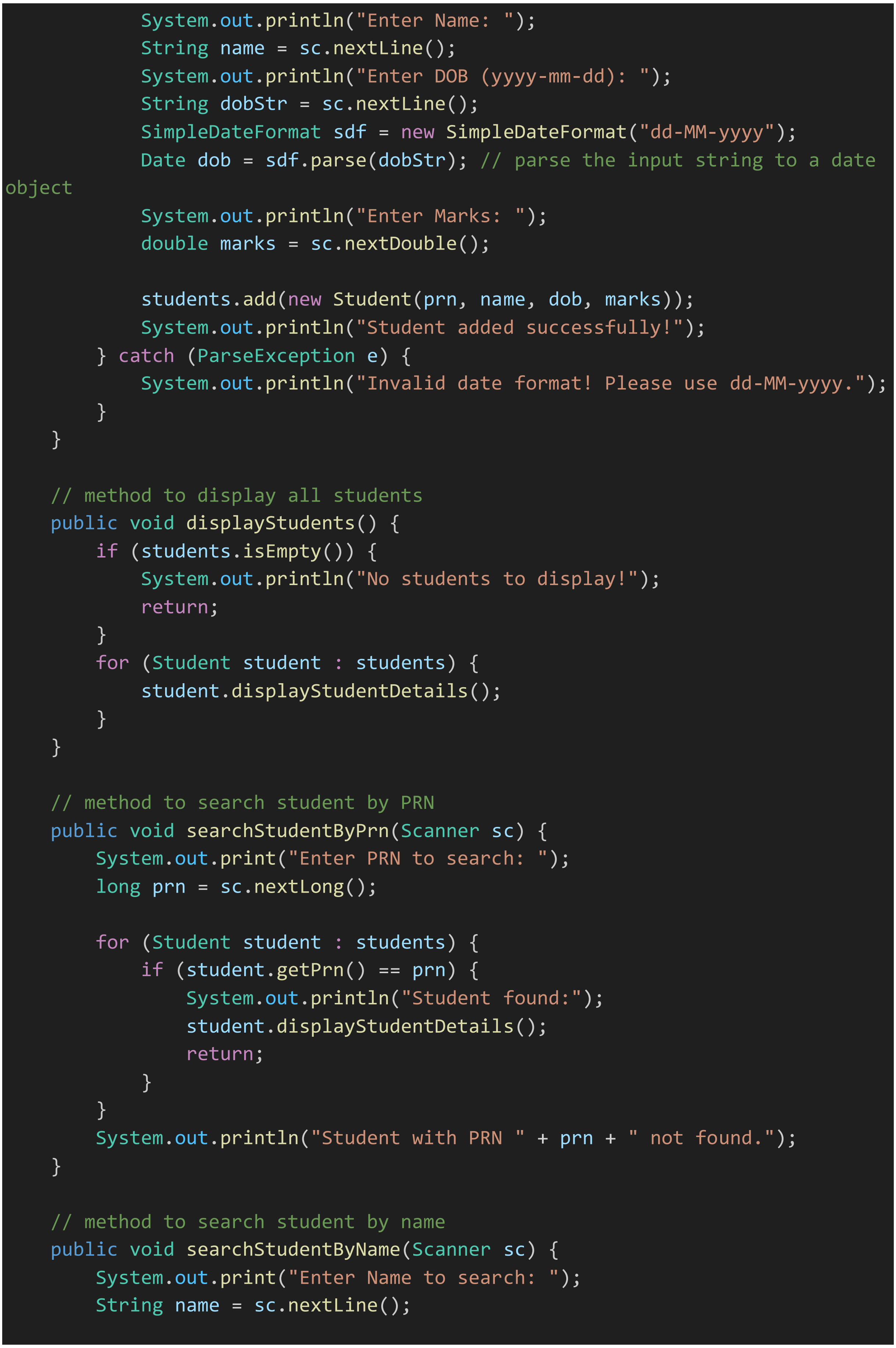
PIJ Assignment 3: Student Management System

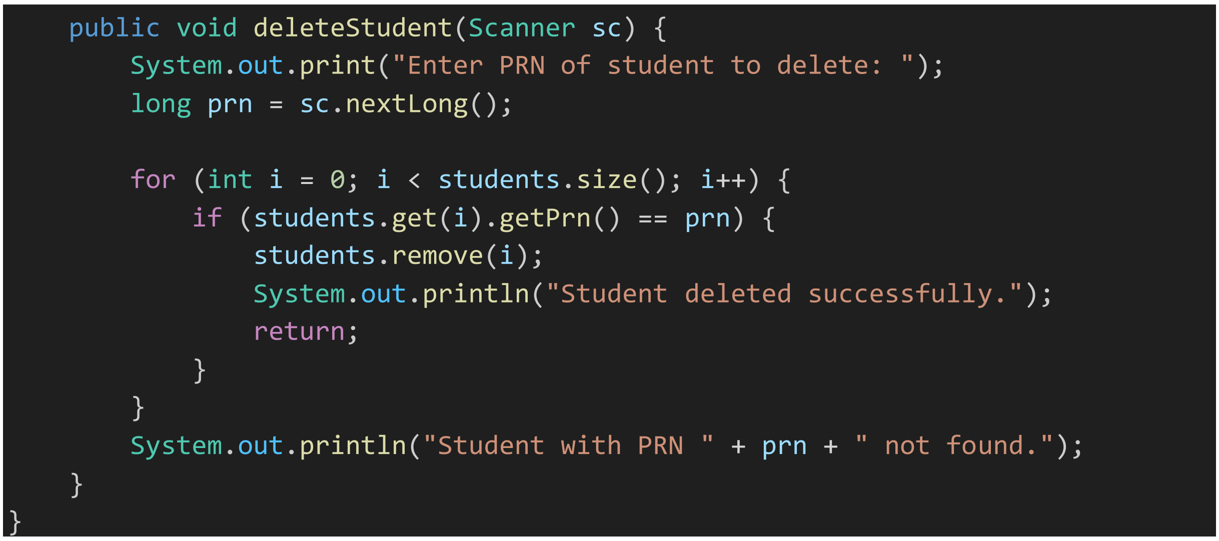
```java
// Name: Samarth Bhadane
// PRN: 24070126503
// Batch: A2

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        // creating Scanner object
        Scanner sc = new Scanner(System.in);
        // creating StudentOperations object
        StudentOperations operations = new StudentOperations();
        int choice;

        // displaying the menu
        do {
            System.out.println("\n=== Student Management System ===");
            System.out.println("1. Add Student");
            System.out.println("2. Display Students");
            System.out.println("3. Search Student by PRN");
            System.out.println("4. Search Student by Name");
            System.out.println("5. Update Student");
            System.out.println("6. Delete Student");
            System.out.println("7. Exit");
            System.out.print("Enter your choice: ");
            choice = sc.nextInt();
            sc.nextLine();

            // switch case to handle operations according to input
            switch (choice) {
                case 1:
                    operations.addStudent(sc);
                    break;
                case 2:
                    operations.displayStudents();
                    break;
                case 3:
                    operations.searchStudentByPrn(sc);
                    break;
                case 4:
                    operations.searchStudentByName(sc);
                    break;
                case 5:
                    operations.updateStudent(sc);
                    break;
                case 6:
                    operations.deleteStudent(sc);
                    break;
                case 7:
                    System.out.println("Exiting...");
                    break;
                default:
                    System.out.println("Invalid choice! Try again.");
            }
        } while (choice != 7);
        sc.close();
    }
}
```

```
// Student.java
import java.text.SimpleDateFormat;
import java.util.Date;

public class Student {
    // declaring the instance members
    private long prn;
    private String name;
    private Date dob;
    private double marks;

    // declaring constructors
    public Student(long prn, String name, Date dob, double marks) {
        this.prn = prn;
        this.name = name;
        this.dob = dob;
        this.marks = marks;
    }

    // getters and setters
    public long getPrn() {
        return prn;
    }

    public void setPrn(long prn) {
        this.prn = prn;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
    public Date getDob() {
        return dob;
    }

    public void setDob(Date dob) {
        this.dob = dob;
    }

    public double getMarks() {
        return marks;
    }

    public void setMarks(double marks) {
        this.marks = marks;
    }

    // display method
    public void displayStudentDetails() {
        SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");
        System.out.println("PRN: " + prn);
        System.out.println("Name: " + name);
        System.out.println("DOB: " + sdf.format(dob));
        System.out.println("Marks: " + marks);
    }
}
```

```
// StudentOperations.java
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.Scanner;

public class StudentOperations {
    private ArrayList<Student> students;

    // constructor
    public StudentOperations() {
        students = new ArrayList<>();
    }

    // method to add student
    public void addStudent(Scanner sc) {
        try {
            System.out.println("Enter PRN: ");
            long prn = sc.nextLong();
            sc.nextLine();
```



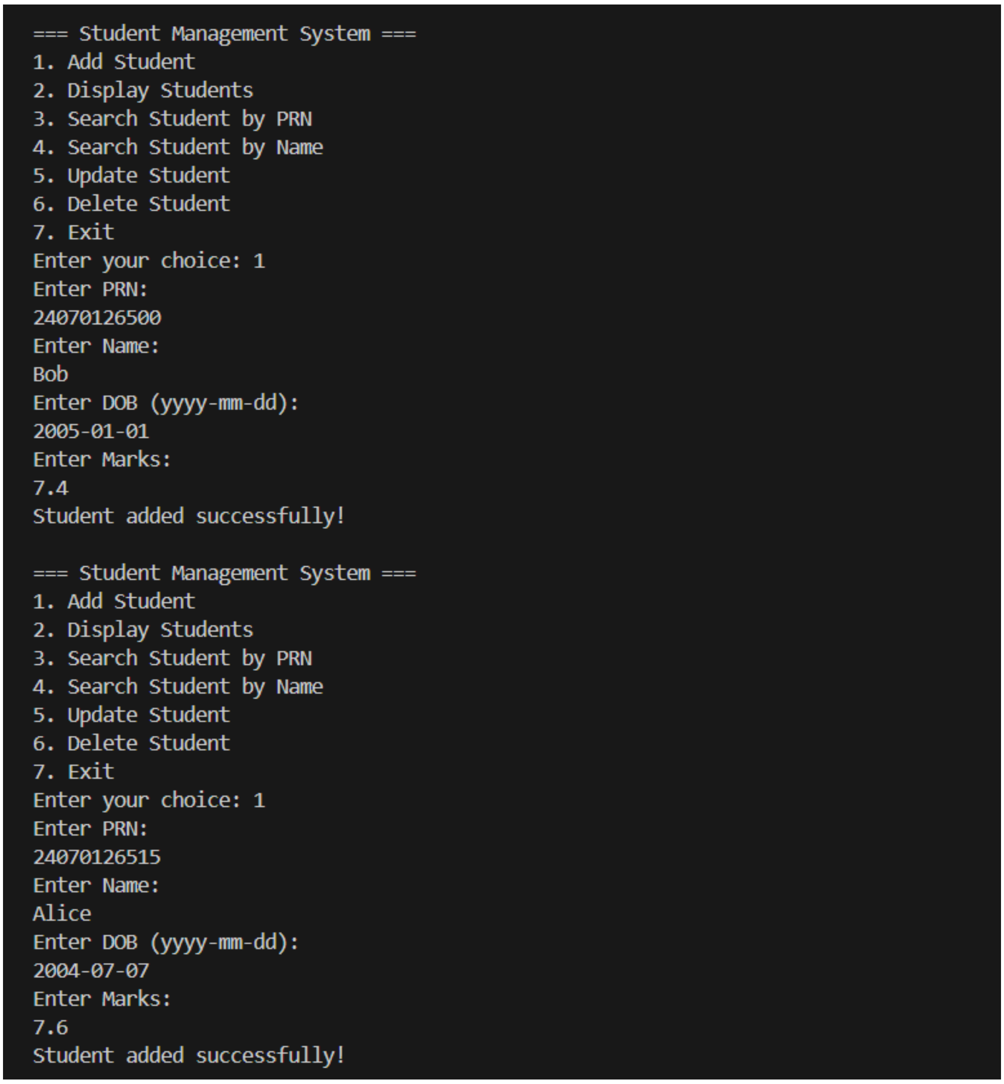
```
            System.out.println("Enter Name: ");
            String name = sc.nextLine();
            System.out.println("Enter DOB (yyyy-mm-dd): ");
            String dobStr = sc.nextLine();
            SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");
            Date dob = sdf.parse(dobStr); // parse the input string to a date
object
            System.out.println("Enter Marks: ");
            double marks = sc.nextDouble();

            students.add(new Student(prn, name, dob, marks));
            System.out.println("Student added successfully!");
        } catch (ParseException e) {
            System.out.println("Invalid date format! Please use dd-MM-yyyy.");
        }
    }

    // method to display all students
    public void displayStudents() {
        if (students.isEmpty()) {
            System.out.println("No students to display!");
            return;
        }
        for (Student student : students) {
            student.displayStudentDetails();
        }
    }

    // method to search student by PRN
    public void searchStudentByPrn(Scanner sc) {
        System.out.print("Enter PRN to search: ");
        long prn = sc.nextLong();

        for (Student student : students) {
            if (student.getPrn() == prn) {
                System.out.println("Student found:");
                student.displayStudentDetails();
                return;
            }
        }
        System.out.println("Student with PRN " + prn + " not found.");
    }

    // method to search student by name
    public void searchStudentByName(Scanner sc) {
        System.out.print("Enter Name to search: ");
        String name = sc.nextLine();
        for (Student student : students) {
            if (student.getName().equalsIgnoreCase(name)) {
                System.out.println("Student found:");
                student.displayStudentDetails();
                return;
            }
        }
        System.out.println("Student with Name " + name + " not found.");
    }

    // method to update student details
    public void updateStudent(Scanner sc) {
        System.out.print("Enter PRN of student to update: ");
        long prn = sc.nextLong();
        sc.nextLine();

        for (Student student : students) {
            if (student.getPrn() == prn) {
                System.out.print("Enter new Name: ");
                String name = sc.nextLine();
                student.setName(name);

                System.out.print("Enter new Date of Birth (yyyy-mm-dd): ");
                String dobStr = sc.nextLine();
                if (!dobStr.isEmpty()) {
                    SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");
                    try {
                        Date dob = sdf.parse(dobStr);
                        student.setDob(dob);
                    } catch (ParseException e) {
                        System.out.println("Invalid date format! Please use
dd-MM-yyyy.");
                        return;
                    }
                }

                System.out.print("Enter new Marks: ");
                double marks = sc.nextDouble();
                student.setMarks(marks);

                System.out.println("Student updated successfully.");
                return;
            }
        }
        System.out.println("Student with PRN " + prn + " not found.");
    }

    // method to delete student
```



```
    public void deleteStudent(Scanner sc) {
        System.out.print("Enter PRN of student to delete: ");
        long prn = sc.nextLong();

        for (int i = 0; i < students.size(); i++) {
            if (students.get(i).getPrn() == prn) {
                students.remove(i);
                System.out.println("Student deleted successfully.");
                return;
            }
        }
        System.out.println("Student with PRN " + prn + " not found.");
    }
}
```

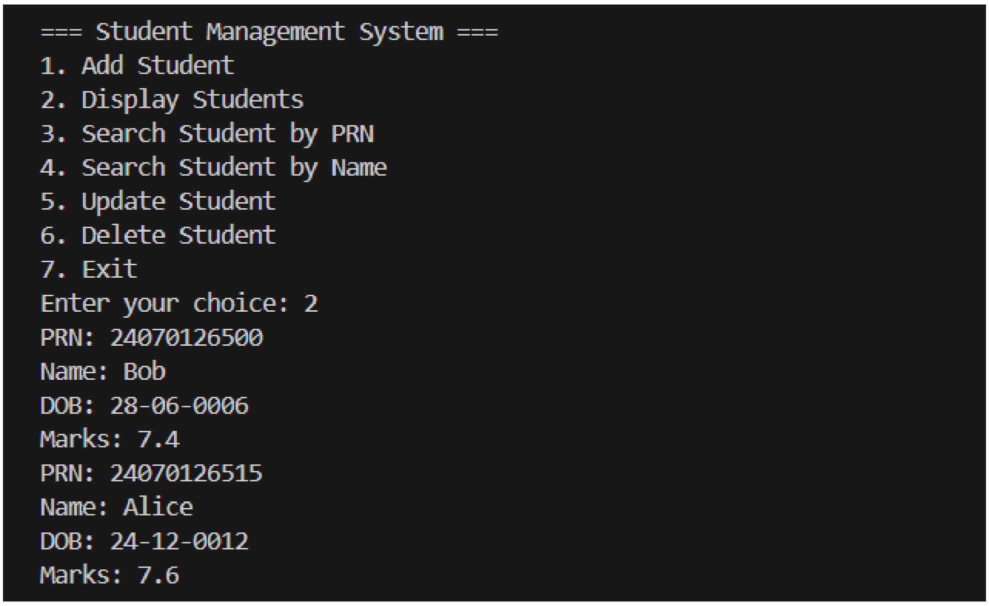
Output-

1. Add Student



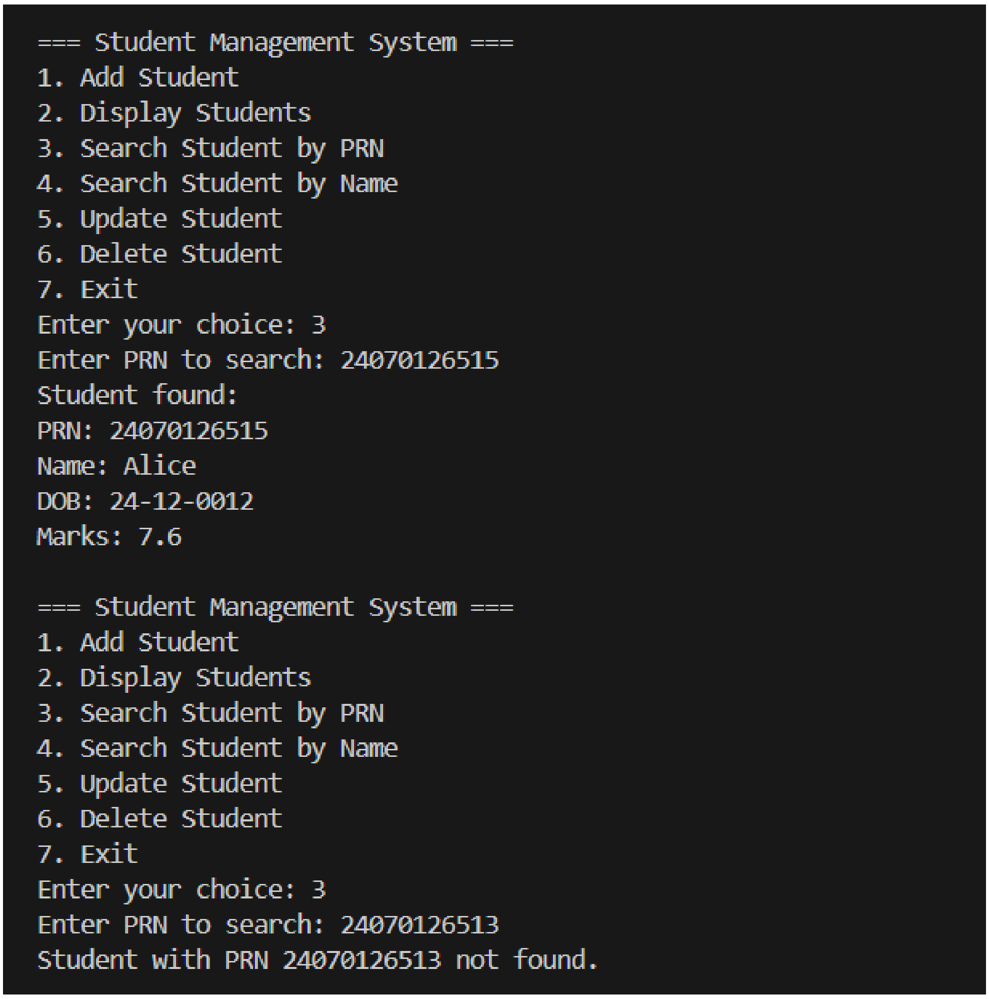
```
=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 1
Enter PRN:
24070126500
Enter Name:
Bob
Enter DOB (yyyy-mm-dd):
2005-01-01
Enter Marks:
7.4
Student added successfully!

=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 1
Enter PRN:
24070126515
Enter Name:
Alice
Enter DOB (yyyy-mm-dd):
2004-07-07
Enter Marks:
7.6
Student added successfully!
```

2. Display Students



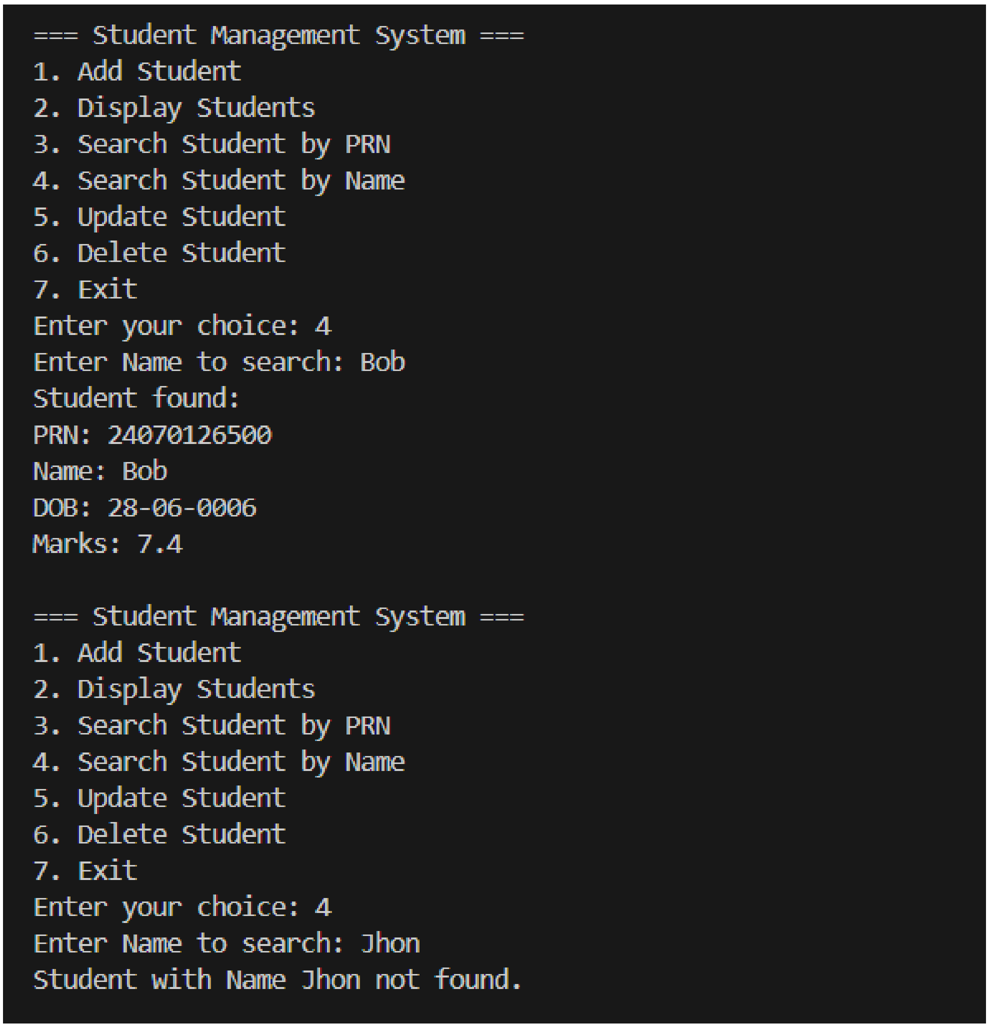
```
=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 2
PRN: 24070126500
Name: Bob
DOB: 28-06-0006
Marks: 7.4
PRN: 24070126515
Name: Alice
DOB: 24-12-0012
Marks: 7.6
```

3. Search Student by PRN



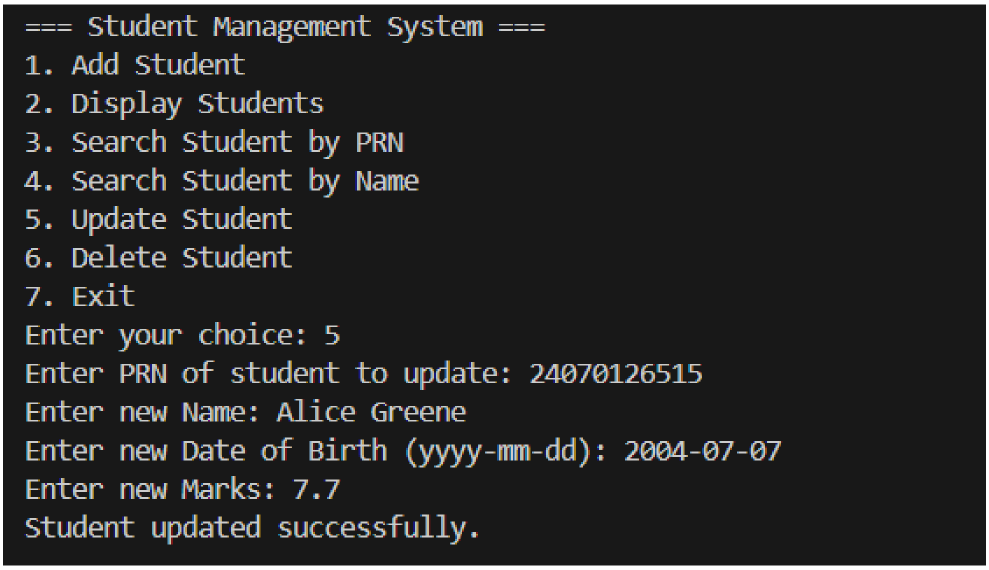
```
=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 3
Enter PRN to search: 24070126515
Student found:
PRN: 24070126515
Name: Alice
DOB: 24-12-0012
Marks: 7.6

=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 3
Enter PRN to search: 24070126513
Student with PRN 24070126513 not found.
```

4. Search student by name



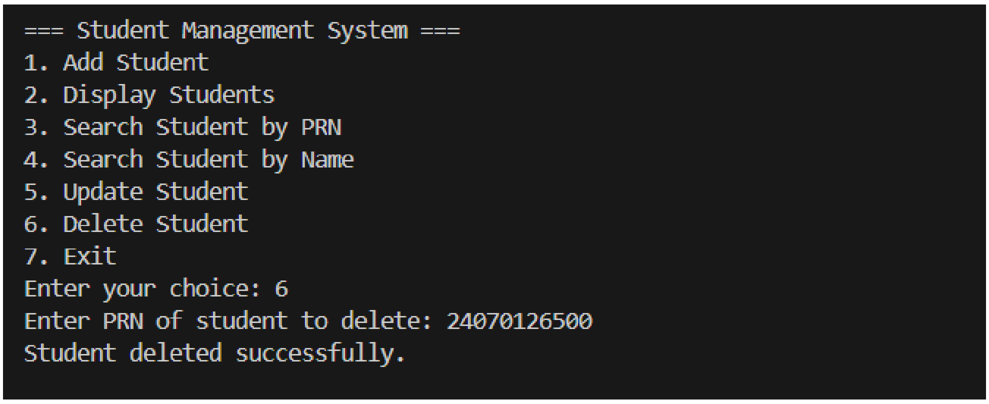
```
=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 4
Enter Name to search: Bob
Student found:
PRN: 24070126500
Name: Bob
DOB: 28-06-0006
Marks: 7.4

=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 4
Enter Name to search: Jhon
Student with Name Jhon not found.
```

5. Update Student



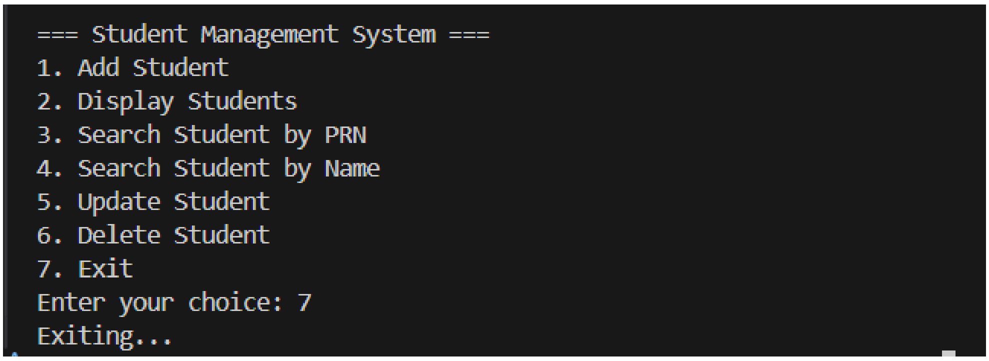
```
=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 5
Enter PRN of student to update: 24070126515
Enter new Name: Alice Greene
Enter new Date of Birth (yyyy-mm-dd): 2004-07-07
Enter new Marks: 7.7
Student updated successfully.
```

6. Delete Student



```
=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 6
Enter PRN of student to delete: 24070126500
Student deleted successfully.
```

7. Exit



```
=== Student Management System ===
1. Add Student
2. Display Students
3. Search Student by PRN
4. Search Student by Name
5. Update Student
6. Delete Student
7. Exit
Enter your choice: 7
Exiting...
```

Link to the repository: https://github.com/samarthsb4real/PIJ-Assignment-3.git