**Exercise 1: Implementing the Single Pattern**

**CODE:**

**Logger.java:**

package SingletonPattern;

public class Logger {

private static Logger *instance*;

private Logger() {

System.*out*.println("Private Constructor");

}

public static Logger getInstance()

{

if(*instance* == null)

{

*instance* = new Logger();

}

return *instance*;

}

public static void showMessage()

{

System.*out*.println("This is a Singleton Pattern Class");

}

}

**LoggerTest.java:**

package SingletonPattern;

public class LoggerTest {

public static void main(String args[])

{

Logger loggerInstance1=Logger.*getInstance*();

Logger loggerInstance2=Logger.*getInstance*();

if(loggerInstance1 == loggerInstance2)

{

System.*out*.println("Logger class follows Singleton design pattern: It has only one instance");

}

else

{

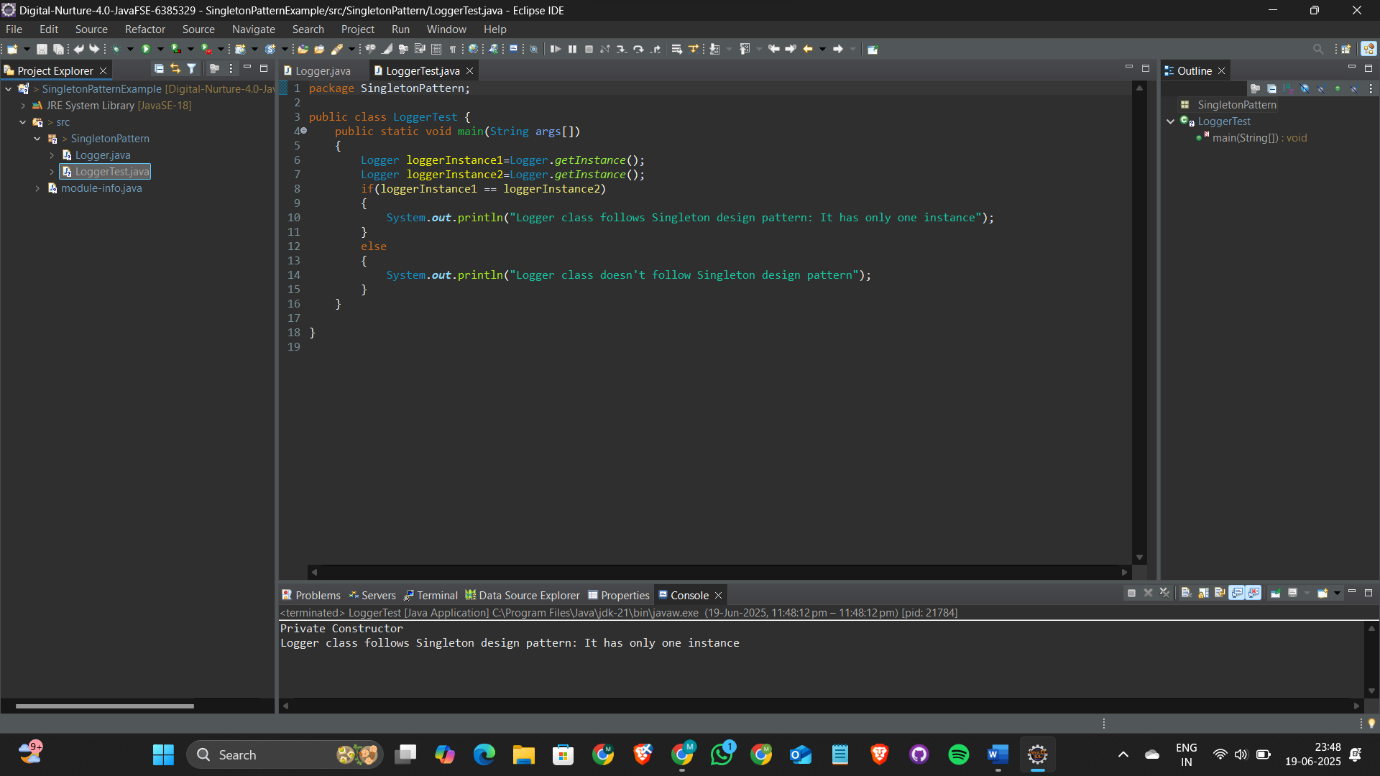
System.*out*.println("Logger class doesn't follow Singleton design pattern");

}

}

}

**OUTPUT:**

****

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

**CODE:**

**Document.java:**

package FactoryMethod;

public abstract class Document {

public abstract void print();

}

**WordDocument.java:**

package FactoryMethod;

public class WordDocument extends Document {

*@Override*

public void print()

{

System.*out*.println("This is a Word document");

}

}

**PdfDocument.java:**

package FactoryMethod;

public class PdfDocument extends Document {

*@Override*

public void print()

{

System.*out*.println("This is a Pdf document");

}

}

**ExcelDocument.java:**

package FactoryMethod;

public class ExcelDocument extends Document {

*@Override*

public void print()

{

System.*out*.println("This is a Excel document");

}

}

**DocumentFactory.java:**

package FactoryMethod;

public abstract class DocumentFactory {

public abstract Document createDocument();

}

**WordDocumentFactory.java:**

package FactoryMethod;

public class WordDocumentFactory extends DocumentFactory{

*@Override*

public Document createDocument()

{

return new WordDocument();

}

}

**PdfDocumentFactory.java:**

package FactoryMethod;

public class PdfDocumentFactory extends DocumentFactory{

*@Override*

public Document createDocument()

{

return new PdfDocument();

}

}

**ExcelDocumentFactory.java:**

package FactoryMethod;

public class ExcelDocumentFactory extends DocumentFactory{

*@Override*

public Document createDocument()

{

return new ExcelDocument();

}

}

**FactoryMethodTest.java:**

package FactoryMethod;

public class FactoryMethodTest {

public static void main(String args[])

{

DocumentFactory wordFactory=new WordDocumentFactory();

Document word=wordFactory.createDocument();

DocumentFactory pdfFactory=new PdfDocumentFactory();

Document pdf=pdfFactory.createDocument();

DocumentFactory excelFactory=new ExcelDocumentFactory();

Document excel=excelFactory.createDocument();

System.***out***.println("Factory Method Pattern Example:");

word.print();

pdf.print();

excel.print();

}

}

**OUTPUT:**

