



HARBIN INSTITUTE OF TECHNOLOGY

哈尔滨工业大学

Java 程序设计实验报告

学号: 1xxxxxxxxx

姓名: xxxxxxxxxx

专业: xxxxxxxxxxxx

班级: xxxxxxxxxxxx

实验四：面向对象程序设计

一、实验目的

- 1) 掌握面向对象的基本概念（成员变量、成员函数等）
- 2) 掌握类的定义、内部类的定义
- 3) 掌握对象的声明
- 4) 掌握对象数组的使用
- 5) 基本算法的设计

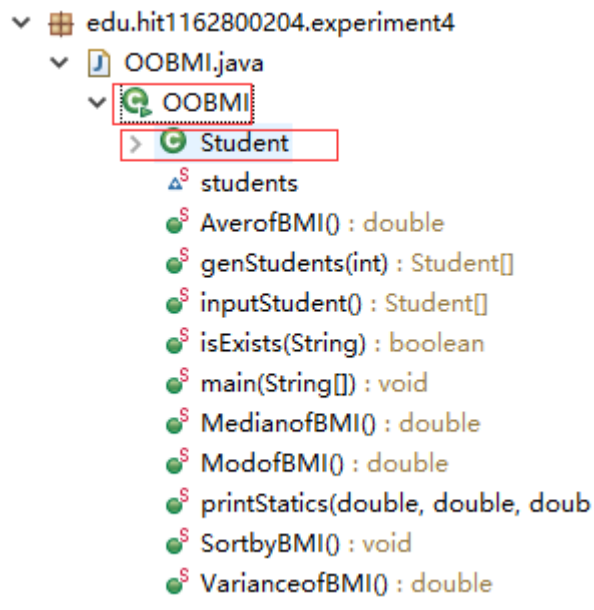
二、实验内容

- 1) 编写 OOBMI 类；并在其中定义内部类 Student，该类包含学号、姓名、身高、体重和 bmi 等属性。
- 2) 在 OOBMI 类中增加成员函数 inputStudent，从键盘输入学生信息，函数返回值为 Student 类型。
- 3) 在 OOBMI 中增加成员属性 Student[] students；编写 inputStudents 函数，提示学生人数，然后循环调用 inputStudent 函数输入多个学生，并保存到 students 数组中。
- 4) 在 OOBMI 类中增加 isExists(String id) 函数，判断该学生是否已经输入（是否在 students 数组中），函数返回值为 boolean 类型，如果已经存在，返回 false；否则，返回 true。
- 5) 在 OOBMI 中增加 genStudents 函数，参数为整数，能够随机生成指定数量的名学生对象，并保存到 students 数组中。
- 6) 修改 OOBMI 中的 genStudents 和 inputStudent 函数，调用 isExists 函数避免输入或生成重复的学号的学生。
- 7) 在 OOBMI 中增加 4 个成员函数，分别统计 bmi 的均值、中值、众数、方差等统计信息。
- 8) 在 OOBMI 中增加 printStatics 函数，该函数可以打印所有学生基本信息，以及统计结果信息。打印时，每个学生的信息打印为一行，为了清晰，学号、姓名、身高、体重和计算后的 bmi 值之间用制表符(\t)隔开；打印完学生信息后，打印 BMI 统计信息。思考：如何将统计信息传递给 printStatics 函数？
- 9) 在 OOBMI 的 main 函数中，调用上述函数，完成输入、统计结果输出的完整过程。

三、实验步骤

- 1) 编写 OOBMI 类；并在其中定义内部类 Student，该类包含学号、姓名、身高、体重和 bmi 等属性。

Step1.定义该类在 OOBMI 类下



Step2.编写代码如下：

```
public class Student{
    String number="",name="";
    double height=0.0,weight=0.0,bmi=0.0;

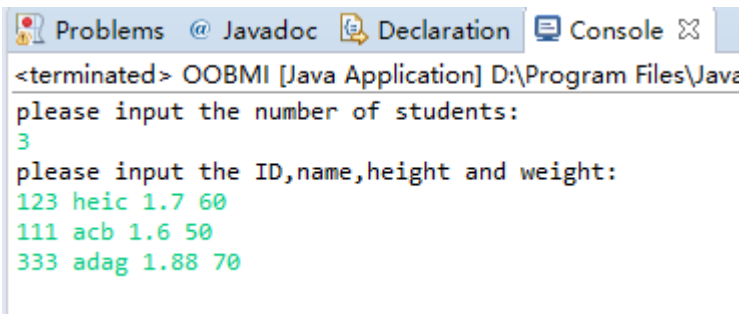
    public Student(){
    public Student(String number,String name,double height,double weight){
        this.number=number;
        this.name=name;
        this.height=(int)(height*100)/100.0;
        this.weight=(int)(weight*100)/100.0;
        this.bmi=(int)(weight/(height*height))*100/100.0;
    }
}
```

2) 在 OOBMI 类中增加成员函数 inputStudent，从键盘输入学生信息，函数返回值为 Student 类型。

Step1.定义函数 inputStudent () 如下

```
public static Student inputStudent(Scanner in){
    String number=in.next();
    String name=in.next();
    double height=in.nextDouble();
    double weight=in.nextDouble();
    Student student=new OOBMI().new Student(number,name,height,weight);
    return student;
}
```

Step2.运行结果如下



```
<terminated> OOBMI [Java Application] D:\Program Files\Java;
please input the number of students:
3
please input the ID,name,height and weight:
123 heic 1.7 60
111 acb 1.6 50
333 adag 1.88 70
```

3) 在 OOBMI 中增加成员属性 Student[] students ; 编写 inputStudents 函数, 提示学生人数, 然后循环调用 inputStudent 函数输入多个学生, 并保存到 students 数组中。

Step1.定义成员变量 Student[] students 如下, 并在 main()中调用 inputStudent () ;

```
public class OOBMI {
    static Student[] students;
    public OOBMI(){ }
    public OOBMI(int n){
        students=new Student[n];
        for(int i=0;i<students.length;i++)students[i]=new OOBMI().new Student();
    }

    public static void main(String[] agrv){
        Scanner in=new Scanner(System.in);
        System.out.println("please input the number of students:");

        int n=in.nextInt();
        new OOBMI(n);

        System.out.println("please input the ID,name,height and weight:");
        for(int i=0;i<n;i++)students[i]=inputStudent(in);

        //genStudents(n);
        in.close();
    }
}
```

Step2.运行结果如实验 3) 所示, 在此不再赘述。

4) 在 OOBMI 类中增加 isExists(String id) 函数, 判断该学生是否已经输入 (是否在 students 数组中), 函数返回值为 boolean 类型, 如果已经存在, 返回 false ; 否则, 返回 true。

Step1. 增加 isExists(String id) 函数如下

```
public static boolean isExists(String id){
    for(int i=0;i<students.length;i++){
        if(id.equals(students[i].number))return true;
    }
    return false;
}
```

5) 在 OOBMI 中增加 genStudents 函数，参数为整数，能够随机生成指定数量的名学生对象，并保存到 students 数组中。

Step1. 增加 genStudents 函数如下

```
public static void genStudents(int n){
    Random ron=new Random();
    char Char[]={
        'A','B','C','D','E','F','G',
        'H','I','J','K','L','M','N',
        'O','P','Q','R','S','T',
        'U','V','W','X','Y','Z',
        'a','b','c','d','e','f','g',
        'h','i','j','k','l','m','n',
        'o','p','q','r','s','t',
        'u','v','w','x','y','z'};

    for(int i=0;i<n;i++){
        String number=String.valueOf(100+ron.nextInt()%100);
        if(isExists(number))continue;
        String name="";
        for(int j=0;j<5;j++){name+=Char[ron.nextInt(46)];}
        double height=ron.nextDouble()+1;
        double weight=ron.nextDouble()*60+40;
        students[i]=new OOBMI().new Student(number,name,height,weight);
    }
    System.out.println("Success!");
}
```

Step2. 运行结果如下

```
<terminated> OOBMI [Java Application] D:\Program Files\Java\jre1.
please input the number of students:
3
Success!
```

6) 修改 OOBMI 中的 genStudents 和 inputStudent 函数，调用 isExists 函数避免输入或生成重复的学号的学生。

Step1. 在实验 5) 中已完成，不再赘述 !!

7) 在 OOBMI 中增加 4 个成员函数，分别统计 bmi 的均值、中值、众数、方差等统计信息。

Step1. 增加各个函数如下（另加了排序函数用于查中位数）

```
public static double AverofBMI(){
    double sum=0;
    for(int i=0;i<students.length;i++)sum+=students[i].bmi;

    return sum/students.length;
}

public static double ModofBMI(){
    int hash[]=new int[10000];
    for(int i=0;i<students.length;i++)hash[(int)students[i].bmi*100-1]++;
    int max=0;
    for(int i=1;i<hash.length;i++){
        if(hash[i]>hash[max])max=i;
    }
    return max/100.0;
}

public static double MedianofBMI(){
    SortbyBMI();
    if(students.length%2==0)
        return students[students.length/2].bmi+students[students.length/2-1].bmi;

    return students[students.length/2].bmi;
}

public static double VarianceofBMI(){
    double sum=0;
    double aver=AverofBMI();
    for(int i=0;i<students.length;i++)
        sum+=(students[i].bmi-aver)*(students[i].bmi-aver);

    return sum/students.length;
}
```

```

public static void SortbyBMI(){
    for(int i=0;i<students.length;i++){
        int min=i;
        for(int j=i+1;j<students.length;j++){
            if(students[i].bmi<students[min].bmi)min=j;
        }
        if(min!=i){
            Student temp=students[i];
            students[i]=students[min];
            students[min]=temp;
        }
    }
}

```

8) 在 OOBMI 中增加 printStatics 函数, 该函数可以打印所有学生基本信息, 以及统计结果信息。打印时, 每个学生的信息打印为一行, 为了清晰, 学号、姓名、身高、体重和计算后的 bmi 值之间用制表符(\t)隔开; 打印完学生信息后, 打印 BMI 统计信息。思考: 如何将统计信息传递给 printStatics 函数?

Step1. 增加 printStatics 函数如下

```

public static void printStatics(double aver,double mod,double median,double variance){
    System.out.printf("Average is:%.2f\tMode is:%.2f\tMedian is:%.2f\tVariance is:%.2f\n",
        ,aver,mod,median,variance);
    System.out.printf("ID\tNAME\tHeight\tWeight\tBMI\n");
    for(int i=0;i<students.length;i++){
        System.out.printf("%s\t%s\t%.2f\t%.2f\t%.2f\n",
            students[i].number,students[i].name,
            students[i].height,students[i].weight,students[i].bmi);
    }
}

```

Step2. 运行结果如下

```

<terminated> OOBMI [Java Application] D:\Program Files\Java\jre1.8.0_121\bin\javaw.exe (2017年5月24
please input the number of students:
5
Success!
Average is:27.40      Mode is:12.99      Median is:38.00 Variance is:214.64
ID    NAME    Height  Weight  BMI
30    PQKWI    1.72    51.97    17.00
46    hf1ST    1.95    51.18    13.00
124   nmLAq    1.20    56.16    38.00
11    hVaGo    1.25    81.44    51.00
32    CCTMp    1.58    46.26    18.00

```

Step3. 思考：传参给 printStatics 函数即可

9) 在 OOBMI 的 main 函数中，调用上述函数，完成输入、统计结果输出的完整过程。

Step1. 修改 main () 函数如下，其中注释掉部分是手动输入学生数据的部分，该部分与 genStudents 函数自动生成二选一即可

```
public static void main(String[] agrv){
    Scanner in=new Scanner(System.in);
    System.out.println("please input the number of students:");

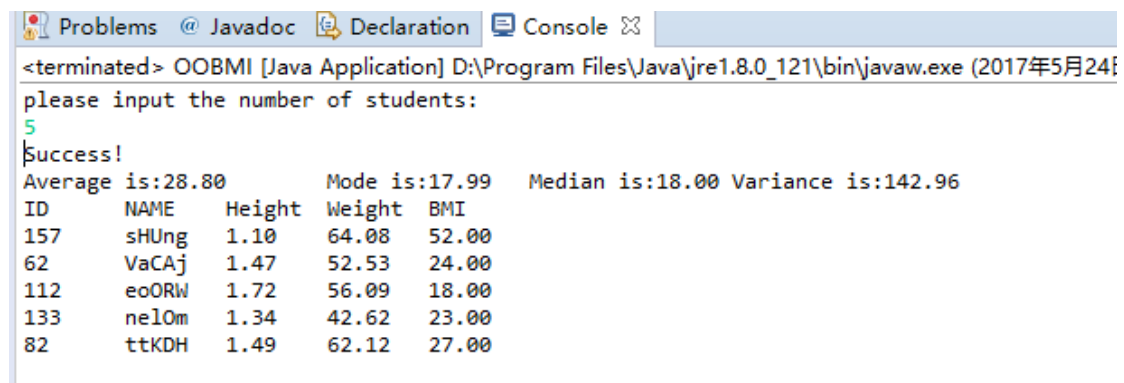
    int n=in.nextInt();
    new OOBMI(n);

    //System.out.println("please input the ID,name,height and weight:");
    //for(int i=0;i<n;i++)students[i]=inputStudent(in);

    genStudents(n);
    printStatics(AverofBMI(),ModofBMI(),MedianofBMI(),VarianceofBMI());

    in.close();
}
```

Step2. 运行结果如下（以自动生成的数据为例）：



```
<terminated> OOBMI [Java Application] D:\Program Files\Java\jre1.8.0_121\bin\javaw.exe (2017年5月24日)
please input the number of students:
5
Success!
Average is:28.80      Mode is:17.99      Median is:18.00 Variance is:142.96
ID    NAME    Height  Weight  BMI
157   sHUnG    1.10    64.08   52.00
62    VaCAj    1.47    52.53   24.00
112   eoORW    1.72    56.09   18.00
133   neLOm    1.34    42.62   23.00
82    ttKDH    1.49    62.12   27.00
```