  for(vari = 0; i < len - 1; i++) {

        for(varj = 0; j < len - 1 - i; j++) {

            if(arr[j] > arr[j+1]) {

                Var temp = arr[j+1];

                arr[j+1] = arr[j];

                arr[j] = temp;

            }

        }

    }

returnarr;

 for(vari = 0; i < len - 1; i++) {

        minIndex = i;

        for(varj = i + 1; j < len; j++) {

            if(arr[j] < arr[minIndex]) {

                minIndex = j;

            }

        }

        temp = arr[i];

        arr[i] = arr[minIndex];

        arr[minIndex] = temp;

    }

returnarr;

上述两段代码分别是选择排序和冒泡排序算法

1. 请你分析两段代码，说明哪一段是选择排序哪一段是冒泡排序
2. 请分别分析两个排序算法的时间复杂度
3. 现有一段数列【5,6,12,3,56,78,35,12,8】请你分别写出第一次排序后的数列排列情况

public （1） ICalculate {

final float PI = 3.1415f;

float getArea(float r);

float getCircumference(float r);   
}

public class Calculate implements ICalculate {

@Override

public float getArea(float r) {

float area = PI\*r\*r;

return area;

}

@Override

public float getCircumference(float r) {

float circumference = 2\*PI\*r;

return circumference;

}

}

public class Bird {

String color = "灰色"; // 颜色

String skin = "羽毛"; // 皮毛

}

public class Pigeon （2）Bird {

public static void main(String[] args) {

Pigeon pigeon = new Pigeon();

System.out.println(pigeon.color);

}

}

public （3） Animal {

　　public Animal() {}

　　public void voice() {

System.out.println("make some voice..");

}

}

public class Dog extends Animal{

public Dog() {}

@Override

public void voice(){

System.out.println("woof...");

}

}

public class Cat （4）Animal {

public Cat() {}

@Override

public void voice(){

System.out.println("nya...");

}

}

public class Zoo {

public static void main(String[] args) {

（5）

dog.voice();

Cat cat = new Cat();

cat.voice();

Fish fish = new Fish();

（6）

}

}

1. 请补充完成代码
2. 请指出代码中的类和接口，并且写出属性和方法