**WEEK 1 ASSIGNMENT**

**Design Pattern & Principles**

**- Aritra Das**

**Superset ID: 6362294**

**Exercise 1: Implementing the Singleton Pattern**

1. **Code:**

package Design\_Patterns\_Principles.Exercise\_1;

class Logger {

private static volatile Logger loggerInstance;

private Logger() {

System.out.println("Logger Class is initialized");

}

public static Logger getInstance() {

if (loggerInstance == null) {

synchronized (Logger.class) {

if (loggerInstance == null) {

loggerInstance = new Logger();

}

}

}

return loggerInstance;

}

public void log(String message) {

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

}

}

public class Main {

public static void main(String[] args) {

Logger obj1 = Logger.getInstance();

Logger obj2 = Logger.getInstance();

obj1.log("Message from first object");

obj2.log("Message from second object");

if (obj1 == obj2)

System.out.println("Singleton Confirmed");

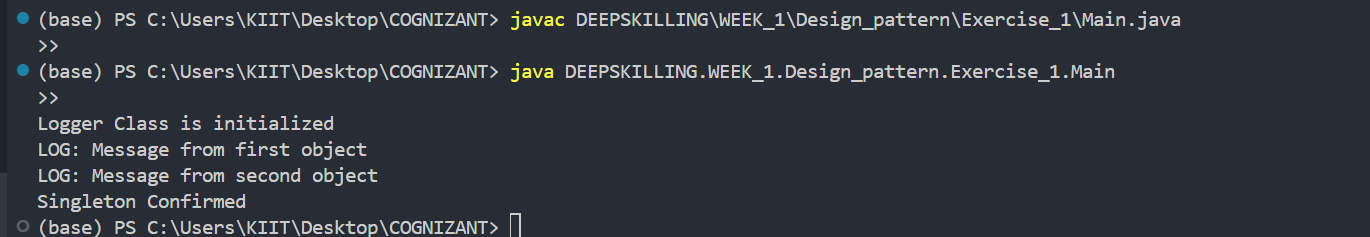
else

System.out.println("Singleton Failed");

}

}

1. **Output:**



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

1. **Code:**

package Design\_Patterns\_Principles.Exercise\_2;

interface Document {

void open();

}

class WordDocument implements Document {

@Override

public void open() {

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

}

}

class PdfDocument implements Document {

@Override

public void open() {

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

}

}

class ExcelDocument implements Document {

@Override

public void open() {

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

}

}

abstract class DocumentFactory {

public abstract Document createDocument();

}

class WordDocumentFactory extends DocumentFactory {

@Override

public Document createDocument() {

return new WordDocument();

}

}

class PdfDocumentFactory extends DocumentFactory {

@Override

public Document createDocument() {

return new PdfDocument();

}

}

class ExcelDocumentFactory extends DocumentFactory {

@Override

public Document createDocument() {

return new ExcelDocument();

}

}

public class Main {

public static void main(String[] args) {

openDocument(new WordDocumentFactory());

openDocument(new PdfDocumentFactory());

openDocument(new ExcelDocumentFactory());

}

private static void openDocument(DocumentFactory factory) {

Document doc = factory.createDocument();

doc.open();

}

}

1. **Output:**

