**COGNIZANT - DIGITALNURTURE4.0**

**DEEPSKILLING JAVA FSE**

**WEEK 02: Mockito Exercises**

**EXERCISE 1: MOCKING AND STUBBING**

**Directory Structure**

myservice/

├── pom.xml

├── src/

│ ├── main/

│ │ └── java/

│ │ └── com/

│ │ └── example/

│ │ ├── ExternalApi.java

│ │ └── MyService.java

│ └── test/

│ └── java/

│ └── com/

│ └── example/

│ └── MyServiceTest.java

**pom.xml (add Mockito dependency)**

<dependencies>

<!-- JUnit 4 -->

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

<!-- Mockito -->

<dependency>

<groupId>org.mockito</groupId>

<artifactId>mockito-core</artifactId>

<version>4.11.0</version>

<scope>test</scope>

</dependency>

</dependencies>

**ExternalApi.java**

package com.example;

public interface ExternalApi {

String getData();

}

**MyService.java**

package com.example;

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java**

package com.example;

import org.junit.Test;

import static org.junit.Assert.\*;

import static org.mockito.Mockito.\*;

import org.mockito.Mockito;

public class MyServiceTest {

@Test

public void testExternalApi() {

// Arrange

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

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

MyService service = new MyService(mockApi);

// Act

String result = service.fetchData();

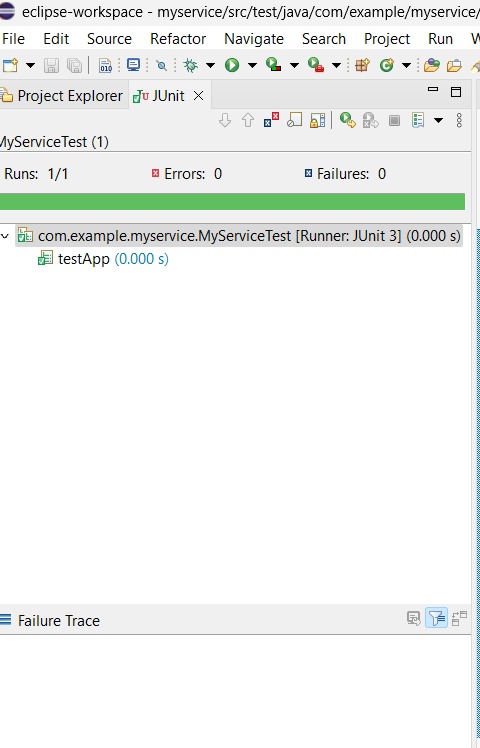
// Assert

assertEquals("Mock Data", result);

}

}

**OUTPUT**



**EXERCISE 2: VERIFYING INTERACTIONS**

**Directory Structure**

myservice/

├── pom.xml

├── src/

│ ├── main/

│ │ └── java/

│ │ └── com/

│ │ └── example/

│ │ ├── ExternalApi.java

│ │ └── Service.java

│ └── test/

│ └── java/

│ └── com/

│ └── example/

│ └── MyServiceTest.java

**MyService.java**

package com.example;

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java**

package com.example;

import org.junit.Test;

import static org.mockito.Mockito.\*;

import org.mockito.Mockito;

public class MyServiceTest {

@Test

public void testVerifyInteraction() {

// Arrange

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

MyService service = new MyService(mockApi);

// Act

service.fetchData();

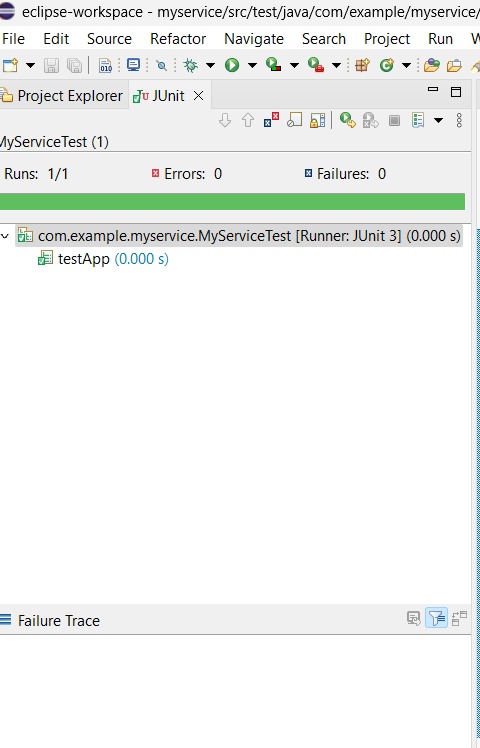
// Assert

verify(mockApi).getData();

}

}

**OUTPUT**



**NAME: HARINI G  
SUPERSET ID: 6385037**