Week_2_ Exercise 4: Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and Teardown Methods in JUnit

src > test > java > AssertionsTest.java > AssertionsTest > tearDown()

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

class Calculator {
    int add(int a, int b) {
        return a + b;
    }

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

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

    int divide(int a, int b) {
        if (b == 0)
            throw new IllegalArgumentException(s:"Cannot divide by zero");
        return a / b;
    }
}

public class AssertionsTest {

    Calculator calc;

    @BeforeEach
    void setUp() {
        calc = new Calculator();
    }

    @AfterEach
    void tearDown() {
        calc = null;
    }

    @Test
    void additionTest() {
        int result = calc.add(a:10, b:5);
        assertEquals(expected:15, result);
    }

    @Test
    void subtractionTest() {
        int result = calc.subtract(a:10, b:5);
        assertEquals(expected:5, result);
    }

    @Test
    void multiplicationTest() {
        int result = calc.multiply(a:3, b:4);
        assertEquals(expected:12, result);
    }

    @Test
    void divisionTest() {
        int result = calc.divide(a:20, b:4);
        assertEquals(expected:5, result);
    }

    @Test
    void divideByZeroTest() {
        assertThrows(expectedType:IllegalArgumentException.class, () -> calc.divide(a:10, b:0));
    }
}
```