactId>second</artifactId>

<version>1.0-SNAPSHOT</version>

<properties>

<maven.compiler.source>22</maven.compiler.source>

<maven.compiler.target>22</maven.compiler.target>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

</properties>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

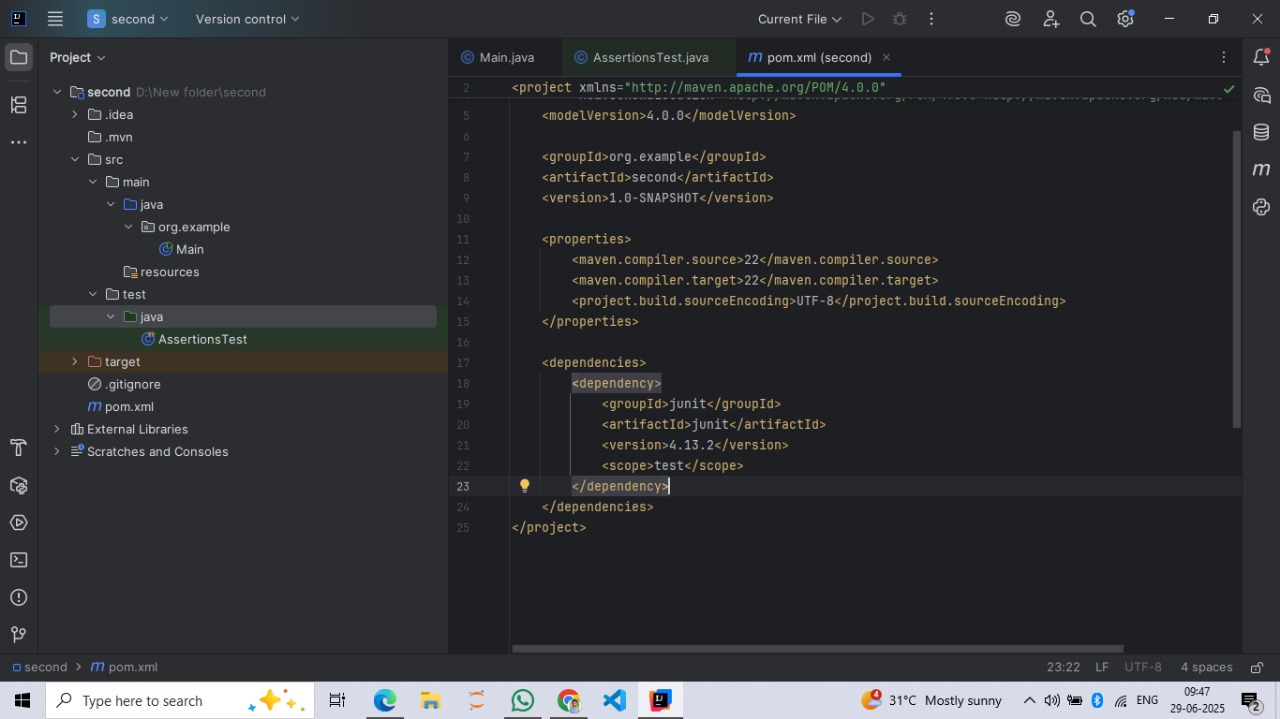
<scope>test</scope>

</dependency>

</dependencies>

</project>

OUTPUT:



EXERCISE 3: ASSERTIONS IN JUNIT

import org.example.Main;

import org.junit.\*;

import static org.junit.Assert.\*;

public class AssertionsTest {

int expected, actual;

@Before

public void before(){

System.out.println("This is before phase\nThe values are Arranged and Acted");

Main m = new Main();

int a=5,b=10;

expected=15;

actual = m.addition(a,b);

}

@Test

public void testAssertions() {

// Assert equals

assertEquals(expected, actual);

// Assert true

assertTrue(expected >= actual);

// Assert false

assertFalse(expected < actual);

// Assert null

assertNull(null);

// Assert not null

assertNotNull(new Object());

}

@After

public void after(){

System.out.println("\nThis is after phase");

System.out.println("Assert Equals: " + (expected==actual) +

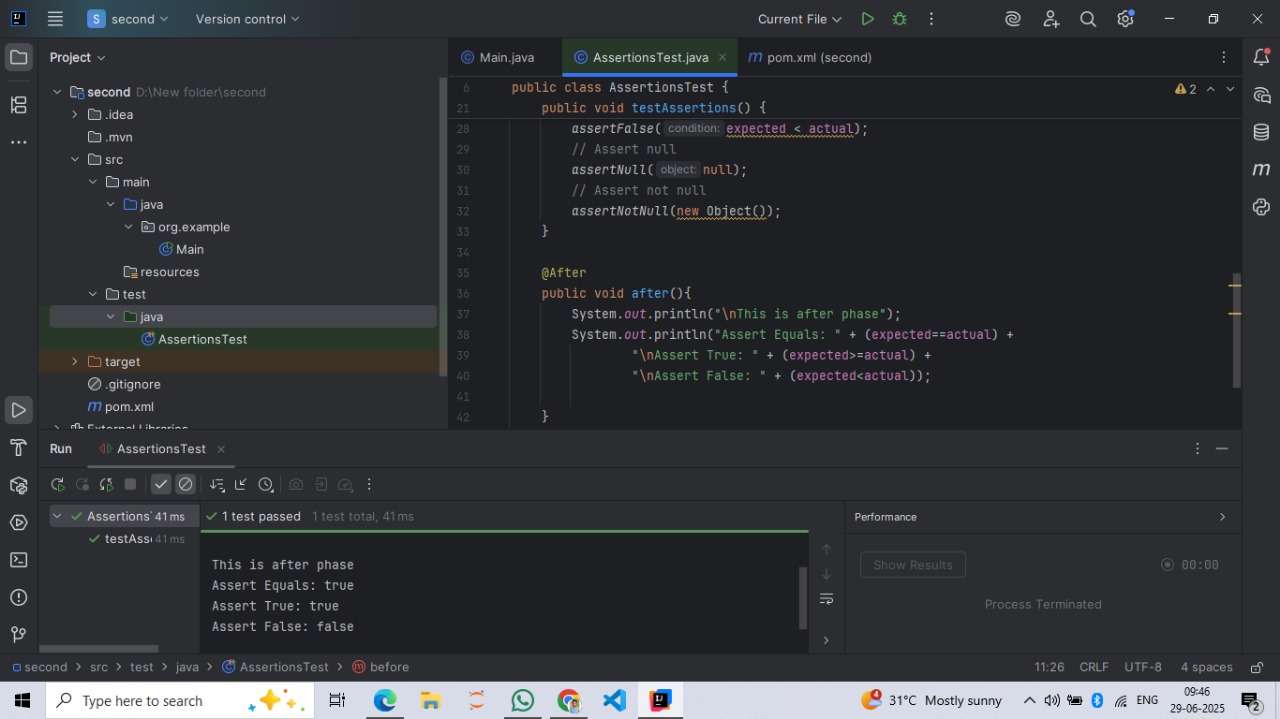
"\nAssert True: " + (expected>=actual) +

"\nAssert False: " + (expected<actual));

}

}

OUTPUT:



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

package org.example;

//TIP To <b>Run</b> code, press <shortcut actionId="Run"/> or

// click the <icon src="AllIcons.Actions.Execute"/> icon in the gutter.

public class Main {

public static void main(String[] args) {

System.out.println("Hello World!");

}

public int addition(int a, int b){

return a+b;

}

}

OUTPUT:

