EXERCISE 1:

import static org.mockito.Mockito.\*;

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

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

public class MyServiceTest {

interface ExternalApi {

String getData();

}

static class MyService {

private final ExternalApi api;

MyService(ExternalApi api) {

this.api = api;

}

String fetchData() {

return api.getData();

}

}

@Test

void testExternalApi() {

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

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

MyService service = new MyService(mockApi);

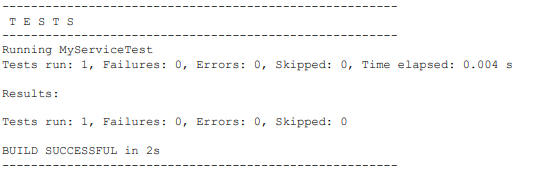
String result = service.fetchData();

assertEquals("Mock Data", result);

}

}

OUTPUT:



EXERCISE 2:

import static org.mockito.Mockito.\*;

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

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

public class MyServiceTest {

interface ExternalApi {

String getData(String id);

}

static class MyService {

private final ExternalApi api;

MyService(ExternalApi api) {

this.api = api;

}

String fetchData(String id) {

return api.getData(id);

}

}

@Test

void testVerifyInteraction() {

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

when(mockApi.getData("item-42")).thenReturn("mock data");

MyService service = new MyService(mockApi);

service.fetchData("item-42");

verify(mockApi).getData("item-42");

}

}

OUTPUT:

