Junit Basic Testing Exercises

Exercise 1:SETTING UP JUNIT:

Scenario:

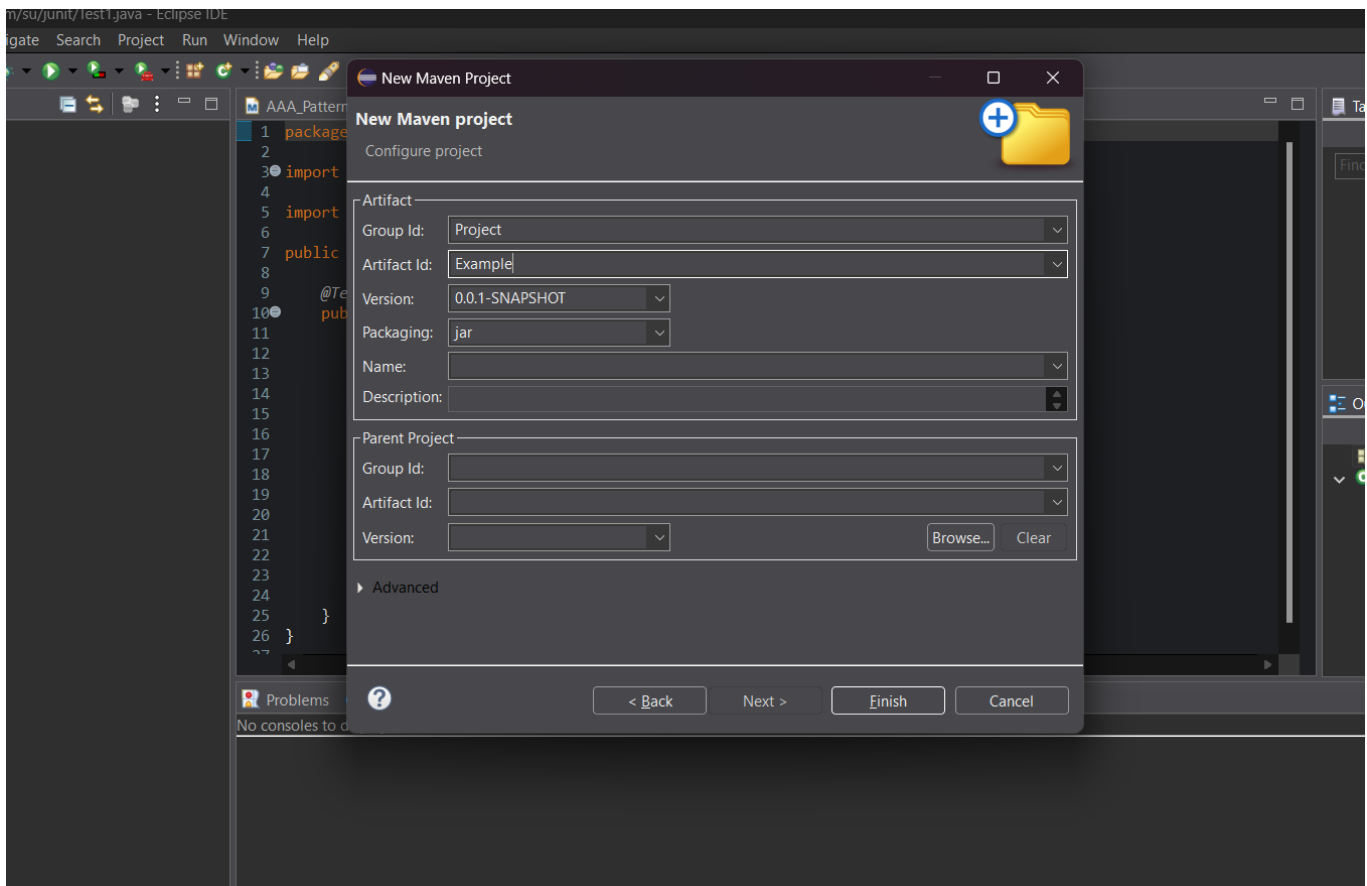**You need to set up JUnit in your Java project to start writing unit tests.**

SOLUTION:

Step 1. Create or Open a **Maven Project** :

　You can use an IDE like **IntelliJ IDEA**, **Eclipse**, or set it up manually.

Step 2. Click File-> New-> Maven Project(Click on create a simple project)

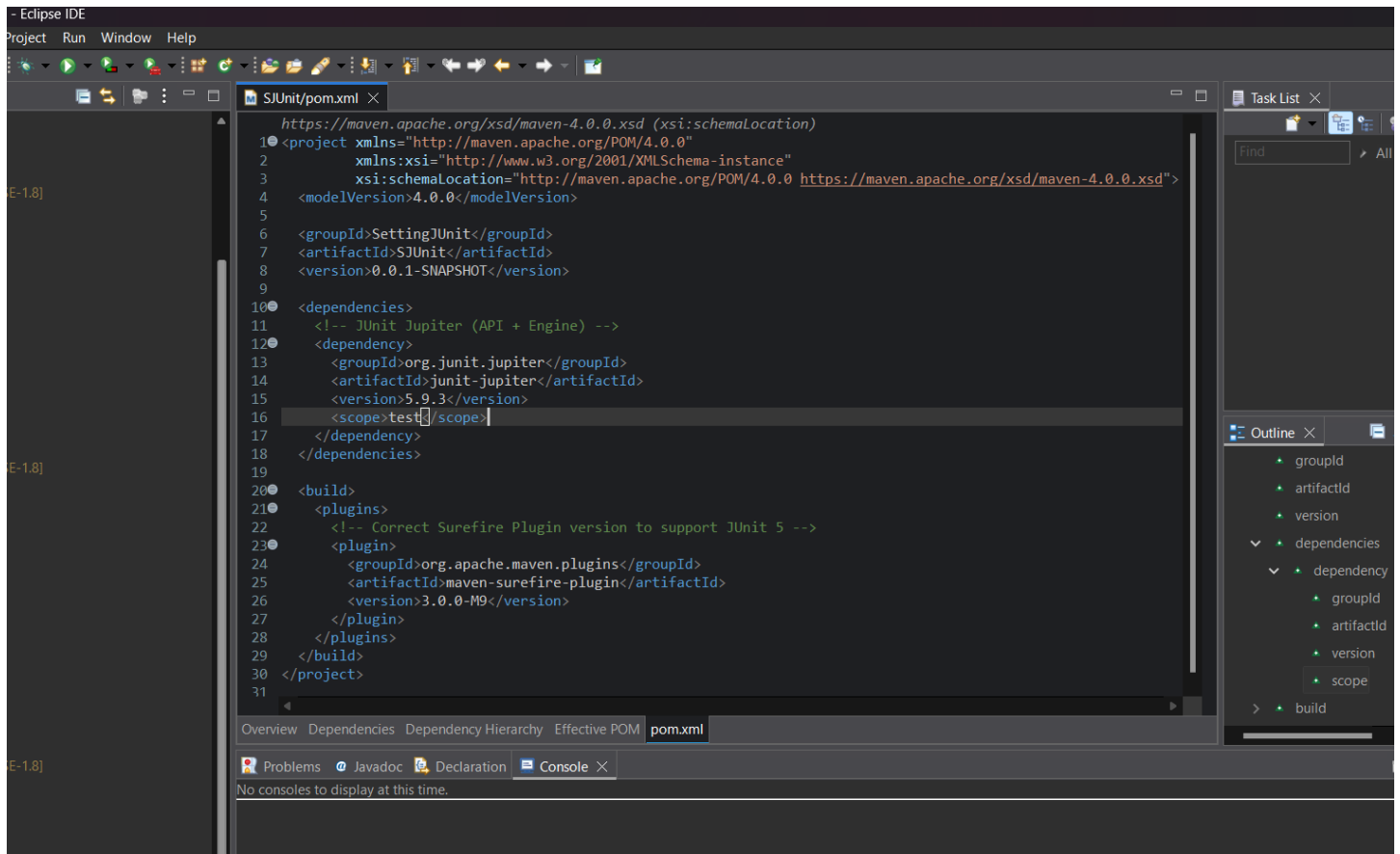Then write the name of group id and artifact id and then click on finish button.

Step 3. Add **Junit Dependency** to your project . Add following to your **pom.xml** file:

Pom.xml:
```xml
<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 https://maven.apache.org/xsd/maven-4.0.0.xsd">
 <modelVersion>4.0.0</modelVersion>
 <groupId>Project</groupId>
 <artifactId>Example</artifactId>
 <version>0.0.1-SNAPSHOT</version>

 <dependencies>
  <!-- JUnit Jupiter (API + Engine) -->
  <dependency>
   <groupId>org.junit.jupiter</groupId>
   <artifactId>junit-jupiter</artifactId>
   <version>5.9.3</version>
   <scope>test</scope>
  </dependency>
 </dependencies>
 <build>
  <plugins>
   <!-- Correct Surefire Plugin version to support JUnit 5 -->
   <plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-surefire-plugin</artifactId>
    <version>3.0.0-M9</version>
   </plugin>
  </plugins>
 </build>
</project>
```
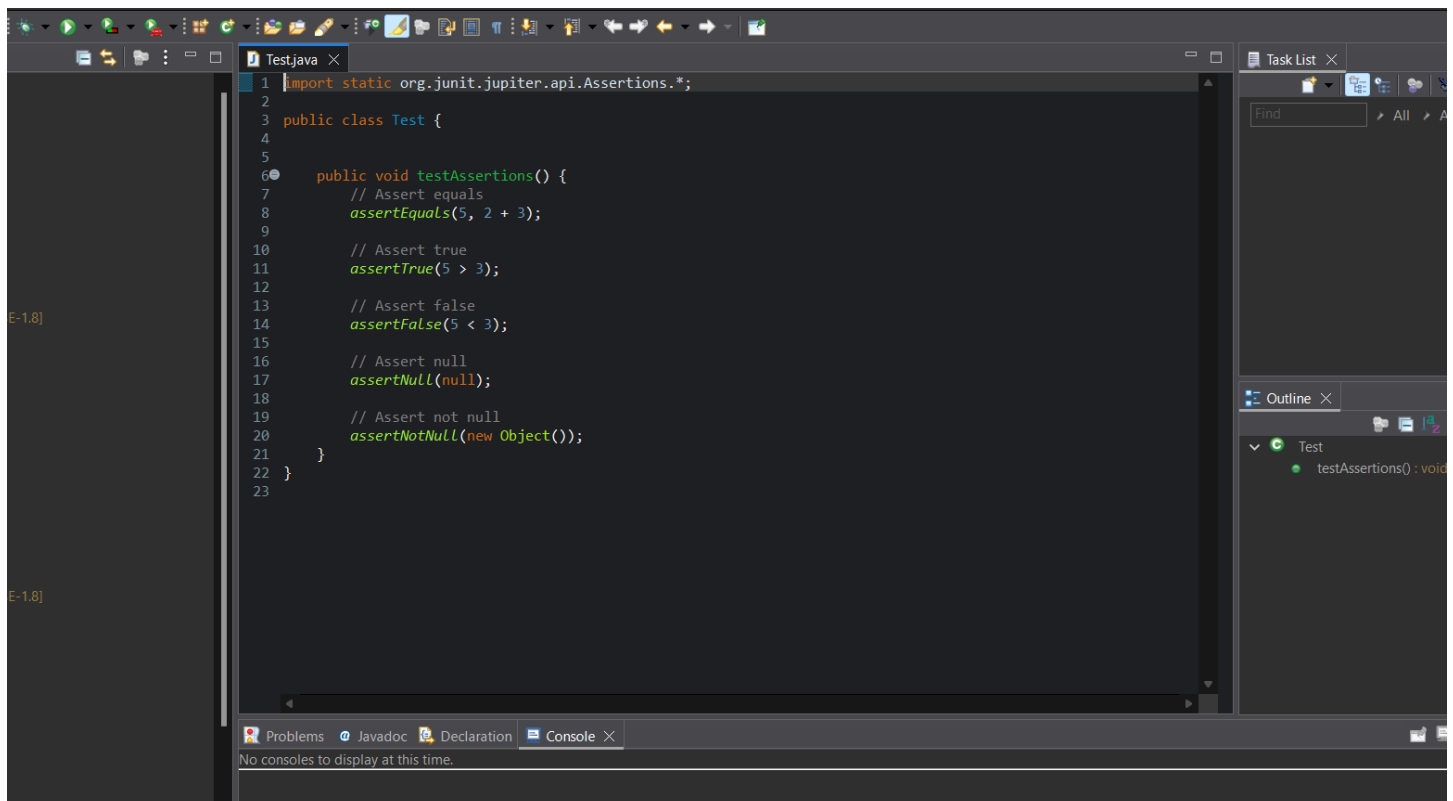
Then run this by maven test.

```xml
        https://maven.apache.org/xsd/maven-4.0.0.xsd (xsi:schemaLocation)
1 <project xmlns="http://maven.apache.org/POM/4.0.0"
2         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
4   <modelVersion>4.0.0</modelVersion>
5
6   <groupId>SettingJUnit</groupId>
7   <artifactId>SJUnit</artifactId>
8   <version>0.0.1-SNAPSHOT</version>
9
10  <dependencies>
11    <!-- JUnit Jupiter (API + Engine) -->
12    <dependency>
13      <groupId>org.junit.jupiter</groupId>
14      <artifactId>junit-jupiter</artifactId>
15      <version>5.9.3</version>
16      <scope>test</scope>
17    </dependency>
18  </dependencies>
19
20  <build>
21    <plugins>
22      <!-- Correct Surefire Plugin version to support JUnit 5 -->
23      <plugin>
24        <groupId>org.apache.maven.plugins</groupId>
25        <artifactId>maven-surefire-plugin</artifactId>
26        <version>3.0.0-M9</version>
27      </plugin>
28    </plugins>
29  </build>
30 </project>
31
```

Step 4. Create your test classes in **src/test/java**.

Write a sample test case:


```java
1 import static org.junit.jupiter.api.Assertions.*;
2
3 public class Test {
4
5
6     public void testAssertions() {
7         // Assert equals
8         assertEquals(5, 2 + 3);
9
10        // Assert true
11        assertTrue(5 > 3);
12
13        // Assert false
14        assertFalse(5 < 3);
15
16        // Assert null
17        assertNull(null);
18
19        // Assert not null
20        assertNotNull(new Object());
21    }
22 }
23
```

CODE:
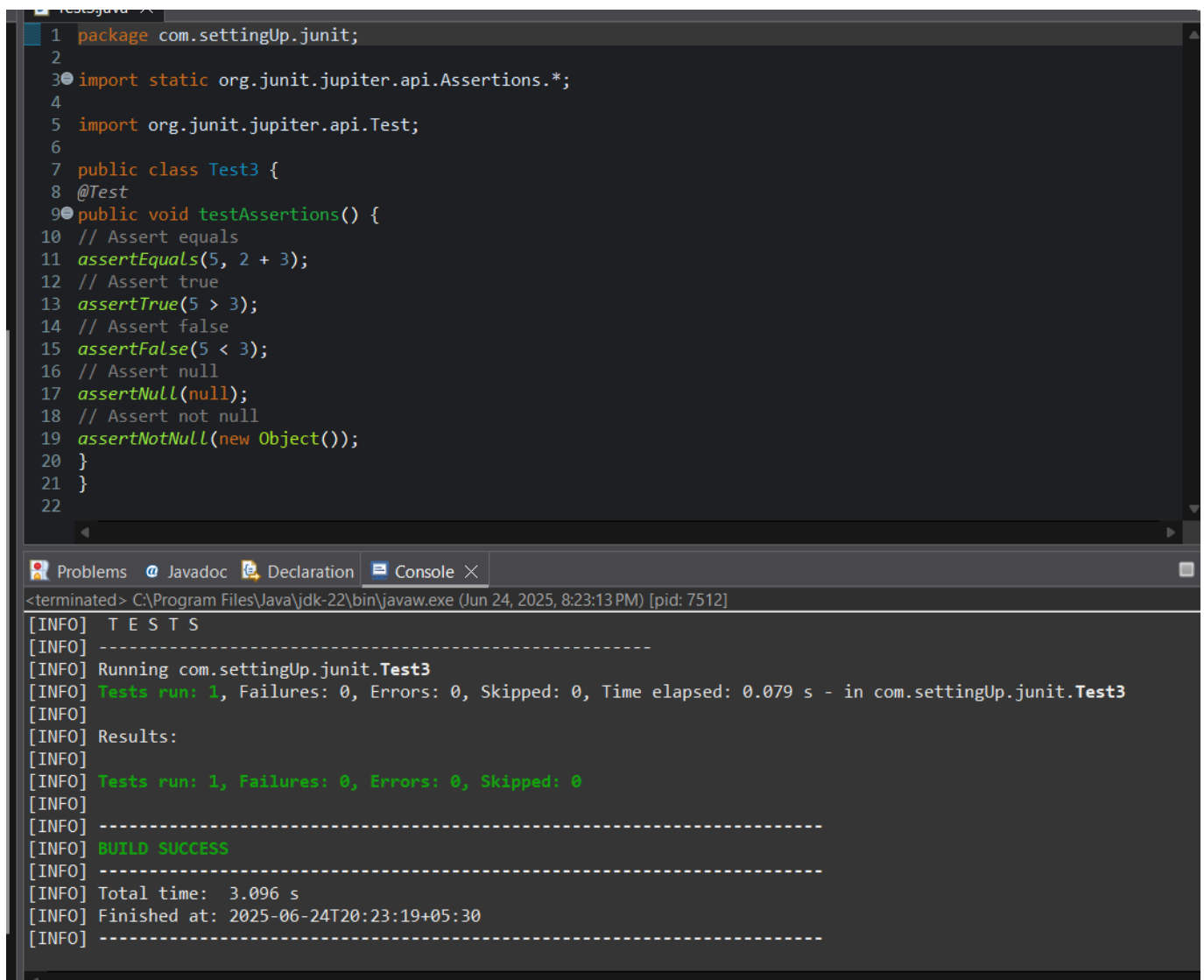
```
import static org.junit.jupiter.api.Assertions.*;
public class Test {
    public void testAssertions() {
        assertEquals(5, 2 + 3);
        assertTrue(5 > 3);
        assertFalse(5 < 3);
        assertNull(null);
        assertNotNull(new Object());
    }
}
```

Step 5. Then run the test class in **maven test** and then **Junit test** .
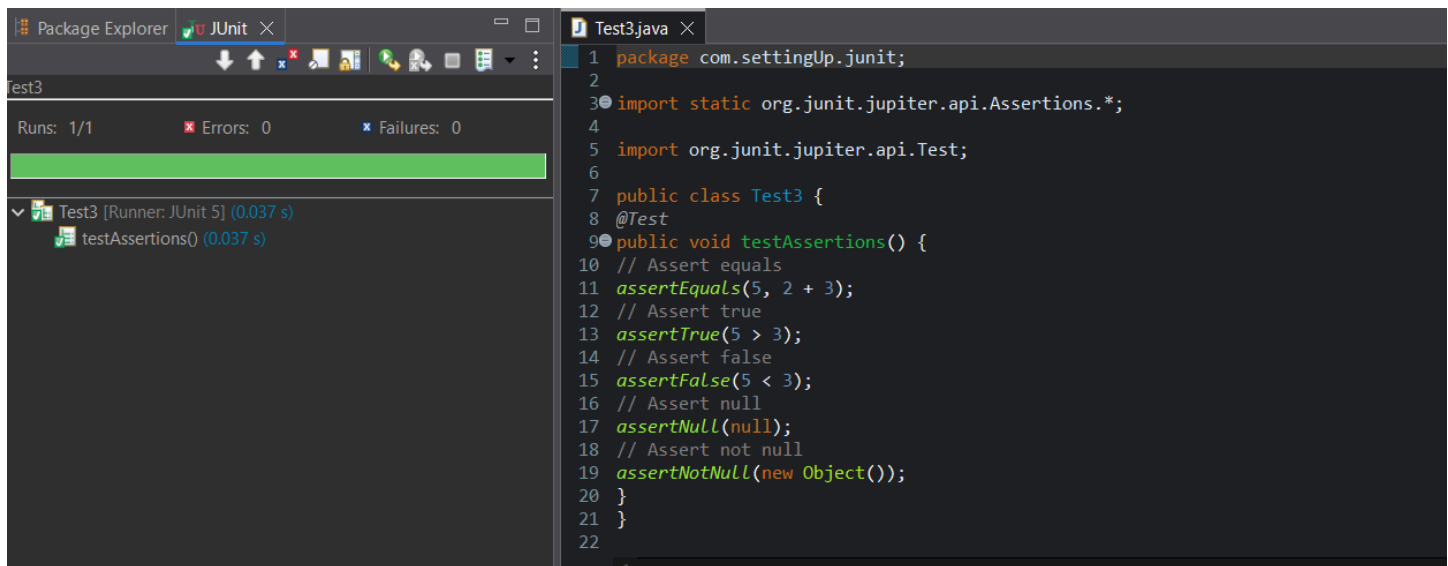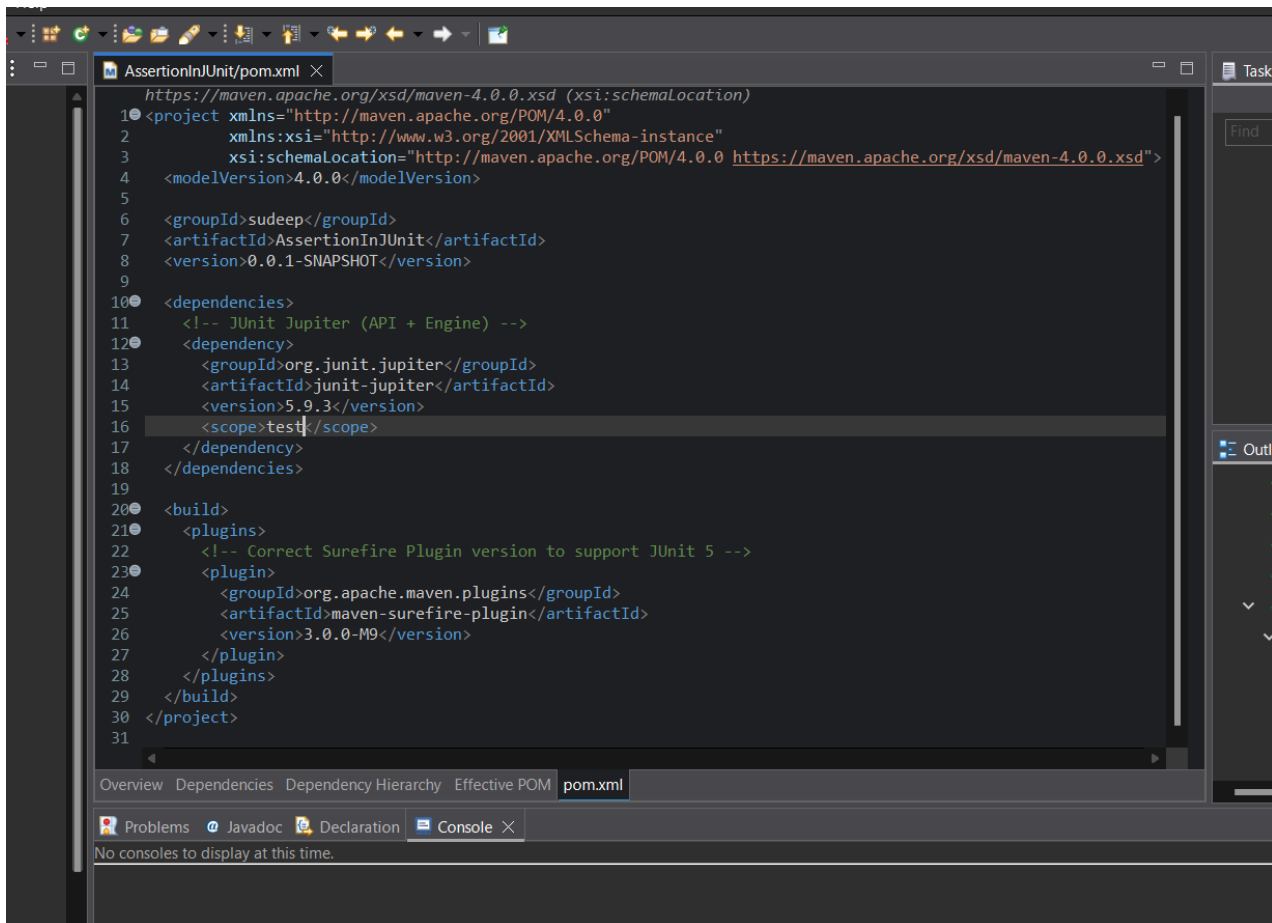
OUTPUT:

Maven Test:

Junit Test:


```
package com.settingUp.junit;

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

import org.junit.jupiter.api.Test;

public class Test3 {
@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());
}
}
```
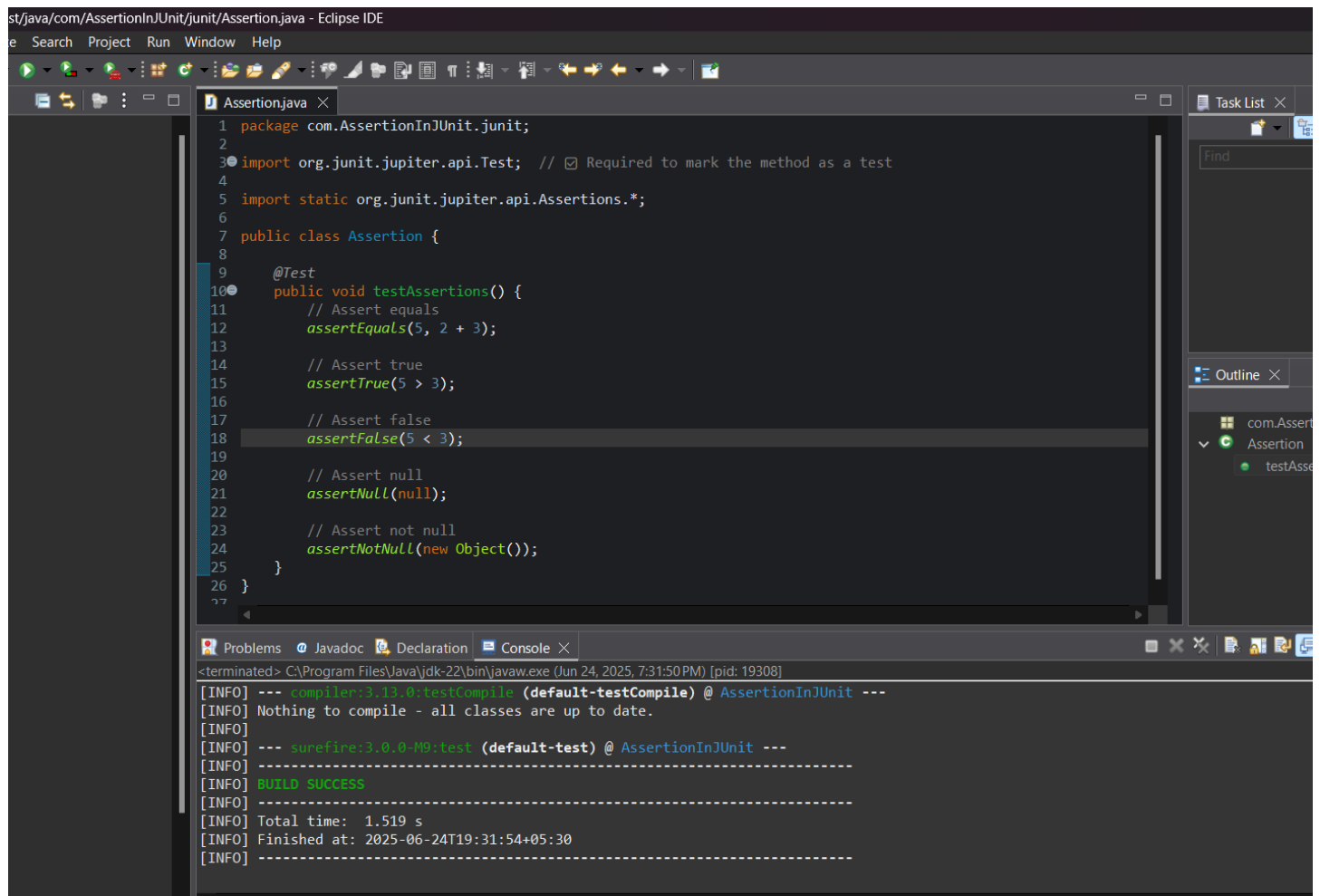
Exercise 3: Assertions in Junit

Scenario:

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

SOLUTION:

Step 1:Add the Junit Dependency in pom.xml

pom.xml :

```
<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 https://maven.apache.org/xsd/maven-4.0.0.xsd">
 <modelVersion>4.0.0</modelVersion>
 <groupId>sudeep</groupId>
 <artifactId>AssertionInJUnit</artifactId>
 <version>0.0.1-SNAPSHOT</version>
 <dependencies>
  <!-- JUnit Jupiter (API + Engine) -->
  <dependency>
   <groupId>org.junit.jupiter</groupId>
   <artifactId>junit-jupiter</artifactId>
   <version>5.9.3</version>
   <scope>test</scope>
  </dependency>
 </dependencies>
 <build>
  <plugins>
```

```xml
    <!-- Correct Surefire Plugin version to support JUnit 5 -->
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-surefire-plugin</artifactId>
      <version>3.0.0-M9</version>
    </plugin>
   </plugins>
  </build>
</project>
```


Step 2: Create your test classes in **src/test/java**.

Write the required test case and run as maven test and Junit test:

Code of Assertion.java:

package com.AssertionInJUnit.junit;

import org.junit.jupiter.api.Test;

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

public class Assertion {

  @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());
    }
}
```
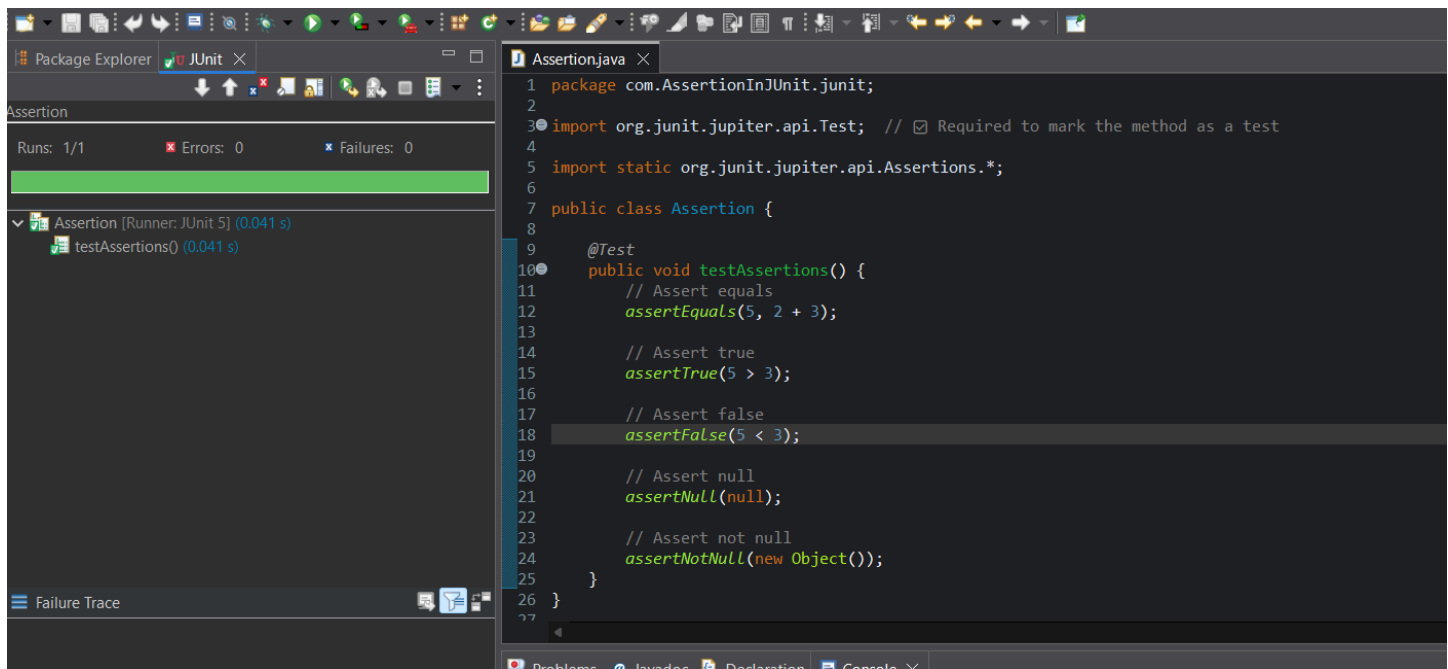
OUTPUT:

Maven test:

Junit test:

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.

SOLUTION:

Step 1:Add the Junit Dependency in pom.xml

pom.xml :

```xml
<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 https://maven.apache.org/xsd/maven-4.0.0.xsd">
 <modelVersion>4.0.0</modelVersion>
 <groupId>AAA</groupId>
 <artifactId>AAA_Pattern</artifactId>
 <version>0.0.1-SNAPSHOT</version>
 <dependencies>
  <!-- JUnit Jupiter (API + Engine) -->
  <dependency>
    <groupId>org.junit.jupiter</groupId>
    <artifactId>junit-jupiter</artifactId>
    <version>5.9.3</version>
```

```xml
      <scope>test</scope>
   </dependency>
  </dependencies>

  <build>
   <plugins>
    <!-- Correct Surefire Plugin version to support JUnit 5 -->
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-surefire-plugin</artifactId>
      <version>3.0.0-M9</version>
    </plugin>
   </plugins>
  </build>
</project>
```
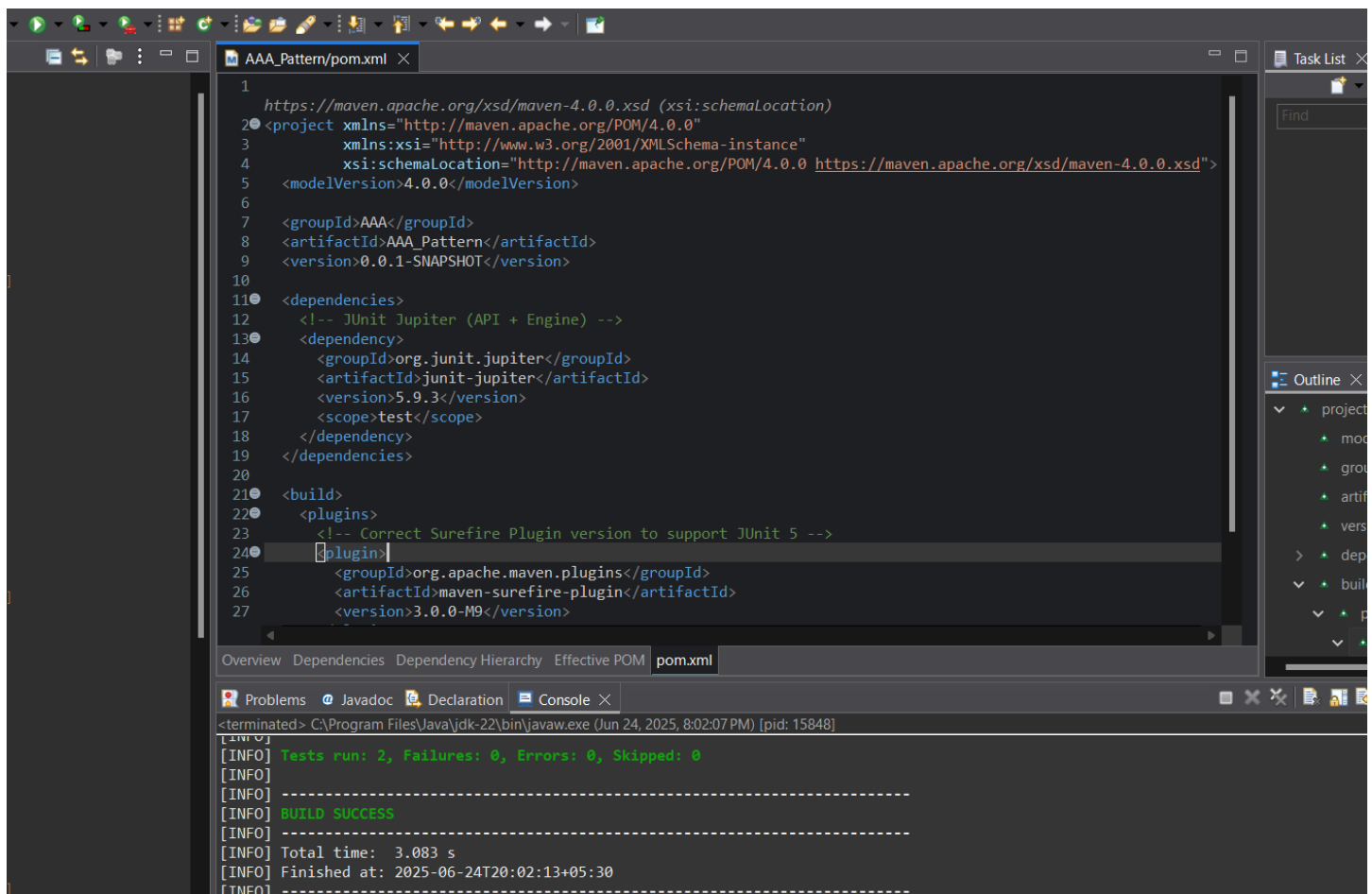
OUTPUT:

Step 2: Create your test classes in **src/test/java**.

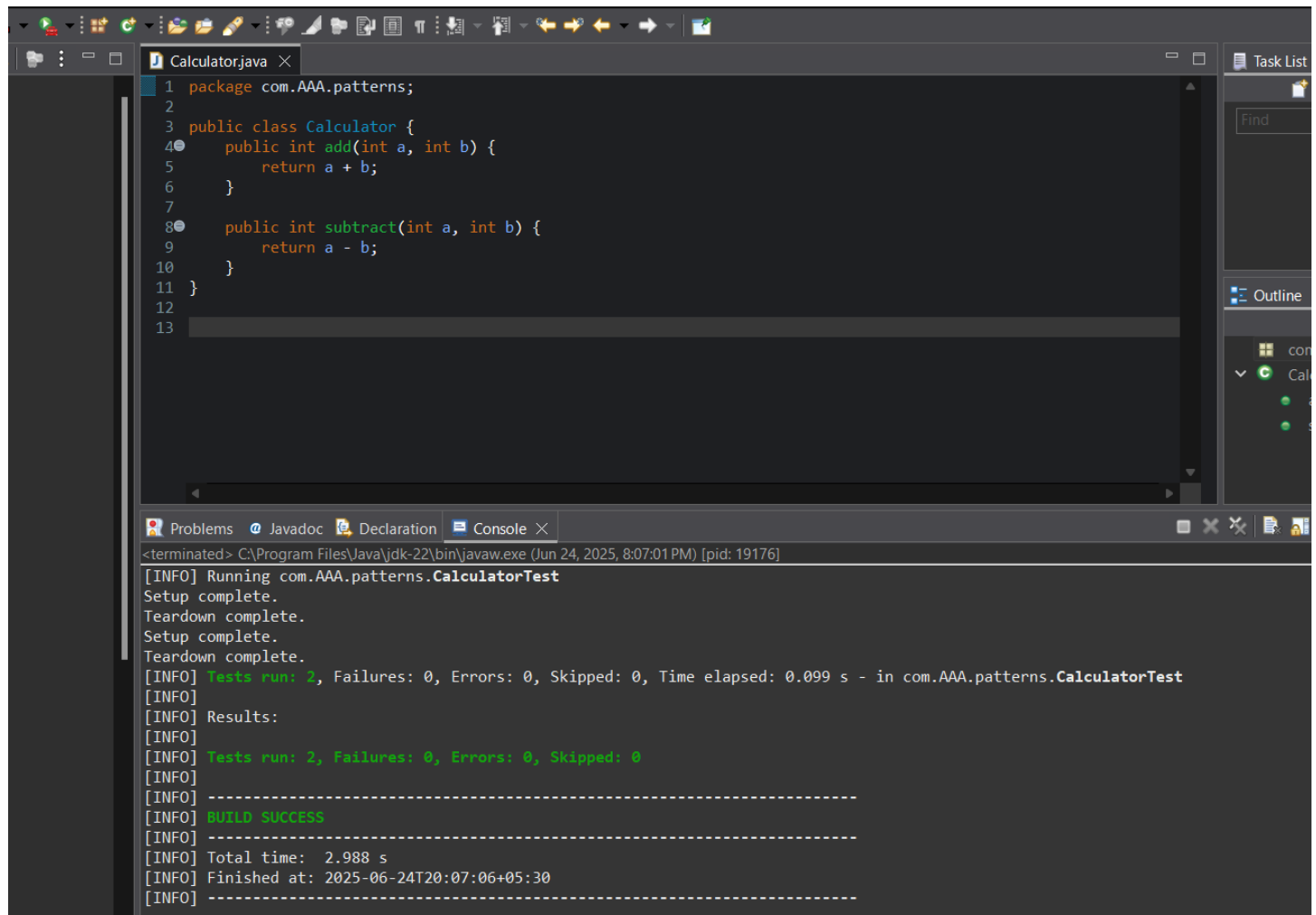Write the required test case and run as maven test and Junit test:

CODE for Calculator.java:

```
package com.AAA.patterns;
public class Calculator {
    public int add(int a, int b) {
        return a + b;
    }
    public int subtract(int a, int b) {
        return a - b;
    }
}
```

OUTPUT:

Maven Test for Calculator.java:

CODE for CalculatorTest.java:

package com.AAA.patterns;

import org.junit.jupiter.api.*;

```java
import static org.junit.jupiter.api.Assertions.*;
public class CalculatorTest {
    private Calculator calculator;
    @BeforeEach
    void setUp() {
        calculator = new Calculator();
        System.out.println("Setup complete.");
    }
    @AfterEach
    void tearDown() {
        calculator = null;
        System.out.println("Teardown complete.");
    }
    @Test
    void testAddition() {
        // Arrange
        int a = 10;
        int b = 5;
        // Act
        int result = calculator.add(a, b);
        // Assert
        assertEquals(15, result);
    }
    @Test
    void testSubtraction() {
        // Arrange
        int a = 10;
        int b = 3;
        // Act
        int result = calculator.subtract(a, b);
        // Assert
        assertEquals(7, result);
    }
}
```

OUTPUT:

Maven Test for CalculatorTest.java:

```java
6  public class CalculatorTest {
7
8      private Calculator calculator;
9
10     @BeforeEach
11     void setUp() {
12         calculator = new Calculator();
13         System.out.println("Setup complete.");
14     }
15
16     @AfterEach
17     void tearDown() {
18         calculator = null;
19         System.out.println("Teardown complete.");
20     }
21
22     @Test
23     void testAddition() {
24         // Arrange
25         int a = 10;
26         int b = 5;
27
```
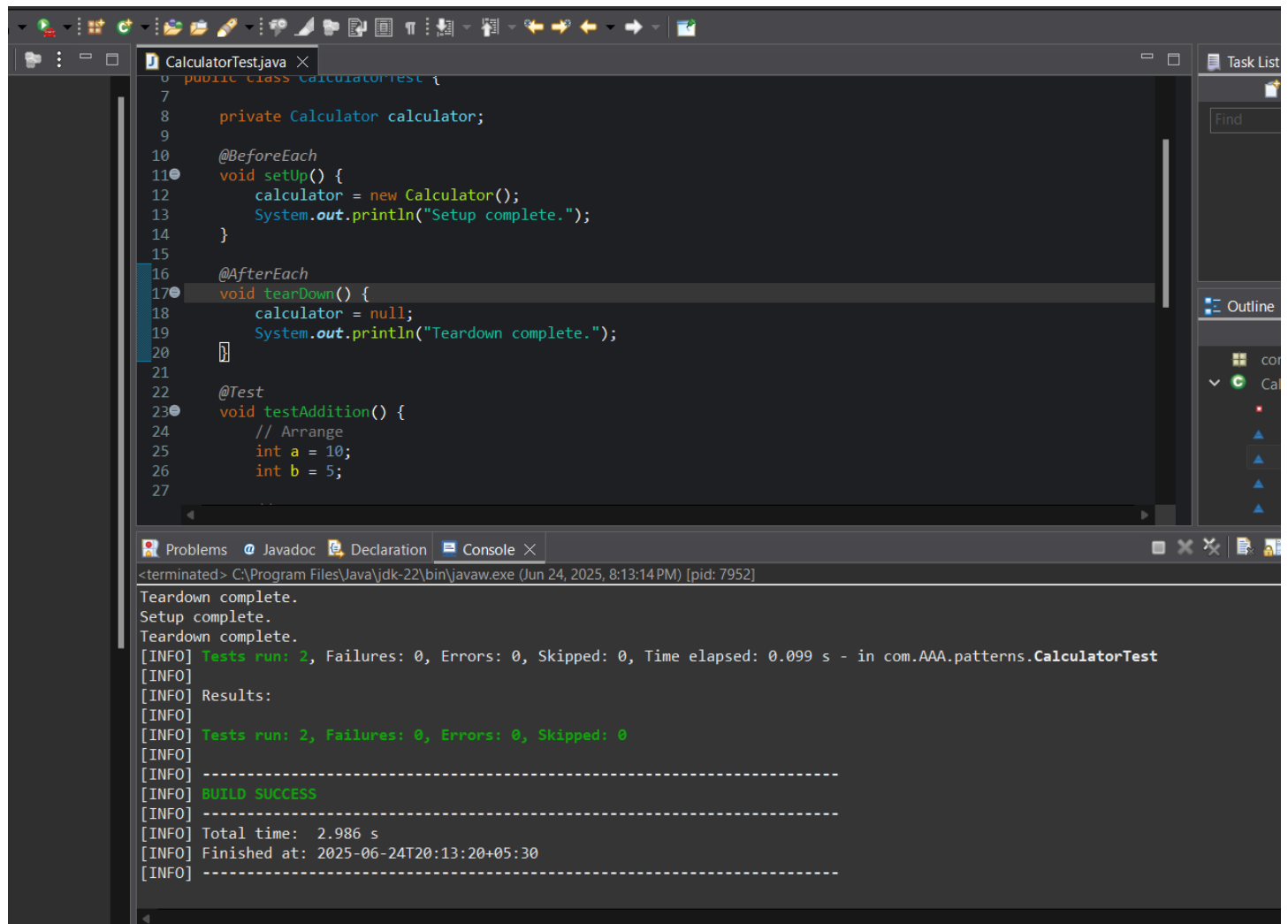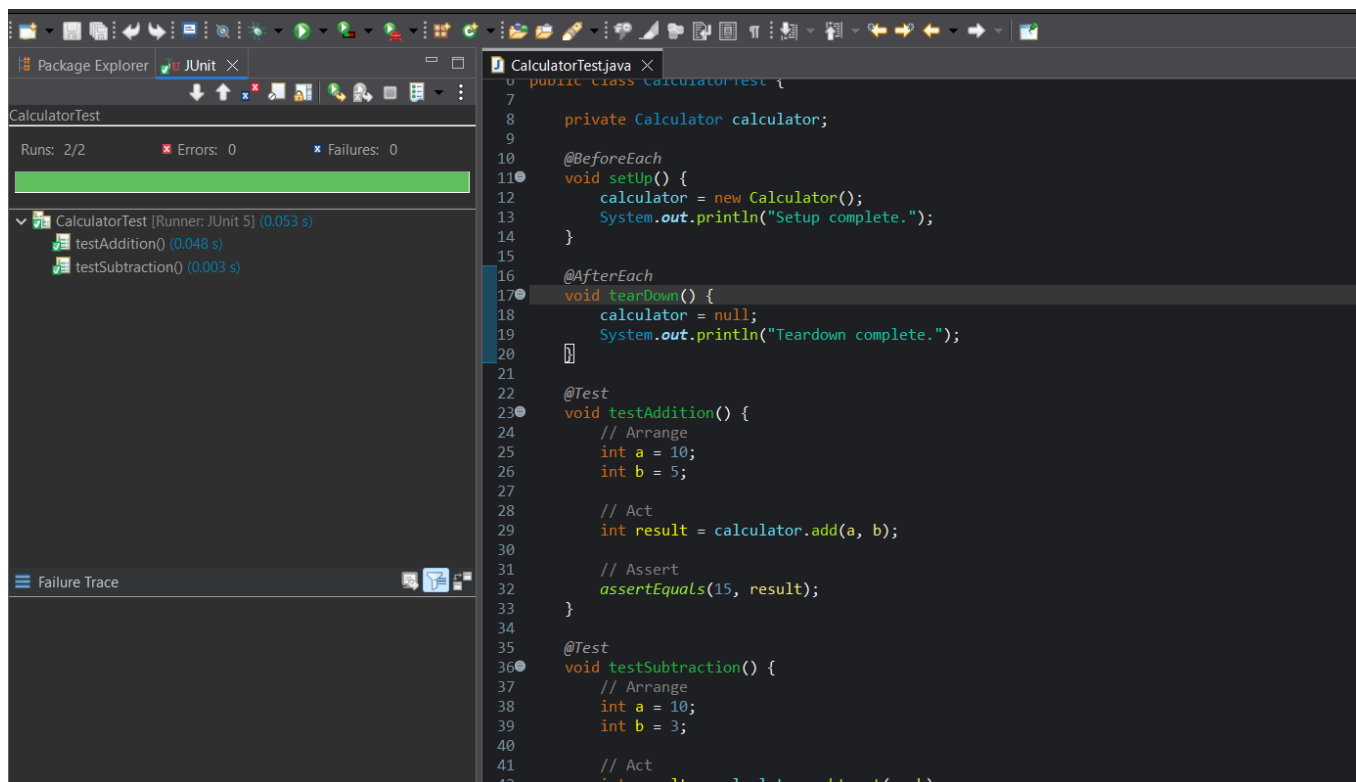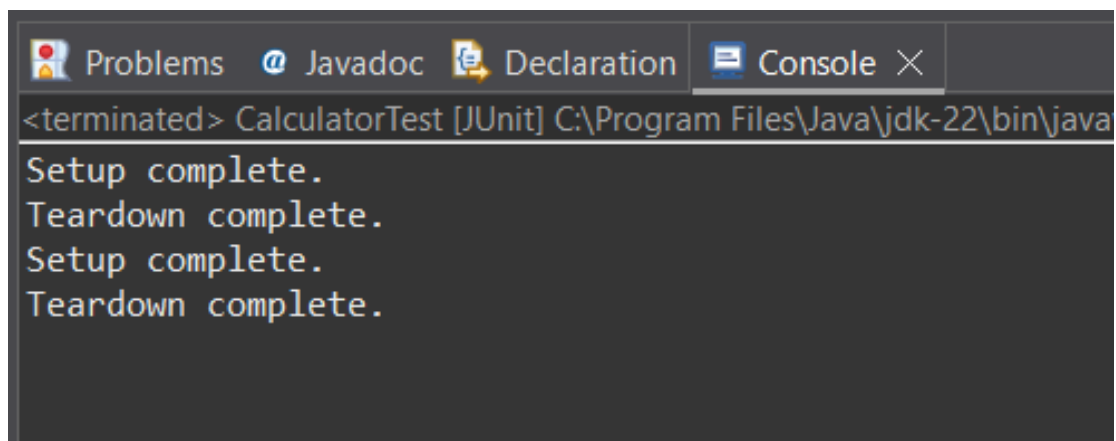
```
Problems  @ Javadoc  Declaration  Console ×

<terminated> C:\Program Files\Java\jdk-22\bin\javaw.exe (Jun 24, 2025, 8:13:14 PM) [pid: 7952]
Teardown complete.
Setup complete.
Teardown complete.
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.099 s - in com.AAA.patterns.CalculatorTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  2.986 s
[INFO] Finished at: 2025-06-24T20:13:20+05:30
[INFO] ------------------------------------------------------------------------
```

Junit Test for CalculatorTest.java:


```
6    public class CalculatorTest {
7
8        private Calculator calculator;
9
10       @BeforeEach
11       void setUp() {
12           calculator = new Calculator();
13           System.out.println("Setup complete.");
14       }
15
16       @AfterEach
17       void tearDown() {
18           calculator = null;
19           System.out.println("Teardown complete.");
20       }
21
22       @Test
23       void testAddition() {
24           // Arrange
25           int a = 10;
26           int b = 5;
27
28           // Act
29           int result = calculator.add(a, b);
30
31           // Assert
32           assertEquals(15, result);
33       }
34
35       @Test
36       void testSubtraction() {
37           // Arrange
38           int a = 10;
39           int b = 3;
40
41           // Act
42           int result = calculator.subtract(a, b);
```

Runs: 2/2   Errors: 0   Failures: 0

CalculatorTest [Runner: JUnit 5] (0.053 s)
testAddition() (0.048 s)
testSubtraction() (0.003 s)

Problems   @ Javadoc   Declaration   Console ×

<terminated> CalculatorTest [JUnit] C:\Program Files\Java\jdk-22\bin\java

```
Setup complete.
Teardown complete.
Setup complete.
Teardown complete.
```

NAME-RITIKA KUMARI

SUPERSET ID-6392654