JUnit Testing Exercises

Exercise 1: Setting Up Junit

Pom.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<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
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>org.example</groupId>
    <artifactId>ex1junitt</artifactId>
    <version>1.0-SNAPSHOT</version>

    <properties>
        <maven.compiler.source>24</maven.compiler.source>
        <maven.compiler.target>24</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>
    <dependencies>
        <!-- JUnit 4.13.2 -->
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.13.2</version>
            <scope>test</scope>
        </dependency>
    </dependencies>
</project>
```

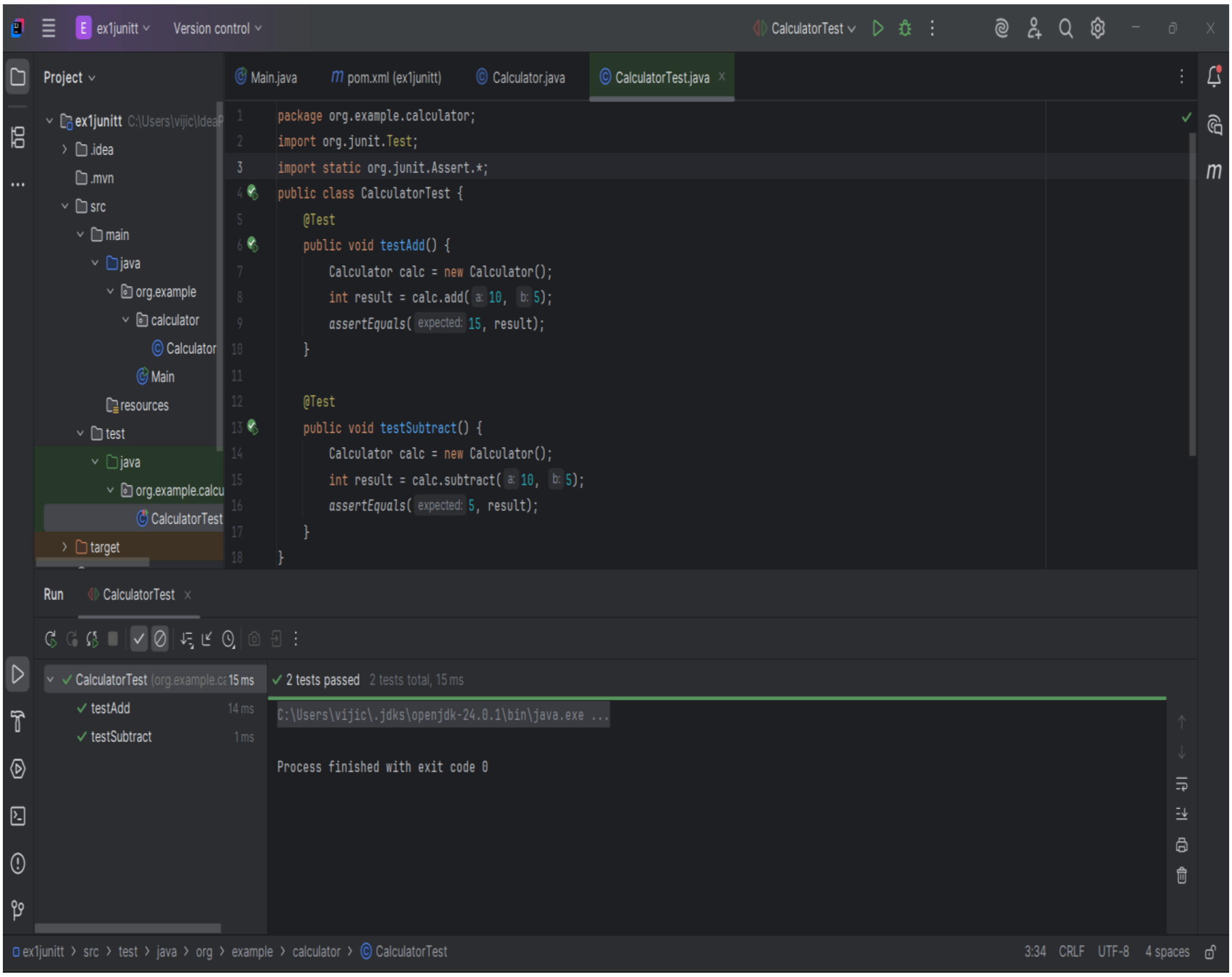
Calculator.java :

```
package org.example.calculator;
public class Calculator {
    public int add(int a, int b) {
        return a + b;
    }

    public int subtract(int a, int b) {
        return a - b;
    }
}
```

CalculatorTest.java:

```
package org.example.calculator;
import org.junit.Test;
import static org.junit.Assert.*;
public class CalculatorTest {
    @Test
    public void testAdd() {
        Calculator calc = new Calculator();
        int result = calc.add(10, 5);
        assertEquals(15, result);
    }

    @Test
    public void testSubtract() {
        Calculator calc = new Calculator();
        int result = calc.subtract(10, 5);
        assertEquals(5, result);
    }
}
```

Output:



```
package org.example.calculator;
import org.junit.Test;
import static org.junit.Assert.*;
public class CalculatorTest {
    @Test
    public void testAdd() {
        Calculator calc = new Calculator();
        int result = calc.add( a: 10,  b: 5);
        assertEquals( expected: 15, result);
    }

    @Test
    public void testSubtract() {
        Calculator calc = new Calculator();
        int result = calc.subtract( a: 10,  b: 5);
        assertEquals( expected: 5, result);
    }
}
```

```
✓ CalculatorTest (org.example.ca 15 ms
    ✓ testAdd        14 ms
    ✓ testSubtract    1 ms
```

```
✓ 2 tests passed  2 tests total, 15 ms
C:\Users\vijic\.jdks\openjdk-24.0.1\bin\java.exe ...

Process finished with exit code 0
```

Exercise 2: Writing Basic JUnit Tests

Pom.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<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
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>org.example</groupId>
    <artifactId>ex2write</artifactId>
    <version>1.0-SNAPSHOT</version>
    <properties>
        <maven.compiler.source>24</maven.compiler.source>
        <maven.compiler.target>24</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>
    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.13.2</version>
            <scope>test</scope>
        </dependency>
    </dependencies>

</project>
```

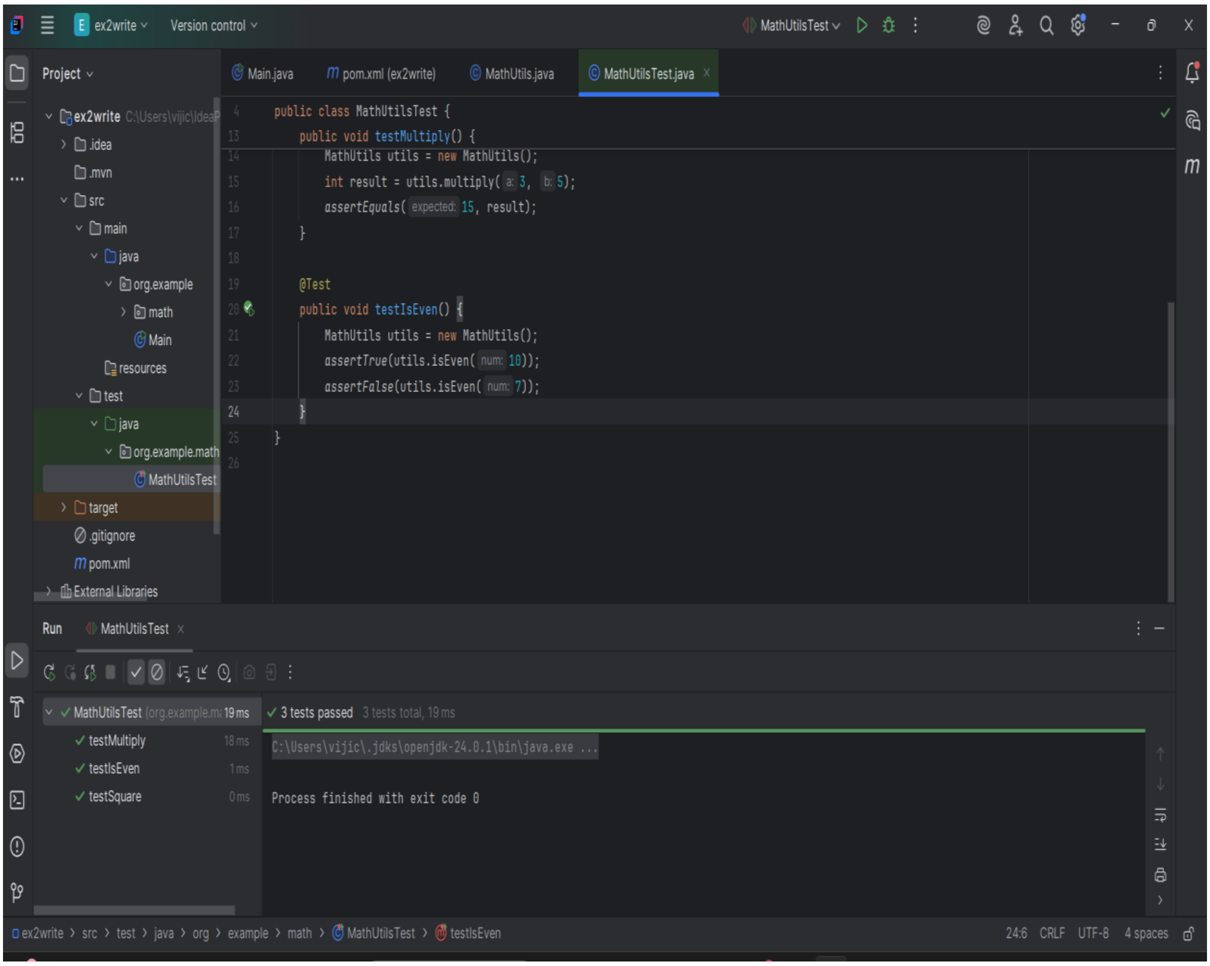
MathUtils.java:

```
package org.example.math;

public class MathUtils {
    public int square(int num) {
        return num * num;
    }

    public int multiply(int a, int b) {
        return a * b;
    }

    public boolean isEven(int num) {
        return num % 2 == 0;
    }
}
```

MathUtilsTest.java :

```
package org.example.math;

public class MathUtils {
    public int square(int num) {
        return num * num;
    }

    public int multiply(int a, int b) {
        return a * b;
    }

    public boolean isEven(int num) {
        return num % 2 == 0;
    }
}
```

Output:



```
public class MathUtilsTest {
    public void testMultiply() {
        MathUtils utils = new MathUtils();
        int result = utils.multiply(a: 3, b: 5);
        assertEquals(expected: 15, result);
    }

    @Test
    public void testIsEven() {
        MathUtils utils = new MathUtils();
        assertTrue(utils.isEven(num: 10));
        assertFalse(utils.isEven(num: 7));
    }
}
```

```
✓ MathUtilsTest (org.example.ma 19 ms    ✓ 3 tests passed  3 tests total, 19 ms
    ✓ testMultiply   18 ms                C:\Users\vijic\.jdks\openjdk-24.0.1\bin\java.exe ...
    ✓ testIsEven      1 ms
    ✓ testSquare      0 ms                Process finished with exit code 0
```

Exercise 3: Assertions in Junit:

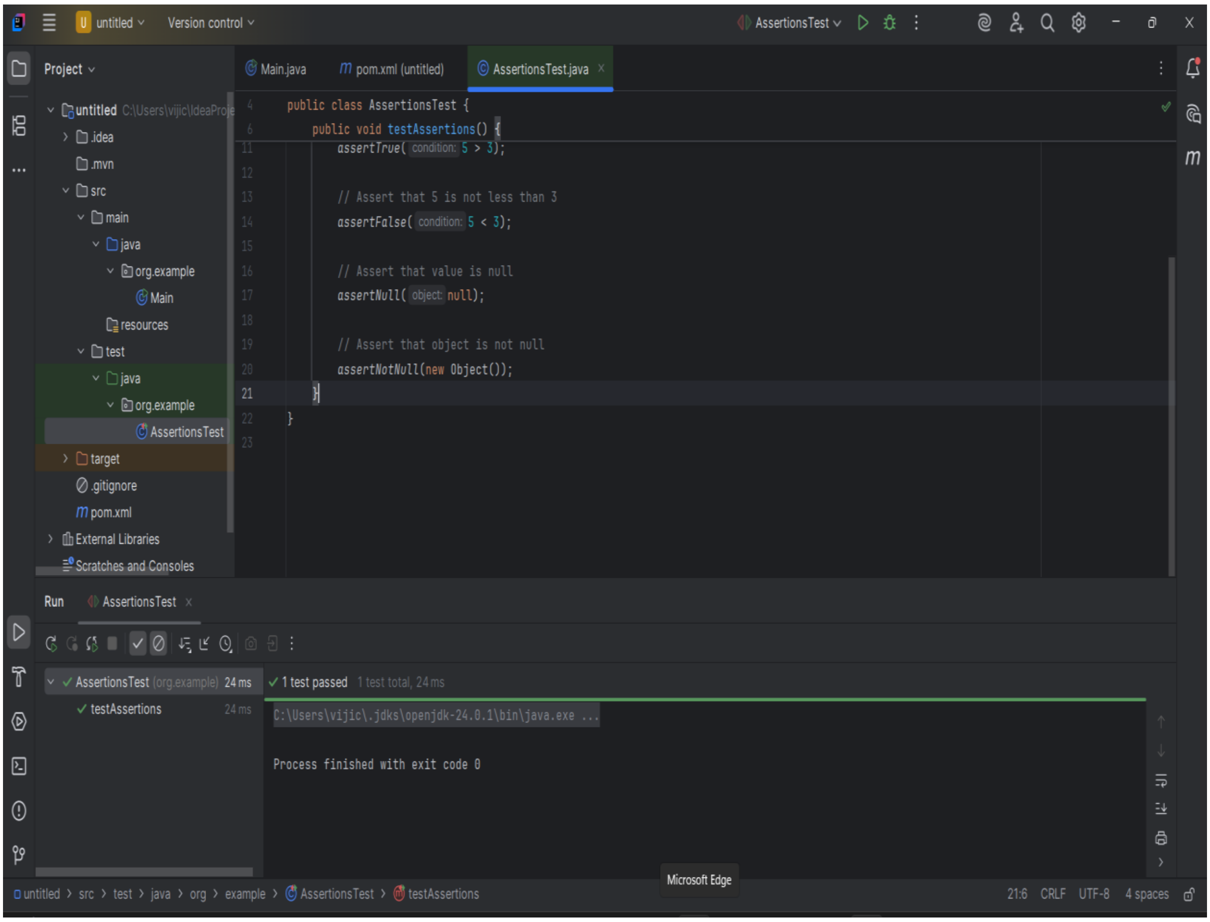
Pom.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<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
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>org.example</groupId>
    <artifactId>untitled</artifactId>
    <version>1.0-SNAPSHOT</version>

    <properties>
        <maven.compiler.source>24</maven.compiler.source>
        <maven.compiler.target>24</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>
    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.13.2</version>
            <scope>test</scope>
        </dependency>
    </dependencies>

</project>
```

AssertionTest.java:

```
package org.example;
import org.junit.Test;
import static org.junit.Assert.*;
public class AssertionsTest {
    @Test
    public void testAssertions() {
        // Assert that 2 + 3 = 5
        assertEquals(5, 2 + 3);

        // Assert that 5 is greater than 3
        assertTrue(5 > 3);

        // Assert that 5 is not less than 3
        assertFalse(5 < 3);

        // Assert that value is null
        assertNull(null);

        // Assert that object is not null
        assertNotNull(new Object());
    }
}
```

Output:



```
public class AssertionsTest {
    public void testAssertions() {
        assertTrue( condition: 5 > 3);

        // Assert that 5 is not less than 3
        assertFalse( condition: 5 < 3);

        // Assert that value is null
        assertNull( object: null);

        // Assert that object is not null
        assertNotNull(new Object());
    }
}
```

✓ AssertionsTest (org.example) 24 ms
✓ testAssertions 24 ms

✓ 1 test passed 1 test total, 24 ms

```
C:\Users\vijic\.jdks\openjdk-24.0.1\bin\java.exe ...

Process finished with exit code 0
```

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

Pom.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<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
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>org.example</groupId>
    <artifactId>ex4</artifactId>
    <version>1.0-SNAPSHOT</version>
    <properties>
        <maven.compiler.source>24</maven.compiler.source>
        <maven.compiler.target>24</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>
    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.13.2</version>
            <scope>test</scope>
        </dependency>
    </dependencies>
</project>
```

Calculator.java:

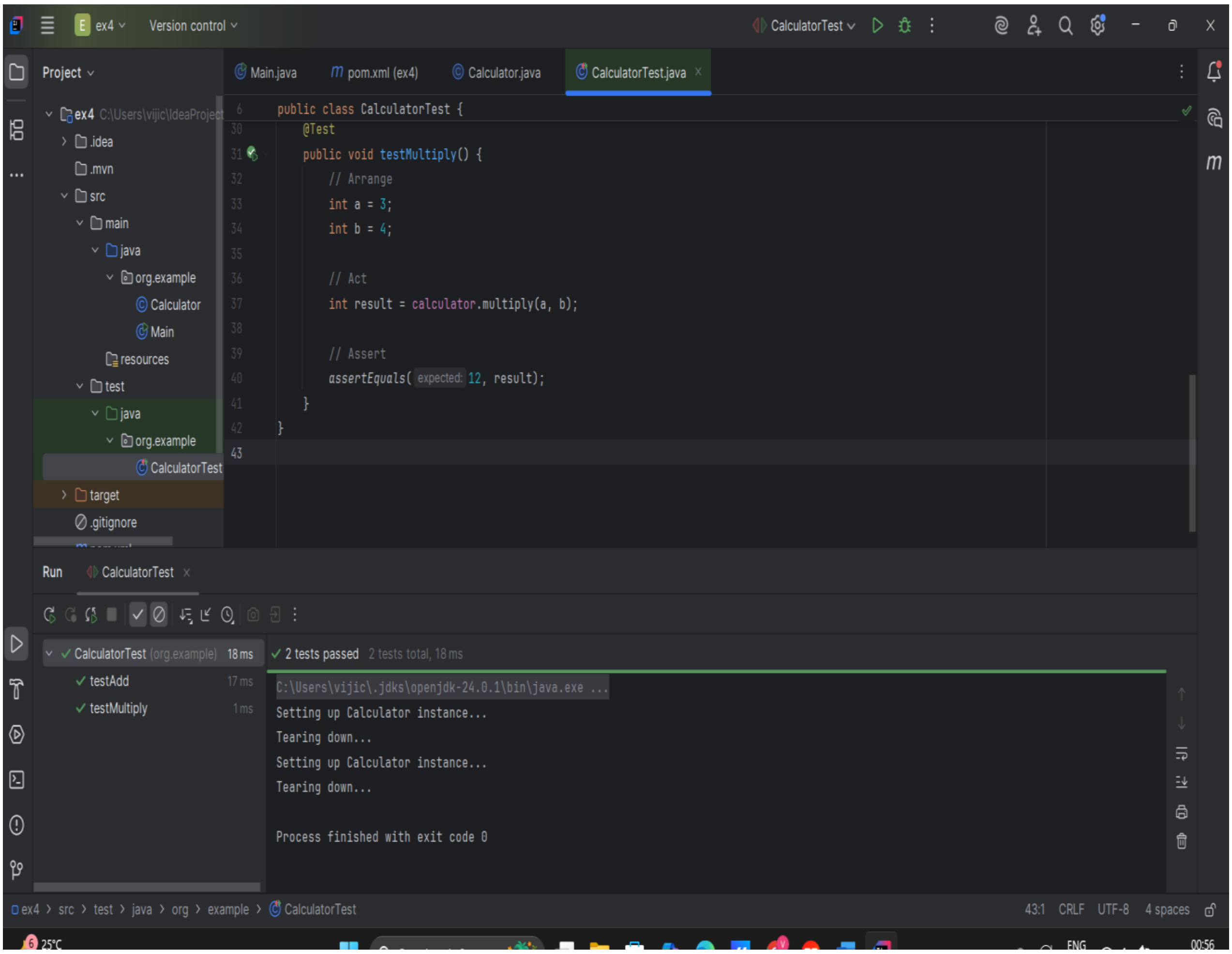
```
package org.example;
public class Calculator {
    public int add(int a, int b) {
```

```
    return a + b;
    }

    public int multiply(int a, int b) {
        return a * b;
    }
}
```

CalculatorTest.java:

```
package org.example;
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 instance...");
        calculator = new Calculator();
    }
    @After
    public void tearDown() {
        System.out.println("Tearing down...");
        calculator = null;
    }
    @Test
    public void testAdd() {
        // Arrange
        int a = 10;
        int b = 5;

        // Act
        int result = calculator.add(a, b);
        // Assert
        assertEquals(15, result);
    }
    @Test
    public void testMultiply() {
        // Arrange
        int a = 3;
        int b = 4;
        // Act
        int result = calculator.multiply(a, b);
        // Assert
        assertEquals(12, result);
    }
}
```

Output:



```
public class CalculatorTest {
    @Test
    public void testMultiply() {
        // Arrange
        int a = 3;
        int b = 4;

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

        // Assert
        assertEquals( expected: 12, result);
    }
}
```

✓ CalculatorTest (org.example) 18 ms
✓ testAdd 17 ms
✓ testMultiply 1 ms

✓ 2 tests passed 2 tests total, 18 ms

```
C:\Users\vijic\.jdks\openjdk-24.0.1\bin\java.exe ...
Setting up Calculator instance...
Tearing down...
Setting up Calculator instance...
Tearing down...

Process finished with exit code 0
```