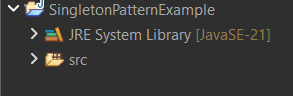
**Exercise 1: Implementing the Singleton Pattern**

**Scenario:**

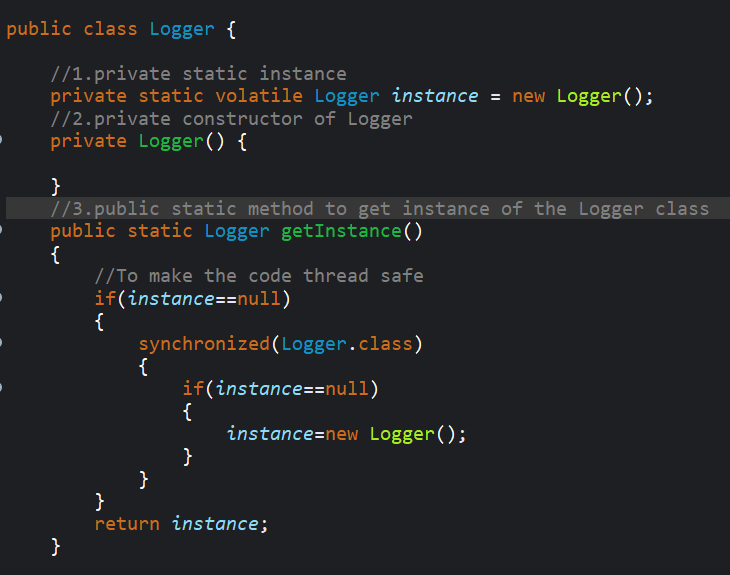
You need to ensure that a logging utility class in your application has only one instance throughout the application lifecycle to ensure consistent logging.

**Steps:**

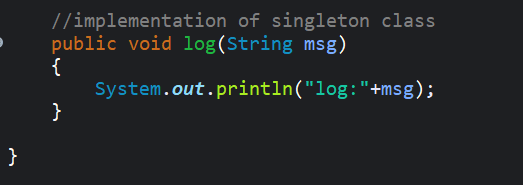
1. **Create a New Java Project:**
   1. Create a new Java project named **SingletonPatternExample**.



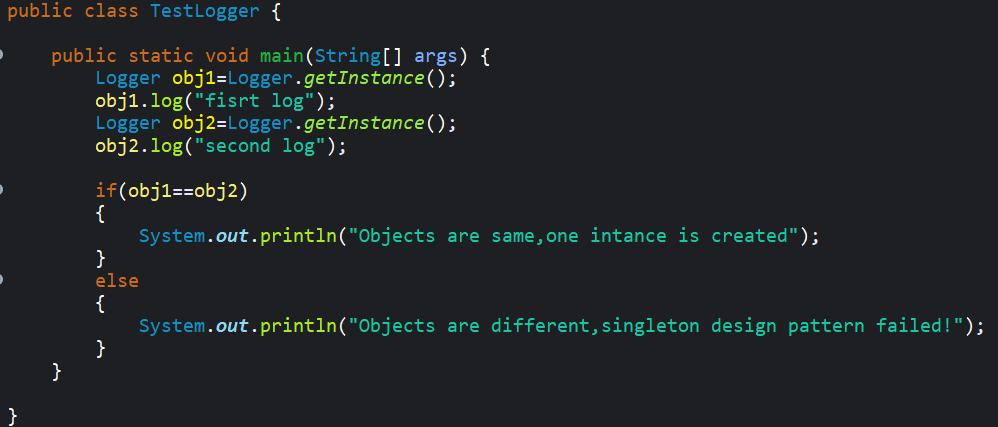
1. **Define a Singleton Class:**
   1. Create a class named Logger that has a private static instance of itself.
   2. Ensure the constructor of Logger is private.
   3. Provide a public static method to get the instance of the Logger class.



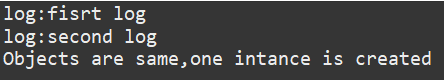
1. **Implement the Singleton Pattern:**
   1. Write code to ensure that the Logger class follows the Singleton design pattern.



1. **Test the Singleton Implementation:**
   1. Create a test class to verify that only one instance of Logger is created and used across the application.



**Code Output**

****

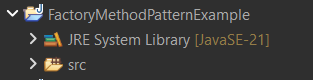
**Exercise 2: Implementing the Factory Method Pattern**

**Scenario:**

You are developing a document management system that needs to create different types of documents (e.g., Word, PDF, Excel). Use the Factory Method Pattern to achieve this.

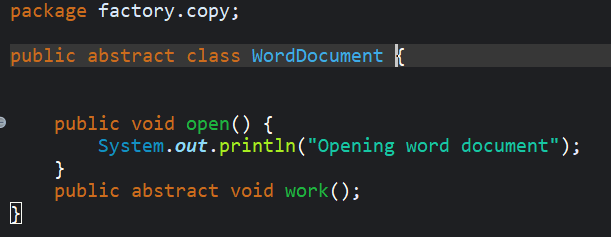
**Steps:**

1. **Create a New Java Project:**
   1. Create a new Java project named **FactoryMethodPatternExample**.

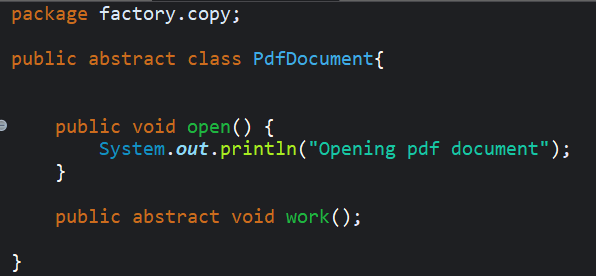


1. **Define Document Classes:**
   1. Create interfaces or abstract classes for different document types such as **WordDocument**, **PdfDocument**, and **ExcelDocument**.

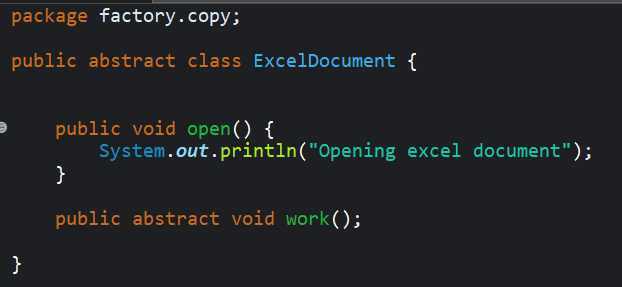
**WordDocument**



**PdfDocument**

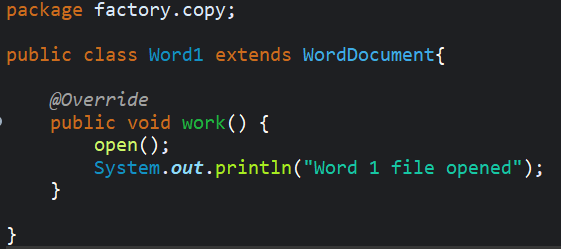


**ExcelDocument**

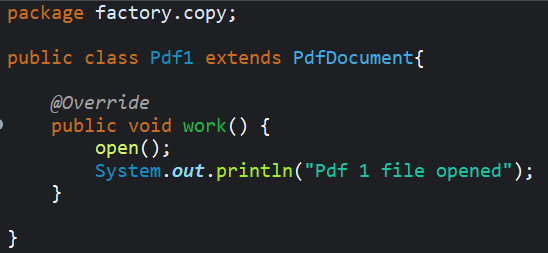


1. **Create Concrete Document Classes:**
   1. Implement concrete classes for each document type that implements or extends the above interfaces or abstract classes.

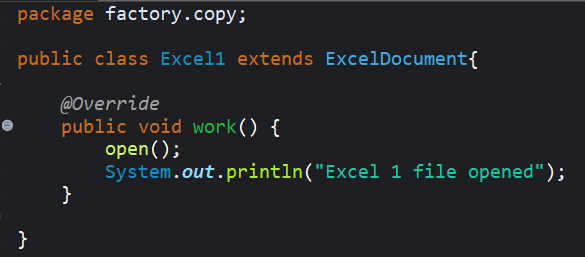
**Word1**



**Pdf1**

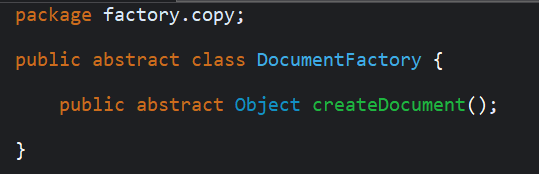


**Excel1**



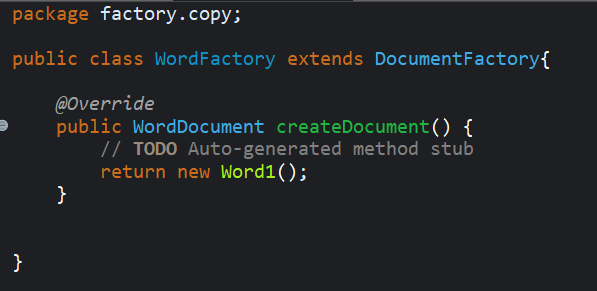
1. **Implement the Factory Method:**
   1. Create an abstract class **DocumentFactory** with a method **createDocument()**.

**DocumentFactory**

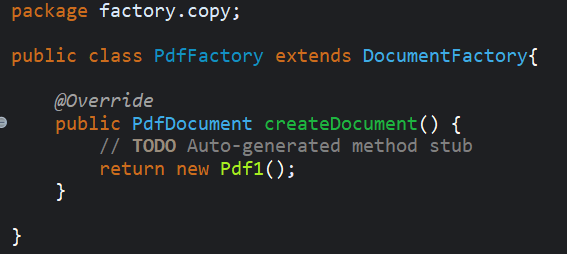


* 1. Create concrete factory classes for each document type that extends DocumentFactory and implements the **createDocument()** method.

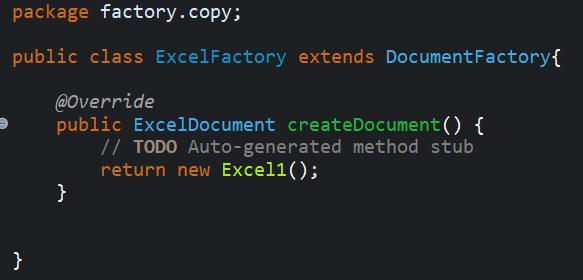
**WordFactory**



**PdfFactory**

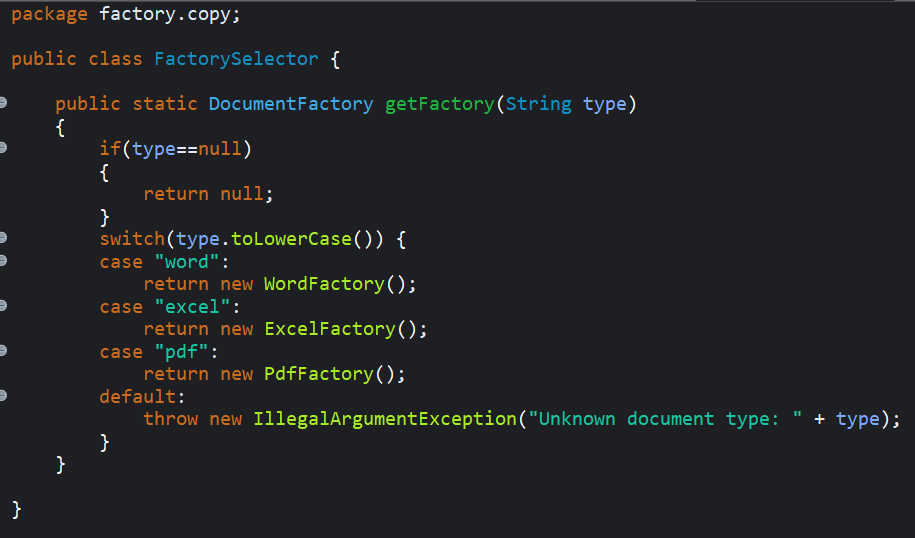


**ExcelFactory**

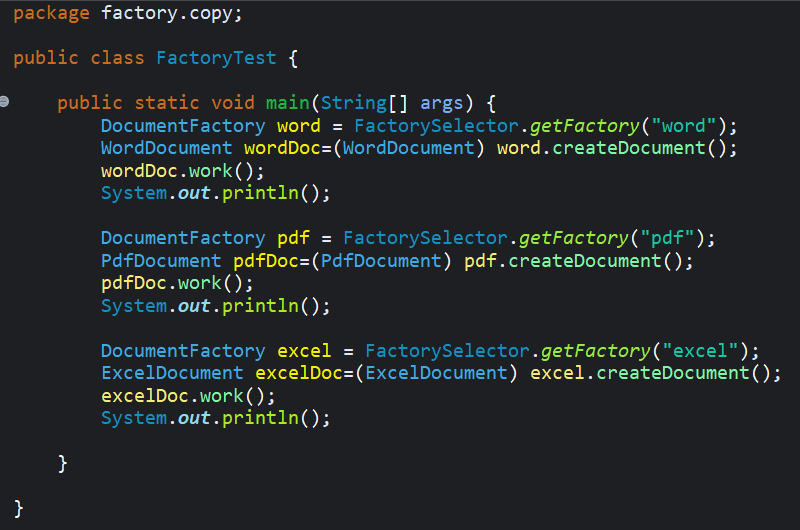


1. **Test the Factory Method Implementation:**
   1. Create a test class to demonstrate the creation of different document types using the factory method.

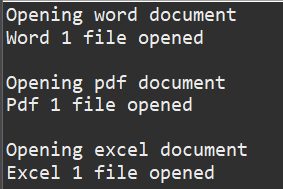
**FactorySelector**



**FactoryTest**



**Code Output**

****