Practical 2

Q. Create a student result database in Java. Calculate the grades of students. Decide criteria for best student and short-list students who satisfy the criteria.

a) A student has a rollNo, name, marks in five courses and a grade. A student list has many students, If a student has grade equal or beyond 9, he is considered as a top band student.

b) Create at least ten students. From these, find all such students which satisfy the criteria for top band student. Create a list of such students and display the students in the list.

database.fmt

```
//ROLL:NAME:S1:S2:S3:S4:S5
603231:DEV:98:96:98:99:98
603927:ABD:85:34:56:75:97
423987:YTG:76:86:88:87:98
432675:HRG:63:87:98:64:67
343255:KJH:98:63:98:24:65
524663:HBY:59:59:35:70:90
432096:UVG:34:65:23:67:65
596973:POM:53:98:32:65:65
698347:UNS:57:64:74:96:23
602485:YSN:65:73:34:64:87
```

Main.java

```
import java.io.FileReader;
import java.io.IOException;
import java.nio.file.Paths;
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.util.Scanner;
import java.util.List;
import java.util.ArrayList;
import java.util.Collections;
```

Main.java

```java
public class Main {
    public static void main(String[] args) throws IOException, FileNotFoundException {
        String databaseFileName = Paths.get(System.getProperty("user.dir"),
"database.fmt").toString();
        // Read Database file, import into ArrayList
        List<Student> studentList = new ArrayList<Student>();
        try (
                BufferedReader databaseFile = new BufferedReader(
                        new FileReader(databaseFileName))) {

            String databaseLine = databaseFile.readLine();

            while (databaseLine != null) {
                // If the line is a comment, move on
                if (databaseLine.startsWith("//") || databaseLine.length() == 0) {
                    databaseLine = databaseFile.readLine();
                    continue;
                }

                try (Scanner readLine = new Scanner(databaseLine)) {
                    readLine.useDelimiter(":");
                    int roll = readLine.nextInt();
                    String name = readLine.next();

                    int[] marks = {
                            readLine.nextInt(), // S1
                            readLine.nextInt(), // S2
                            readLine.nextInt(), // S3
                            readLine.nextInt(), // S4
                            readLine.nextInt(), // S5
                    };

                    studentList.add(new Student(roll, name, marks));
                    databaseLine = databaseFile.readLine();
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

        // Get the students sorted by grade (descending)
        Collections.sort(studentList);
        Collections.reverse(studentList);

        List<Student> topBandStudent = new ArrayList<Student>();

        // Create list of top band students
        for (Student student : studentList) {
            if (student.getGradePoint() >= 8.0) {
                topBandStudent.add(student);
            }
        }

        // Print the top band students
        System.out.println("Roll.  Name [S1, S2, S3, S4, S5] Grade");
        for (Student student : topBandStudent) {
            System.out.println(student);
        }
    }
}
```

Student.java

```java
import java.util.Arrays;

public class Student implements Comparable<Student> {
    int studentRollNumber = 0;
    String studentName = "";
    // MAX MARKS : 100 per subject
    int[] studentMarks = {0, 0, 0, 0, 0};
    double studentGrade = 0;

    public Student(int roll, String name, int[] marks) {
        studentRollNumber = roll;
        studentName = name;
        studentMarks = marks;

        // Calc Grade
        int total_marks = 0;
        for (int i : marks) {
            total_marks += i;
        }

        studentGrade = total_marks / 50.0;
    }

    // @Override
    public String toString() {
        return  studentRollNumber + " " + studentName + "   " + Arrays.toString(studentMarks) + " " +
studentGrade;
    }

    // @Override
    public String toFileString() {
        return studentRollNumber + ":" + studentName + ":" + studentMarks[0] + ":" + studentMarks[1]
+ ":" + studentMarks[2] + ":" + studentMarks[3] + ":" + studentMarks[4];
    }

    public double getGradePoint() {
        return studentGrade;
    }

    @Override
    public int compareTo(Student s) {
        return Double.compare(this.studentGrade, s.getGradePoint());
    }
}

// ROLL:NAME:S1:S2:S3:S4:S5
```

OUTPUT

```
> javac Main.java && java Main
Roll.  Name [S1, S2, S3, S4, S5] Grade
603231 DEV  [98, 96, 98, 99, 98] 9.78
423987 YTG  [76, 86, 88, 87, 98] 8.7
```

Result : The program for creating a student result database has been successfully executed.