**CODE:**

package com.example;

public interface FileReader {

String read();

}

package com.example;

public class FileService {

private FileReader reader;

private FileWriter writer;

public FileService(FileReader reader, FileWriter writer) {

this.reader = reader;

this.writer = writer;

}

public String processFile() {

String content = reader.read();

String processed = "Processed " + content;

writer.write(processed);

return processed;

}

}

package com.example;

public interface FileWriter {

void write(String data);

}

package com.example;

import static org.mockito.Mockito.\*;

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

import org.junit.jupiter.api.Test;

public class FileServiceTest {

@Test

public void testServiceWithMockFileIO() {

// Step 1: Create mock reader and writer

FileReader mockFileReader = mock(FileReader.class);

FileWriter mockFileWriter = mock(FileWriter.class);

// Step 2: Stub read method

when(mockFileReader.read()).thenReturn("Mock File Content");

// Step 3: Inject into FileService

FileService fileService = new FileService(mockFileReader, mockFileWriter);

// Step 4: Process file and verify result

String result = fileService.processFile();

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

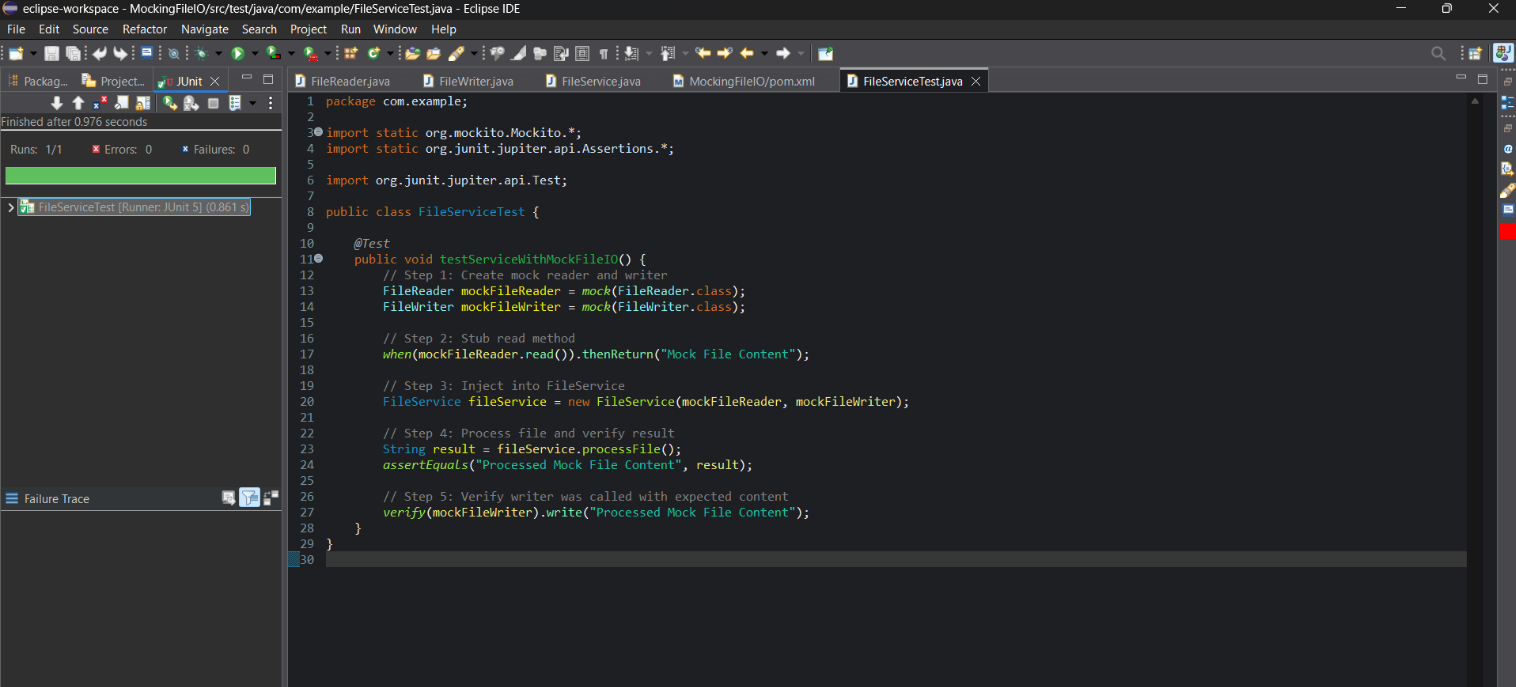
// Step 5: Verify writer was called with expected content

verify(mockFileWriter).write("Processed Mock File Content");

}

}

**OUTPUT:**

****