Pratical 8

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

class Student {
    int studentId;
    String name;
    int rollNo;
    String studentClass;
    double marks;
    String address;

    Student(int studentId, String name, int rollNo, String studentClass, double marks, String address) {
        this.studentId = studentId;
        this.name = name;
        this.rollNo = rollNo;
        this.studentClass = studentClass;
        this.marks = marks;
        this.address = address;
    }

    @Override
    public String toString() {
        return studentId + "," + name + "," + rollNo + "," + studentClass + "," + marks + "," + address;
    }
}

public class StudentDatabase {
    private static final String FILE_NAME = "student_database.txt";

    public static void createDatabase(List<Student> students) {
```

```
        try (BufferedWriter writer = new BufferedWriter(new FileWriter(FILE_NAME))) {
            for (Student student : students) {
                writer.write(student.toString());
                writer.newLine();
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    public static void displayDatabase() {
        try (BufferedReader reader = new BufferedReader(new FileReader(FILE_NAME))) {
            String line;
            while ((line = reader.readLine()) != null) {
                String[] data = line.split(",");
                System.out.println("ID: " + data[0] + ", Name: " + data[1] + ", Roll No: " + data[2] + ", Class: "
+ data[3] + ", Marks: " + data[4] + ", Address: " + data[5]);
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    public static void deleteRecord(int studentId) {
        List<Student> students = new ArrayList<>();
        try (BufferedReader reader = new BufferedReader(new FileReader(FILE_NAME))) {
            String line;
            while ((line = reader.readLine()) != null) {
                String[] data = line.split(",");
                if (Integer.parseInt(data[0]) != studentId) {
                    students.add(new Student(Integer.parseInt(data[0]), data[1], Integer.parseInt(data[2]),
data[3], Double.parseDouble(data[4]), data[5]));
```

```
            }
        }
        try (BufferedWriter writer = new BufferedWriter(new FileWriter(FILE_NAME))) {
            for (Student student : students) {
                writer.write(student.toString());
                writer.newLine();
            }
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}


public static void updateRecord(int studentId, String newName, String newClass, double
newMarks, String newAddress) {
    List<Student> students = new ArrayList<>();
    try (BufferedReader reader = new BufferedReader(new FileReader(FILE_NAME))) {
        String line;
        while ((line = reader.readLine()) != null) {
            String[] data = line.split(",");
            if (Integer.parseInt(data[0]) == studentId) {
                students.add(new Student(studentId, newName, Integer.parseInt(data[2]), newClass,
newMarks, newAddress));
            } else {
                students.add(new Student(Integer.parseInt(data[0]), data[1], Integer.parseInt(data[2]),
data[3], Double.parseDouble(data[4]), data[5]));
            }
        }
        try (BufferedWriter writer = new BufferedWriter(new FileWriter(FILE_NAME))) {
            for (Student student : students) {
                writer.write(student.toString());
                writer.newLine();
```

```
            }
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}

public static void searchRecord(int studentId) {
    try (BufferedReader reader = new BufferedReader(new FileReader(FILE_NAME))) {
        String line;
        while ((line = reader.readLine()) != null) {
            String[] data = line.split(",");
            if (Integer.parseInt(data[0]) == studentId) {
                System.out.println("ID: " + data[0] + ", Name: " + data[1] + ", Roll No: " + data[2] + ", Class: " + data[3] + ", Marks: " + data[4] + ", Address: " + data[5]);
                return;
            }
        }
        System.out.println("Student record not found.");
    } catch (IOException e) {
        e.printStackTrace();
    }
}

public static void main(String[] args) {
    List<Student> students = new ArrayList<>();
    students.add(new Student(1, "Alice", 101, "10th", 89.5, "123 Street"));
    students.add(new Student(2, "Bob", 102, "10th", 75.0, "456 Avenue"));
    students.add(new Student(3, "Charlie", 103, "11th", 92.0, "789 Road"));

    createDatabase(students);
```

```
        System.out.println("Database after creation:");
        displayDatabase();

        System.out.println("\nDeleting record with ID 2:");
        deleteRecord(2);
        displayDatabase();

        System.out.println("\nUpdating record with ID 3:");
        updateRecord(3, "Charlie Updated", "12th", 95.0, "101 Boulevard");
        displayDatabase();

        System.out.println("\nSearching for record with ID 1:");
        searchRecord(1);
    }
}
```