**WEEK-1**

**Design Patterns and Principles**

**Exercise 1: Implementing the Singleton Pattern**

**Code:**

**Logger.java:**

public class Logger {

private static Logger instance = new Logger();

private Logger() {

System.out.println("Logger instance created.");

}

public static Logger getInstance() {

return instance;

}

public void log(String message) {

System.out.println("LOG: " + message);

}

}

**Test.java:**

pack public class Main {

public static void main(String[] args) {

Logger logger1 = Logger.getInstance();

Logger logger2 = Logger.getInstance();

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

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

if (logger1 == logger2) {

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

} else {

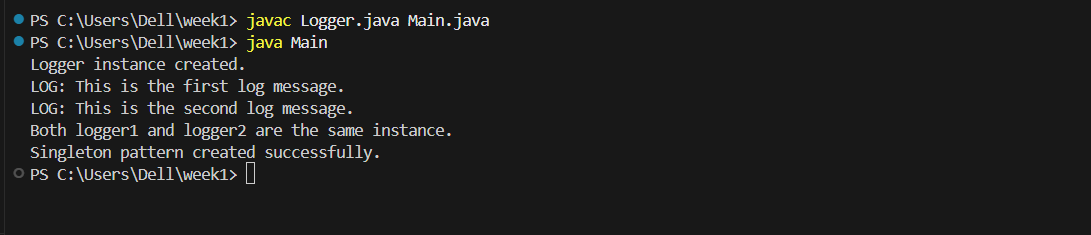
System.out.println("logger1 and logger2 are different instances.");

}

System.out.println("Singleton pattern created successfully.");

}

}

**Output:**

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

**Code:**

// Document.java

public interface Document {

void open();

}

// WordDocument.java

public class WordDocument implements Document {

public void open() {

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

}

}

// PdfDocument.java

public class PdfDocument implements Document {

public void open() {

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

}

}

// ExcelDocument.java

public class ExcelDocument implements Document {

public void open() {

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

}

}

// DocumentFactory.java

public abstract class DocumentFactory {

public abstract Document createDocument();

}

// WordDocumentFactory.java

public class WordDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new WordDocument();

}

}

// PdfDocumentFactory.java

public class PdfDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

// ExcelDocumentFactory.java

public class ExcelDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new ExcelDocument();

}

}

**Main.java**

public class Main {

public static void main(String[] args) {

DocumentFactory wordFactory = new WordDocumentFactory();

Document wordDoc = wordFactory.createDocument();

wordDoc.open();

DocumentFactory pdfFactory = new PdfDocumentFactory();

Document pdfDoc = pdfFactory.createDocument();

pdfDoc.open()

DocumentFactory excelFactory = new ExcelDocumentFactory();

Document excelDoc = excelFactory.createDocument();

excelDoc.open();

System.out.println("Factory method pattern implemented successfully.");

}

}

**Output:**

