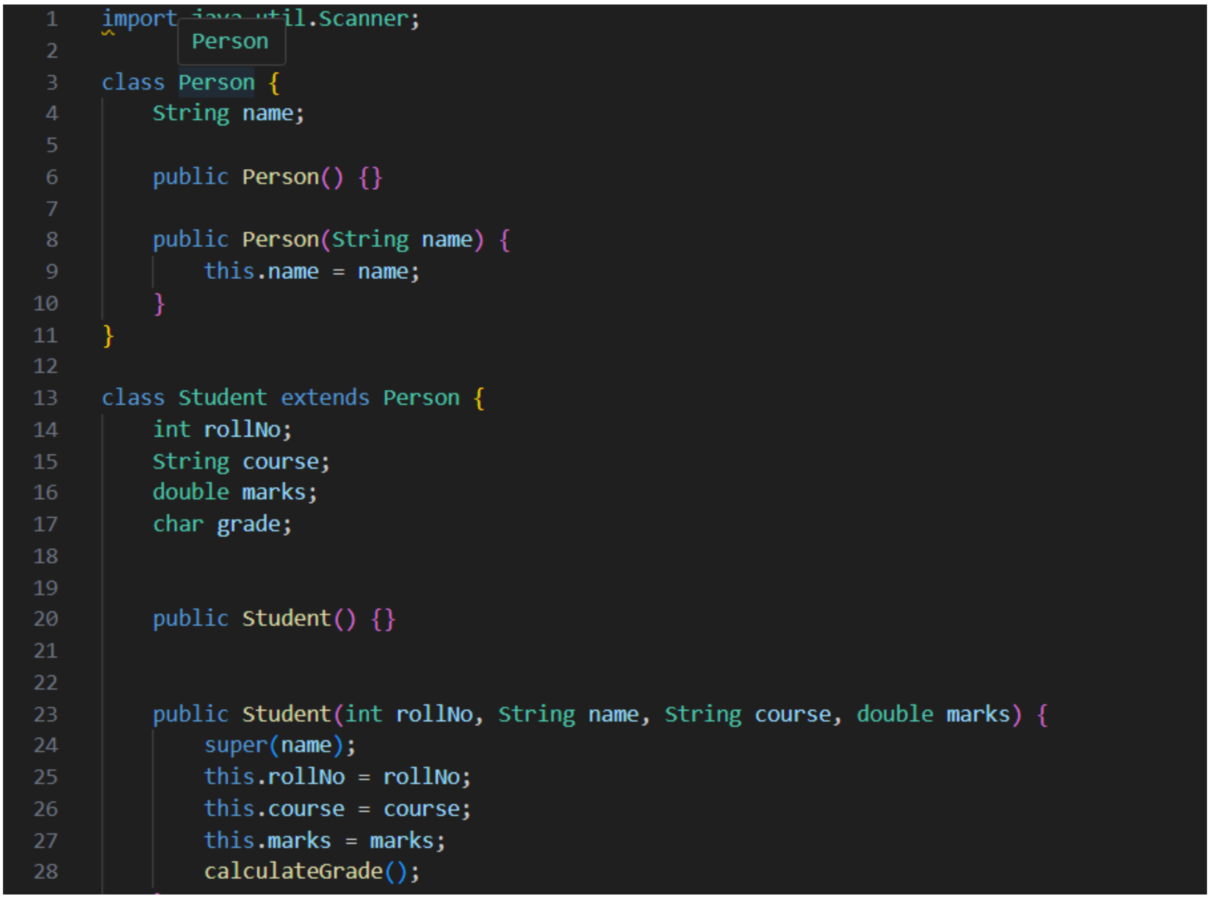
LAB ASSIGNMENT-1

Name-Rajat Rao

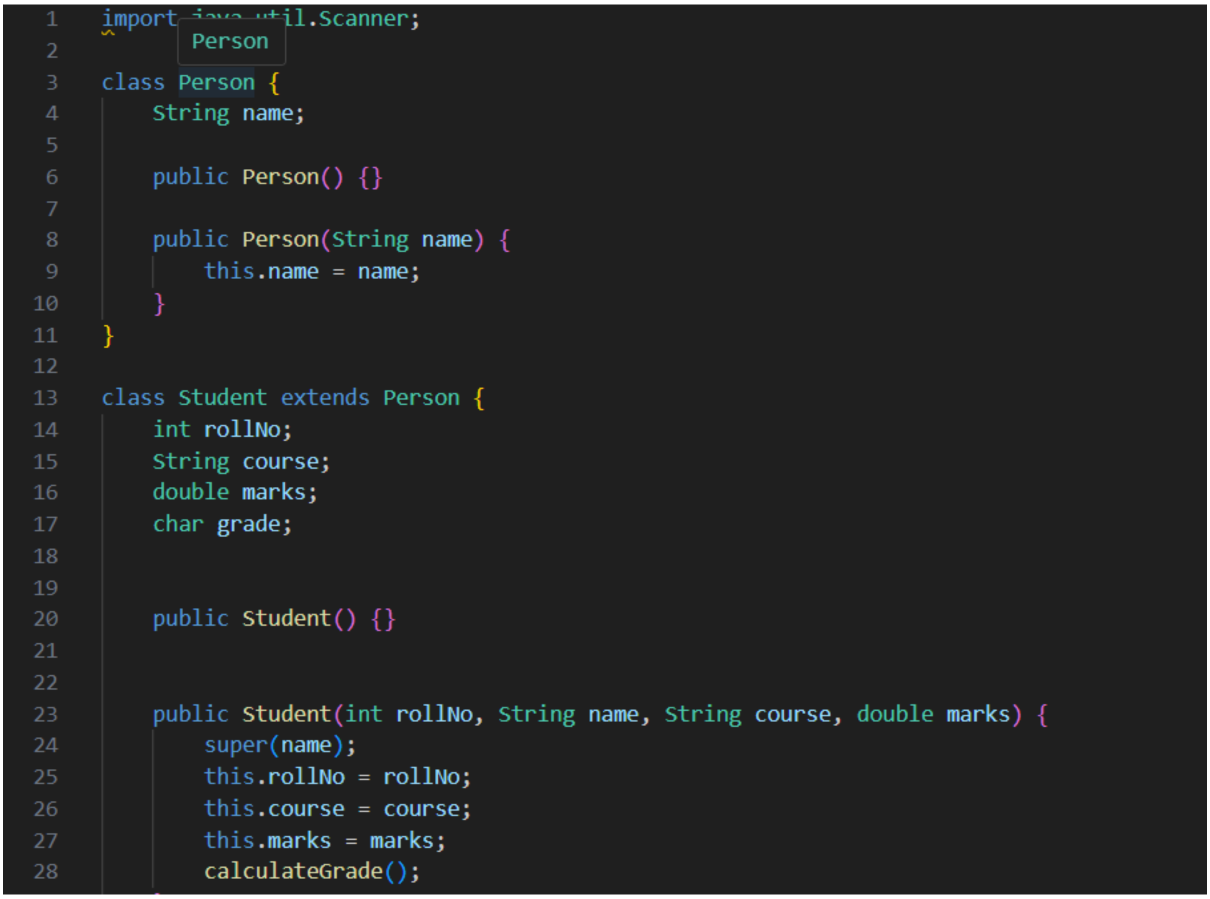
Roll no-2401201067

Course-BCA(AI&DS)

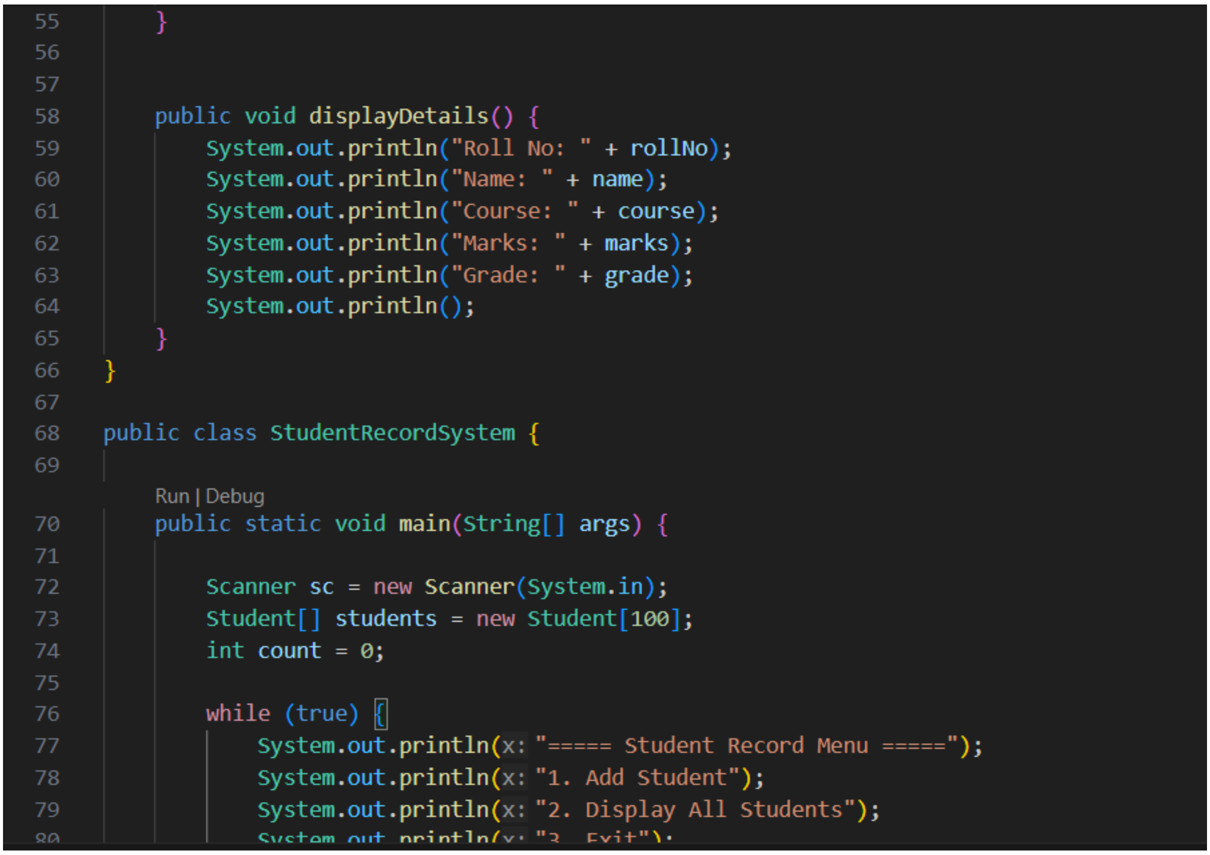
Input-



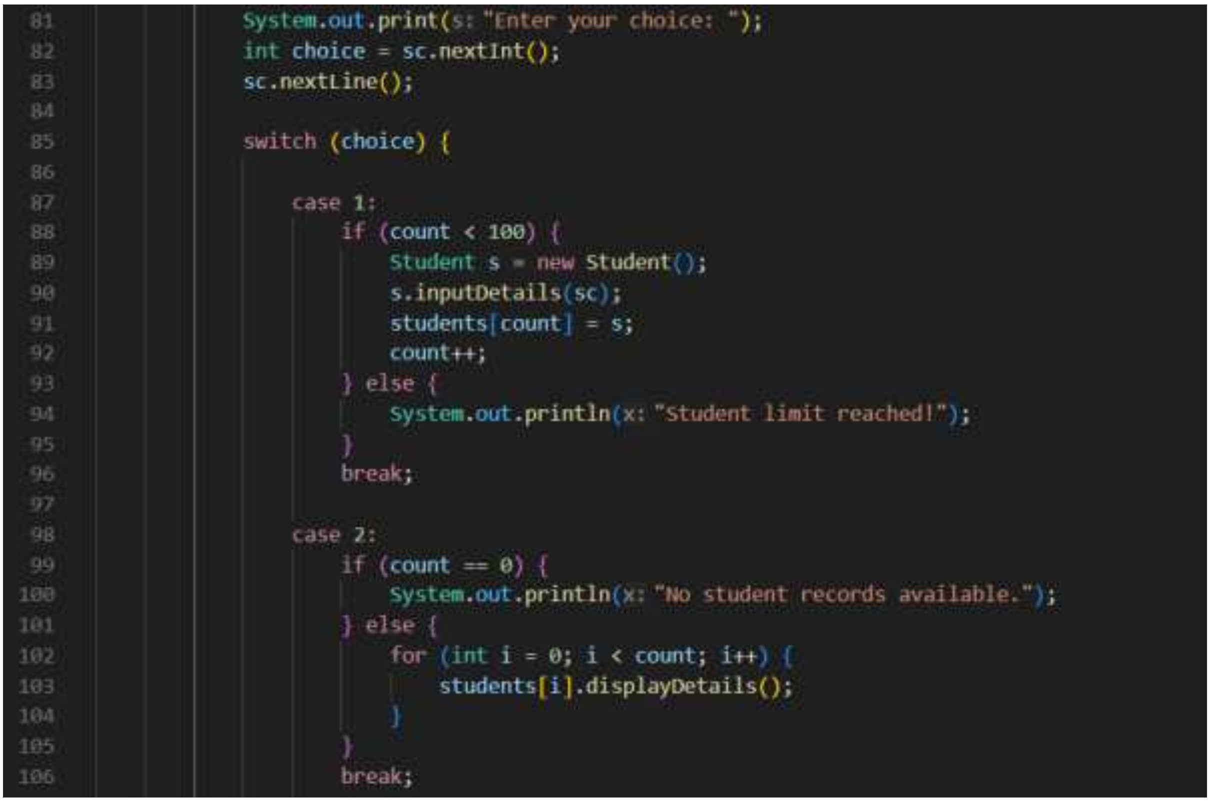
```java
import java.util.Scanner;

class Person {
    String name;

    public Person() {}

    public Person(String name) {
        this.name = name;
    }
}

class Student extends Person {
    int rollNo;
    String course;
    double marks;
    char grade;


    public Student() {}


    public Student(int rollNo, String name, String course, double marks) {
        super(name);
        this.rollNo = rollNo;
        this.course = course;
        this.marks = marks;
        calculateGrade();
```



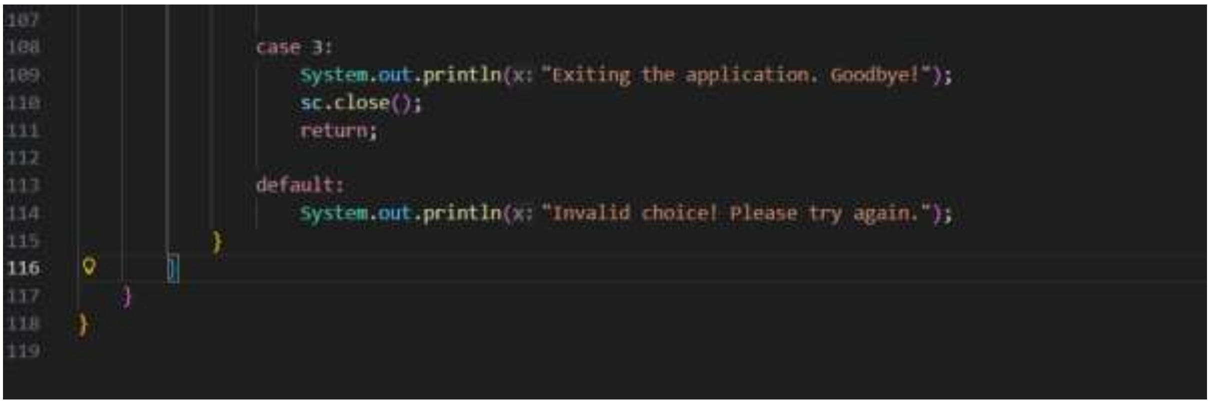
```java
import java.util.Scanner;

class Person {
    String name;

    public Person() {}

    public Person(String name) {
        this.name = name;
    }
}

class Student extends Person {
    int rollNo;
    String course;
    double marks;
    char grade;


    public Student() {}


    public Student(int rollNo, String name, String course, double marks) {
        super(name);
        this.rollNo = rollNo;
        this.course = course;
        this.marks = marks;
        calculateGrade();
```



```java
    }


    public void displayDetails() {
        System.out.println("Roll No: " + rollNo);
        System.out.println("Name: " + name);
        System.out.println("Course: " + course);
        System.out.println("Marks: " + marks);
        System.out.println("Grade: " + grade);
        System.out.println();
    }
}

public class StudentRecordSystem {

    Run | Debug
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        Student[] students = new Student[100];
        int count = 0;

        while (true) {
            System.out.println(x: "===== Student Record Menu =====");
            System.out.println(x: "1. Add Student");
            System.out.println(x: "2. Display All Students");
            System.out.println(x: "3. Exit");
```

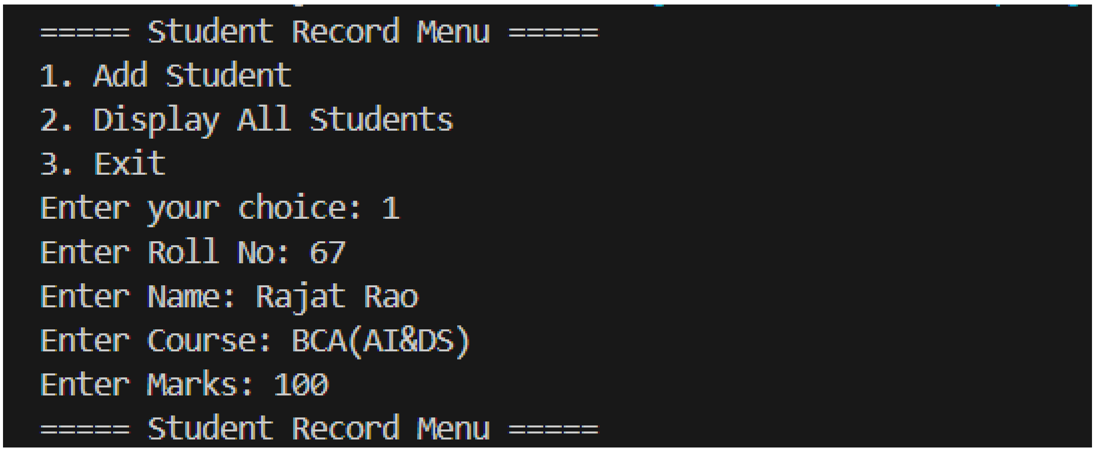


```java
                System.out.print(s: "Enter your choice: ");
                int choice = sc.nextInt();
                sc.nextLine();

                switch (choice) {

                    case 1:
                        if (count < 100) {
                            Student s = new Student();
                            s.inputDetails(sc);
                            students[count] = s;
                            count++;
                        } else {
                            System.out.println(x: "Student limit reached!");
                        }
                        break;

                    case 2:
                        if (count == 0) {
                            System.out.println(x: "No student records available.");
                        } else {
                            for (int i = 0; i < count; i++) {
                                students[i].displayDetails();
                            }
                        }
                        break;
```



```java

                    case 3:
                        System.out.println(x: "Exiting the application. Goodbye!");
                        sc.close();
                        return;

                    default:
                        System.out.println(x: "Invalid choice! Please try again.");
                }
            }
        }
    }

```

Output-



```
===== Student Record Menu =====
1. Add Student
2. Display All Students
3. Exit
Enter your choice: 1
Enter Roll No: 67
Enter Name: Rajat Rao
Enter Course: BCA(AI&DS)
Enter Marks: 100
===== Student Record Menu =====
```

EXPLANATION-

1.Person Class

- **A simple parent class that stores the student's name.**
- **Student class inherits this class to show basic inheritance.**

2. Student Class (inherits Person)

- **Stores student details: roll number, name, course, marks, grade.**
- **Has two constructors — default and parameterized.**
- **inputDetails() → takes student details from the user.**
- **calculateGrade() → assigns grade (A, B, C, D) based on marks.**
- **displayDetails() → prints all stored student information.**

3. StudentRecordSystem Class

- **Contains the main() method.**
- **Uses an array of Student objects to store multiple students.**
- **Uses a menu to let the user:**
    1. **Add a student**
    2. **Display all students**
    3. **Exit the program**
- **Uses loops and switch-case to control program flow.**