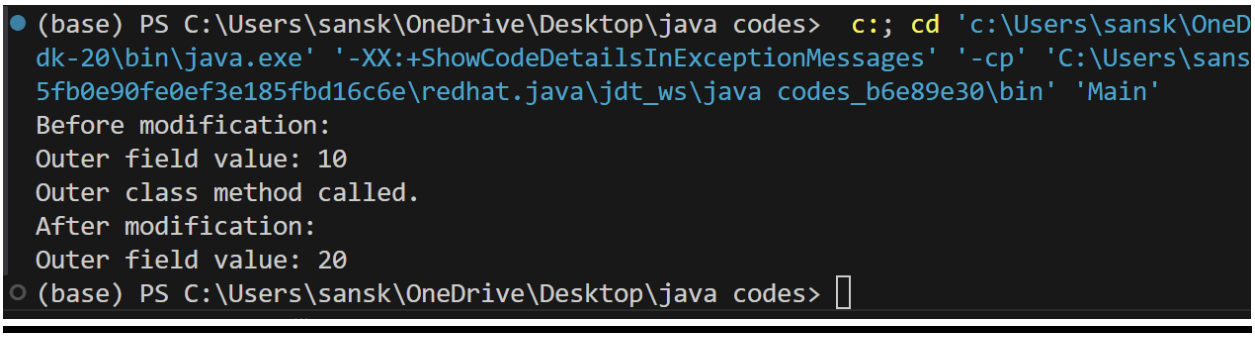
Lab 6

Q1. Create a class with a private field and a private method. Create an inner class with a method that modifies the outer-class field and calls the outer-class method. In a second outer-class method, create an object of the inner class and call its method, then show the effect on the outer-class object.

```
class OuterClass {
    private int outerField = 10;
    private void outerMethod() {
        System.out.println("Outer class method called.");
    }
    class InnerClass {
        void modifyOuterClass() {
            outerField = 20;
            outerMethod();
        }
    }
    void createInnerAndModify() {
        InnerClass inner = new InnerClass();
        inner.modifyOuterClass();
    }
    void display() {
        System.out.println("Outer field value: " + outerField);
    }
}

public class Main {
    public static void main(String[] args) {
        OuterClass outer = new OuterClass();
        System.out.println("Before modification:");
        outer.display();
        outer.createInnerAndModify();
        System.out.println("After modification:");
        outer.display();
    }
}
```

Output :


(base) PS C:\Users\sansk\OneDrive\Desktop\java codes> c:; cd 'c:\Users\sansk\OneD
dk-20\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\sans
5fb0e90fe0ef3e185fbd16c6e\redhat.java\jdt_ws\java codes_b6e89e30\bin' 'Main'
Before modification:
Outer field value: 10
Outer class method called.
After modification:
Outer field value: 20
(base) PS C:\Users\sansk\OneDrive\Desktop\java codes>

Q2. Determine whether an outer class has access to the private elements of its inner class.

```java
class OuterClass2 {
    public class InnerClass2 {
        private int innerField = 30;
        private void innerMethod() {
            System.out.println("Inner class method called.");
        }
    }
    void accessInnerClass() {
        InnerClass2 inner = new InnerClass2();
        System.out.println("Inner field value: " + inner.innerField);
        inner.innerMethod();
        System.out.println("Cannot access inner class's private field or method
directly.");
    }
}

public class Main2 {
    public static void main(String[] args) {
        OuterClass2 outer2 = new OuterClass2();
        outer2.accessInnerClass();
    }
}
```

Output :

```
(base) PS C:\Users\sansk\OneDrive\Desktop\java codes>  c:; cd 'c:\
dk-20\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp
5fb0e90fe0ef3e185fbd16c6e\redhat.java\jdt_ws\java codes_b6e89e30\b
Inner field value: 30
Inner class method called.
Cannot access inner class's private field or method directly.
(base) PS C:\Users\sansk\OneDrive\Desktop\java codes>
```