Mockito Hands-On Exercises

# 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 org.junit.jupiter.api.Test; import org.mockito.Mockito;

public class MyServiceTest { @Test

public 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);

}

}

**Program:**

**ExternalApi.java:**

public interface ExternalApi {

String getData();

}

**MyService.java:**

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java:**

import static org.mockito.Mockito.\*;

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

import org.junit.jupiter.api.Test;

public class MyServiceTest {

@Test

public void testExternalApi() {

ExternalApi mockApi = mock(ExternalApi.class);

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

MyService service = new MyService(mockApi);

String result = service.fetchData();

System.out.println("Result = " + result);

assertEquals("Mock Data", result);

}

}

**pom.xml:**

<project>

<modelVersion>4.0.0</modelVersion>

<groupId>mygroupid</groupId>

<artifactId>myartifactid</artifactId>

<version>0.0-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>org.junit.jupiter</groupId>

<artifactId>junit-jupiter</artifactId>

<version>5.7.1</version>

<scope>test</scope>

</dependency>

<dependency>

<groupId>org.mockito</groupId>

<artifactId>mockito-core</artifactId>

<version>4.0.0</version>

<scope>test</scope>

</dependency>

</dependencies>

</project>

**.replit:**

run = "javac -cp '.:junit-platform-console-standalone-1.7.1.jar:mockito-all-2.0.2-beta.jar' \*.java && java -jar junit-platform-console-standalone-1.7.1.jar --class-path .:mockito-all-2.0.2-beta.jar --select-class MyServiceTest"

**replit.nix:**

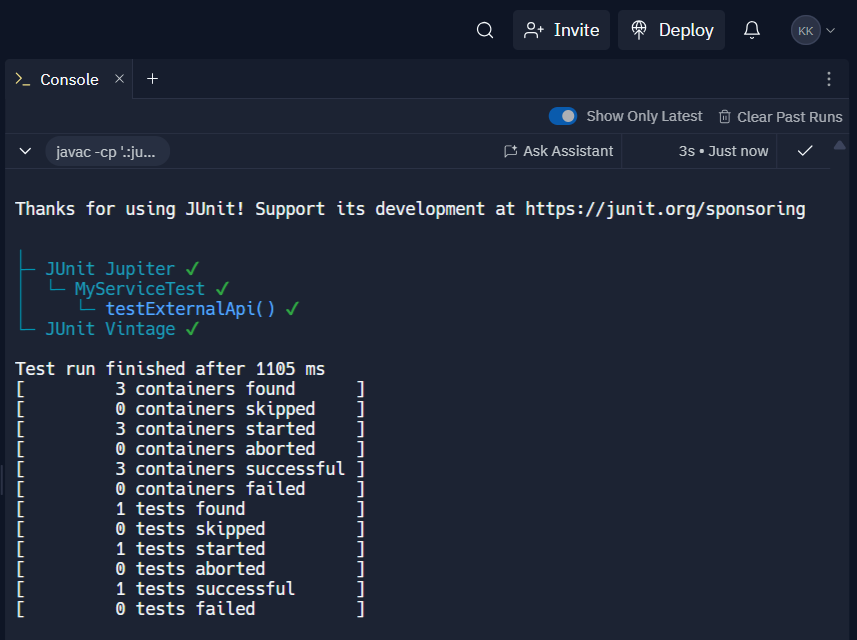
{ pkgs }: {

deps = [

pkgs.openjdk

];)

**Output:**

****

# Exercise 2: Verifying Interactions

Scenario:

You need to ensure that a method is called with specific arguments.

Steps:

1. Create a mock object.
2. Call the method with specific arguments.
3. Verify the interaction.

Solution Code:

import static org.mockito.Mockito.\*;

import org.junit.jupiter.api.Test; import org.mockito.Mockito;

public class MyServiceTest { @Test

public void testVerifyInteraction() {

ExternalApi mockApi = Mockito.mock(ExternalApi.class); MyService service = new MyService(mockApi); service.fetchData();

verify(mockApi).getData();

}

}

**Program:**

**ExternalApi.java:**

public interface ExternalApi {

String getData();

}

**MyService.java:**

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java:**

import static org.mockito.Mockito.\*;

import org.junit.jupiter.api.Test;

public class MyServiceTest {

@Test

public void testVerifyInteraction() {

ExternalApi mockApi = mock(ExternalApi.class);

MyService service = new MyService(mockApi);

service.fetchData();

verify(mockApi).getData();

}

}

**pom.xml:**

<project>

<modelVersion>4.0.0</modelVersion>

<groupId>mygroupid</groupId>

<artifactId>myartifactid</artifactId>

<version>0.0-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.12</version>

<type>jar</type>

</dependency>

</dependencies>

</project>

**.replit:**

run = "javac -cp '.:junit-platform-console-standalone-1.7.1.jar:mockito-core-4.0.0.jar:objenesis-3.2.jar:byte-buddy-1.11.22.jar' \*.java && java -jar junit-platform-console-standalone-1.7.1.jar --class-path .:mockito-core-4.0.0.jar:objenesis-3.2.jar:byte-buddy-1.11.22.jar --select-class MyServiceTest"

**replit.nix:**

{ pkgs }: {

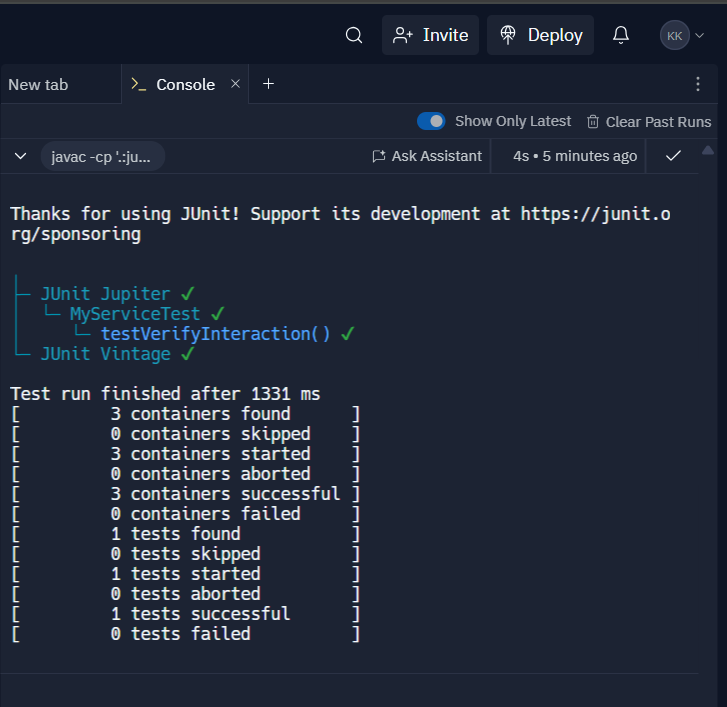
deps = [

pkgs.openjdk

];

}

**Output:**

****