**Exercise 1: Setting Up JUnit**

* **Pom.xml file**

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

* **MainTest.java**

package org.example

public class MainTest {

}

**Exercise 3: Assertions in JUnit**

import org.junit.jupiter.api.Test;

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

public class AssertionsTest {

@Test

public void basicAssertions() {

assertEquals(5, 2 + 3);

assertTrue(5 > 3);

assertFalse(5 < 3);

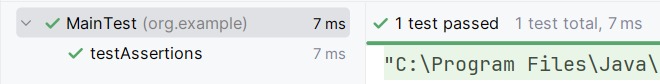
assertNull(null);

assertNotNull(new Object());

}

}

* Output



**Exercise 4: Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup andTeardown Methods in JUnit**

* Main.java

package org.example;

public class Main {

public int sum(int x, int y) {

return x + y;

}

public int difference(int x, int y) {

return x - y;

}

public static void main(String[] args) {

System.out.println("Hey there!");

}

}

* MainTest.java

package org.example;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

public class MainTest {

private Main mainApp;

@Before

public void init() {

mainApp = new Main();

System.out.println("Setup is ready");

}

@After

public void cleanup() {

mainApp = null;

System.out.println("Cleanup done");

}

@Test

public void checkSum() {

assertEquals(8, mainApp.sum(5, 3));

}

@Test

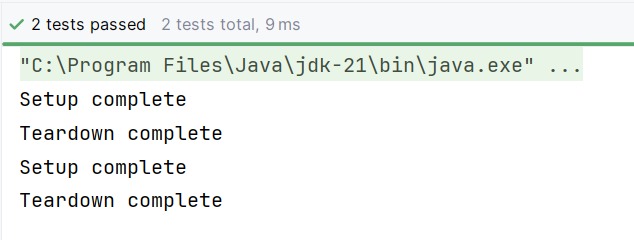
public void checkDifference() {

assertEquals(6, mainApp.difference(10, 4));

}

}

* **Output**

****

**Exercise 1: Mocking and Stubbing**

* **ForMock.java**

package org.example;

public class ForMock {

private final ExternalAPI externalAPI;

public ForMock(ExternalAPI externalAPI) {

this.externalAPI = externalAPI;

}

public int addLocally(int a, int b) {

return a + b;

}

public int addWithRemoteValue(int base) {

int fetched = externalAPI.fetchRemoteValue();

return base + fetched;

}

public int subtractValues(int a, int b) {

return a - b;

}

public static void main(String[] args) {

System.out.println("Hey there!");

}

}

* **ForMockTest.java**

package org.example;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.assertEquals;

import static org.mockito.Mockito.\*;

public class ForMockTest {

private ExternalAPI mockApi;

private ForMock forMock;

@Before

public void prepare() {

mockApi = mock(ExternalAPI.class);

when(mockApi.fetchRemoteValue()).thenReturn(5);

forMock = new ForMock(mockApi);

}

@Test

public void testAdditionWithRemoteValue() {

int result = forMock.addWithRemoteValue(10);

assertEquals(15, result); // 10 + mocked 5

}

}

* **Output**

WhatsApp Image 2025-06-29 at 10.02.19 AM.jpeg

**Exercise 2: Verifying Interactions**

* **ExternalApi2.java**

// ExternalApi2.java

package org.example;

public interface ExternalApi2 {

String getData();

}

* **MyService.java**

// MyService.java

package org.example;

public class MyService {

private final ExternalApi2 externalApi;

public MyService(ExternalApi2 externalApi) {

this.externalApi = externalApi;

}

public void callFetch() {

externalApi.getData();

}

}

* **Output**

WhatsApp Image 2025-06-29 at 10.02.45 AM.jpeg