

Java程序设计实验报告

实验报告

【实验题目】 Java实验3

#### 【实验内容】

##### 1. 练习1

###### **实验目的** 熟练掌握函数重载的方法

**实验要求**编写一个 java 程序来计算两个整数及两个浮点数的加法运算。如果未提供参数，请提供默 认值以计算总和。使用函数重载。测试相关的用例并输出结果。

###### 代码：

import java.util.Scanner;

public class Summation {

    public static void Sum(int a, int b) { // 原本的加法函数

        int sum = a + b;// 计算

        System.out.println(a + "+" + b + "=" + sum);// 输出

    }

    private static void Sum(double a, double b) { // 重载函数

        double sum = a + b;// 计算

        System.out.print(a + "+" + b + "=");// 输出

        System.out.println(String.format("%.3f", sum));// 只输出三位小数

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int a = 0, b = 0;// 初始值

        double c = 0.0, d = 0.0;// 初始值

        System.out.println("是否提供参数：（Y/N）");// 是否输入参数判断

        char choice = sc.next().charAt(0); // 输入

        if (choice == 'Y' || choice == 'y') {// 要输入参数

            System.out.println("整数加法：");

            a = sc.nextInt(); // 输入语句

            b = sc.nextInt();// 输入语句

            Sum(a, b);// 调用函数求和

            System.out.println("浮点数加法：");

            c = sc.nextDouble();// 输入语句

            d = sc.nextDouble();// 输入语句

            Sum(c, d);// 调用函数求和

        } else {// 默认参数计算

            System.out.println("整数加法：");

            Sum(a, b);

            System.out.println("浮点数加法：");

            Sum(c, d);

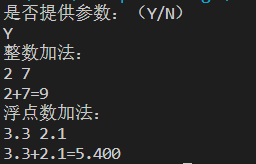
        }

    }

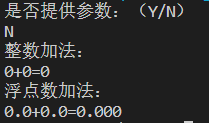
}

###### 输出结果：

##### 提供参数



1. 不提供参数



###### 分析：

先写一个方法用来计算整数的加法，之后用同样的方法名字来重写这个方法用于计算浮点数，在主程序中直接调用此方法根据输入的数据类型不同，会自动选择合适的方法进行运算并输出结果

##### 2.练习2

**实验目的**了解类中变量的私有公有，和类中的成员函数的调用

**实验要求**(1)定义圆类 Circle 该类包括以下内容：表示圆心坐标和半径的私有数据 x,y 和 r，为数据成员赋值的静态方法 setCircle()，输出圆的静态方法showCircle()，计算并输出圆的周长和面积的方法 perimeterAera()

(2)编写 main 并运行程序。要求测试相关的方法，并输出测试结果。

###### 代码：

**Circle.java**

public class Circle {

    private static double x = 0;// 初始值

    private static double y = 0;// 初始值

    private static double r = 1;// 初始值

    public static void setCircle(double a, double b, double c) {// 设置Circle的参数

        x = a;

        y = b;

        r = c;

    }

    public static void showCircle() {// 打印

        System.out.println("圆心位置：(" + x + "," + y + ")");// 圆的圆心位置

        System.out.println("圆的半径：" + r);// 圆的半径

    }

    public static void perimeterAera() {// 计算面积和周长

        double d = 2 \* Math.PI \* r;// 周长

        double s = Math.PI \* r \* r;// 面积

        System.out.println(String.format("圆的周长：%.2f", d));// 打印周长

        System.out.println(String.format("圆的面积：%.2f", s));// 打印面积

    }

}

**Test1.java**

import java.util.Scanner;

public class Test1 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("是否提供参数：（Y/N）");// 是否输入参数判断

        char choice = sc.next().charAt(0); // 输入

        if (choice == 'Y' || choice == 'y') {// 要输入参数

            System.out.println("请输入圆心位置:");

            double x = sc.nextDouble();// 输入x坐标

            double y = sc.nextDouble();// 输入y坐标

            System.out.println("请输入半径：");

            double r = sc.nextDouble();// 输入半径r

            Circle.setCircle(x, y, r);// 设置圆的参数

        }

        Circle.showCircle();// 打印圆的信息

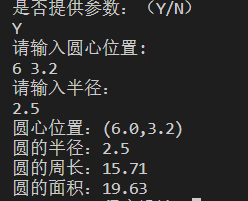
        Circle.perimeterAera();// 计算周长和面积

    }

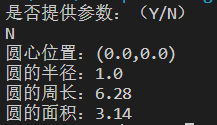
}

###### 输出结果：

1.提供参数



2.不提供参数



###### 分析：

Circle类中先设置私有变量x，y，r，并初始化这三个变量，再写一个成员方法设置这三个变量，之后写一个成员方法来打印圆的相关信息，包括圆心位置和半径，在写一个成员方法用来计算圆的周长和面积。在测试类中让用户决定是否自行输入参数，如果输入则就调用成员方法来设置新的参数，否则直接调用打印方法，直接打印默认值即可。

##### 练习3

###### 实验目的 熟练掌握类的继承，子类调用父类方法的使用

**实验要求**设计一个类 A，该类有一个方法 f(a, b)，返回 a,b 的最大公约数。设计一个子类 B，重写 祖先的 f 方法，且返回 a，b 的最小公倍数。要求在子类重写父类方法时，首先调用父类的 方法 f 获得最大公约数 m，然后再用公(a\*b)/m 获得最小公倍数。最后写一个测试程序，分别调用父类和子类的方法。测试 6 和 8 的最大公约数以 及 10 和 18 的最小公倍数

###### 代码：

**A.java**

public class A {

    int x = 1, y = 1; // 初始化

    public int f(int a, int b) {

        x = a;// 赋值

        y = b;

        int t;

        t = a > b ? a : b;// 判断哪个数字大

        while (t > 0) { // 从最大数开始递减就可以知道最大公约数

            if (x % t == 0 && y % t == 0) // 两个数取余都为0就是公约数

                break;

            --t; // 否则就算下一个小1的数字

        }

        return t;

    }

}

**B.java**

public class B extends A {

    public void g(int x, int y) {

        int m = super.f(x, y);// 调用父类方法

        int n = (x \* y) / m;// 计算最大公倍数

        System.out.println(x + "和" + y + "的最小公倍数数为：" + n);// 打印

    }

}

Test2.java

import java.util.Scanner;

public class Test2 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        A num1 = new A();// 定义A类变量num1

        B num2 = new B();// 定义B类变量num2

        int a, b, c, d;

        // 下面是求最大公约数

        System.out.println("请输入两个数求最大公约数：");

        a = sc.nextInt();

        b = sc.nextInt();

        num1.f(a, b);// 求6,8的最大公约数

        System.out.println(num1.x + "和" + num1.y + "的最大公约数为：" + num1.f(a, b));

        // 下面是求最小公倍数

        System.out.println("请输入两个数求最小公倍数：");

        c = sc.nextInt();

        d = sc.nextInt();

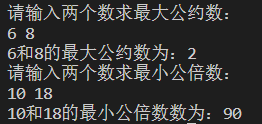
        // 测试最小公倍数

        num2.g(c, d);

    }

}

###### 输出结果：



###### 分析：

A类就是算最大公因数，公因数简单理解就是一个数能使两个数对它取余是0，所以我们就能写出相应代码，只要从最大那个数开始递减1，每一次都判断两个数对它取余是否都为0即可。B类就用super来调用父类A中的方法，之后按照公式计算即可，测试类就是数据输出等基本功能。

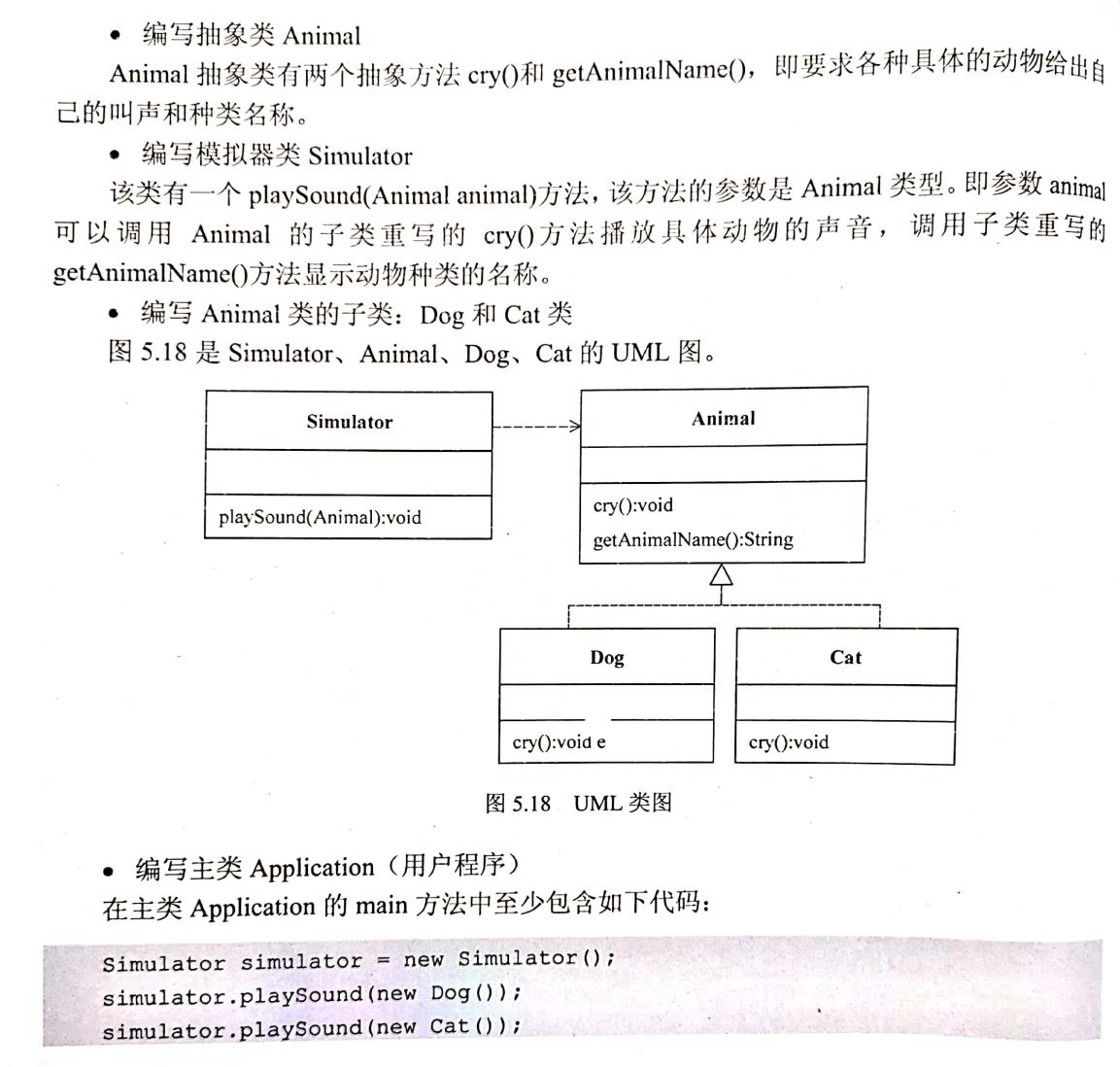
##### 练习4

###### 实验目的

熟练掌握类的继承和父类方法重写，以及抽象类的使用

###### 实验要求

###### 



###### 代码：

**Animal.java**

public abstract class Animal {

    public abstract void cry();// 抽象方法cry

    public abstract void getAnimalName();// 抽象方法getAnimalName

}

**Simulator.java**

public class Simulator {

    Animal animal;// 成员变量

    public void playSound(Animal animal) {

        this.animal = animal;

        animal.cry();// 调用cry方法

        animal.getAnimalName();// 调用getAnimalName方法

    }

}

**Dog.java**

public class Dog extends Animal {// Animal的子类Dog

    public void cry() {// 重写方法cry

        System.out.println("wang wang wang!");

    }

    public void getAnimalName() {// 重写方法getAnimalName

        System.out.println("这是狗");

    }

}

**Cat.java**

public class Cat extends Animal {// Animal的子类Cat

    public void cry() { // 重写方法cry

        System.out.println("miao miao miao!");

    }

    public void getAnimalName() {// 重写方法getAnimalName

        System.out.println("这是猫");

    }

}

**Test3.java**

public class Test3 {

    public static void main(String[] args) {

        Simulator simulator = new Simulator();// 定义新对象

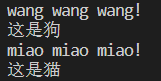
        simulator.playSound(new Dog());// 调用方法

        simulator.playSound(new Cat());// 调用方法

    }

}

###### 输出结果：



###### 分析：

构造抽象类Animal，含有cry和getAnimalName两个成员方法，Dog类和Cat类都是Animal类的子类，所以他们也都含有这两个成员方法，我们只需要在子类中对父类的方法进行重写即可，Simulator类只是调用相应的成员方法。

#### 【小结与讨论】

1. 在这次编写代码的过程中我学会了在父类中变量如果用private修饰就不可以在子类中使用了
2. 在调用父类方法的时候发现，static修饰的方法中不可以使用this和super不可以直接访问它所在类的成员变量和成员方法

#### 