

一、继承性的作用

二、继承性的格式

三、继承性的体现

四、Java中关于继承性的规定

五、java.lang.Object的理解

六、例题

1、作业练习

2、圆柱的体积

一、继承性的作用

- 1、继承的出现减少了代码冗余，提高了代码的复用性。
- 2、继承的出现，更有利于功能的扩展。
- 3、提供了多态的前提。

二、继承性的格式

- 1、格式：class A extends B
- 2、说明：

A：子类、派生类、subclass

B：父类、超类、基类、superclass

三、继承性的体现

1、一旦子类A继承了父类B之后，子类A就获取了父类B中声明的所有结构、属性、方法。特别的，父类中声明为private的属性或方法，子类继承父类以后，仍然认为获取了父类中私有的结构。只是因为封装性的影响，使得子类不能直接调用父类的结构而已。

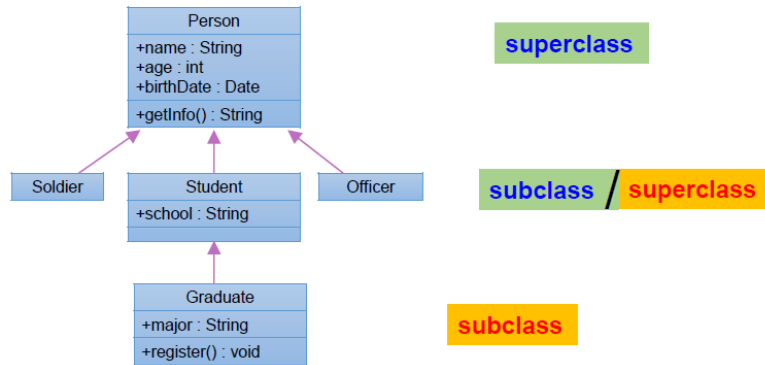
- 2、子类A继承了父类B之后，还可以声明自己特有的属性或方法，实现功能的拓展。

子类和父类的关系，不同于子集和集合的关系。

四、Java中关于继承性的规定

- 1、一个类可以被多个子类继承
- 2、Java中类的单继承性：一个类只能有一个父类
- 3、子父类是相对的概念
- 4、子类直接继承的父类称为：直接父类；间接继承的父类称为：间接父类
- 5、子类继承父类以后，就获取了所有的直接父类以及所有间接父类中声明的属性和方法

单继承与多层继承举例



五、java.lang.Object的理解

- 1、如果没有显示的声明一个类的父类的话，则此类继承于java.lang.Object类。
- 2、所有的java类（除java.lang.Object类以外）都直接或间接继承于java.lang.Object类。
- 3、意味着：所有的java类都具有java.lang.Object类所声明的功能

六、例题

1、作业练习

练习1

2. (1)定义一个ManKind类， 包括

- 成员变量int sex和int salary;
- 方法void manOrWoman(): 根据sex的值显示 “man” (sex==1)或者 “woman”(sex==0);
- 方法void employeeed(): 根据salary的值显示 “no job” (salary==0)或者 “ job”(salary!=0)。

(2)定义类Kids继承ManKind， 并包括

- 成员变量int yearsOld;
- 方法printAge()打印yearsOld的值。

(3)定义类KidsTest， 在类的main方法中实例化Kids的对象someKid， 用该对象访问其父类的成员变量及方法。

```
1
2  /*
3   * (1) 定义一个 ManKind 类， 包括：
4   * 成员变量 int sex 和 int salary
5   * 方法 void manOrWoman ()(): 根据 sex 的值显示“ man ”(sex==1) 或者 woman ”(sex==0)
6   * 方法 void employeeed ()(): 根据 salary 的值显示“ no job ”(salary==0) 或者“ job”(salary!=0) 。
7   *
8   * */
9  class ManKind {
10     private int sex;// 性别
11     private int salary; //薪资
12
13     public ManKind() {
14     }
15 }
```

```

16 public ManKind(int sex, int salary) {
17     this.sex = sex;
18     this.salary = salary;
19 }
20
21 public int getSex() {
22     return sex;
23 }
24
25 public void setSex(int sex) {
26     this.sex = sex;
27 }
28
29 public int getSalary() {
30     return salary;
31 }
32
33 public void setSalary(int salary) {
34     this.salary = salary;
35 }
36
37 public void manOrWoman (){
38     if(sex == 1){
39         System.out.println("man");
40     }else if(sex == 0){
41         System.out.println("woman");
42     }
43 }
44
45 public void employeeed(){
46     if(salary == 0){
47         System.out.println("no job");
48     }else if(salary != 0){
49         System.out.println("job");
50     }
51 }
52 }
53
54 /*
55  * (2).定义类 Kids 继承 ManKind ， 并包括
56  * 成员变量 int yearsOld
57  * 方法 printAge 打印 yearsOld 的值。
58  */
59
60
61 class Kids extends ManKind{
62     private int yearsOld; // 年龄
63
64     public Kids() {
65

```

```

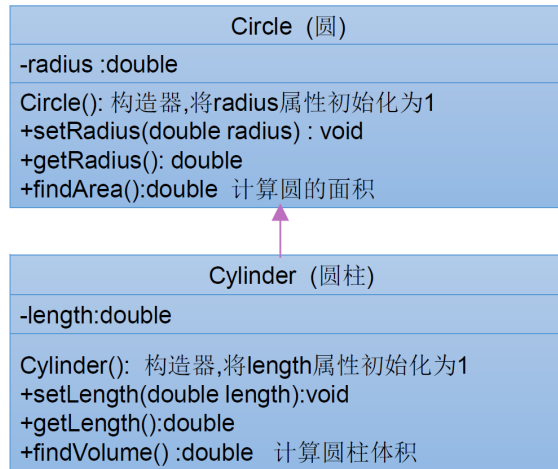
66
67 public Kids(int yearsOld) {
68     this.yearsOld = yearsOld;
69 }
70
71 public int getYearsOld() {
72     return yearsOld;
73 }
74
75 public void setYearsOld(int yearsOld) {
76     this.yearsOld = yearsOld;
77 }
78
79 // 获取年龄
80 public void printAge(){
81     System.out.println("I am " + yearsOld + " years old");
82 }
83
84 }
85
86 /*
87  * (3) 定义类 KidsTest，在类的 main 方法中实例化 Kids 的对象 someKid，
88  * 用该对象访问其父类的成员变量及方法。
89  */
90
91 public class KidsTest {
92     public static void main(String[] args) {
93         Kids someKid = new Kids();
94         someKid.setYearsOld(18);
95         someKid.printAge();
96
97         someKid.setSex(1);
98         someKid.manOrWoman();
99
100         someKid.setSalary(3232);
101         someKid.employeed();
102     }
103 }

```

2、圆柱的体积

练习1

3. 根据下图实现类。在**CylinderTest**类中创建**Cylinder**类的对象，设置圆柱的底面半径和高，并输出圆柱的体积。



让天下没有难学的技术。

```
1
2 class Circle {
3     private double radius; // 半径
4
5     // 将 radius 属性初始化为 1
6     public Circle() {
7         this.radius = 1.0;
8     }
9
10    public double getRadius() {
11        return radius;
12    }
13
14    public void setRadius(double radius) {
15        this.radius = radius;
16    }
17
18    public double findArea () {
19        return Math.PI * radius * radius;
20    }
21 }
22
23 class Cylinder extends Circle {
24     private double length;
25
26     public Cylinder() {
27         this.length = 1.0;
28     }
29
30    public double getLength() {
31        return length;
32    }
33 }
```

```

34 public void setLength(double length) {
35     this.length = length;
36 }
37
38 public double findVolume () {
39     //return Math.PI * getRadius() * getRadius() * getLength();
40     return findArea() * getLength();
41 }
42 }
43
44 public class CylinderTest {
45     public static void main(String[] args) {
46         Cylinder cy = new Cylinder();
47
48         cy.setRadius(2.0);
49         double area = cy.findArea();
50         System.out.println("圆的面积: " + area);
51
52         cy.setLength(5.0);
53         double volume = cy.findVolume();
54         System.out.println("圆柱的体积为: " + volume);
55
56     }
57 }

```

