**1. Implementing the Singleton Pattern**

**CODE:**

class AppLogger

{

private static AppLogger instance;

private AppLogger()

{

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

}

public static AppLogger getInstance()

{

if(instance==null)

{

instance=new AppLogger();

}

return instance;

}

public void log(String msg)

{

System.out.println(msg);

}

}

public class LoggerTest

{

public static void main(String[] args)

{

AppLogger loggerOne=AppLogger.getInstance();

loggerOne.log("Application started");

AppLogger loggerTwo=AppLogger.getInstance();

loggerTwo.log("Another message logged successfully!");

if(loggerOne==loggerTwo)

{

System.out.println("Only one instance is used!");

}

else

{

System.out.println("Different instances are used!");

}

}

}

**OUTPUT:**

**AppLogger instance created**

**Application started**

**Another message logged successfully!**

**Only one instance is used!**

**2. Implementing the Factory Method Pattern**

**CODE:**

interface AppDocument {

void open();

void save();

void print();

}

class WordFile implements AppDocument {

public void open() {

System.out.println("Opening the word document....");

}

public void save() {

System.out.println("Saving the word document....");

}

public void print() {

System.out.println("Printing the word document....");

}

}

class PdfFile implements AppDocument {

public void open() {

System.out.println("Opening the Pdf document....");

}

public void save() {

System.out.println("Saving the pdf document....");

}

public void print() {

System.out.println("Printing the pdf document....");

}

}

class ExcelFile implements AppDocument {

public void open() {

System.out.println("Opening the excel document....");

}

public void save() {

System.out.println("Saving the excel document....");

}

public void print() {

System.out.println("Printing the excel document....");

}

}

class DocFactory {

public AppDocument createDocument(String type) {

if (type.equalsIgnoreCase("WORD")) {

return new WordFile();

} else if (type.equalsIgnoreCase("PDF")) {

return new PdfFile();

} else if (type.equalsIgnoreCase("EXCEL")) {

return new ExcelFile();

}

return null;

}

}

public class DocumentTest {

public static void main(String[] args) {

DocFactory factory = new DocFactory();

AppDocument word = factory.createDocument("word");

word.open();

word.save();

word.print();

AppDocument pdf = factory.createDocument("pdf");

pdf.open();

pdf.save();

pdf.print();

AppDocument excel = factory.createDocument("excel");

excel.open();

excel.save();

excel.print();

}

}**OUTPUT:**

Opening the word document....

Saving the word document....

Printing the word document....

Opening the Pdf document....

Saving the pdf document....

Printing the pdf document....

Opening the excel document....

Saving the excel document....

Printing the excel document....