**Exercise 2: Verifying Interactions**

* **Understand the Purpose**

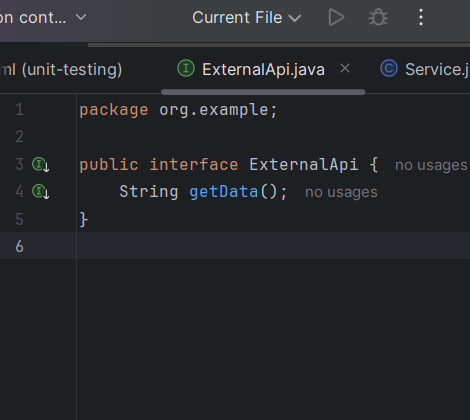
In this exercise, we want to test whether a specific method — like getData() — was actually called during execution.

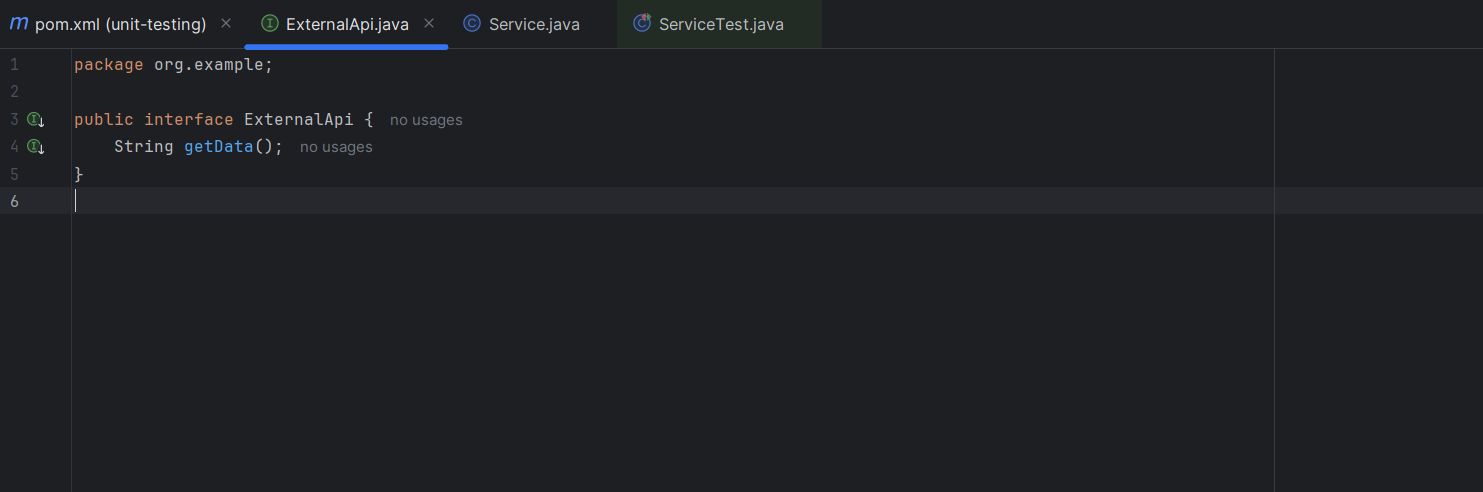
This is especially useful when:

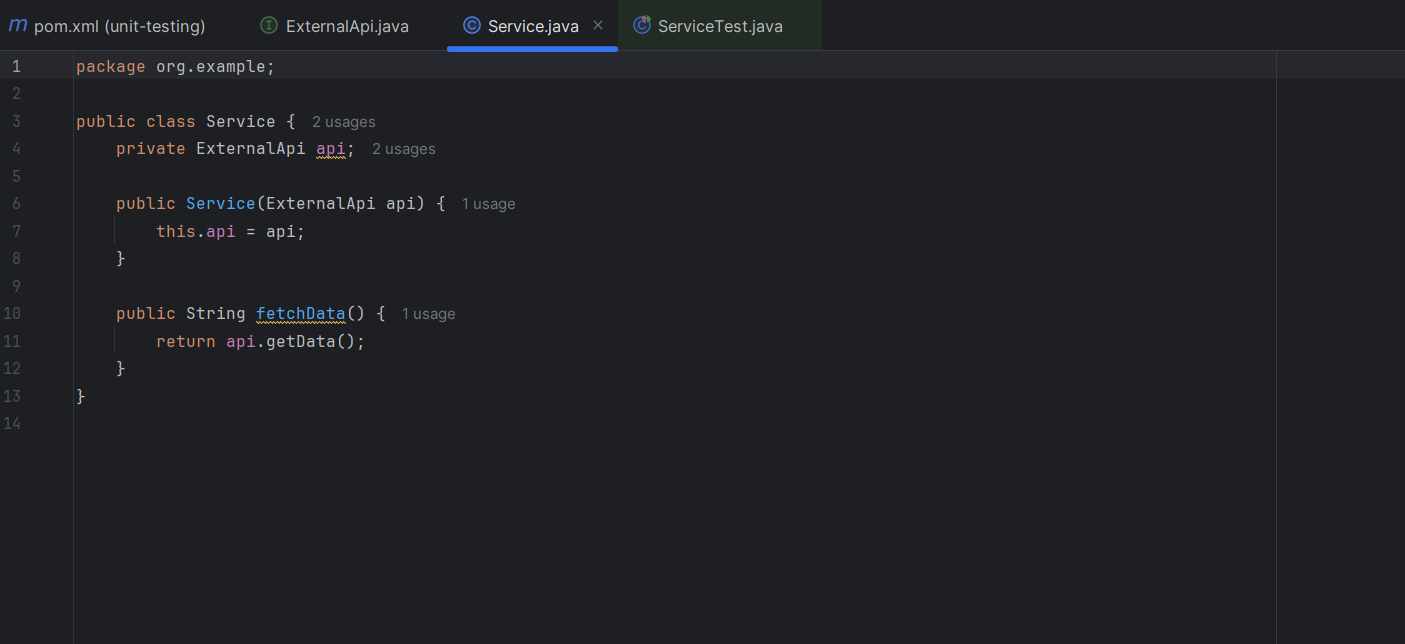
* The method returns void.
* The result is not as important as the interaction (e.g., sending an email, calling an external API).
* You want to verify behavior, not data.

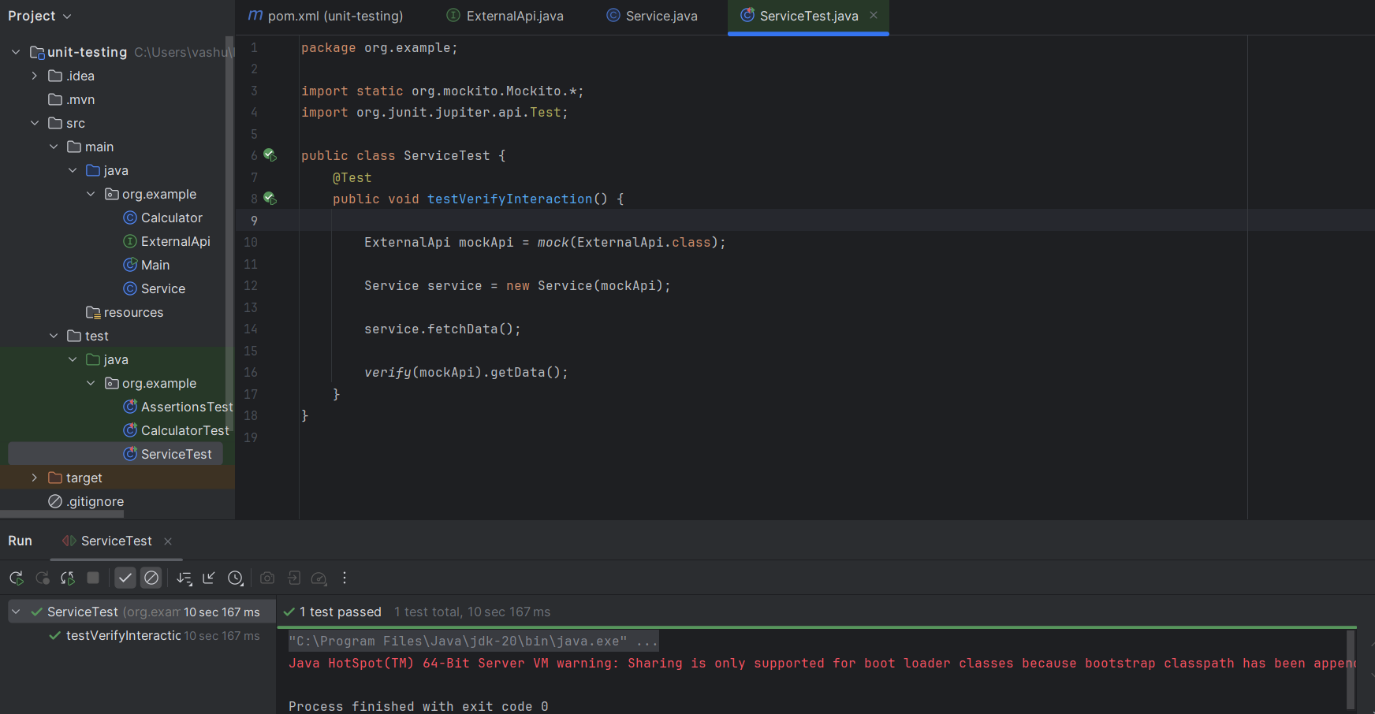
Mockito allows us to verify if a method was called, how many times it was called, and with what arguments.

* **Working Process**
* Create a Mock Object -We start by mocking the external dependency, which in this case is an interface called ExternalApi.
* Inject the Mock into the Service - We inject the mock into the service class we want to test. Suppose the service class depends on ExternalApi.
* Call the Method Being Tested - Now we call the actual method in Service that is expected to interact with the mock
* Verify the Interaction - Finally, we use verify() to check that getdata() was actually called on the mock
* **NOTE**
* **Mockito's verify() method** allows us to confirm interactions between objects.
* It’s especially useful when testing **void methods** or methods that produce **side effects**.
* This keeps our tests **behaviour-focused** rather than value-focused.

****

****

****

****