Exercise 3: Mocking File I/O

**FileReader.java:**

package com.example.mockito;

public interface FileReader {

String read();

}

**FileWriter.java:**

package com.example.mockito;

public interface FileWriter {

void write(String data);

}

**FileService.java:**

package com.example.mockito;

public class FileService {

private final FileReader reader;

private final FileWriter writer;

public FileService(FileReader reader, FileWriter writer) {

this.reader = reader;

this.writer = writer;

}

public String processFile() {

String content = reader.read();

writer.write("Processed " + content);

return "Processed " + content;

}

}

**FileServiceTest.java:**

package com.example.mockito;

import org.junit.jupiter.api.Test;

import static org.mockito.Mockito.\*;

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

public class FileServiceTest {

@Test

public void testServiceWithMockFileIO() {

FileReader mockReader = *mock*(FileReader.class);

FileWriter mockWriter = *mock*(FileWriter.class);

*when*(mockReader.read()).thenReturn("Mock File Content");

FileService fileService = new FileService(mockReader, mockWriter);

String result = fileService.processFile();

*assertEquals*("Processed Mock File Content", result);

*verify*(mockWriter).write("Processed Mock File Content");

}

}

**Output:**

