**Exercise 1: Setting Up Junit**

**Main class**

package com.example;

public class App

{

    public static void main( String[] args )

    {

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

    }

}

**Test class**

package com.example;

import junit.framework.Test;

import junit.framework.TestCase;

import junit.framework.TestSuite;

public class AppTest

    extends TestCase

{

    public AppTest( String testName )

    {

        super( testName );

    }

    public static Test suite()

    {

        return new TestSuite( AppTest.class );

    }

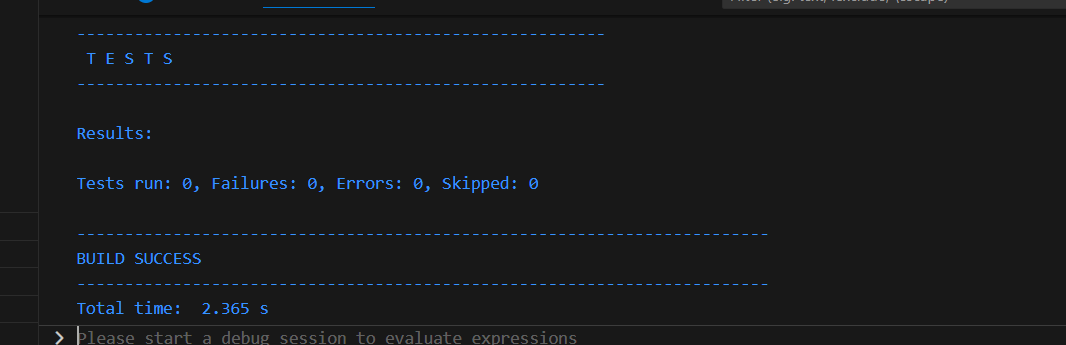
    public void testApp()

    {

        assertTrue( true );

    }}

**OUTPUT**



**Exercise 3: Assertions in Junit**

**Test class**

package com.example;

import static org.junit.Assert.assertEquals;

import static org.junit.Assert.assertFalse;

import static org.junit.Assert.assertNotNull;

import static org.junit.Assert.assertNull;

import static org.junit.Assert.assertTrue;

import org.junit.Test;

 public class AssertionsTest {

    @Test

    public void testAssertions() {

        assertEquals(5, 2 + 3);

        assertTrue(5 > 3);

        assertFalse(5 < 3);

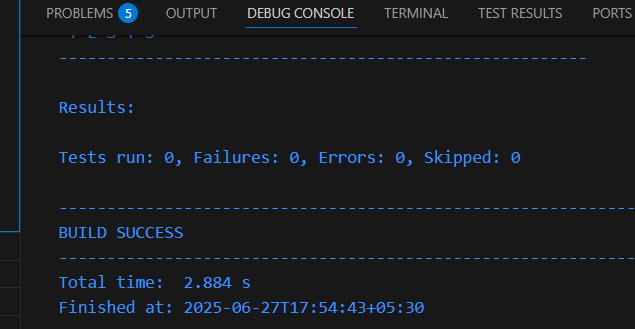
        assertNull(null);

        assertNotNull(new Object());

    }

}

**OUTPUT**



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

**Main class**

public class Calculator {

public int add(int a, int b) {

return a + b;

}

public int subtract(int a, int b) {

return a - b;

}

public int multiply(int a, int b) {

return a \* b;

}

public int divide(int a, int b) {

if (b == 0) {

throw new IllegalArgumentException("Cannot divide by zero");

}

return a / b;

}

}

**Test class**

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

public class CalculatorTest {

private Calculator calculator;

@Before

public void setUp() {

System.out.println("Setting up...");

calculator = new Calculator();

}

@After

public void tearDown() {

System.out.println("Cleaning up...");

calculator = null;

}

@Test

public void testAdd() {

// Arrange

int a = 5;

int b = 3;

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

assertEquals(8, result);

}

@Test

public void testSubtract() {

// Arrange

int a = 10;

int b = 4;

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

assertEquals(6, result);

}

@Test

public void testMultiply() {

// Arrange

int a = 7;

int b = 6;

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

assertEquals(42, result);

}

@Test

public void testDivide() {

// Arrange

int a = 20;

int b = 4;

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

assertEquals(5, result);

}

@Test(expected = IllegalArgumentException.class)

public void testDivideByZero() {

// Arrange

int a = 10;

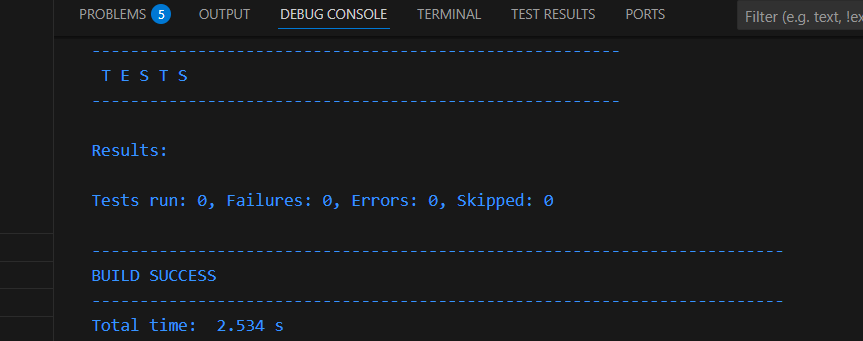
int b = 0;

calculator.divide(a, b);

}

}

**OUTPUT**

****