**Exercise 1: Implementing the Singleton Pattern**

**Code:**

*Logger.java*

**package** com.example.singleton;

**public** **class** Logger {

**private** **static** Logger *loggerInstance*;

**private** Logger() {

System.***out***.println("Logger is Intantiated");

}

**public** **static** Logger getLogger() {

**if**(*loggerInstance*==**null**) {

*loggerInstance* = **new** Logger();

}

**return** *loggerInstance*;

}

**public** **void** log(String str) {

System.***out***.println("Log : "+str);

}

}

*SingletonTest.java*

**package** com.example.singleton;

**public** **class** SingletonTest {

**public** **static** **void** main(String args[]) {

Logger l1 = Logger.*getLogger*();

l1.log("Log is used in the main");

Logger l2 = Logger.*getLogger*();

l2.log("Log is used again");

**if**(l1.equals(l2)) {

System.***out***.println("Instantiated only once as it is a singleton.");

}

**else** {

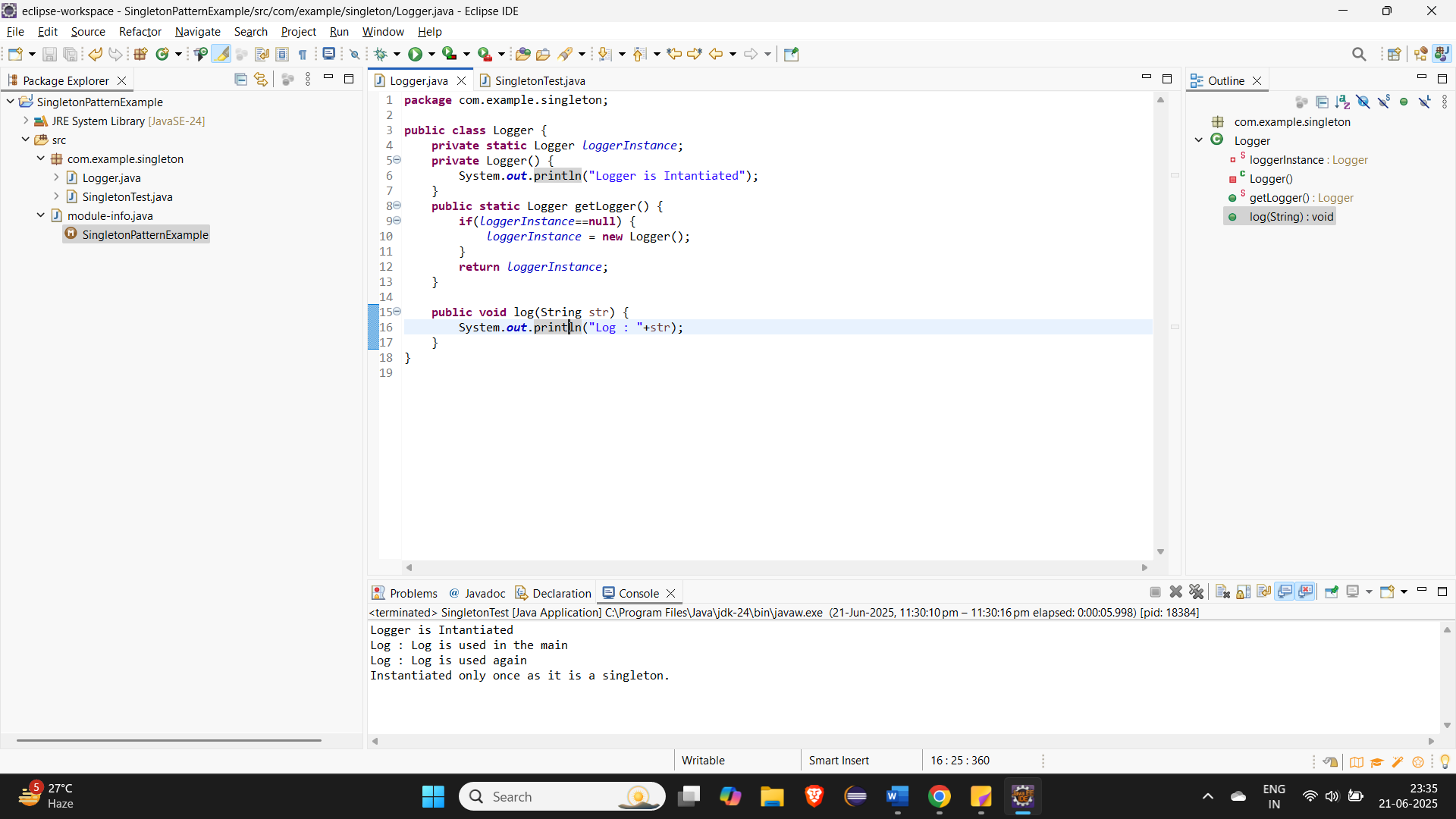
System.***out***.println("Different instances");

}

}

}

**Output:**



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

**Code:**

**Document Interface(***Document.java)*:

**package** com.example.factory;

**public** **interface** Document {

String create();

}

**Concrete Document Classes:**

1. *WordDocument.java*

**package** com.example.factory;

**public** **class** WordDocument **implements** Document{

**public** WordDocument() {

System.***out***.println(**this**.create());

}

@Override

**public** String create() {

**return** "Word Document Created";

}

}

1. *PdfDocument.java*

**package** com.example.factory;

**public** **class** WordDocument **implements** Document{

**public** WordDocument() {

System.***out***.println(**this**.create());

}

@Override

**public** String create() {

**return** "Word Document Created";

}

}

1. *ExcelDocument.java*

**package** com.example.factory;

**public** **class** ExcelDocument **implements** Document{

**public** ExcelDocument() {

System.***out***.println(**this**.create());

}

@Override

**public** String create() {

**return** "Excel Document Created";

}

}

**Abstract Factory (***DocumentFactory.java***):**

**package** com.example.factory;

**public** **abstract** **class** DocumentFactory {

**public** **abstract** Document createDocument();

}

**Concrete Factories**

1. *WordFactory.java:*

**package** com.example.factory;

**public** **class** WordFactory **extends** DocumentFactory {

@Override

**public** Document createDocument() {

**return** **new** WordDocument();

}

}

1. *PdfFactory.java:*

**package** com.example.factory;

**public** **class** PdfFactory **extends** DocumentFactory{

@Override

**public** Document createDocument(){

**return** **new** PdfDocument();

}

}

1. *ExcelFactory.java:*

**package** com.example.factory;

**public** **class** ExcelFactory **extends** DocumentFactory{

@Override

**public** Document createDocument(){

**return** **new** ExcelDocument();

}

}

*FactoryTest.java:*

