

第 4 次实验类的三大特性

实验内容：建立类继承的层次结构。

1 创建人-教师/学生类的继承关系，定义 Test 类创建对象

属性要是 private 私有属性

构造函数 2 个

一组 get 方法和一组 set 方法

输出属性所有属性的 print 方法

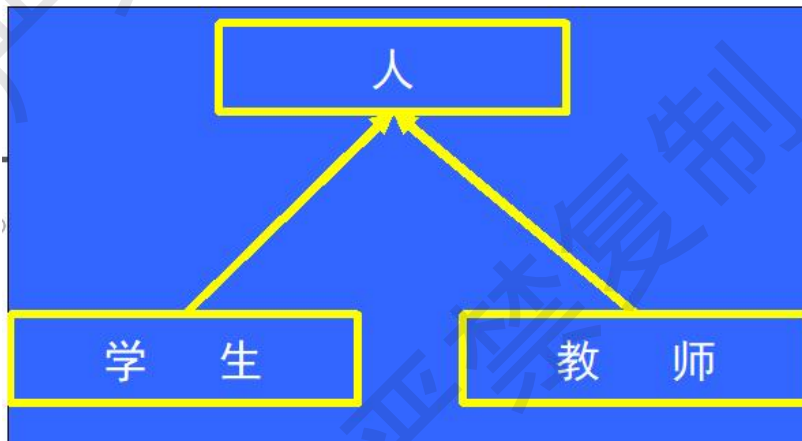
其他方法自己想想

在子类教师/子类学生中使用 super() 调用父类（人类）构造函数

一组 get 方法和一组 set 方法

输出属性所有属性的 print 方法//print 方法实现覆盖（父类和子类都有）

其他方法自己想想



程序 2：点-圆-圆柱的继承关系，

点类 Point

用 private 定义坐标属性 x , y

构造函数 2 个

一组 get 方法和一组 set 方法

输出属性所有属性的 print 方法

其他方法自己想想

子类 Circle

用 private 定义坐标属性 radius

构造函数 2 个

一组 get 方法和一组 set 方法

计算圆的周长的方法

计算圆的面积的方法

输出所有属性及相关信息 print 方法//print 方法实现覆盖（父类和子类都有）

其他方法自己想想

注意：使用 super 调用父类的构造函数，super 调用父类的同名方法

子类 Cylinder

用 private 定义坐标属性 h

构造函数 2 个

一组 get 方法和一组 set 方法

计算圆柱的表面积的方法

计算圆的体积的方法

输出所有属性及相关信息的 print 方法//print 方法实现覆盖(父类和子类都有)

注意：使用 super 调用父类的构造函数，super 调用父类的同名方法

3 (选做，下次的作业会有该题) 填写如下程序，并在需要修改的地方进行修改,后,分析程序的运行结果,以及运行错误的原因。

```
class Person{
    private String name;
    private int ID;
//补充无参的构造函数
    public Person(String theName){
        this.name = theName;
    }
//补充 2 个参数的构造函数
    public String getName(){
        return name;
    }
//补充一组 set 方法和一组 get 方法
    void print(){
        System.out.println("name :"+name+"ID:"+ID );
    }
}
public class Employee extends Person{
    private String company;
//补充无参的构造函数

    public Employee(String name, int ID, String company){
        this.name = name;    this.ID=ID                // error, 请修改
        this. company = company;
    }
//补充 set 方法
    public int get Company (){}
```

```
        return company;
    }
    void print(){} //补充能输出所有属性

}

public class Main{
    public static void main(String[] args){
        //创建 Person 类的对象 p1,他的名字是 Mike,ID 是 123453
        System.out.println(p1.getName()); // Mike
        //创建 Person 类的对象 p2,他的名字是 Dani,ID 是 123456

        System.out.println(p2.getName()); //输出什么?
        System.out.println(p2.getId()); // 输出什么

        Person p3 = new Employee("Tom", 1211,"Microsoft");
        System.out.println(p2.getName()); //输出什么?
        System.out.println(p2.getId()); // 输出什么

        Employee p4 = new Person("Mike"); //error 为什么?

        Employee emp = new Employee("Dani", 213223,"DEC");
        System.out.println(emp.getName()); ///输出什么?
        System.out.println(emp.getId()); // /输出什么?

    }
}
```