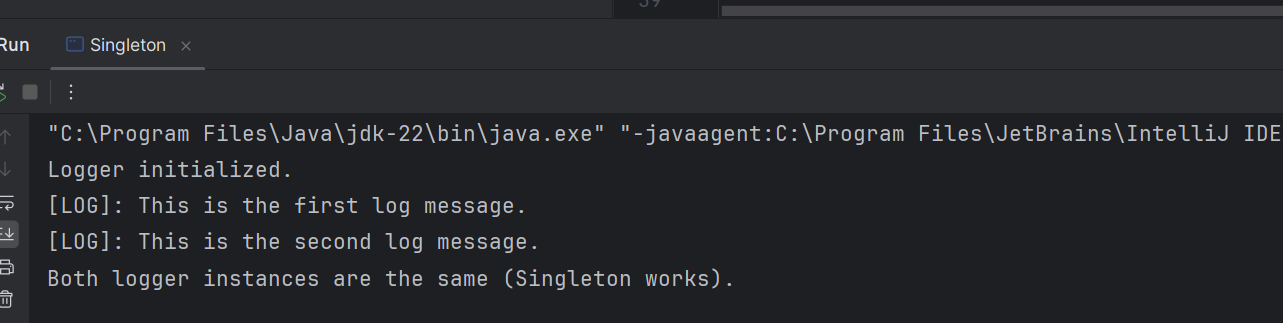
**Ques 1 : Implementing the Singleton Pattern**

**Code :**

public class Singleton {  
  
 // Inner static Logger class implementing Singleton  
 static 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);  
 }  
 }  
  
 // Main method to test singleton behavior  
 public static void main(String[] args) {  
 Logger logger1 = Logger.*getInstance*();  
 Logger logger2 = Logger.*getInstance*();  
  
 logger1.log("This is the first log message.");  
 logger2.log("This is the second log message.");  
  
 if (logger1 == logger2) {  
 System.*out*.println("Both logger instances are the same (Singleton works).");  
 } else {  
 System.*out*.println("Logger instances are different (Singleton failed).");  
 }  
 }  
}

**Output :**



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

**Code :**

// Factory Method Pattern Example for Document Management  
  
public class DocumentFactoryExample {  
  
 // Step 1: Product Interface  
 interface Document {  
 void open();  
 }  
  
 // Step 2: Concrete Products  
 static class WordDocument implements Document {  
 public void open() {  
 System.*out*.println("Opening Word document...");  
 }  
 }  
  
 static class PdfDocument implements Document {  
 public void open() {  
 System.*out*.println("Opening PDF document...");  
 }  
 }  
  
 static class ExcelDocument implements Document {  
 public void open() {  
 System.*out*.println("Opening Excel document...");  
 }  
 }  
  
 // Step 3: Creator (Factory)  
 abstract static class DocumentFactory {  
 public abstract Document createDocument();  
 }  
  
 // Step 4: Concrete Factories  
 static class WordDocumentFactory extends DocumentFactory {  
 public Document createDocument() {  
 return new WordDocument();  
 }  
 }  
  
 static class PdfDocumentFactory extends DocumentFactory {  
 public Document createDocument() {  
 return new PdfDocument();  
 }  
 }  
  
 static class ExcelDocumentFactory extends DocumentFactory {  
 public Document createDocument() {  
 return new ExcelDocument();  
 }  
 }  
  
 // Step 5: Client Code  
 public static void main(String[] args) {  
 DocumentFactory wordFactory = new WordDocumentFactory();  
 Document wordDoc = wordFactory.createDocument();  
 wordDoc.open();  
  
 DocumentFactory pdfFactory = new PdfDocumentFactory();  
 Document pdfDoc = pdfFactory.createDocument();  
 pdfDoc.open();  
  
 DocumentFactory excelFactory = new ExcelDocumentFactory();  
 Document excelDoc = excelFactory.createDocument();  
 excelDoc.open();  
 }  
}

**Output :**

