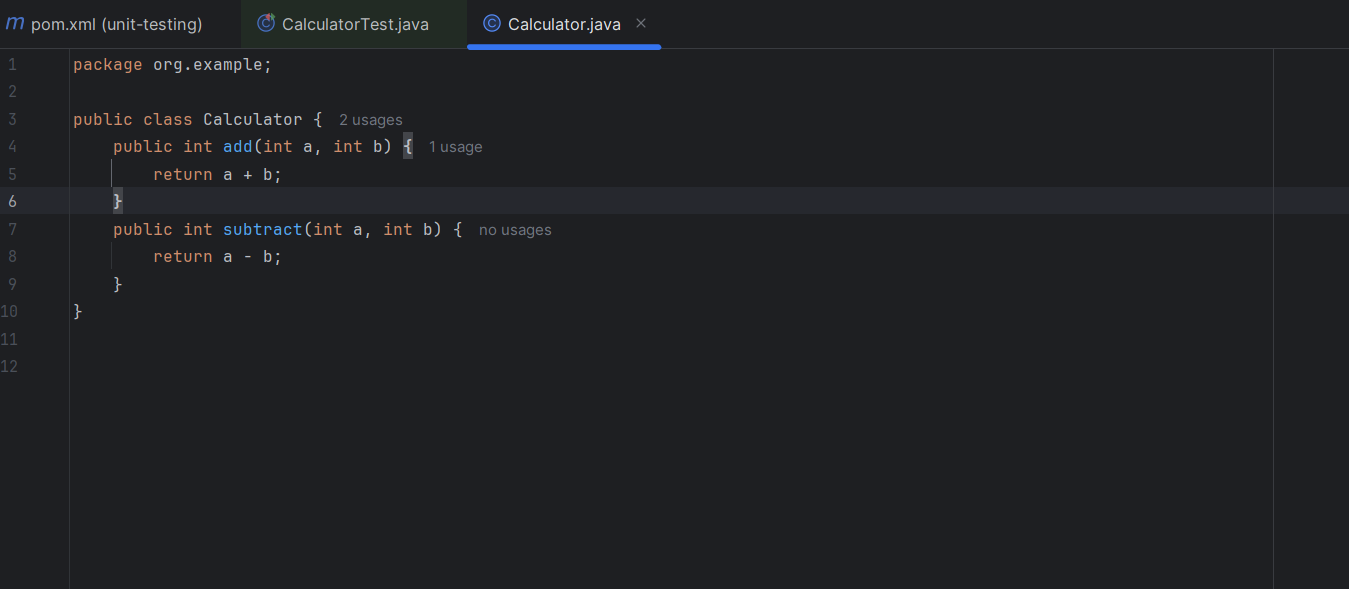
**Exercise 4: Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and Teardown Methods in Junit**

* Create a simple class(Calculator.java) whose methods will be tested.

****

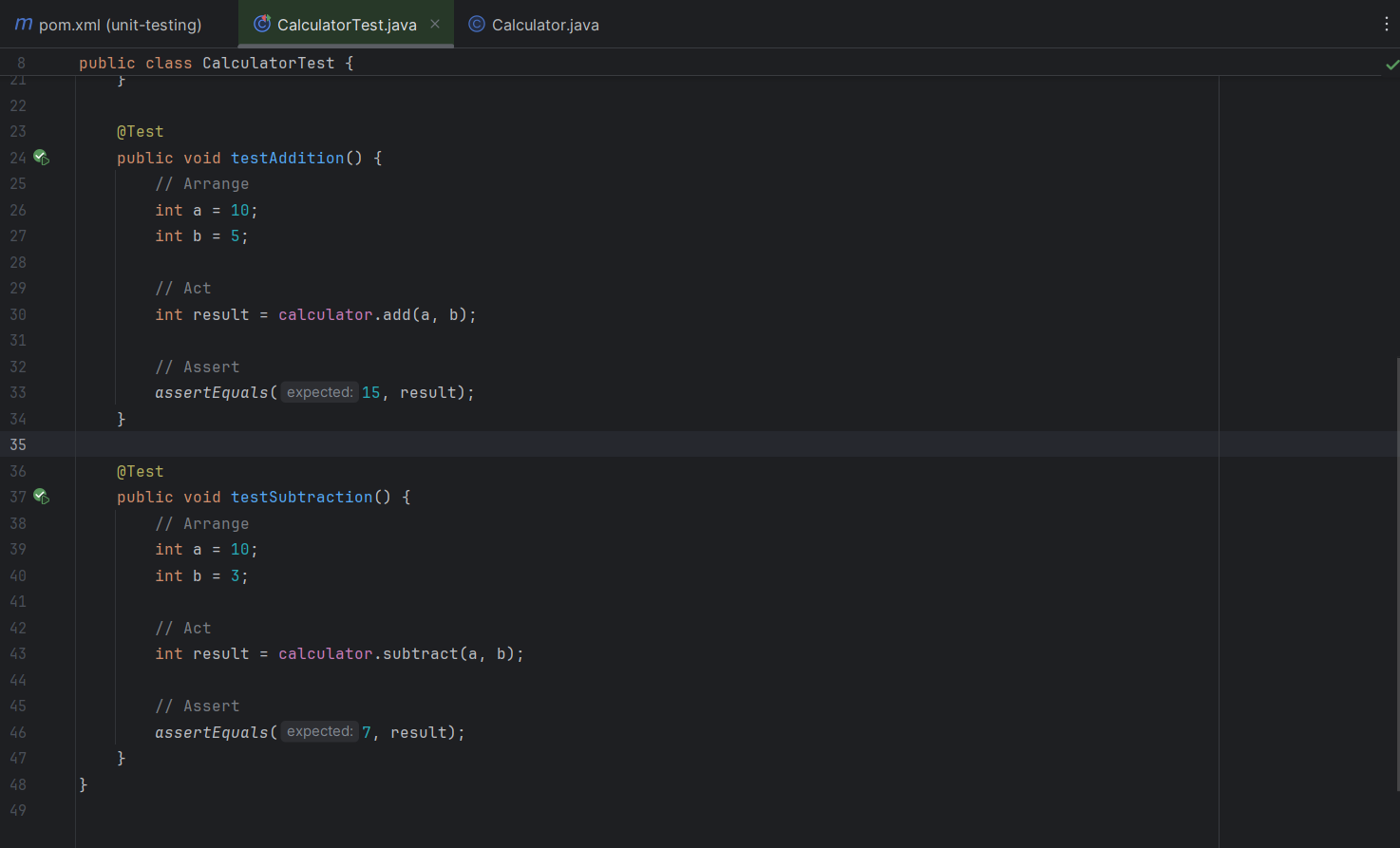
****

* JUnit provides special annotations to help prepare and clean the environment before and after each test - Used @BeforeEach and @AfterEach for Setup and Teardown
* @BeforeEach
* This method runs before every test method.
* Used to initialize test data or create reusable objects.

Example: Creating a new Calculator or Service object.

* @AfterEach
* This method runs after every test method.
* Used to clean up, reset, or release resources after the test.

Example: Setting the object to null.

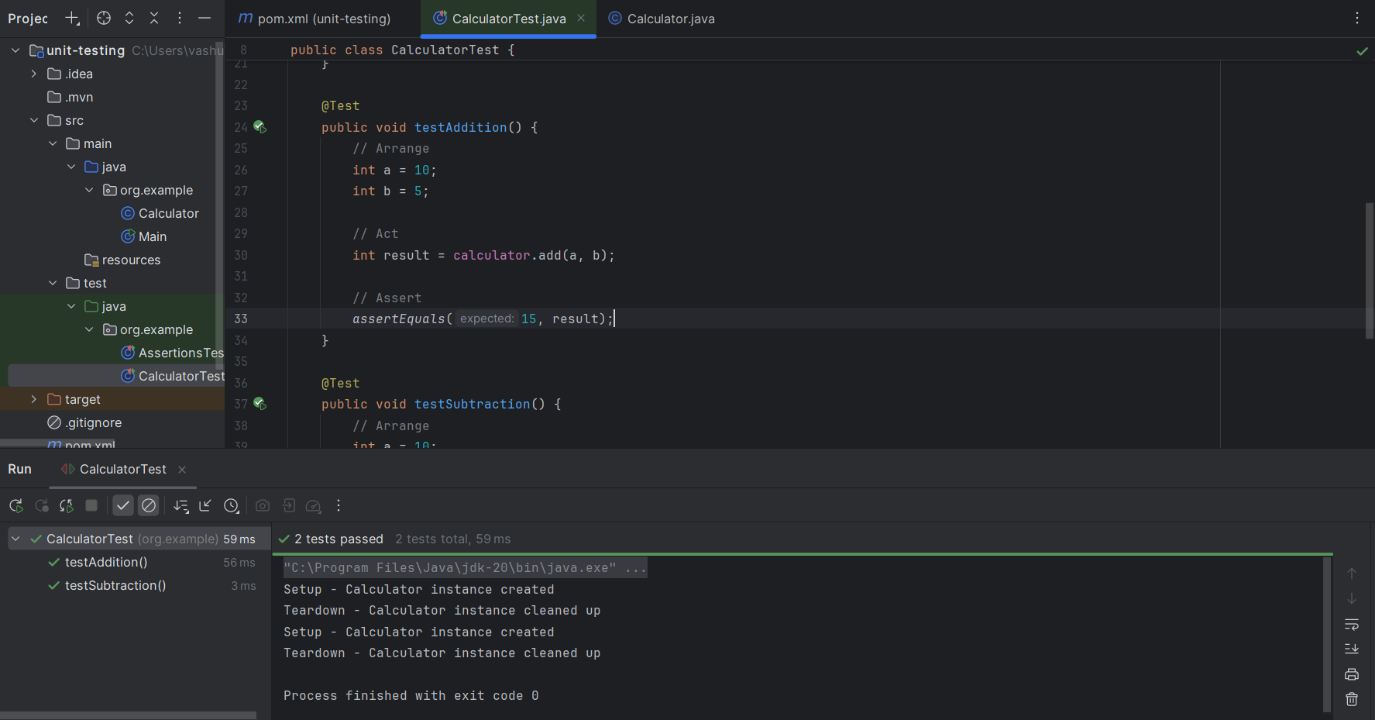
****

* The Arrange-Act-Assert (AAA) pattern is a widely used structure for writing clean and readable test methods. It divides the test into three logical parts:

● Arrange: Prepare all necessary objects, inputs, or dependencies required to perform the test.

● Act: Call the method or perform the operation that needs to be tested.

● Assert: Verify that the output or state is as expected.

****