**WEEK-2**

**MODULE-4**

**MOCKITO EXERCISES**

**SUPERSET ID:6407550**

**Exercise 6: Verifying Interaction Order**

**pom.xml**

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.example</groupId>

<artifactId>JUnitDemo</artifactId>

<version>0.0.1-SNAPSHOT</version>

<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>

<build>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.1</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

</plugins>

</build>

</project>

**src/main/java**

**FileHandler.java**

package com.example;

public interface FileHandler {

void openFile();

void writeFile(String data);

void closeFile();

}

**FileProcessor.java**

package com.example;

public class FileProcessor {

private FileHandler handler;

public FileProcessor(FileHandler handler) {

this.handler = handler;

}

public void processFile(String content) {

handler.openFile();

handler.writeFile(content);

handler.closeFile();

}

}

**src/test/java**

**FileProcessorTest.java**

package com.example;

import static org.mockito.Mockito.\*;

import org.junit.Test;

import org.mockito.InOrder;

public class FileProcessorTest {

@Test

public void testInteractionOrder() {

// Step 1: Create mock

FileHandler mockHandler = mock(FileHandler.class);

// Step 2: Use mock in real object

FileProcessor processor = new FileProcessor(mockHandler);

processor.processFile("Hello, file!");

// Step 3: Verify call order

InOrder inOrder = inOrder(mockHandler);

inOrder.verify(mockHandler).openFile();

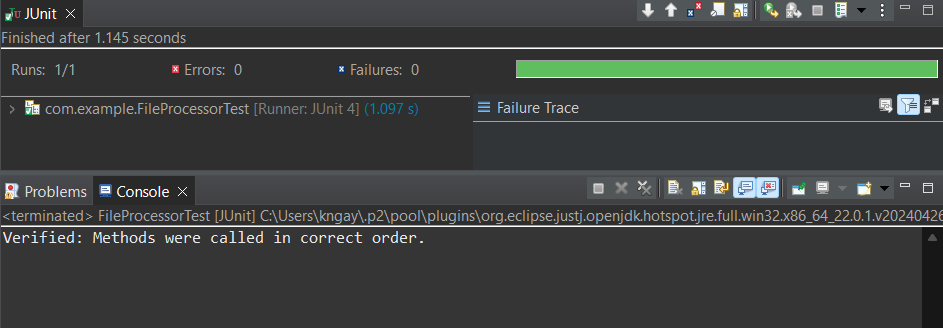
inOrder.verify(mockHandler).writeFile("Hello, file!");

inOrder.verify(mockHandler).closeFile();

System.out.println("Verified: Methods were called in correct order.");

}

}

**OUTPUT:**