**Design Patterns and Principles**

**Exercise 1: Implementing the Singleton Pattern**

**Class 1: Logger.java**package SingletonPatternExample;

public class Logger {

private static Logger *instance*;

private Logger() {

System.*out*.println("Logger initialized");

}

public static Logger getInstance() {

if (*instance* == null) {

*instance* = new Logger();

}

return *instance*;

}

public void log(String message) {

System.*out*.println("Log: " + message);

}

}

**Class 2: LoggerTest.java**

package SingletonPatternExample;

public class LoggerTest {

public static void main(String[] args) {

Logger logger1 = Logger.*getInstance*();

logger1.log("First message");

Logger logger2 = Logger.*getInstance*();

logger2.log("Second message");

if (logger1 == logger2) {

System.*out*.println("Both logger instances are the same (Singleton verified).");

} else {

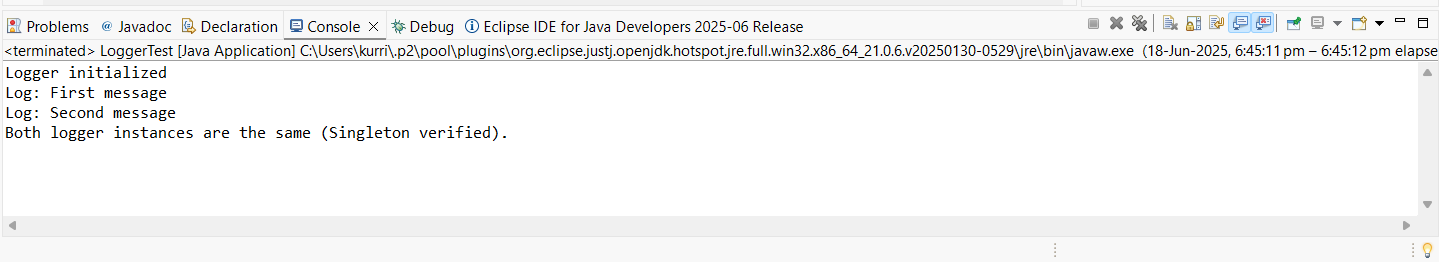
System.*out*.println("Logger instances are different (Singleton failed).");

}

}

}

**Output:**

****

**Exercise 2: Implementing the Factory Method Pattern**

**Package 1: document**

**Class 1: Document.java**package document;

public interface Document {

void open();

}

**Class 2: WordDocument.java**

package document;

public class WordDocument implements Document {

public void open() {

System.out.println("Opening Word document.");

}

}

**Class 3: PdfDocument.java**

package document;

public class PdfDocument implements Document {

public void open() {

System.out.println("Opening PDF document.");

}

}

**Class 4: ExcelDocument.java**

package document;

public class ExcelDocument implements Document {

public void open() {

System.out.println("Opening Excel document.");

}

}

**Package 2: factory**

**Class 1: DocumentFactory.java (interface)**

package document;

public interface Document {

void open();

}

**Class 2: WordFactory.java**

package factory;

import document.Document;

import document.WordDocument;

public class WordFactory extends DocumentFactory {

public Document createDocument() {

return new WordDocument();

}

}

**Class 3: PdfFactory.java**

package factory;

import document.Document;

import document.PdfDocument;

public class PdfFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

**Class 4: ExcelFactory.java**

package factory;

import document.Document;

import document.ExcelDocument;

public class ExcelFactory extends DocumentFactory {

public Document createDocument() {

return new ExcelDocument();

}

}

**Package 3: main**

**Class 1: FactoryTest.java**

package main;

import factory.\*;

import document.Document;

public class FactoryTest {

public static void main(String[] args) {

DocumentFactory wordFactory = new WordFactory();

Document wordDoc = wordFactory.createDocument();

wordDoc.open();

DocumentFactory pdfFactory = new PdfFactory();

Document pdfDoc = pdfFactory.createDocument();

pdfDoc.open();

DocumentFactory excelFactory = new ExcelFactory();

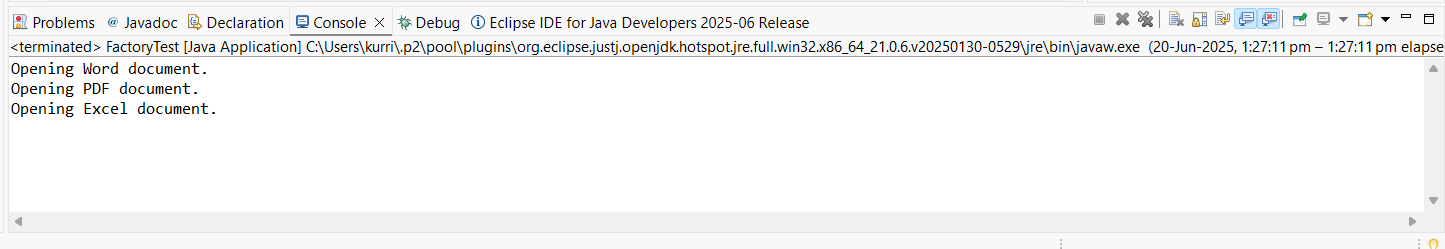
Document excelDoc = excelFactory.createDocument();

excelDoc.open();

}

}

**Output:**

****