



HARBIN INSTITUTE OF TECHNOLOGY

哈尔滨工业大学

Java 程序设计实验报告

学号： XXXXXXXXXX

姓名： XXXXXX

专业： XXXXXXXXXX

班级： 1XXXXXX

实验六：文件读写程序设计与异常

一、实验目的

- 1) 了解流输入输出的基本概念。
- 2) 了解异常的概念。
- 3) 掌握 File、FileReader、FileWriter 等文件读写类库的使用。
- 3) 掌握 IOException、FileNotFoundException 等异常的使用。

二、实验内容

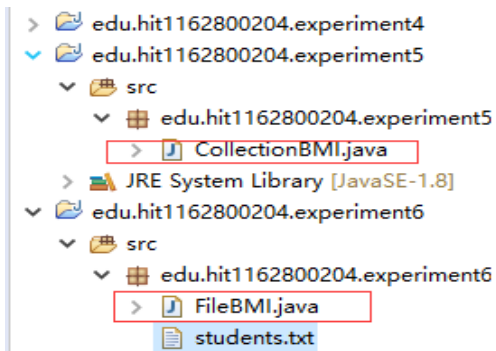
- 1) 将 CollectionBMI 类改造为 FileBMI 类；
- 2) 在 FileBMI 类中增加方法 saveFile(ArrayList<Student> students, String filename)，该方法可以将学生信息 students 写入到指定的文本文件中，每一行写入一个学生。
- 3) 在 FileBMI 类中增加方法 ArrayList<Student> readFile(String filename) 读文件中的数据到学生 ArrayList 中，并返回该 ArrayList。
- 4) 增加 findStudent、inputStudent、modifyStudent、deleteStudent 四个函数。findStudent 可根据学号查询学生信息并显示，然后提示用户是否增加、修改或删除学生。如果用户选择在增加，调用 inputStudent 函数增加学生；如果用户选择修改，调用 modifyStudent 函数修改学生信息，注意学号不能修改；如果用户选择删除学生，则调用 deleteStudent 函数删除该学生。注意，所有增删改操作完成后，都应重新保存到文件。
- 5) 增加 menu 函数，能够根据用户的选择，调用相关函数执行相应功能：a) 完成学生信息生成及文件保存；b) 学生查询、增加、删除及修改；c) 计算并显示统计结果。
- 6) 在 FileBMI 中的 main 函数中，调用 menu 函数，根据提示进行操作。

恭喜大家完成 1.0 版 BMI 统计分析系统!

三、实验步骤

- 1) 将 CollectionBMI 类改造为 FileBMI 类；

Step1. 改造 FileBMI 类如下：

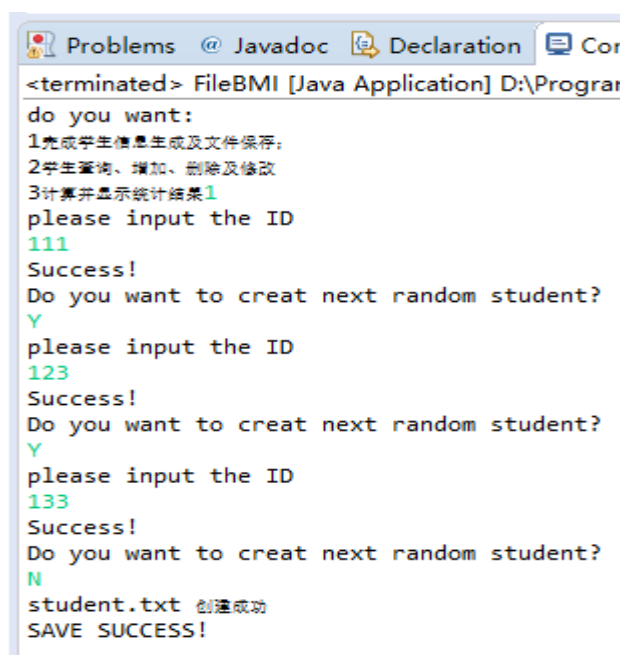


2) 在 FileBMI 类中增加方法 saveFile(ArrayList<Student> students, String filename), 该方法可以将学生信息 students 写入到指定的文本文件中, 每一行写入一个学生。

Step1.增加 saveFile 方法如下：

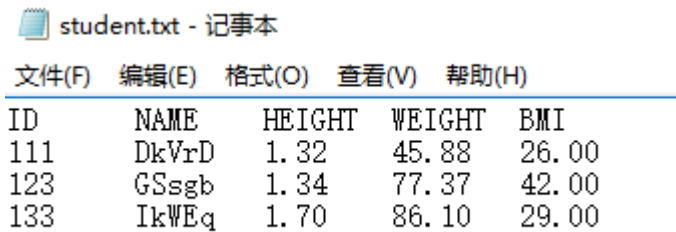
```
public static void saveFile(List<Student> students2, String filename){
    File file = new File("."+File.separator+filename);
    if(!file.exists()) {
        try {
            file.createNewFile();
            System.out.println(file.getName() + " 创建成功");
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    FileWriter writer;
    try {
        writer = new FileWriter(filename, false);
        writer.write(String.format("ID\tNAME\tHEIGHT\tWEIGHT\tBMI\n"));
        for(Student st: students2){
            writer.write(String.format("%s\t%s\t%.2f\t%.2f\t%.2f\n",
                st.number,st.name,st.height,st.weight,st.bmi));
        }
        writer.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
    System.out.println("SAVE SUCCESS!");
}
```

Step2.运行测试结果如下：



```
<terminated> FileBMI [Java Application] D:\Program
do you want:
1完成学生信息生成及文件保存;
2学生查询、增加、删除及修改
3计算并显示统计结果1
please input the ID
111
Success!
Do you want to creat next random student?
Y
please input the ID
123
Success!
Do you want to creat next random student?
Y
please input the ID
133
Success!
Do you want to creat next random student?
N
student.txt 创建成功
SAVE SUCCESS!
```

Step3.输出文件如下：



student.txt - 记事本

文件(F)	编辑(E)	格式(O)	查看(V)	帮助(H)
ID	NAME	HEIGHT	WEIGHT	BMI
111	DkVrD	1.32	45.88	26.00
123	GSsGb	1.34	77.37	42.00
133	IkWEq	1.70	86.10	29.00

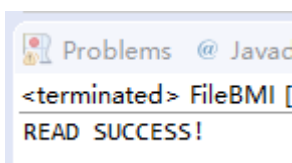
3) 在 FileBMI 类中增加方法 `ArrayList<Student> readFile(String filename)`

读文件中的数据到学生 ArrayList 中，并返回该 ArrayList。

Step1.增加造 `readFile` 方法如下：

```
public static ArrayList<Student> readFile(String filename){
    ArrayList<Student> buff=new ArrayList<Student>();
    File file = new File("."+File.separator+filename);
    if(file.exists()) {
        try {
            BufferedReader reader =new BufferedReader(new FileReader(file));
            String tempString = null;
            while ((tempString = reader.readLine()) != null) {
                String[] a= tempString.split(" ");
                Student st= new FileBMI().new Student(a[0],a[1],
                    Double.parseDouble(a[2]),Double.parseDouble(a[3]));
                buff.add(st);
            }
            reader.close();
            System.out.println("READ SUCCESS!");
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (NumberFormatException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
    } else {
        System.out.println("the file have noe been found!!");
    }
    return buff;
}
```

Step2.运行测试结果如下：



Problems @ Java

<terminated> FileBMI [

READ SUCCESS!

4) 增加 findStudent、inputStudent、modifyStudent、deleteStudent 四个函数。findStudent 可根据学号查询学生信息并显示，然后提示用户是否增加、修改或删除学生。如果用户选择在增加，调用 inputStudent 函数增加学生；如果用户选择修改，调用 modifyStudent 函数修改学生信息，注意学号不能修改；如果用户选择删除学生，则调用 deleteStudent 函数删除该学生。注意，所有增删改操作完成后，都应重新保存到文件。

Step1.增加造 findStudent 方法如下（本人认为 modifyStudent、deleteStudent 函数可以用 List 类中的方法更轻易实现，故无需赘添函数，inputStudent 本已存在）：

```
public static void findStudent(List<Student> stu,Scanner in){
    System.out.println("please input the id:");
    String id=in.next();

    for(int i=0;i<students.size();i++){

        if(id.compareTo(students.get(i).number)==0){
            System.out.printf("The ID has been found,do you want:\n"
                + "1 modify\n"
                + "2 delete\n");
            switch(in.nextInt()){
                case 1:students.set(i,inputStudent(in));break;
                case 2:students.remove(i);break;
                default:System.out.println("ERROR CHIOSE");break;
            }
            return;
        }
    }
    System.out.printf("The ID has not been found,do you want creat?input 1 to do it\n");
    if(in.nextInt()==1)stu.add(inputStudent(in));
}
```

（注，保存文件功能经考虑更适合放置于下一实验的 menu 函数中，以减少代码的调用次数，与后续功能统筹）

5) 增加 menu 函数，能够根据用户的选择，调用相关函数执行相应功能：

a)完成学生信息生成及文件保存；b)学生查询、增加、删除及修改；c)计算并显示统计结果。

Step1.增加 menu 函数如下：

```

public static void menu(Scanner in){
    System.out.printf("do you want:\n"
        + "1生成学生信息生成及文件保存: \n"
        + "2学生查询、增加、删除及修改\n"
        + "3计算并显示统计结果\n"
        + "4读取学生信息\n");
    switch(in.nextInt()){
        case 1: String str;
            do{
                System.out.println("please input the ID");
                genStudents(in.next());
                System.out.println("Do you want to creat next random student?");
                str=in.next();
            }while(str.compareTo("Y")==0||str.compareTo("y")==0);
            break;
        case 2: findStudent(students,in);
            break;
        case 3:
            double aver=AverofBMI(),mod=ModofBMI(),med=MedianofBMI(),var=VarianceofBMI();
            System.out.printf("\nsort by bmi\n");
            printStatics(aver,mod,med,var);
            break;
        default: readFile("student.txt");break;
    }saveFile(students, "student.txt");
}

```

(该函数功能将在下一步测试，在此不再赘述)

6) 在 FileBMI 中的 main 函数中，调用 menu 函数，根据提示进行操作。

Step1.在 main () 函数循环中调用如下：

```

public static void main(String[] agrv){
    new FileBMI();
    Scanner in=new Scanner(System.in);

    while(true)menu(in);
}

```

Step2.运行测试结果如下：

随机生成了 111,222,333 三位同学，保存文件

```

FileBMI [Java Application] D:\Program Files\Java\jre1
do you want:
1生成学生信息生成及文件保存:
2学生查询、增加、删除及修改
3计算并显示统计结果
4读取学生信息
1
please input the ID
222
Success!
Do you want to creat next random student?
y
please input the ID
111
Success!
Do you want to creat next random student?
y
please input the ID
333

```

修改 111 为 444 保存文件

删除 222 保存文件

```

Problems @ Javadoc Declaration Console
FileBMI [Java Application] D:\Program Files\Java\jre1.8.0_1
student.txt 创建成功
SAVE SUCCESS!
do you want:
1完成学生信息生成及文件保存;
2学生查询、增加、删除及修改
3计算并显示统计结果
4读取学生信息
2
please input the id:
111
The ID has been found,do you want:
1 modify
2 delete
1
please input the ID,name,height and weight:
444 asda 1.7 70
SAVE SUCCESS!
do you want:
1完成学生信息生成及文件保存;
2学生查询、增加、删除及修改
3计算并显示统计结果
4读取学生信息
2
please input the id:
222
The ID has been found,do you want:
1 modify
2 delete
2
SAVE SUCCESS!

```

打印结果成绩单

```

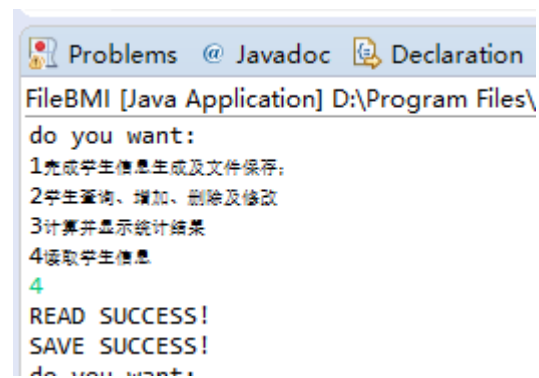
do you want:
1完成学生信息生成及文件保存;
2学生查询、增加、删除及修改
3计算并显示统计结果
4读取学生信息
3
|
sort by bmi
Average is:33.50      Mode is:23.99      Median is:67.00 Variance is:90.25
ID      NAME      Height      Weight      BMI
444      asda      1.70      70.00      24.00
333      TjfoQ      1.03      46.89      43.00
SAVE SUCCESS!
do you want:
1完成学生信息生成及文件保存;
2学生查询、增加、删除及修改
3计算并显示统计结果
4读取学生信息

```

文件如下

ID	NAME	HEIGHT	WEIGHT	BMI
444	asda	1.70	70.00	24.00
333	TjfoQ	1.03	46.89	43.00

下次读取结果如下



```
FileBMI [Java Application] D:\Program Files\
do you want:
1完成学生信息生成及文件保存:
2学生查询、增加、删除及修改
3计算并显示统计结果
4读取学生信息
4
READ SUCCESS!
SAVE SUCCESS!
do you want:
```