Exercise 1

Setting Up JUnit

Scenario:

You need to set up JUnit in your Java project to start writing unit tests.

Steps:

1. Create a new Java project in your IDE (e.g., IntelliJ IDEA, Eclipse).

2. Add JUnit dependency to your project. If you are using Maven, add the following to your

pom.xml

src/main/java/com/example/MyClass.java

package com.example;

public class MyClass {

public int add(int a, int b) {

return a + b;

}

}

src/test/java/com/example/MyClassTest.java

package com.example;

import org.junit.Test;

import static org.junit.Assert.\*;

public class MyClassTest {

@Test

public void testAdd() {

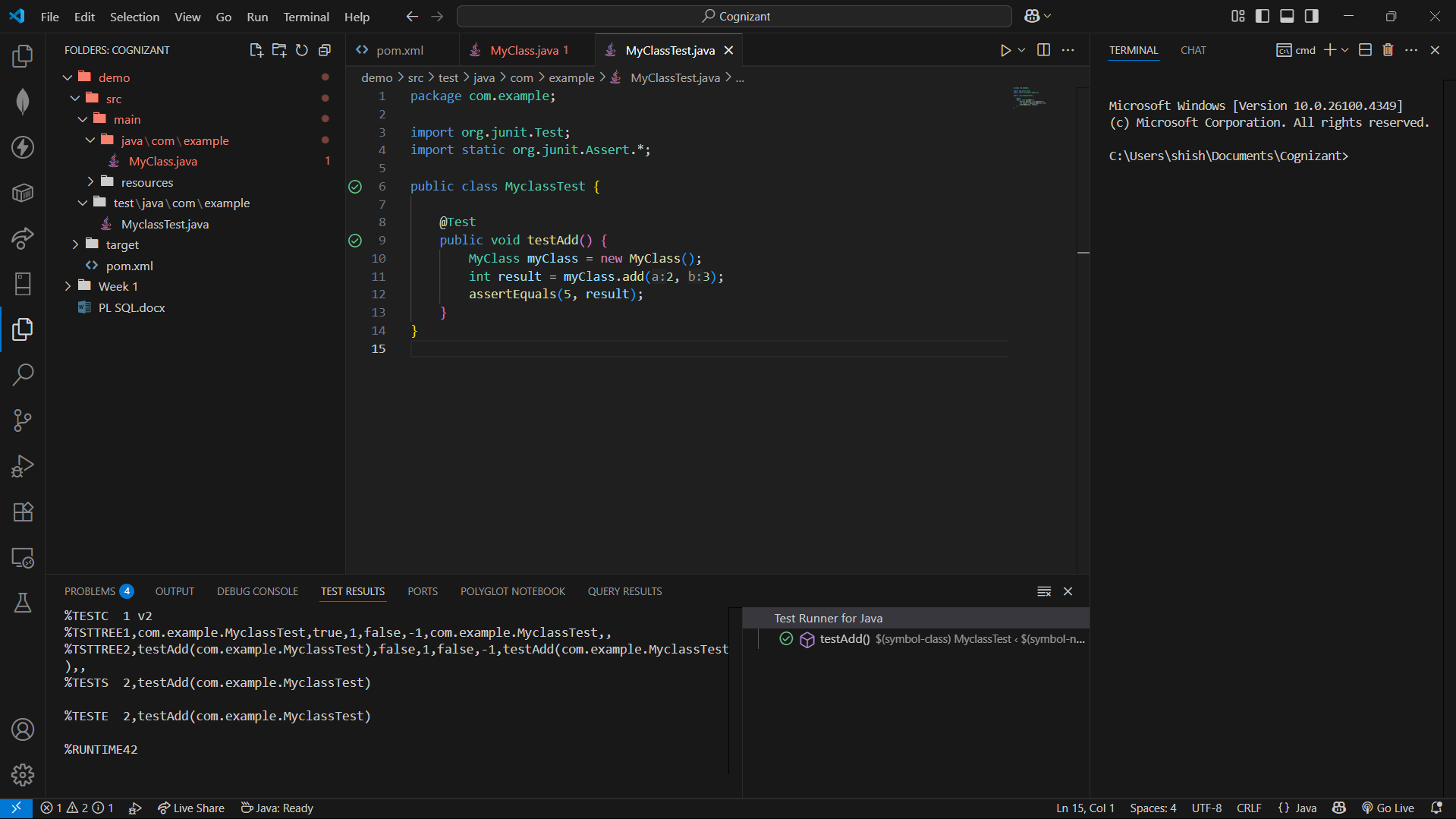
MyClass myClass = new MyClass();

int result = myClass.add(2, 3);

assertEquals(5, result);

}

}



Exercise 3

Assertions in JUnit

Scenario:

You need to use different assertions in JUnit to validate your test results.

Steps:

1. Write tests using various JUnit assertions.

AssertionsTest.java

package com.example;

import org.junit.Test;

import static org.junit.Assert.\*;

public class AssertionsTest {

    @Test

    public void testAssertions() {

        // Assert equals

        assertEquals(5, 2 + 3);

        // Assert true

        assertTrue(5 > 3);

        // Assert false

        assertFalse(5 < 3);

        // Assert null

        Object obj = null;

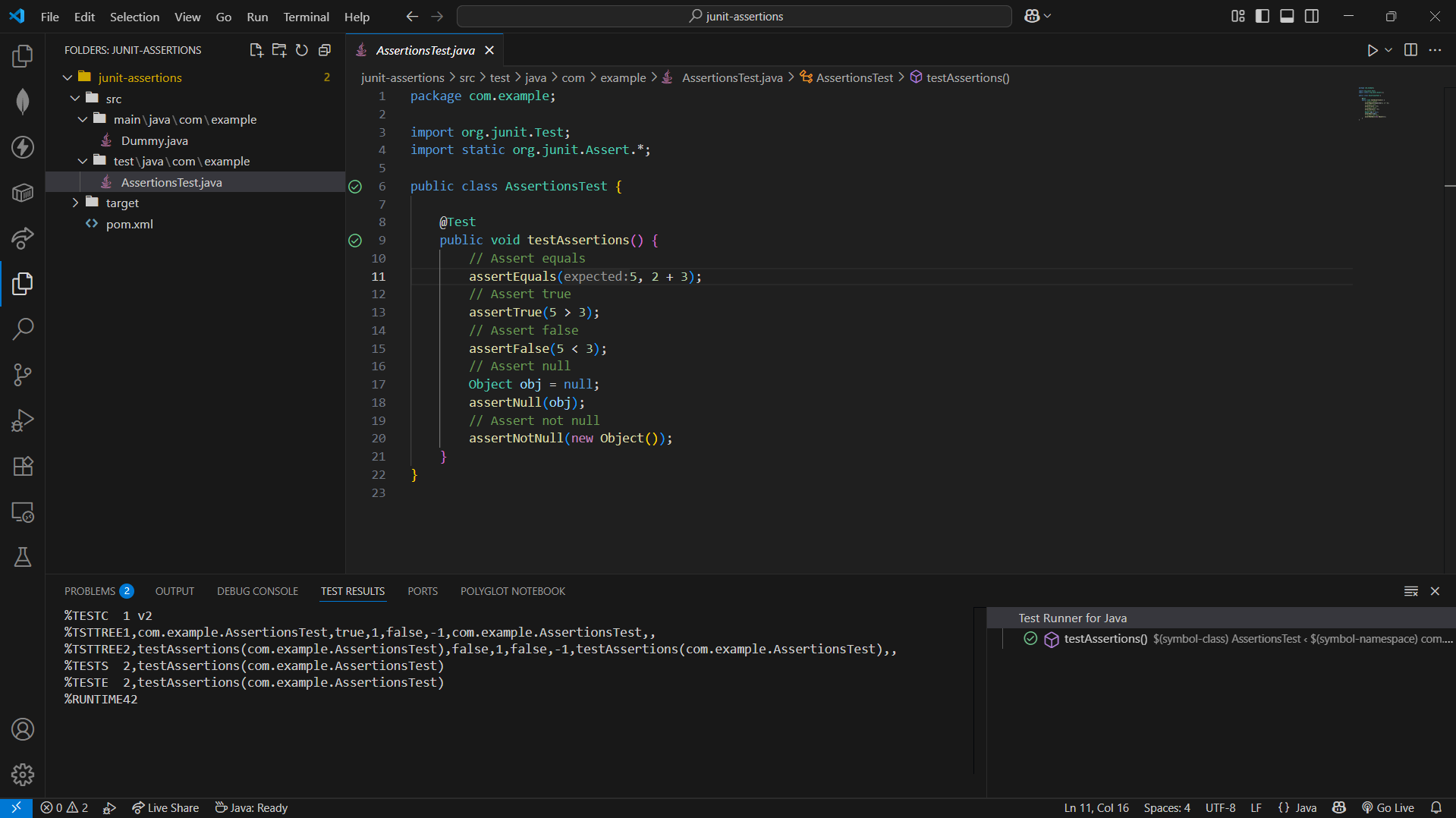
        assertNull(obj);

        // Assert not null

        assertNotNull(new Object());

    }

}



Exercise 4

Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and

Teardown Methods in JUnit

Scenario:

You need to organize your tests using the Arrange-Act-Assert (AAA) pattern and use setup

and teardown methods.

Steps:

1. Write tests using the AAA pattern.

2. Use @Before and @After annotations for setup and teardown methods.

Calculator.java

package com.example;

public class Calculator {

public int multiply(int a, int b) {

return a \* b;

}

public int subtract(int a, int b) {

return a - b;

}

}

CalculatorTest.java

package com.example;

import org.junit.Before;

import org.junit.After;

import org.junit.Test;

import static org.junit.Assert.\*;

public class CalculatorTest {

private Calculator calculator;

// Setup: runs before each test

@Before

public void setUp() {

calculator = new Calculator();

System.out.println("Setup: Calculator created");

}

// Teardown: runs after each test

@After

public void tearDown() {

calculator = null;

System.out.println("Teardown: Calculator destroyed");

}

@Test

public void testMultiply() {

// Arrange

int a = 4;

int b = 5;

// Act

int result = calculator.multiply(a, b);

// Assert

assertEquals(20, result);

}

@Test

public void testSubtract() {

// Arrange

int a = 10;

int b = 3;

// Act

int result = calculator.subtract(a, b);

// Assert

assertEquals(7, result);

}

}

