Digital Nurture 4.0 – Deep Skilling Program

Java Full Stack Engineer (FSE)

Week 1 – Mandatory Exercises Submission

Name: Khushi

GitHub Profile: https://github.com/KhushiSGowda

Email:khushisnagraj12@gmail.com

Skills Covered:

• Design Patterns and Principles

• Data Structures and Algorithms

Design Patterns and Principles

**Exercise 1: Implementing the Singleton Pattern**

**Program:**

public class SingletonPatternDemo {

// Logger class using Singleton Pattern

static class Logger {

private static Logger instance;

// Private constructor

private Logger() {

System.out.println("Logger Initialized");

}

// Public method to get the single instance

public static Logger getInstance() {

if (instance == null) {

instance = new Logger();

}

return instance;

}

public void log(String message) {

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

}

}

// Main method to test Singleton

public static void main(String[] args) {

Logger logger1 = Logger.getInstance();

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

Logger logger2 = Logger.getInstance();

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

System.out.println("logger1 and logger2 are same instance: " + (logger1 == logger2));

}

}

**Output:**

****

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

**Program:**

public class FactoryMethodPatternDemo {

// Document interface

interface Document {

void open();

}

// Concrete Document classes

static class WordDocument implements Document {

public void open() {

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

}

}

static class PdfDocument implements Document {

public void open() {

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

}

}

static class ExcelDocument implements Document {

public void open() {

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

}

}

// Abstract Factory

static abstract class DocumentFactory {

public abstract Document createDocument();

}

// Concrete Factories

static class WordFactory extends DocumentFactory {

public Document createDocument() {

return new WordDocument();

}

}

static class PdfFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

static class ExcelFactory extends DocumentFactory {

public Document createDocument() {

return new ExcelDocument();

}

}

// Main method

public static void main(String[] args) {

DocumentFactory pdfFactory = new PdfFactory();

Document pdf = pdfFactory.createDocument();

pdf.open();

DocumentFactory wordFactory = new WordFactory();

Document word = wordFactory.createDocument();

word.open();

DocumentFactory excelFactory = new ExcelFactory();

Document excel = excelFactory.createDocument();

excel.open();

} }

**Output:**

****