**Design Patterns and Principles – Mandatory Exercises**

**Exercise 1: Implementing the Singleton Pattern**

**Code:**

//Logger.java

**package** SinglePattern;

**public** **class** Logger {

**private** **static** Logger *instance*;

**private** Logger(){

System.***out***.println("Logger initialized");

}

**public** **static** Logger getInstance(){

**if** (*instance* == **null**){

*instance* = **new** Logger();

}

**return** *instance*;

}

**public** **void** log(String message){

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

}

}

//LoggerTest.java

**package** SinglePattern;

**public** **class** LoggerTest{

**public** **static** **void** main(String[] args){

Logger logger1 = Logger.*getInstance*();

Logger logger2 = Logger.*getInstance*();

logger1.log("Logging from logger1");

logger2.log("Logging from logger2");

**if**(logger1 == logger2){

System.***out***.println("Same instance confirmed");

}**else** {

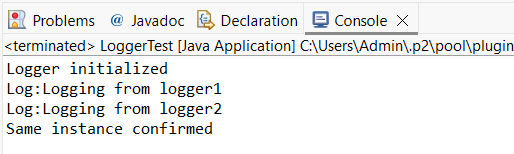
System.***out***.println("Different instances found");

}

}

}

**Output:**



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

**Code:**

//Document.java

**package** FactoryMethodPattern;

**public** **interface** Document{

**void** open();

}

// WordDocument.java

**package** FactoryMethodPattern;

**public** **class** WordDocument **implements** Document{

**public** **void** open(){

System.***out***.println("Opening Word Document.");

}

}

// PdfDocument.java

**package** FactoryMethodPattern;

**public** **class** PdfDocument **implements** Document{

**public** **void** open(){

System.***out***.println("Opening PDF Document.");

}

}

// ExcelDocument.java

**package** FactoryMethodPattern;

**public** **class** ExcelDocument **implements** Document{

**public** **void** open(){

System.***out***.println("Opening Excel Document.");

}

}

// DocumentFactory.java

**package** FactoryMethodPattern;

**public** **abstract** **class** DocumentFactory{

**public** **abstract** Document createDocument();

}

// WordFactory.java

**package** FactoryMethodPattern;

**public** **class** WordFactory **extends** DocumentFactory{

**public** Document createDocument(){

**return** **new** WordDocument();

}

}

// PdfFactory.java

**package** FactoryMethodPattern;

**public** **class** PdfFactory **extends** DocumentFactory{

**public** Document createDocument(){

**return** **new** PdfDocument();

}

}

// ExcelFactory.java

**package** FactoryMethodPattern;

**public** **class** ExcelFactory **extends** DocumentFactory{

**public** Document createDocument(){

**return** **new** ExcelDocument();

}

}

// DocumentTest.java

**package** FactoryMethodPattern;

**public** **class** DocumentTest{

**public** **static** **void** main(String[] args){

DocumentFactory wordFactory = **new** WordFactory();

Document wordDoc = wordFactory.createDocument();

wordDoc.open();

DocumentFactory pdfFactory = **new** PdfFactory();

Document pdfDoc = pdfFactory.createDocument();

pdfDoc.open();

DocumentFactory excelFactory = **new** ExcelFactory();

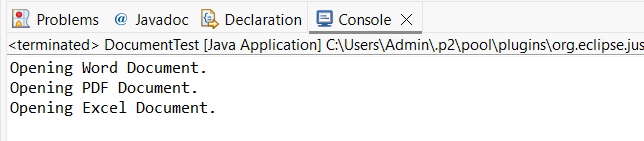
Document excelDoc = excelFactory.createDocument();

excelDoc.open();

}

}

**Output:**

****