**SUPERSET ID: 6391783**

**Exercise 1: Implementing the Singleton Pattern**

**CODE:**

**Singleton.java**

public class Singleton {

private static volatile Singleton instance;

private Singleton() {

System.out.println("Singleton Instance Created");

}

public static Singleton getInstance() {

if (instance == null) {

synchronized (Singleton.class) {

if (instance == null) {

instance = new Singleton();

}

}

}

return instance;

}

public void showMessage() {

System.out.println("Hello from Singleton!");

}

}

**Main.java**

public class Main {

public static void main(String[] args) {

Singleton s1 = Singleton.getInstance();

Singleton s2 = Singleton.getInstance();

s1.showMessage();

if (s1 == s2) {

System.out.println("Both are same instances.");

} else {

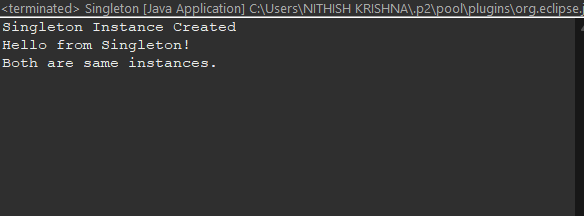
System.out.println("Different instances (error in Singleton).");

}

}

}

**OUTPUT SCREENSHOT:**

****