1. **Setting up Junit**

Sample Code(Main.java):

package com.example;

public class Main {

    public static void main(String[] args) {

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

    }

}

Dependency added in Pom.xml:

<dependency>

            <groupId>junit</groupId>

            <artifactId>junit</artifactId>

            <version>4.13.2</version>

            <scope>test</scope>

</dependency>

Testing Code(MainTest.java):

package com.example;

import org.junit.Test;

import static org.junit.Assert.\*;

public class MainTest {

    @Test

    public void testMainRunsWithoutException() {

        try {

            Main.main(new String[]{});  // run your main method

        } catch (Exception e) {

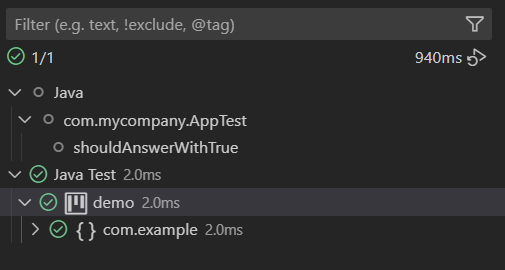
            fail("Main method threw an exception: " + e.getMessage());

        }

    }

}

Output:



1. **Assertions in Junit**

Code(AssertionTest.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

        assertNull(null);

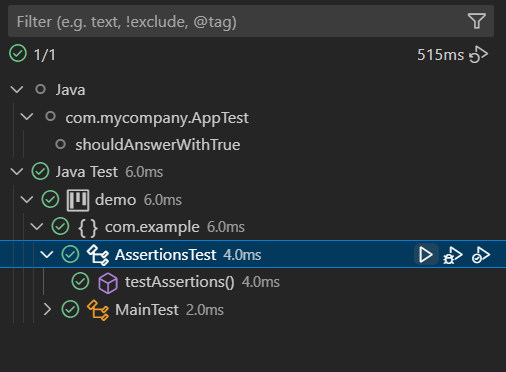
        // Assert not null

        assertNotNull(new Object());

    }

}

Output:



1. **Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and Teardown Methods in Junit**

Sample Code(Adder.java):

package com.example;

public class Adder {

    public int add(int a, int b) {

        return a + b;

    }

    public void close() {

        System.out.println("Adder closed.");

    }

}

Testing Code(AdderTest.java):

package com.example;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

public class AdderTest {

    private Adder calculator;

    @Before

    public void setUp() {

        calculator = new Adder();

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

    }

    @Test

    public void testAdd() {

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

        assertEquals(5, result);

    }

    @After

    public void tearDown() {

        calculator.close();

        System.out.println("Teardown: Calculator cleaned up.");

    }

}

Output:

