**Example 1 : Implementing the Singleton Pattern**

**Project Name: SingletonPatternExample**

**Program:**

**LOGGER CLASS**

public class Main {

    public static void main(String[] args) {

        Logger logger1 = Logger.getInstance();

        Logger logger2 = Logger.getInstance();

        logger1.log("This is the first log.");

        logger2.log("This is the second log.");

        if (logger1 == logger2) {

            System.out.println("Both logger1 and logger2 are the same instance.");

        } else {

            System.out.println("Different instances detected! Singleton failed.");

        }

    }

}

**Main.java**

public class Main {

public static void main(String[] args) {

// Getting instances

Logger logger1 = Logger.getInstance();

Logger logger2 = Logger.getInstance();

// Logging messages

logger1.log("This is the first log.");

logger2.log("This is the second log.");

// Test if both logger1 and logger2 refer to the same object

if (logger1 == logger2) {

System.out.println("Both logger1 and logger2 are the same instance.");

} else {

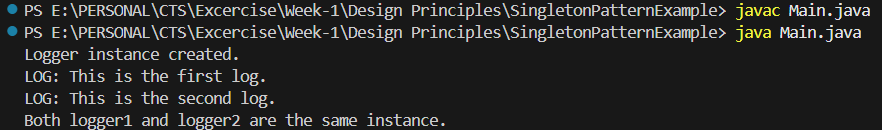
System.out.println("Different instances detected! Singleton failed.");

}

}

}

**OUTPUT:**

****

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

**Project Name:** **FactoryMethodPatternExample**

**Program:**

public class FactoryMethodPatternExample {

// Document Interface

interface Document {

void open();

}

// Word Document

static class WordDocument implements Document {

public void open() {

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

}

}

// PDF Document

static class PdfDocument implements Document {

public void open() {

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

}

}

// Excel Document

static class ExcelDocument implements Document {

public void open() {

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

}

}

// Abstract Factory

abstract static class DocumentFactory {

public abstract Document createDocument();

}

// Word Factory

static class WordFactory extends DocumentFactory {

public Document createDocument() {

return new WordDocument();

}

}

// PDF Factory

static class PdfFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

// Excel Factory

static class ExcelFactory extends DocumentFactory {

public Document createDocument() {

return new ExcelDocument();

}

}

// Main Method

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();

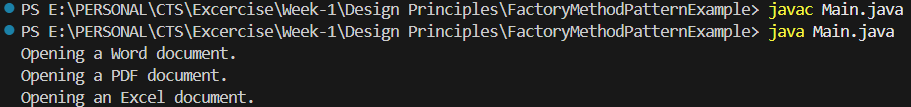
Document excelDoc = excelFactory.createDocument();

excelDoc.open();

}

}

**Output:**

****