**Exercise 4.3: Mocking File I/O**

**CODE:**

**FileReader Interface: -**

package org.example;  
  
public interface FileReader{  
 String read();  
}

**FileWriter Interface: -**

package org.example;  
  
public interface FileWriter{  
 void write(String content);  
}

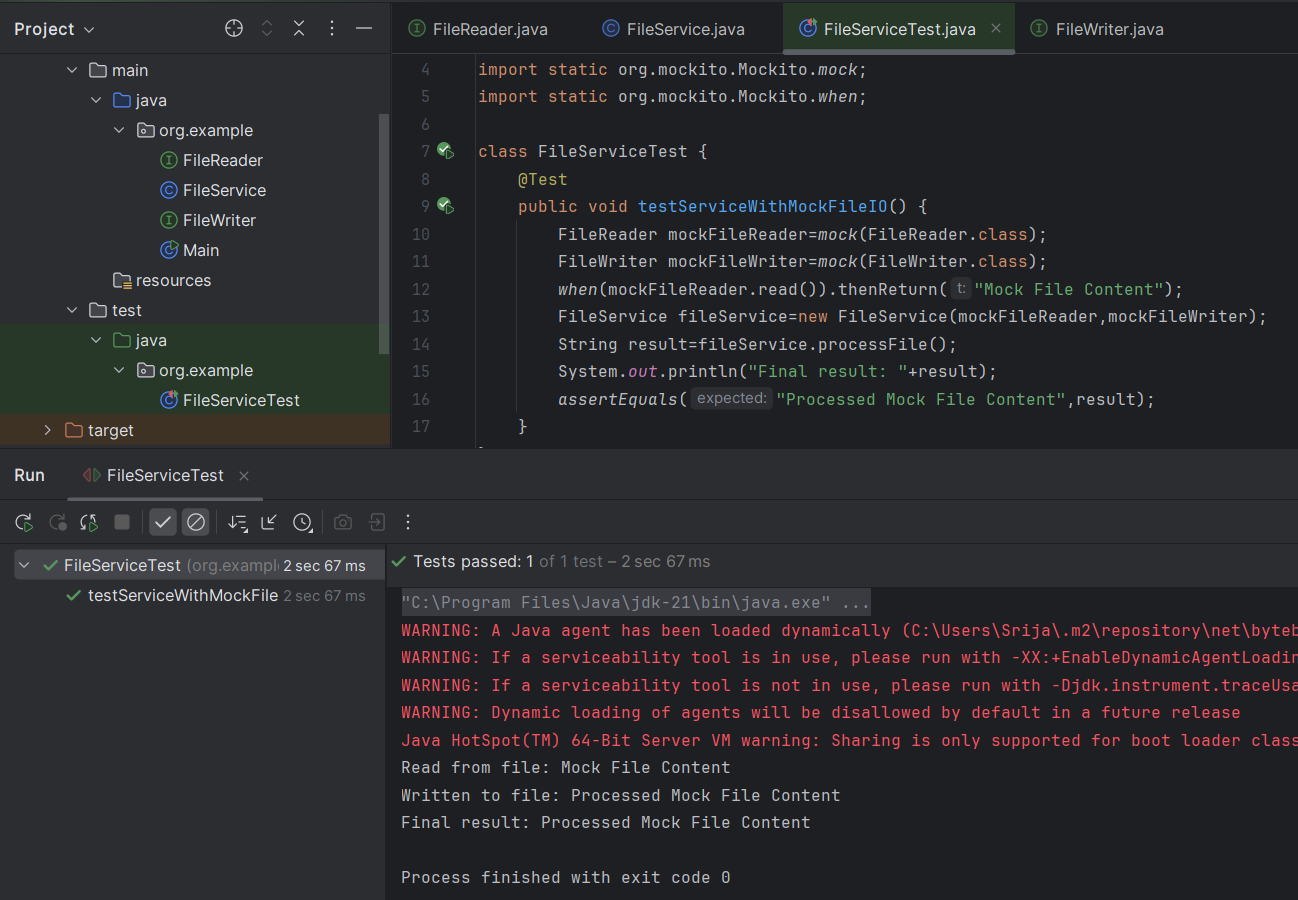
**FileService Class: -**

package org.example;  
  
public class FileService{  
 private FileReader fileReader;  
 private FileWriter fileWriter;  
 public FileService(FileReader fileReader, FileWriter fileWriter) {  
 this.fileReader=fileReader;  
 this.fileWriter=fileWriter;  
 }  
 public String processFile() {  
 String content=fileReader.read();  
 System.*out*.println("Read from file: "+content);  
 String processed="Processed "+content;  
 fileWriter.write(processed);  
 System.*out*.println("Written to file: "+processed);  
 return processed;  
 }  
}

**FileServiceTest Class: -**

package org.example;  
import org.junit.jupiter.api.Test;  
import static org.junit.jupiter.api.Assertions.\*;  
import static org.mockito.Mockito.*mock*;  
import static org.mockito.Mockito.*when*;  
  
class FileServiceTest {  
 @Test  
 public void testServiceWithMockFileIO() {  
 FileReader mockFileReader=*mock*(FileReader.class);  
 FileWriter mockFileWriter=*mock*(FileWriter.class);  
 *when*(mockFileReader.read()).thenReturn("Mock File Content");  
 FileService fileService=new FileService(mockFileReader,mockFileWriter);  
 String result=fileService.processFile();  
 System.*out*.println("Final result: "+result);  
 *assertEquals*("Processed Mock File Content",result);  
 }  
}

**OUTPUT:**

****