实验一：Java 基本程序设计

一、实验目的

1）掌握标准输入输出函数的使用。
2）静态函数的定义和使用（本实验要求所有函数均为静态函数）；
2）掌握简单排序算法；
3）Java 基础语法综合运行（非面向对象版本 BMI 程序）；

二、实验内容

1)编写 BMI 类，在 main 函数中增加数组 String[] ids，String[] names，float[] heights, float[] weights, float[] bmis，分别存储学生们的学号、姓名、身高、体重、计算后的 bmi 值和胖瘦健康状况。注意，上述数值均需保留两位小数存储。

2）定义 inputStudents 函数，该函数的参数为上述数组，该函数的功能是输入多个学生的相关信息，并将相关数据存储到上述数组中；

3）在 BMI 类中，增加一个函数 checkHealth，函数参数为 bmi 值，该函数按下表中 BMI 取值范围判断胖瘦健康状况，该函数的返回值为字符串，返回结果即下表中的第一列中的值，并在 inputStudents 函数中调用该函数，获得学生的胖瘦健康状况。

Category	BMI (kg/m^2)	
	from	to
Underweight		18.5
Normal Range	18.5	23
Overweight—At Risk	23	25
Overweight—Moderately Obese	25	30
Overweight—Severely Obese	30	

4）在 BMI 类中，增加 5 个排序 sortByXXX 函数，XXX 表示排序属性，可以分别按照学生学号、姓名、身高、体重、BMI 进行由小到大排序，排序算法可以利用简单排序、选择排序、冒泡排序算法或其他算法（选择其中一种算法实现即可）。排序前后必须保证同一个学生在所有数组中对应相同的下标！**为了方便实现上述功能,可定义一个排序数组 int sortedIndex[]，该数组中保存了进行排序的数组排序后的下标，排序结束后，返回该数组，以便根据该数据进行打印显示。**

5）在 BMI 类中，增加 printStudents 函数，该函数的参数含有 int sortedIndex[],该函数可以打印排序前和排序后的结果。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。

6）定义 menu 函数，提供输入学生、打印学生，5 种排序、程序退出等 8 种选项，用户输入指定选项后，运行相应函数功能。**注意，在调用 inputStudents 函数前，需先提示用户数输入指定人数。**

7）在 BMI 类的 main 函数中，调用 menu 函数，测试运行各项功能。

注意，身高、体重、及 bmi 等数值均需保留两位小数的格式进行存储和显示。

三、实验代码

注意：将程序代码和运行结果截图粘贴在此处，注意源代码中注释行数不少于全部代码的 1/3，程序源代码请压缩后上传，压缩文件按照 学号.zip 进行命名，注意源程序于报告请分别上传到不同的文件夹中！

```
package edu.java.exp1;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.Comparator;

import java.util.Scanner;

public class l170300901  {

    static Scanner in = new Scanner(System.in);

    // 기존의 ids, names, heights, weights, bmis 배열 대신 Student 데이터 클래스 이용

    public static final class Student {

    private String id;
    private String name;
    private float height;
    private float weight;
    private float bmi;
    public Student() {}

        // 이클립스에서는 Ctrl + Space를 이용해 getter와 setter 메서드를 자동 완성할 수 있습니다.

      // 또는 Source -> Generate Getters and Setters...를 이용할 수도 있습니다.

    public float getBmi() {

        return bmi;

      }
```

```
public String getId() {
    return id;
}
public void setId(String id) {
    this.id = id;
}
public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}
public void setHeight(float height) {
    this.height = height;
}
 public void setBmi(float bmi) {
    this.bmi = bmi;
}
public float getHeight() {
    return height;
}
public float getWeight() {
    return weight;
```

```
    }

    public void setWeight(float weight) {

      this.weight = weight;

    }

  }

// 기존의 StudentsSum과 sortedIndex 대신 ArrayList 컨테이너에 일괄 저장

private static final ArrayList<Student> students = new ArrayList<>();

// 배열로도 가능

private static final Student[] students2 = new Student[3000];

public static void main(String[] args)

{

    System.out.println("Welcome To The Students' Healthy Information System!₩n");

   menu();

    in.close();

}

public static void menu()

{

    // 불필요한 input 지역변수 제거

    while(true)

    {

        System.out.println("1. Input students' information");

        System.out.println("2. Print students' information");
```

```
            System.out.println("3. Sort the students by IDs");

            System.out.println("4. Sort the students by Names");

            System.out.println("5. Sort the students by Heights");

            System.out.println("6. Sort the students by Weights");

            System.out.println("7. Sort the students by BMIs");

            System.out.println("8. Exit the students' healthy information
system₩n");

            System.out.print("Please input the number you want to do: ");

            switch(Integer.parseInt(in.nextLine()))

            {

                case 1: inputStudents(); break;

                case 2: printStudents(); break;

                case 3: sortByIDs(); break; // start, end가 전체 범위로만 이용하는
것 같아 그냥 뺐습니다.

                case 4: sortByNames(); break;

                case 5: sortByHeights(); break;

                case 6: sortByWeights(); break;

                case 7: sortByBMI(); break;

                case 8: System.out.println("Goodbye! Thank you for using.");
return;

                default: System.out.print("You input the wrong number. Please
input again."); break;

            }

            System.out.println();
```

```
        }
    }
    public static void inputStudents()
    {
        System.out.print("Please input the numbers of the students: ");
        int from = students.size() + 1;
        int to = from + Integer.parseInt(in.nextLine());
        System.out.println();
        for(int i = from; i < to; i++)
        {
            Student student = new Student();
            System.out.print("Please input the ID of the No." + i + " student: ");
            student.setId(in.nextLine());
            System.out.print("₩nPlease input the name of the No." + i + " student:
");
            student.setName(in.nextLine());
            System.out.print("₩nPlease input the height of the No." + i + " student:
");
            student.setHeight(Float.parseFloat(in.nextLine()));
            System.out.print("₩nPlease input the weight of the No." + i + " student:
");
            student.setWeight(Float.parseFloat(in.nextLine()));
            System.out.println();
```

```
            student.setBmi(calcBMI(student.getWeight(), student.getHeight()));
            checkHealth(student.getBmi());
            // 배열에 추가
           students2[i-1] = student;
            // 리스트에 추가
            students.add(student);
        }
    }
    public static float calcBMI(float weight, float height) {
        return weight / (height * height);
    }
    public static void checkHealth(float bmis)
    {
        if(bmis <= 18.5)
        System.out.println("Underweight");
        else if(bmis <= 23)
        System.out.println("Normal Range");
        else if(bmis <= 25)
        System.out.println("Overweight--At Risk");
        else if(bmis <= 30)
        System.out.println("Overweight--Moderately Obese");
        else
```

```
            System.out.println("Overweight--Severely Obese");

        }

        public static void printStudents()

       {

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

            System.out.printf(students.get(i).getId() + "₩t" +
students.get(i).getName() + "₩t%.2f₩t%.2f₩t%.2f₩n",
students.get(i).getHeight(), students.get(i).getWeight(),
students.get(i).getBmi());

            // 배열은 아래처럼

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

            // System.out.printf(students2[i].getId() + "₩t" + students2[i].getName()
+ "₩t%.2f₩t%.2f₩t%.2f₩n", students2[i].getHeight(), students2[i].getWeight(),
students2[i].getBmi());

        }

        public static void sortByIDs()

        {

            students.sort(Comparator.comparing(Student::getId));

            // 반대 방향 정렬하려면 아래처럼

            // students.sort(Comparator.comparing(Student::getId).reversed());

            Arrays.sort(students2, 0, students.size(),
Comparator.comparing(Student::getId));

        }

        public static void sortByNames()

        {
```

```
        students.sort(Comparator.comparing(Student::getName));

        Arrays.sort(students2, 0, students.size(),
Comparator.comparing(Student::getName));

    }

    public static void sortByHeights()

    {

        students.sort(Comparator.comparing(Student::getHeight));

        Arrays.sort(students2, 0, students.size(),
Comparator.comparing(Student::getHeight));

    }

    public static void sortByWeights()

    {

        students.sort(Comparator.comparing(Student::getWeight));

        Arrays.sort(students2, 0, students.size(),
Comparator.comparing(Student::getWeight));

    }

    public static void sortByBMI()

    {

        students.sort(Comparator.comparing(Student::getBmi));

        Arrays.sort(students2, 0, students.size(),
Comparator.comparing(Student::getBmi));

    }

 }
Welcome To The Students' Healthy Information System!

1. Input students' information
```

```
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 1
Please input the numbers of the students:  3

Please input the ID of the No.1 student: 11704231231

Please input the name of the No.1 student: DDD

Please input the height of the No.1 student: 176

Please input the weight of the No.1 student: 54

Underweight
Please input the ID of the No.2 student: 11702312344

Please input the name of the No.2 student: WWW

Please input the height of the No.2 student: 175

Please input the weight of the No.2 student: 77

Underweight
Please input the ID of the No.3 student: 11704231323

Please input the name of the No.3 student: KKK

Please input the height of the No.3 student: 156

Please input the weight of the No.3 student: 45

Underweight

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
```

```
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 2
11704231231 DDD 176.00  54.00   0.00
11702312344 WWW 175.00  77.00   0.00
11704231323 KKK 156.00  45.00   0.00

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 3

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 2
11702312344 WWW 175.00  77.00   0.00
11704231231 DDD 176.00  54.00   0.00
11704231323 KKK 156.00  45.00   0.00

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 4
```

```
1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 2
11704231231 DDD 176.00  54.00   0.00
11704231323 KKK 156.00  45.00   0.00
11702312344 WWW 175.00  77.00   0.00

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 5

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 2
11704231323 KKK 156.00  45.00   0.00
11702312344 WWW 175.00  77.00   0.00
11704231231 DDD 176.00  54.00   0.00

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
```

```
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information s ystem

Please input the number you want to do: 6

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 2
11704231323 KKK 156.00  45.00   0.00
11704231231 DDD 176.00  54.00   0.00
11702312344 WWW 175.00  77.00   0.00

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 7

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 2
11704231231 DDD 176.00  54.00   0.00
11704231323 KKK 156.00  45.00   0.00
```

```
11702312344 WWW 175.00  77.00   0.00

1. Input students' information
2. Print students' information
3. Sort the students by IDs
4. Sort the students by Names
5. Sort the students by Heights
6. Sort the students by Weights
7. Sort the students by BMIs
8. Exit the students' healthy information system

Please input the number you want to do: 8
Goodbye! Thank you for using
```