**Exercise 1: Implementing the Singleton PatternCode:**

Class Logger.java:

public class Logger {

private static Logger instance;

private Logger() {

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

}

public static Logger getInstance() {

if (instance == null) {

instance = new Logger();

}

return instance;

}

public void log(String message) {

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

}

}

MainApp.java:

public class Main {

public static void main(String[] args) {

Logger logger1 = Logger.getInstance();

Logger logger2 = Logger.getInstance();

logger1.log("First log message.");

logger2.log("Second log message.");

if (logger1 == logger2) {

System.out.println("Both logger instances are the same. Singleton works!");

} else {

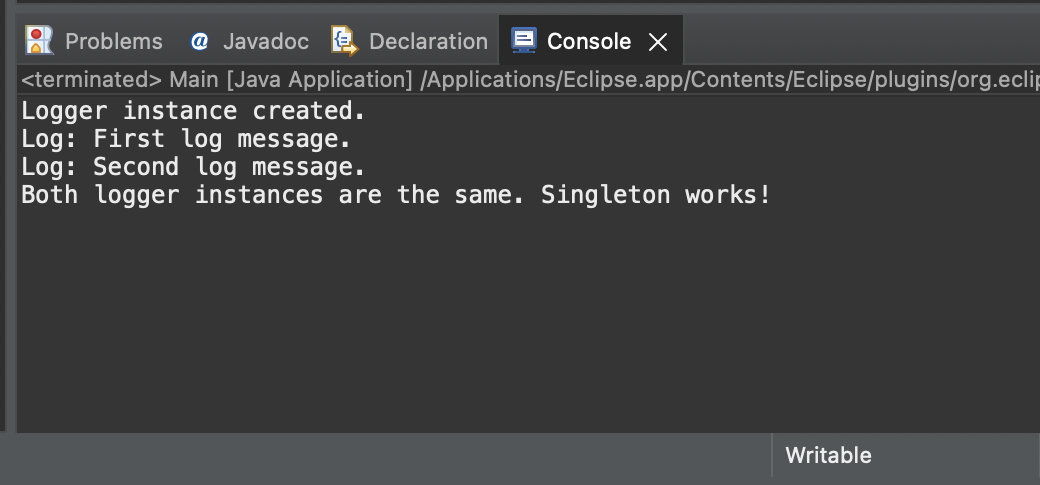
System.out.println("Different logger instances. Singleton failed.");

}

}

}

**Output**:



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

FactoryMethodPatternExample/

│

├── Document.java

├── WordDocument.java

├── PdfDocument.java

├── ExcelDocument.java

├── DocumentFactory.java

├── WordDocumentFactory.java

├── PdfDocumentFactory.java

├── ExcelDocumentFactory.java

└── MainApp.java

Document.java:

public interface Document {

void open();

}

WordDocument.java:

public class WordDocument implements Document {

@Override

public void open() {

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

}

}

PdfDocument.java:

public class PdfDocument implements Document {

@Override

public void open() {

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

}

}

ExcelDocument.java:

public class ExcelDocument implements Document {

@Override

public void open() {

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

}

}

DocumentFactory.java:

public abstract class DocumentFactory {

public abstract Document createDocument();

}

WordDocumentFactory.java:

public class WordDocumentFactory extends DocumentFactory {

@Override

public Document createDocument() {

return new WordDocument();

}

}

PdfDocumentFactory.java:

public class PdfDocumentFactory extends DocumentFactory {

@Override

public Document createDocument() {

return new PdfDocument();}

}

ExcelDocumentFactory.java:

public class ExcelDocumentFactory extends DocumentFactory {

@Override

public Document createDocument() {

return new ExcelDocument();

}

}

Main.java:

public class Main {

public static void main(String[] args) {

DocumentFactory wordFactory = new WordDocumentFactory();

Document word = wordFactory.createDocument();

word.open();

DocumentFactory pdfFactory = new PdfDocumentFactory();

Document pdf = pdfFactory.createDocument();

pdf.open();

DocumentFactory excelFactory = new ExcelDocumentFactory();

Document excel = excelFactory.createDocument();

excel.open();

}

}

**Output:**

