# Exercise 1: Mocking and Stubbing

Scenario:  
You need to test a service that depends on an external API. Use Mockito to mock the external API and stub its methods.  
  
Steps:  
1. Create a mock object for the external API.  
2. Stub the methods to return predefined values.  
3. Write a test case that uses the mock object.

## Solution Code:

import static org.mockito.Mockito.\*;

import static org.junit.jupiter.api.Assertions.\*;

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

// Assuming this is the external API interface

interface ExternalApi {

String getData();

}

// The service class that depends on the External API

class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

// Unit test class

public class MyServiceTest {

@Test

public void testExternalApi() {

// 1. Create mock object

ExternalApi mockApi = Mockito.mock(ExternalApi.class);

// 2. Stub method

when(mockApi.getData()).thenReturn("Mock Data");

// 3. Inject mock into service

MyService service = new MyService(mockApi);

// 4. Call method and verify result

String result = service.fetchData();

// 5. Assert expected outcome

assertEquals("Mock Data", result);

}

}