LAB Assignment – 4

NAME – SHASHANK TYAGI

ROLL NO – 2401201033

COURSE- BCA(AI&DS)

CODE –

Main.java 2 ●

C: > Users > tyagi > OneDrive > Desktop > src > Main.java > Language Support for Java(TM) by Red Hat > Main

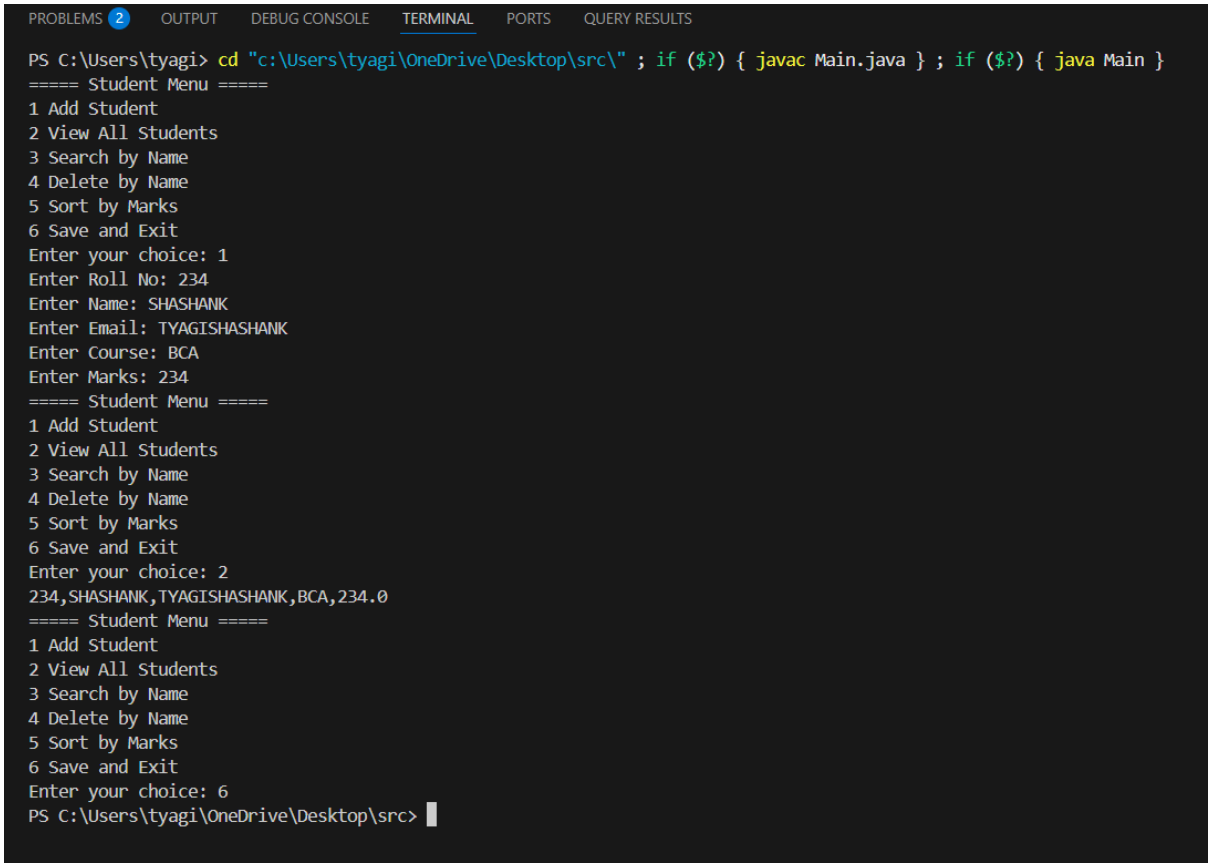
```java
import java.util.*;
public class Main {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        StudentManager sm = new StudentManager();
        String path = "students.txt";
        sm.loadFromFile(path);
        while(true) {
            System.out.println(x: "===== Student Menu =====");
            System.out.println(x: "1 Add Student");
            System.out.println(x: "2 View All Students");
            System.out.println(x: "3 Search by Name");
            System.out.println(x: "4 Delete by Name");
            System.out.println(x: "5 Sort by Marks");
            System.out.println(x: "6 Save and Exit");
            System.out.print(s: "Enter your choice: ");
            int ch = sc.nextInt();
            sc.nextLine();
            if(ch==1) {
                System.out.print(s: "Enter Roll No: ");
                int r=sc.nextInt();
                sc.nextLine();
                System.out.print(s: "Enter Name: ");
                String n=sc.nextLine();
                System.out.print(s: "Enter Email: ");
                String e=sc.nextLine();
                System.out.print(s: "Enter Course: ");
                String c=sc.nextLine();
                System.out.print(s: "Enter Marks: ");
                double m=sc.nextDouble();
                sm.addStudent(new Student(r,n,e,c,m));
            } else if(ch==2) {
                for(Student s:sm.getAll()) System.out.println(s);
            } else if(ch==3) {
                System.out.print(s: "Enter Name: ");
                String n=sc.nextLine();
```

Main.java 2 ●

C: > Users > tyagi > OneDrive > Desktop > src > Main.java > Language Support for Java(TM) by Red Hat > Mai

```java
public class Main {
    public static void main(String[] args) {
                String n=sc.nextLine();
                Student s=sm.searchByName(n);
                if(s!=null) System.out.println(s);
            } else if(ch==4) {
                System.out.print(s: "Enter Name: ");
                String n=sc.nextLine();
                sm.deleteByName(n);
            } else if(ch==5) {
                for(Student s:sm.sortByMarks()) System.out.println(s);
            } else if(ch==6) {
                sm.saveToFile(path);
                break;
            }
        }
    }
}
```

OUTPUT –



EXPLANATION –

The program is a Student Record Management System that stores details of multiple students such as roll number, name, email, course, and marks. It uses Object-Oriented Programming, Collections, and File Handling to manage data efficiently.

- **1. Student Class**

This class represents a single student. It contains variables for storing the student's information and provides methods to access that data. It acts as a blueprint for creating student objects.

- **2. File Handling Class**

This class is responsible for reading student data from a file at the start and writing updated data back to the file when the user exits. It reads each line from the file, converts it into a student object, and stores it in a list. Similarly, it writes each student's details into the file to save them permanently.

- **3. Student Manager Class**

This class manages all student records using an ArrayList.
It can:

- **Add a student**
- **Display all students**
- **Search a student by name**
- **Delete a student**
- **Sort students by marks**

It works as the main controller for performing operations on the data.

- **4. Main Class (User Menu)**

This class shows a menu to the user and asks for their choice. Based on the input, it performs different actions such as adding a student, viewing all records, searching, deleting, sorting, or exiting. When exiting, it saves the updated list of students to the file.