**Design Principles and Patterns**

**Exercise 1: Implementing the Singleton Pattern**

public class Main {

static class Logger {

private static Logger instance;

private Logger() {

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

}

public static synchronized Logger getInstance() {

if (instance == null) {

instance = new Logger();

}

return instance;

}

public void log(String message) {

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

}

}

// Main method to test the Singleton

public static void main(String[] args) {

Logger logger1 = Logger.getInstance();

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

Logger logger2 = Logger.getInstance();

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

if (logger1 == logger2) {

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

} else {

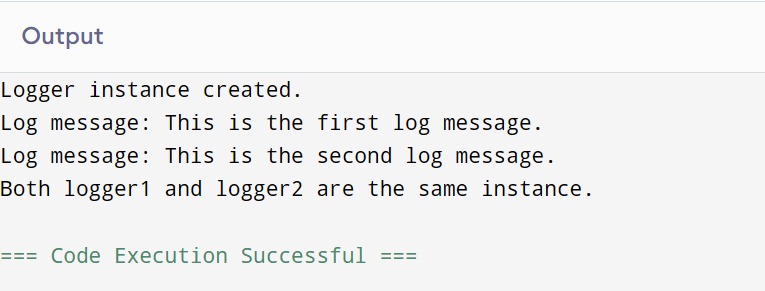
System.out.println("Different instances exist!");

}

}

}

**Output:**



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

public class Main {

public static void main(String[] args) {

DocumentFactory wordFactory = new WordDocumentFactory();

Document word = wordFactory.createDocument();

word.open();

DocumentFactory pdfFactory = new PdfDocumentFactory();

Document pdf = pdfFactory.createDocument();

pdf.open();

DocumentFactory excelFactory = new ExcelDocumentFactory();

Document excel = excelFactory.createDocument();

excel.open();

}

}

abstract class Document {

public abstract void open();

}

class WordDocument extends Document {

public void open() {

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

}

}

class PdfDocument extends Document {

public void open() {

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

}

}

class ExcelDocument extends Document {

public void open() {

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

}

}

abstract class DocumentFactory {

public abstract Document createDocument();

}

class WordDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new WordDocument();

}

}

class PdfDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

class ExcelDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new ExcelDocument();

}

}

**Output :**

