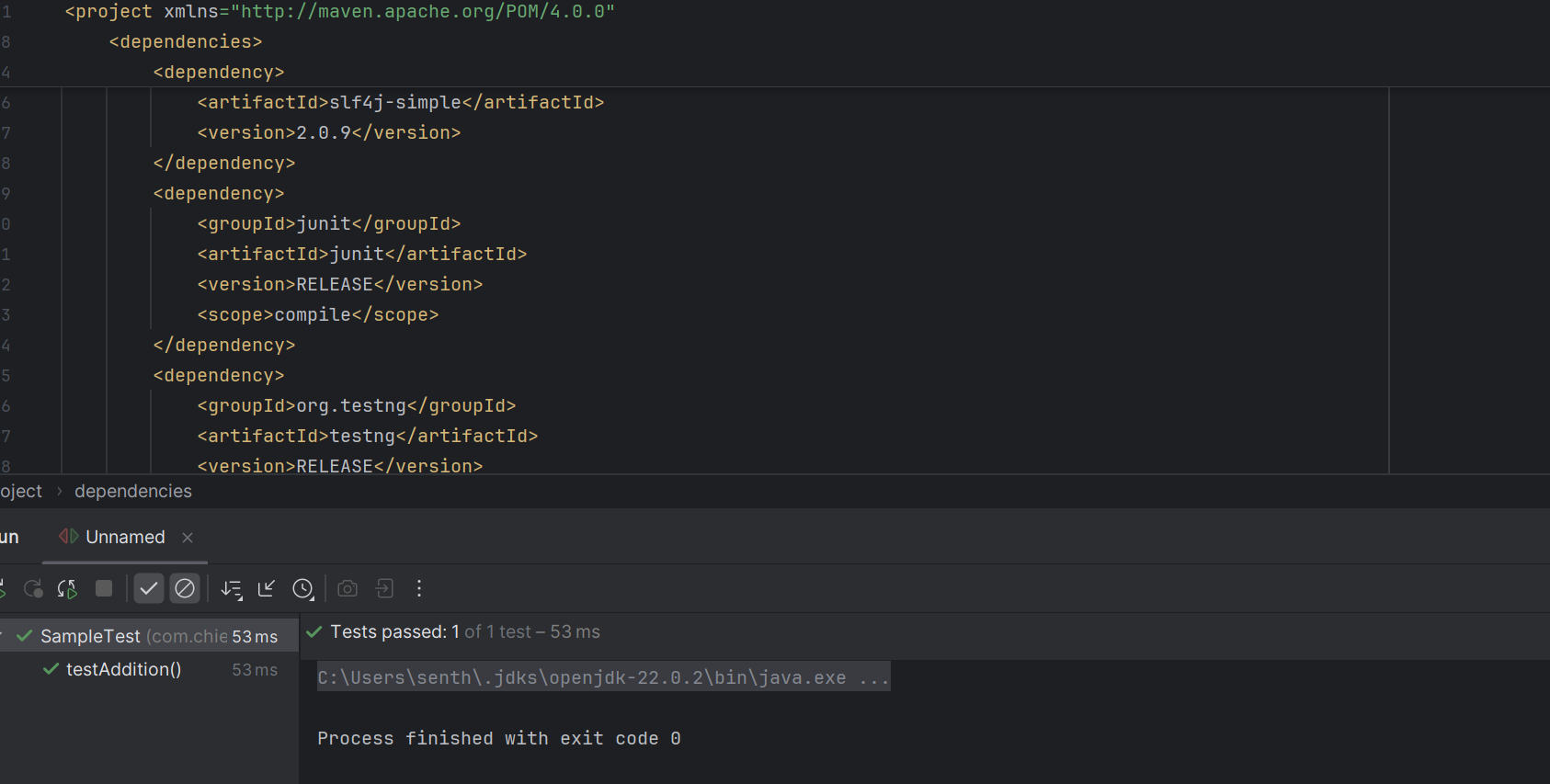
J-Unit

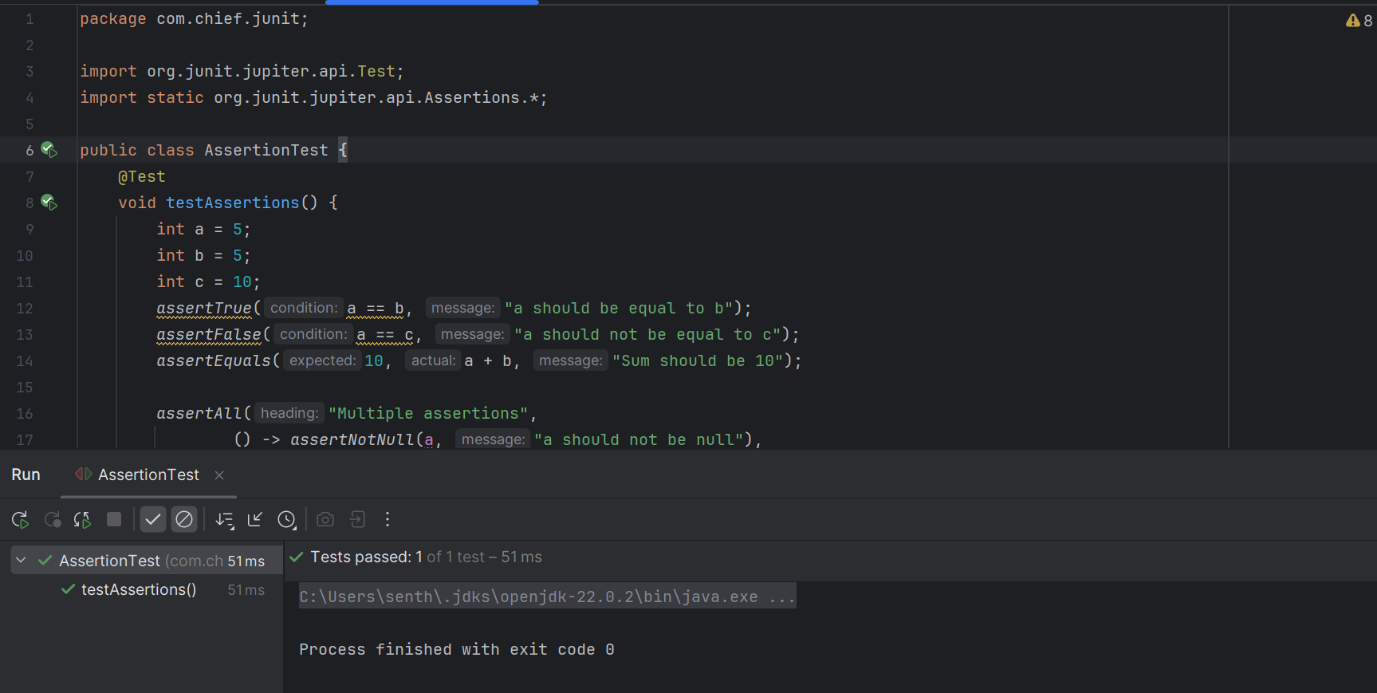
Exercise 1:

package com.chief.junit;  
  
import org.junit.jupiter.api. Test;  
  
import static org.junit.jupiter.api.Assertions.*assertEquals*;  
  
public class SampleTest {  
 @Test  
 void testAddition() {  
 int sum = 2 + 3;  
 *assertEquals*(5, sum, "2 + 3 should equal 5");  
 }  
}



Exercise 3: Assertions

package com.chief.junit;  
  
import org.junit.jupiter.api.Test;  
import static org.junit.jupiter.api.Assertions.\*;  
  
public class AssertionTest {  
 @Test  
 void testAssertions() {  
 int a = 5;  
 int b = 5;  
 int c = 10;  
 *assertTrue*(a == b, "a should be equal to b");  
 *assertFalse*(a == c, "a should not be equal to c");  
 *assertEquals*(10, a + b, "Sum should be 10");  
  
 *assertAll*("Multiple assertions",  
 () -> *assertNotNull*(a, "a should not be null"),  
 () -> *assertNotEquals*(a, c, "a and c should not be equal")  
 );  
  
 Exception exception = *assertThrows*(ArithmeticException.class, () -> {  
 int result = 1 / 0;  
 });  
 *assertEquals*("/ by zero", exception.getMessage());  
 }  
}



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

package com.chief.junit;  
import org.junit.jupiter.api.\*;  
import static org.junit.jupiter.api.Assertions.\*;  
  
class CalculatorTest {  
  
 Calculator calculator;  
  
 @BeforeEach  
 void setup() {  
 calculator = new Calculator();  
 }  
  
 @Test  
 void testAdd() {  
 int result = calculator.add(5, 3);  
 *assertEquals*(8, result);  
 }  
  
 @Test  
 void testSubtract() {  
 *assertEquals*(2, calculator.subtract(5, 3));  
 }  
  
 @AfterEach  
 void teardown() {  
 calculator = null;  
 }  
}

