public class Student {

        private int studentId;

        private String studentName;

        private int studentAge;

        private String gender;

        public Student(int studentId, String studentName, int studentAge, String gender) {

                this.studentId = studentId;

                this.studentName = studentName;

                this.studentAge = studentAge;

                this.gender=gender;

        }

        public String getGender() {

 return gender;

 }

 public void setGender(String gender) {

 this.gender = gender;

 }

 public int getStudentId() {

                return studentId;

        }

        public void setStudentId(int studentId) {

                this.studentId = studentId;

        }

        public String getStudentName() {

                return studentName;

        }

        public void setStudentName(String studentName) {

                this.studentName = studentName;

        }

        public int getStudentAge() {

                return studentAge;

        }

        public void setStudentAge(int studentAge) {

                this.studentAge = studentAge;

        }

}

public class Section {

        private Student arrayStu[];

        private int counter;

        public Section(int size)

        {

                arrayStu=new Student[size];

                counter=0;

        }

        public boolean addStudent(Student stu)

        {

                if(counter < arrayStu.length)

                        {

                        arrayStu[counter] = stu;

                        counter++;

                        return true;

                        }

                else

                        return false;

        }

        public int search(int id)

        {

                for(int i=0;i<counter;i++)

                        if(id==arrayStu[i].getStudentId())

                                return i;

                return -1;

        }

        boolean deleteStu(int id)

        {

                int index=search(id);

                if(index==-1)

                        return false;

                else

                {

                        arrayStu[index]=arrayStu[counter-1];

                        arrayStu[counter-1]=null;

                        counter--;

                        return true;

                }

        }

        public int findMaxAge()

        {

                int max =arrayStu[0].getStudentAge() ;

                for(int i=1; i<counter; i++)

                        if(arrayStu[i].getStudentAge()>max)

                                max = arrayStu[i].getStudentAge();

                return max;

        }

        public int findMinAge()

        {

                int min =arrayStu[0].getStudentAge() ;

                for(int i=1; i<counter; i++)

                        if(arrayStu[i].getStudentAge()<min)

                                min = arrayStu[i].getStudentAge();

                return min;

        }

        public Student findMinStu()

        {

        Student min=arrayStu[0];

        for(int i=1; i<counter; i++)

                if(arrayStu[i].getStudentAge()<min.getStudentAge())

                        min = arrayStu[i];

        return min;

        }

        public Student findMaxStu()

        {

        Student max=arrayStu[0];

        for(int i=1; i<counter; i++)

                if(arrayStu[i].getStudentAge()>max.getStudentAge())

                        max = arrayStu[i];

        return max;

        }

  public double calculateTotalAges()

        {

                double sum = 0.0;

                for(int i=0; i<counter; i++)

                        sum += arrayStu[i].getStudentAge();

                return sum;

        }

        public double calculateAvgAge()

        {

                if (counter==0)

                        return 0.0;

                else

                        return calculateTotalAges()/counter;

        }

        public void printStuData(int id)

        {

                int index=search(id);

                if(index==-1)

                        System.out.println("Student not Found");

                else

                {

                        System.out.println("Id= "+arrayStu[index].getStudentId());

                        System.out.println("Name= "+arrayStu[index].getStudentName());

                        System.out.println("Age= "+arrayStu[index].getStudentAge());

                }

        }

        public Student[] findStudents(int age)

        {

        Student temp[]=new Student[counter];

        int j=0;

        for(int i=0;i<counter;i++)

        if(arrayStu[i].getStudentAge()<age)

        {

        temp[j]=arrayStu[i];

        j++;

        }

        return temp;

        }

        public int getNStudents(String gender)

        {

        int count=0;

        for(int i=0;i<counter;i++)

        if(arrayStu[i].getGender().equalsIgnoreCase(gender))

        count++;

        return count;

        }

}

public class TestSection {

        /\*\*

         \* @param args

         \*/

        public static void main(String[] args) {

                // TODO Auto-generated method stub

                Section sec=new Section(10);

                Student st1=new Student(101,"Aaa",21,"Male");

                Student st2=new Student(102,"Bbb",19,"Male");

                Student st3=new Student(103,"Ccc",22,"Female");

                Student st4=new Student(104,"Ddd",24,"Female");

                Student st5=new Student(105,"Eee",18,"Male");

                System.out.println(sec.addStudent(st1));

                System.out.println(sec.addStudent(st2));

                System.out.println(sec.addStudent(st3));

                System.out.println(sec.addStudent(st4));

                System.out.println(sec.addStudent(st5));

                System.out.println(sec.getNStudents("Male"));

                System.out.println(sec.getNStudents("Female"));

               /\* System.out.println(sec.search(109));

                System.out.println(sec.findMaxAge());

                System.out.println(sec.findMinAge());

                sec.printStuData(109);

                Student st=sec.findMinStu();

                sec.printStuData(st.getStudentId());

                Student[] arStu=sec.findStudents(30);

                if(sec.deleteStu(109)==true)

                        System.out.println("student deleted");

                else

                        System.out.println("not found");\*/

        }

}