**Exercise 1: Implementing the Singleton Pattern**

Logger.java

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:" + message);

    }

}

TestLogger.java

public class TestLogger {

    public static void main(String[] args) {

        Logger log1 = Logger.getInstance();

        log1.log(" This is the first log message.");

        Logger log2 = Logger.getInstance();

        log2.log(" This is the second log message.");

        if (log1 == log2) {

            System.out.println("Both loggers are the same instance.");

        } else {

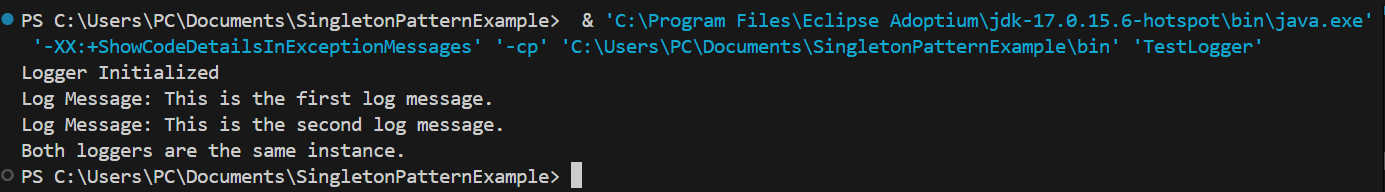
            System.out.println("Different logger instances found.");

        }

    }

}

Output:



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

Document.java

public interface Document {

    void open();

}

WordDocument.java

public class WordDocument implements Document {

    public void open() {

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

    }

}

PdfDocument.java

public class PdfDocument implements Document {

    public void open() {

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

    }

}

ExcelDocument.java

public class ExcelDocument implements Document {

    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 {

    public Document createDocument() {

        return new WordDocument();

    }

}

PdfDocumentFactory.java

public class PdfDocumentFactory extends DocumentFactory {

    public Document createDocument() {

        return new PdfDocument();

    }

}

ExcelDocumentFactory.java

public class ExcelDocumentFactory extends DocumentFactory {

    public Document createDocument() {

        return new ExcelDocument();

    }

}

App.java

public class App {

    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:

