**Exercise 2: Verifying Interactions**

public interface ExternalApi {

String getData();

void deleteData();

void processData();

}

**MyServiceTest.java:**

import static org.mockito.Mockito.\*;

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

import org.junit.jupiter.api.Test;

import org.mockito.InOrder;

import org.mockito.Mockito;

public class MyServiceTest {

@Test

public void testVerifySingleInteraction() {

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

MyService service = new MyService(mockApi);

service.fetchData();

verify(mockApi).getData();

}

@Test

public void testVerifyCallCount() {

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

MyService service = new MyService(mockApi);

service.fetchData();

verify(mockApi, times(1)).getData();

verify(mockApi, never()).deleteData();

}

@Test

public void testVerifyNoMoreInteractions() {

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

MyService service = new MyService(mockApi);

service.fetchData();

verify(mockApi).getData();

verifyNoMoreInteractions(mockApi);

}

@Test

public void testVerifyOrderOfCalls() {

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

MyService service = new MyService(mockApi);

service.process();

InOrder order = inOrder(mockApi);

order.verify(mockApi).getData();

order.verify(mockApi).processData();

}

@Test

public void testWithArgumentMatcher() {

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

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

MyService service = new MyService(mockApi);

String result = service.fetchData();

assertEquals("Any Data", result);

verify(mockApi).getData(); // still verifying exact method call

}

}

**Output:**

Tests run: 5, Failures: 0, Errors: 0, Skipped: 0