

## 1. 介绍下 **static final extends implements abstract** 作用

**static** 是一个修饰符 表示静态的 可以修饰属性 方法 代码块

修饰的属性表示静态属性 表示整个类型共享一份的属性 不是每个对象都有一份的属性

修饰的方法表示静态方法 需要拿着类名去调用 静态方法里面只能直接的访问静态的成员 如果想要访问非静态的成员 需要先创建对象 拿着对象去调用

修饰的代码块表示静态代码块 当类第一次被加载的时候执行 而且只执行一次

**final** 是一个修饰符 表示最终的 可以修饰类 方法 变量

修饰的类叫做最终类 不能有子类 但是可以有父类

修饰的方法叫做静态方法 不能被覆盖 但是可以正常的被继承

修饰的变量表示最终变量 一旦赋值不能再改值

**extends** 表示继承 用该单词实现两个类之间的 is a 的关系 继承是最简单的代码共享方式之一

**implements** 表示实现 用该单词实现一个类和一个接口之间的 is a 的关系 当我们拿着一个类实现一个接口的时候 需要给出接口里面所有抽象方法的具体实现

**abstract** 表示一个修饰符 表示抽象的 可以修饰类和方法

修饰的类叫做抽象类 表示该类型不形象 不具体 不能创建对象

修饰的方法叫做抽象方法 表示该类型一定会这个方法 但是现在给不出具体的实现 需要待留给子类去实现

## 2. 冒泡排序一个数组

```
int[] data = new int[] {45, 66, 82, 10, 22, 50};
for(int x = 0; x < data.length - 1; x++) {
    for(int y = 0; y < data.length - 1 - x; y++) {
        if(data[y] > data[y + 1]) {
            data[y] = data[y] ^ data[y + 1];
            data[y + 1] = data[y] ^ data[y + 1];
            data[y] = data[y] ^ data[y + 1];
        }
    }
}
```

## 3. **int arr[]={12,55,23,36,35,25,4}**求平均值

```
double sum = 0;
for(int score : srr) {
    sum += score;
}
double avg = sum / data.length;
```

## 4. 你知道的数据类型及内存大小

java 中的数据类型分为两大类或者无数类

两大类分为：基本数据类型 和 引用数据类型

基本数据类型分为：

布尔类型：boolean    字符类型：char(16 位)

整数类型：byte (8)    short (16)    int (32)    long (64)

浮点类型：float (32)    double (64)

5. 有一个二维数组，里面都是数字，就像 15 这个数是行中最小并且同时列中最大，用代码实现查找这样的数

```
int[][] data = new int[][]{{12,14,13,22,11},{22,16,5,7,9},{12,7,9,33,28},
{17,16,15,23,36},{1,2,6,9,10}};
int[] max = new int[5];
int[] min = new int[5];
for(int x = 0;x < data.length;x++){
    int min1 = 100;
    int max1 = 0;
    for(int y = 0;y < data[x].length;y++){
        if(data[x][y] < min1){
            min1 = data[x][y];
        }
        if(data[y][x] > max1){
            max1 = data[y][x];
        }
    }
    max[x] = max1;
    min[x] = min1;
}
for(int x = 0;x < max.length;x++){
    for(int y = 0;y < min.length;y++){
        if(max[x] == min[y]){
            System.out.println(max[x]);
        }
    }
}
```