**Exercise 1: Implementing the Singleton Pattern**

Code –

* *SingletonPatternExample/src/com/example/singleton/Logger.java*

package com.example.singleton;

public class Logger {

private static Logger *logger*;

private Logger() {}

public static Logger getInstance() {

if (*logger* == null) {

*logger* = new Logger();

}

return *logger*;

}

public void log(String str) {

System.*out*.println(str);

}

}

* *SingletonPatternExample/src/com/example/singleton/Test.java*

package com.example.singleton;

public class Test {

public static void main(String[] args) {

Logger l1 = Logger.*getInstance*();

Logger l2 = Logger.*getInstance*();

l1.log(“Instance 1 is running…”);

l2.log(“Instance 2 is running…”);

System.*out*.println();

if (l1==l2) {

System.*out*.println(“Both instances are the same”);

} else {

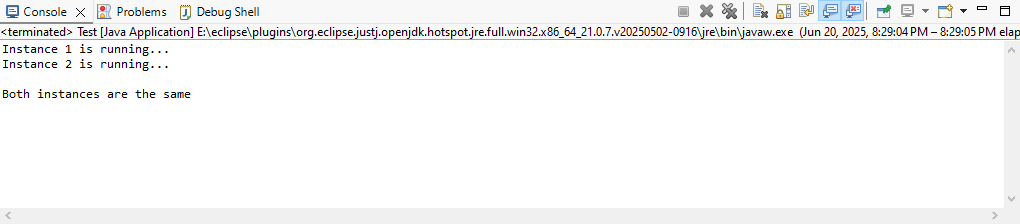
System.*out*.println(“Two different instances created”);

}

}

}

Output –



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

Code –

* *FactoryMethodPatternExample/src/com/example/factory/WordDocument.java*

package com.example.factory;

public interface WordDocument {

public void open(String filename);

}

* *FactoryMethodPatternExample/src/com/example/factory/PdfDocument.java*

package com.example.factory;

public interface PdfDocument {

public void open(String filename);

}

* *FactoryMethodPatternExample/src/com/example/factory/ExcelDocument.java*

package com.example.factory;

public interface ExcelDocument {

public void open(String filename);

}

* *FactoryMethodPatternExample/src/com/example/factory/WordDocClass.java*

package com.example.factory;

public class WordDocClass implements WordDocument {

@Override

public void open(String filename) {

System.*out*.println("Word file opened: " + filename);

}

}

* *FactoryMethodPatternExample/src/com/example/factory/PdfDocClass.java*

package com.example.factory;

public class PdfDocClass implements PdfDocument {

@Override

public void open(String filename) {

System.*out*.println("Pdf file opened: " + filename);

}

}

* *FactoryMethodPatternExample/src/com/example/factory/ExcelDocClass.java*

package com.example.factory;

public class ExcelDocClass implements ExcelDocument {

@Override

public void open(String filename) {

System.*out*.println("Excel file opened: " + filename);

}

}

* *FactoryMethodPatternExample/src/com/example/factory/DocumentFactory.java*

package com.example.factory;

public abstract class DocumentFactory<T> {

public abstract T createDocument();

}

* *FactoryMethodPatternExample/src/com/example/factory/WordDocFactory.java*

package com.example.factory;

public class WordDocFactory extends DocumentFactory<WordDocument> {

@Override

public WordDocument createDocument() {

System.*out*.println("New word document created");

return new WordDocClass();

}

}

* *FactoryMethodPatternExample/src/com/example/factory/PdfDocFactory.java*

package com.example.factory;

public class PdfDocFactory extends DocumentFactory<PdfDocument> {

@Override

public PdfDocument createDocument() {

System.*out*.println("New pdf document created");

return new PdfDocClass();

}

}

* *FactoryMethodPatternExample/src/com/example/factory/ExcelDocFactory.java*

package com.example.factory;

public class ExcelDocFactory extends DocumentFactory<ExcelDocument> {

@Override

public ExcelDocument createDocument() {

System.*out*.println("New excel document created");

return new ExcelDocClass();

}

}

* *FactoryMethodPatternExample/src/com/example/factory/Test.java*

package com.example.factory;

public class Test {

public static void main(String[] args) {

DocumentFactory<WordDocument> wordFac = new WordDocFactory();

DocumentFactory<PdfDocument> pdfFac = new PdfDocFactory();

DocumentFactory<ExcelDocument> excelFac = new ExcelDocFactory();

WordDocument file1 = (WordDocument) wordFac.createDocument();

file1.open("file1.docx");

System.*out*.println();

PdfDocument file2 = (PdfDocument) pdfFac.createDocument();

file2.open("file2.pdf");

System.*out*.println();

ExcelDocument file3 = (ExcelDocument) excelFac.createDocument();

file3.open("file3.xlsx");

System.*out*.println();

}

}

Output –

