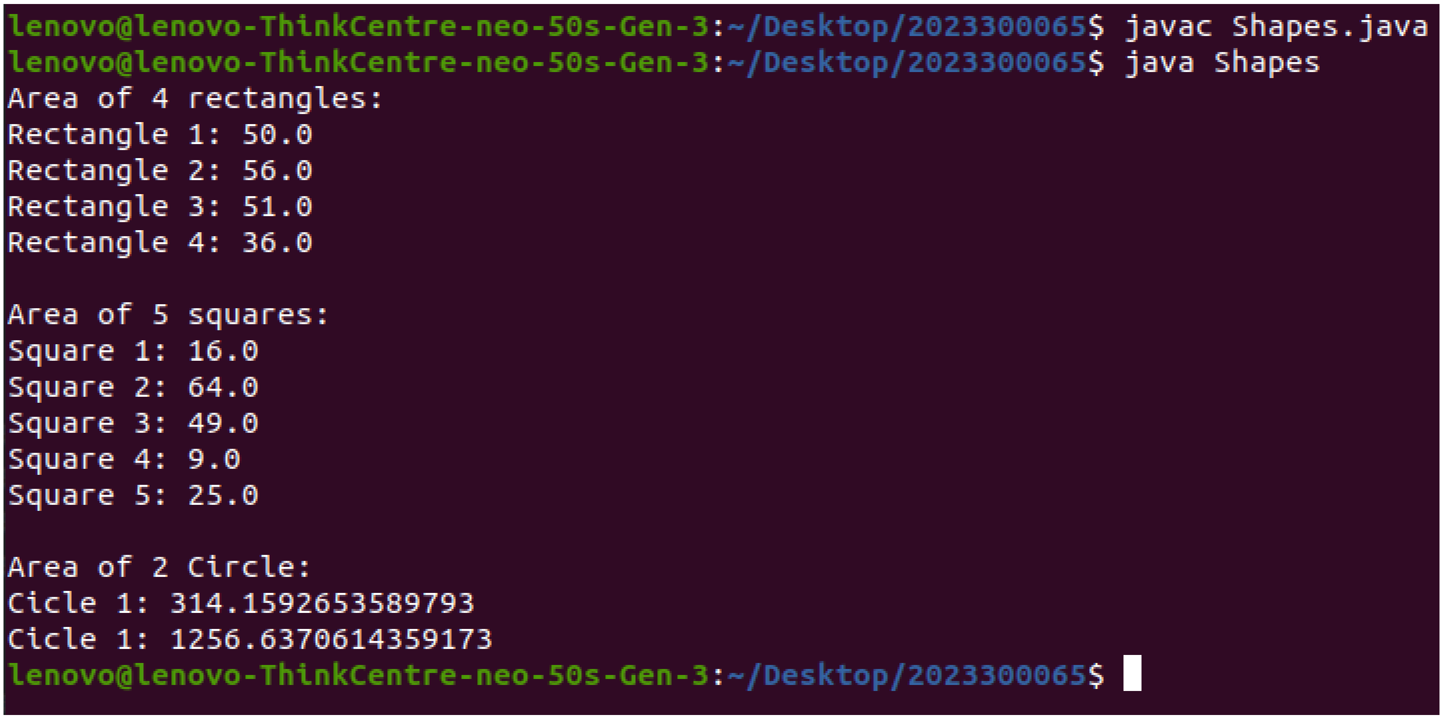
Name:	Debjit Ghosal
UID:	2023300065
Experiment No.	7b

AIM:	Program on Abstraction:Implement a Program to demonstrate Abstraction using abstract class
Program 1	
PROBLEM STATEMENT :	**Problem Statement#1: Shape Program** Write a program to calculate the area of 4 rectangles, 5 squares and 2 circles. Create an abstract class 'Shape' with three abstract methods namely 'RectangleArea' taking two parameters, 'SquareArea' and 'CircleArea' taking one parameter each. The parameters of 'RectangleArea' are its length and breadth, that of 'SquareArea' is its side and that of 'CircleArea' is its radius. Now create another class 'Area' containing all the three methods 'RectangleArea', 'SquareArea' and 'CircleArea' for printing the area of rectangle, square and circle respectively. Create an object of class 'Area' and call all the three methods.
PROGRAM:	/* Problem Statement: Shape Program Write a program to calculate the area of 4 rectangles, 5 squares and 2 circles. Create an abstract class 'Shape' with three abstract methods namely 'RectangleArea' taking two parameters, 'SquareArea' and 'CircleArea' taking one parameter each. The parameters of 'RectangleArea' are its length and breadth, that of 'SquareArea' is its side and that of 'CircleArea' is its radius. Now create another class 'Area' containing all the three methods 'RectangleArea', 'SquareArea' and 'CircleArea' for printing the area of rectangle, square and circle respectively. Create an object of class 'Area' and call all the three methods.*/

```
import java.util.Scanner;

abstract class Shape {

    abstract double RectangleArea(double length, double breadth);
    abstract double SquareArea(double side);
    abstract double CircleArea(double radius);
}

class Area extends Shape {

    @Override
    double RectangleArea(double length, double breadth) {
        return length * breadth;
    }

    @Override
    double SquareArea(double side) {
        return side * side;
    }

    @Override
    double CircleArea(double radius) {
        return Math.PI * radius * radius;
    }
}


public class Shapes{
    public static void main(String[] args){
     Area area = new Area();


        System.out.println("Area of 4 rectangles:");
        System.out.println("Rectangle 1: " + area.RectangleArea(5, 10));
        System.out.println("Rectangle 2: " + area.RectangleArea(7, 8));
        System.out.println("Rectangle 3: " + area.RectangleArea(17, 3));
        System.out.println("Rectangle 4: " + area.RectangleArea(4, 9));
        System.out.println("\nArea of 5 squares:");
        System.out.println("Square 1: " + area.SquareArea(4));
        System.out.println("Square 2: " + area.SquareArea(8));
        System.out.println("Square 3: " + area.SquareArea(7));
        System.out.println("Square 4: " + area.SquareArea(3));
        System.out.println("Square 5: " + area.SquareArea(5));

        System.out.println("\nArea of 2 Circle:");
        System.out.println("Cicle 1: " + area.CircleArea(10));
        System.out.println("Cicle 1: " + area.CircleArea(20));

        }
    }
```

RESULT:



```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Shapes.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Shapes
Area of 4 rectangles:
Rectangle 1: 50.0
Rectangle 2: 56.0
Rectangle 3: 51.0
Rectangle 4: 36.0

Area of 5 squares:
Square 1: 16.0
Square 2: 64.0
Square 3: 49.0
Square 4: 9.0
Square 5: 25.0

Area of 2 Circle:
Cicle 1: 314.1592653589793
Cicle 1: 1256.6370614359173
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$
```

Program 2

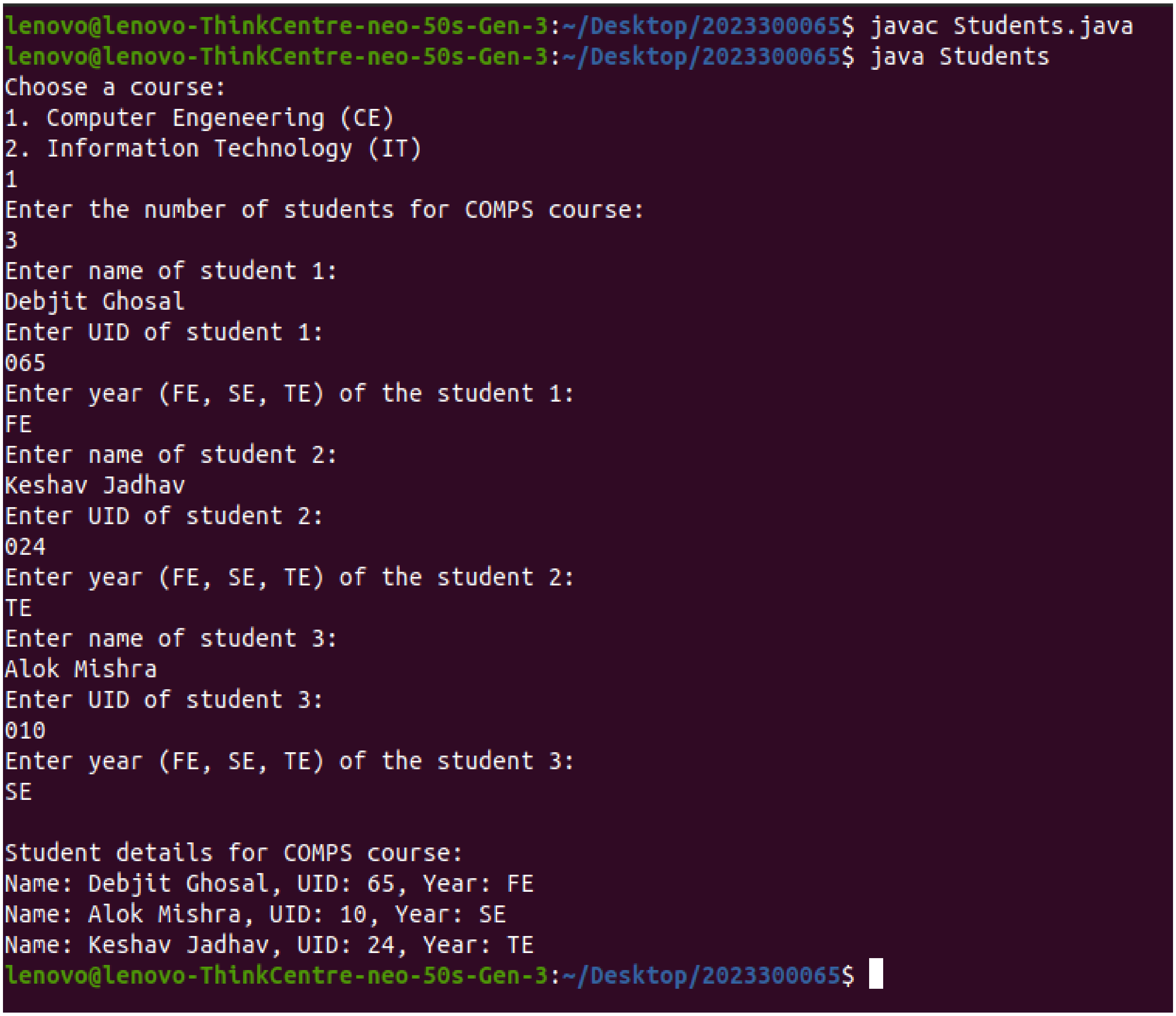
PROBLEM STATEMENT :	**Problem Statement#2: Student Details Program** Write an abstract class Course with an abstract method studentDetails(). Create 2 classes Comps and IT which inherits class Course and implements studentDetails(). The studentDetails() method should print the name, UID, and year (FE, SE, TE) of students

of that Course. These details have to be taken from the user. Write a program that asks user to choose a course and print the details of all students in that course in a sorted manner as per the year(FE,SE,TE)

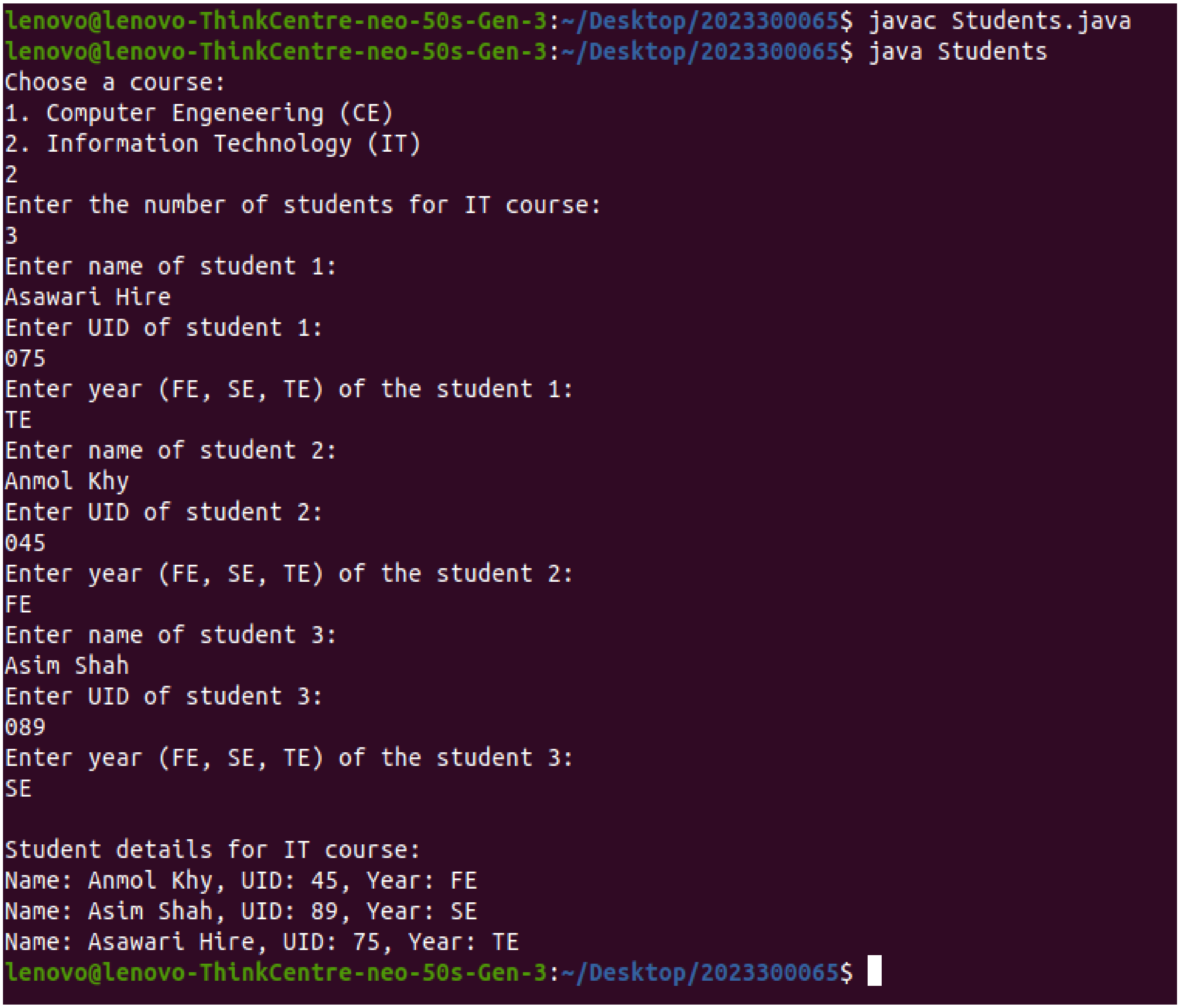
PROGRAM:

```
/*
Problem Statement#: Student Details Program

Write an abstract class Course with an abstract method studentDetails().

Create 2 classes Comps and IT which inherits class Course and implements
studentDetails().

The studentDetails() method should print the name, UID, and year (FE, SE, TE) of
students of that Course.
These details have to be taken from the user.
Write a program that asks user to choose a course  and print the details of all students in
that course in a sorted manner
as per the year( FE,SE,TE)\
*/


import java.util.*;

abstract class Course {
    abstract void studentDetails();
}

class Comps extends Course {
    private List<Student> students = new ArrayList<>();

    @Override
    void studentDetails() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the number of students for COMPS course:");
        int numStudents = scanner.nextInt();
        scanner.nextLine();

        for (int i = 0; i < numStudents; i++) {
            System.out.println("Enter name of student " + (i + 1) + ":");
            String name = scanner.nextLine();
            System.out.println("Enter UID of student " + (i + 1) + ":");
            int uid = scanner.nextInt();
            scanner.nextLine();
            System.out.println("Enter year (FE, SE, TE) of the student " + (i + 1) + ":");
            String year = scanner.nextLine();
            students.add(new Student(name, uid, year));
        }
        students.sort(Comparator.comparing(Student::getYear));
        System.out.println("\nStudent details for COMPS course:");
        for (Student student : students) {
            System.out.println(student);
        }
    }
}

class IT extends Course {
    private List<Student> students = new ArrayList<>();

    @Override
    void studentDetails() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the number of students for IT course:");
        int numStudents = scanner.nextInt();
        scanner.nextLine();

        for (int i = 0; i < numStudents; i++) {
            System.out.println("Enter name of student " + (i + 1) + ":");
            String name = scanner.nextLine();
            System.out.println("Enter UID of student " + (i + 1) + ":");
            int uid = scanner.nextInt();
            scanner.nextLine();
            System.out.println("Enter year (FE, SE, TE) of the student " + (i + 1) + ":");
            String year = scanner.nextLine();
            students.add(new Student(name, uid, year));
        }
        students.sort(Comparator.comparing(Student::getYear));
        System.out.println("\nStudent details for IT course:");
        for (Student student : students) {
            System.out.println(student);
        }
    }
}

class Student {
    private String name;
    private int uid;
    private String year;

    public Student(String name, int uid, String year) {
        this.name = name;
        this.uid = uid;
        this.year = year;
    }

    public String getYear() {
        return year;
    }

    @Override
    public String toString() {
        return "Name: " + name + ", UID: " + uid + ", Year: " + year;
    }
}

public class Students {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Choose a course:\n1. Computer Engeneering (CE)\n2.
Information Technology (IT)");
        int choice = scanner.nextInt();
        Course course;
        switch (choice) {
            case 1:
                course = new Comps();
                break;
            case 2:
                course = new IT();
                break;
            default:
                System.out.println("Invalid choice");
                return;
            }
        course.studentDetails();
    }
}
```

RESULT:



```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Students.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Students
Choose a course:
1. Computer Engeneering (CE)
2. Information Technology (IT)
1
Enter the number of students for COMPS course:
3
Enter name of student 1:
Debjit Ghosal
Enter UID of student 1:
065
Enter year (FE, SE, TE) of the student 1:
FE
Enter name of student 2:
Keshav Jadhav
Enter UID of student 2:
024
Enter year (FE, SE, TE) of the student 2:
TE
Enter name of student 3:
Alok Mishra
Enter UID of student 3:
010
Enter year (FE, SE, TE) of the student 3:
SE

Student details for COMPS course:
Name: Debjit Ghosal, UID: 65, Year: FE
Name: Alok Mishra, UID: 10, Year: SE
Name: Keshav Jadhav, UID: 24, Year: TE
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ 
```



```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Students.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Students
Choose a course:
1. Computer Engeneering (CE)
2. Information Technology (IT)
2
Enter the number of students for IT course:
3
Enter name of student 1:
Asawari Hire
Enter UID of student 1:
075
Enter year (FE, SE, TE) of the student 1:
TE
Enter name of student 2:
Anmol Khy
Enter UID of student 2:
045
Enter year (FE, SE, TE) of the student 2:
FE
Enter name of student 3:
Asim Shah
Enter UID of student 3:
089
Enter year (FE, SE, TE) of the student 3:
SE

Student details for IT course:
Name: Anmol Khy, UID: 45, Year: FE
Name: Asim Shah, UID: 89, Year: SE
Name: Asawari Hire, UID: 75, Year: TE
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ 
```

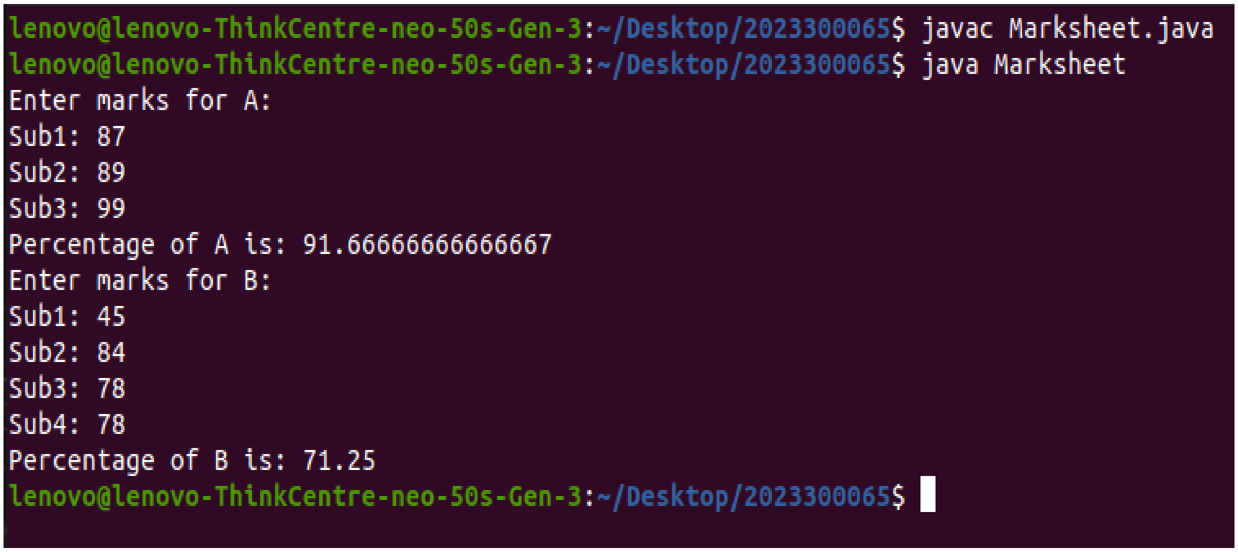
Program 3

PROBLEM STATEMENT:	**Problem Statement#3: Percentage-Marks Program** Write a program to calculate the percentage of marks obtained in three subjects (each out of 100) by student A and in four subjects (each out of 100) by student B. Create an abstract class 'Marks' with an abstract method 'getPercentage'. It is inherited by two other classes 'A' and 'B' each having a method with the same name which returns the percentage of the students. The constructor of student A takes the marks in three subjects as its parameters and the marks in four subjects as its parameters for student B. Input to be taken from the user. Display the marks in highest order of student A and B.

PROGRAM:

```
/*
Write a program to calculate the percentage of marks obtained in three subjects
(each out of 100) by student A and in four subjects (each out of 100) by student B.
Create an abstract class 'Marks' with an abstract method 'getPercentage'.

It is inherited by two other classes 'A' and 'B' each having a method with the same name
which returns the percentage of the students. The constructor of student A takes the
marks in three subjects
as its parameters and the marks in four subjects as its parameters for student B. Input to
be taken
from the user. Display the marks in highest order of student A and B.
 */

import java.util.Scanner;

abstract class Marks {
        abstract double getPercentage();
}

class A extends Marks {
        int sub1;
        int sub2;
        int sub3;

        public A(int sub1, int sub2, int sub3) {
                this.sub1 = sub1;
                this.sub2 = sub2;
                this.sub3 = sub3;
        }

        @Override
        double getPercentage() {
                int sum = sub1 + sub2 + sub3;
                return (double) (sum * 100) / 300;
        }
}


class B extends Marks {
    int sub1;
    int sub2;
    int sub3;
    int sub4;

    public B (int sub1, int sub2, int sub3, int sub4) {
        this.sub1 = sub1;
        this.sub2 = sub2;
        this.sub3 = sub3;
        this.sub4 = sub4;
    }

    @Override
    double getPercentage() {
            int sum = sub1 + sub2 + sub3 + sub4;
            return (double) (sum * 100) / 400;
    }
}

public class Marksheet {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.println("Enter marks for A: ");
        System.out.print("Sub1: ");
        int sub1 = in.nextInt();
        System.out.print("Sub2: ");
        int sub2 = in.nextInt();
        System.out.print("Sub3: ");
        int sub3 = in.nextInt();


        A a = new A(sub1, sub2, sub3);
        System.out.println("Percentage of A is: " + a.getPercentage());

        System.out.println("Enter marks for B: ");
        System.out.print("Sub1: ");
        sub1 = in.nextInt();
        System.out.print("Sub2: ");
        sub2 = in.nextInt();
        System.out.print("Sub3: ");
        sub3 = in.nextInt();
        System.out.print("Sub4: ");
        int sub4 = in.nextInt();

        B b = new B(sub1, sub2, sub3, sub4);
        System.out.println("Percentage of B is: " + b.getPercentage());

    }
}
```

RESULT:



```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Marksheet.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Marksheet
Enter marks for A:
Sub1: 87
Sub2: 89
Sub3: 99
Percentage of A is: 91.66666666666667
Enter marks for B:
Sub1: 45
Sub2: 84
Sub3: 78
Sub4: 78
Percentage of B is: 71.25
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$
```

CONCLUSION:	I have learnt about abstraction, abstract class and method overriding.