

Name:Sumit Kalshetty

Class: SE-IT-A

Roll no. 2125

Subject: OOP LAB

Practical 8: File Handling

Problem Statement:

Implement a program for maintaining a student records database using File Handling. Student has Student_id, name, Roll_no, Class, marks and address. Display the data for five students.

- i) Create Database
- ii) Display Database
- iii) Clear Records
- iv) Modify record
- v) Search Record

*****PROGRAM*****

```
import java.io.*;
import java.util.ArrayList;
import java.util.List;

public class StudentRecords {

    private static final String FILE_NAME = "st.txt";
    static BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
    private List<Student> students = new ArrayList<>();
    // Inner class to represent a student
```

```

static class Student {
String name;
int id;
int rollNo;
String address;
int studentClass;
float marks;

public Student(String name, int id, int rollNo, String address, int studentClass,
float marks) {
this.name = name;
this.id = id;
this.rollNo = rollNo;
this.address = address;
this.studentClass = studentClass;
this.marks = marks;
}

```

@Override

```

public String toString() {
return name + " " + id + " " + rollNo + " " + address + " " + studentClass + " " +
marks;
}
}

```

```

public void addRecords() throws IOException {
do{
System.out.print("\nEnter Student Name: ");
String studentName = br.readLine();
System.out.print("Student Id: ");
int studentId = Integer.parseInt(br.readLine());
System.out.print("Roll no: ");
int rollNo = Integer.parseInt(br.readLine());
System.out.print("Address: ");
String address = br.readLine();

```

```

System.out.print("Class: ");
int studentClass = Integer.parseInt(br.readLine());
System.out.print("Marks: ");
float marks = Float.parseFloat(br.readLine());

// Check for unique ID
if (students.stream().anyMatch(s -> s.id == studentId)) {
System.out.println("ERROR: Student ID must be unique.");
continue;
}

students.add(new Student(studentName, studentId, rollNo, address,
studentClass, marks));
System.out.print("\nRecords added successfully!\n");

System.out.print("\nDo you want to add more records? (y/n): "); }
while (br.readLine().equalsIgnoreCase("y"));

writeToFile();
showMenu();
}

private void writeToFile() throws IOException {
try (PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter(FILE_NAME)))) {
for (Student student : students) {
pw.println(student);
}
}
}

public void readRecords() throws IOException {
if (students.isEmpty()) {
System.out.println("\nNo records found.");
return;
}
System.out.println("\n--- Student Records ---");
for (Student student : students) {

```

```

System.out.println(student);
}
showMenu();
}

public void clearRecords() throws IOException {
students.clear();
writeToFile();
System.out.println("\nAll records cleared successfully!");
showMenu();
}

public void showMenu() throws IOException {

System.out.println("\n1. Add Records");
System.out.println("2. Display Records");
System.out.println("3. Clear All Records");
System.out.println("4. Exit");
System.out.print("\nEnter your choice: "); int
choice = Integer.parseInt(br.readLine());

switch (choice) {
case 1 -> addRecords();
case 2 -> readRecords();
case 3 -> clearRecords();
case 4 -> System.exit(0);
default -> {
System.out.println("\nInvalid choice!");
showMenu();
}
}
}

public static void main(String[] args) throws IOException {
StudentRecords call = new StudentRecords();
call.showMenu();
}
}

```

*****OUTPUT*****

C:\Users\acer\Documents\2104>java file.java

1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit

Enter your choice: 1

Enter Student Name: Jack Ryan

Student Id: 22264

Roll no: 2126

Address: Los Santos

Class: 12

Marks: 99

Records added successfully!

Do you want to add more records? (y/n): y

Enter Student Name: John Wick

Student Id: 1329

Roll no: 2128

Address: High Command

Class: 12

Marks: 98

Records added successfully!

Do you want to add more records? (y/n): y

Enter Student Name: Billy Butcher

Student Id: 1344

Roll no: 2143

Address: America Class: 12 Marks: 97

Records added successfully!

Do you want to add more records? (y/n): n

1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit

Enter your choice: 2

---Student Records---

Jack Ryan 22264 2126 Los Santos 12 99.0

John Wick 1329 2128 High Command 12 98.0

Billy Butcher 1344 2143 America 12 97.0

1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit

Enter your choice: 4

Enter the ID of the student you want to search: 1329 Record
found: John Wick 1329 2128 High Command 12 98.0

1. Add Records
2. Display Records
3. Clear All Records

4. Search Records 5. Delete Records 6. Update Records 7. Exit

Enter your choice: 5

Enter the ID of the student you want to delete:

1329 Record deleted successfully!

1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit

Enter your choice: 6

Enter the ID of the student you want to update:
22264

Record found: Jack Ryan 22264 2126 Los Santos 12

99.0 Enter updated marks: 99

Record updated successfully!

1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit

Enter your choice: 3

All records cleared successfully!

1. Add Records
2. Display Records
3. Clear All Records
4. Search Records

- 5. Delete Records
- 6. Update Records
- 7. Exit

Enter your choice: 2

---Student Records---

- 1. Add Records
- 2. Display Records
- 3. Clear All Records
- 4. Search Records
- 5. Delete Records
- 6. Update Records
- 7. Exit

Enter your choice: 7

C:\Users\acer\Documents\2123

>