**Exercise 1: Implementing the Singleton Pattern**

**Code:**

// Logger.java

public class Logger {

private static Logger instance = new Logger();

private Logger() {

System.out.println("Logger Initialized.");

}

public static Logger getInstance() {

return instance;

}

public void log(String message) {

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

}

}

// Main.java

public class Main {

public static void main(String[] args) {

Logger logger1 = Logger.getInstance();

Logger logger2 = Logger.getInstance();

logger1.log("This is the first message.");

logger2.log("This is the second message.");

if (logger1 == logger2) {

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

} else {

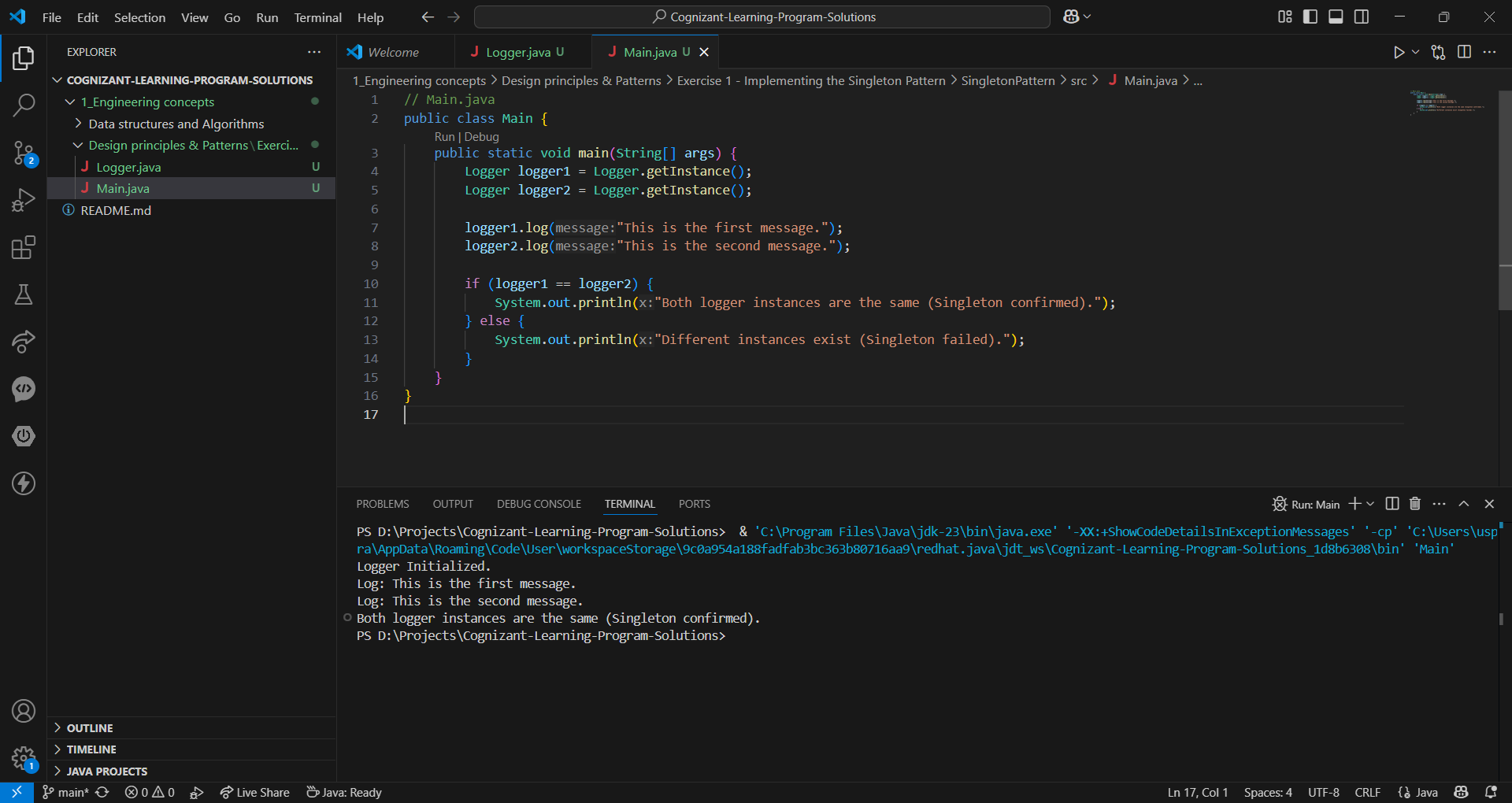
System.out.println("Different instances exist (Singleton failed).");

}

}

}

**Output:**



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

**Code:**

// App.java

public class App {

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();

}

}

// Document.java

public interface Document {

void open();

}

// DocumentFactory.java

public abstract class DocumentFactory {

public abstract Document createDocument();

}

// ExcelDocument.java

public class ExcelDocument implements Document {

public void open() {

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

}

}

// ExcelFactory.java

public class ExcelFactory extends DocumentFactory {

public Document createDocument() {

return new ExcelDocument();

}

}

// PdfDocument.java

public class PdfDocument implements Document {

public void open() {

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

}

}

// PdfFactory.java

public class PdfFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

// WordDocument.java

public class WordDocument implements Document {

public void open() {

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

}

}

// WordFactory.java

public class WordFactory extends DocumentFactory {

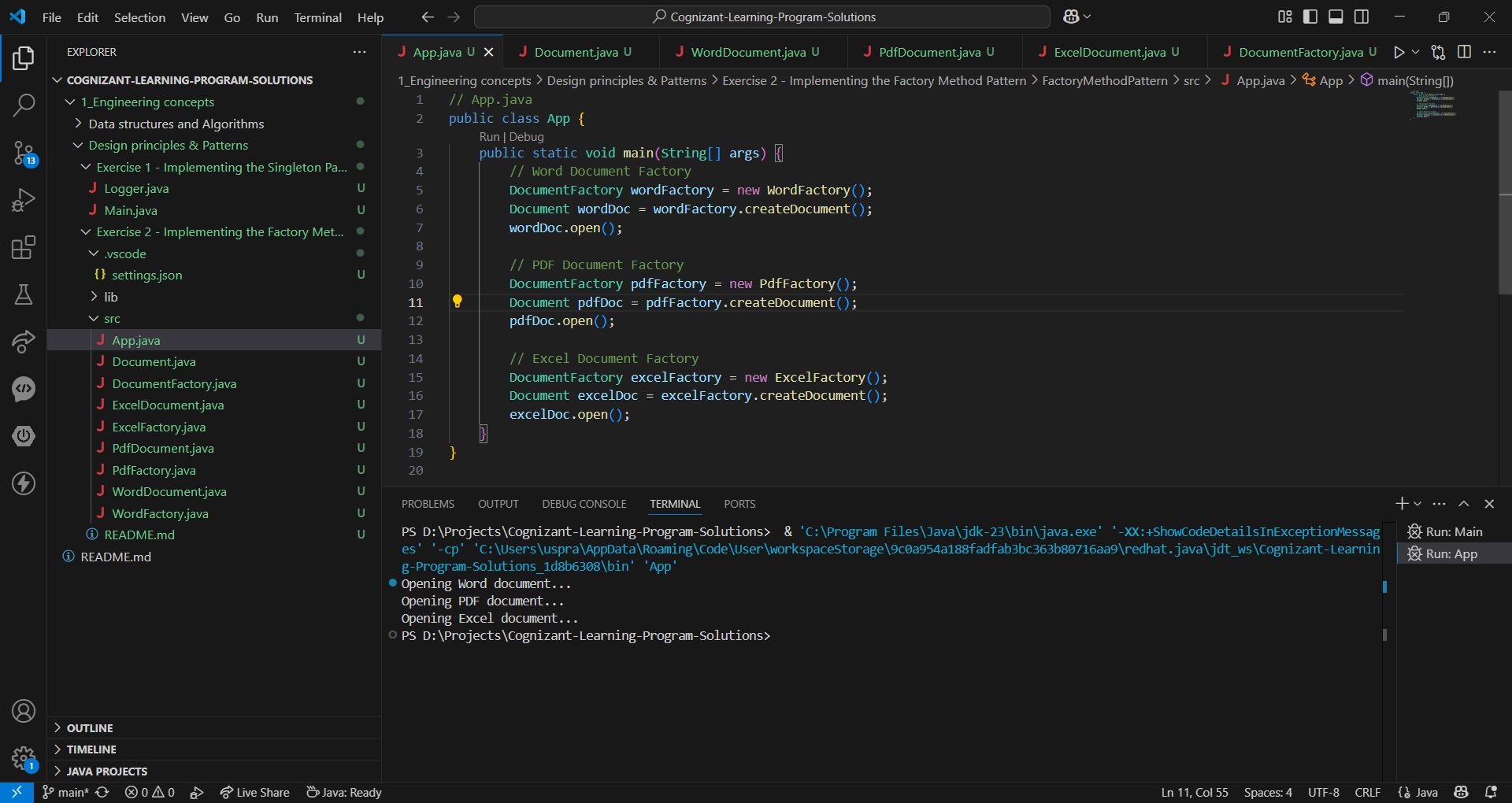
public Document createDocument() {

return new WordDocument();

}

}

**Output:**

****