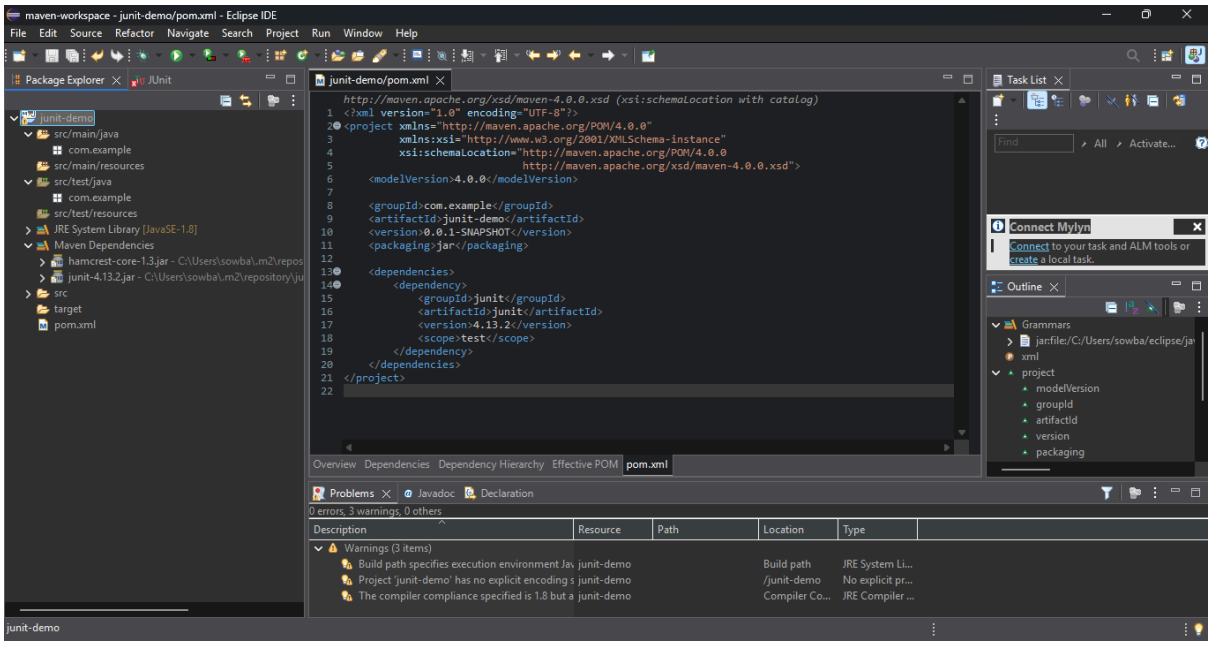
JUnit Testing Exercises Output

Exercise 1: Setting Up Junit

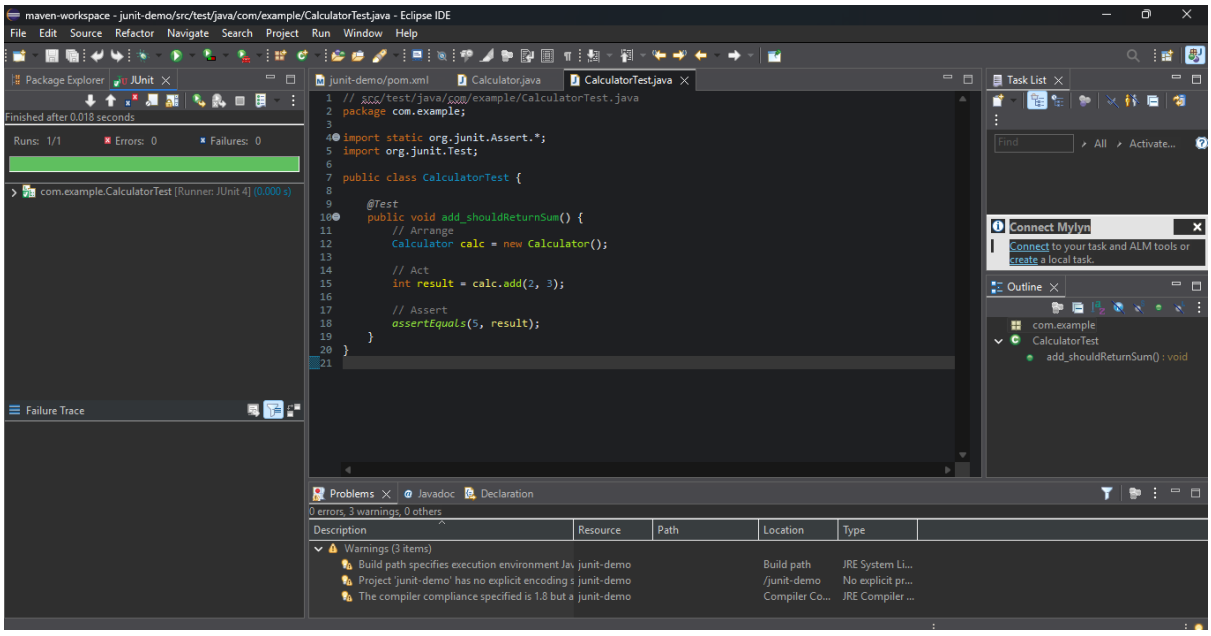
Output:



Set-up: Add JUnit as a *test-scope* dependency (or Gradle equivalent) and create a test class—JUnit is now wired in.

Exercise 2: Writing Basic JUnit Tests

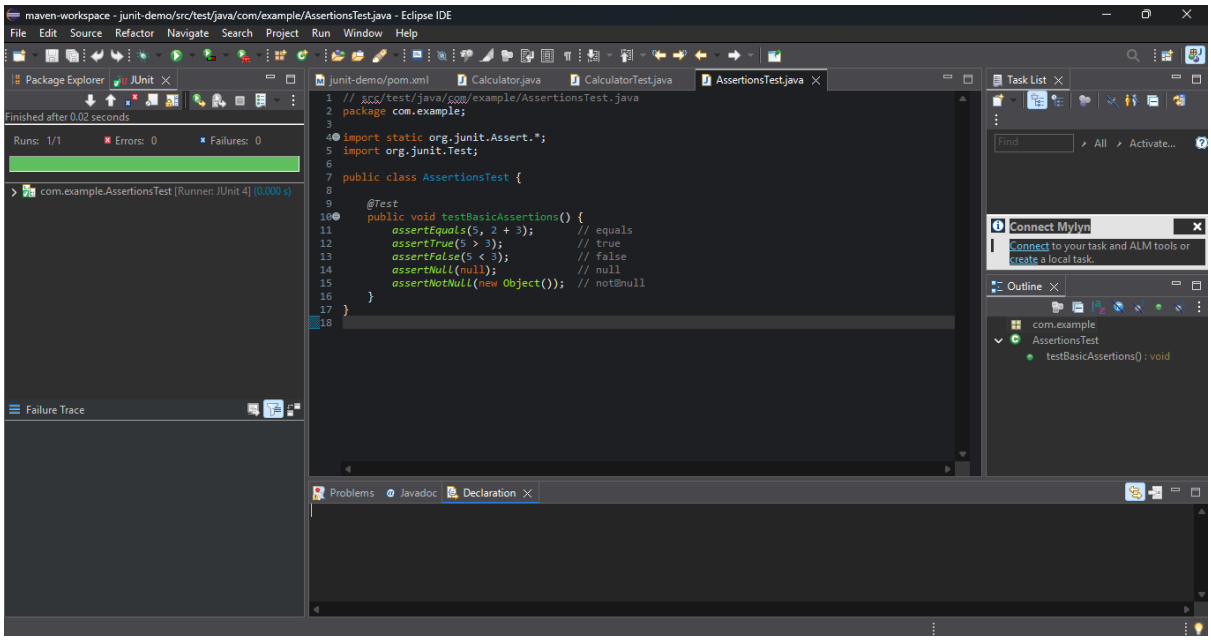
Output:



Basic tests: Write methods annotated with @Test to check the behaviour of your production code; run them via IDE or mvn test

Exercise 3: Assertions in Junit

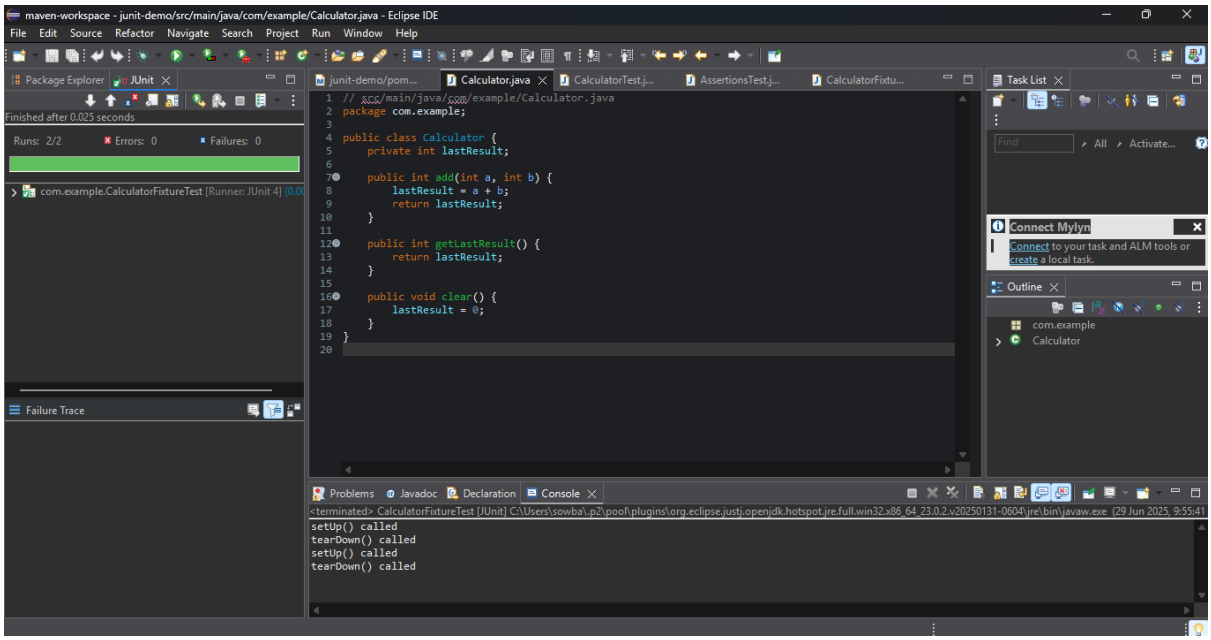
Output:



Assertions: Leverage the full assertion palette—assertEquals, assertTrue/False, assertNull/NotNull, etc.—to express expected outcomes clearly.

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

Output:



Structure & hygiene: Follow Arrange-Act-Assert to keep tests readable, and use @Before/@After (or JUnit 5's @BeforeEach/@AfterEach) to set up and tear down shared test fixtures, ensuring isolation and reliability.