CSDN

博客 学院 下载 图文课 论坛 APP 问答 商城 VIP会员 活动 招聘 ITeye GitChat



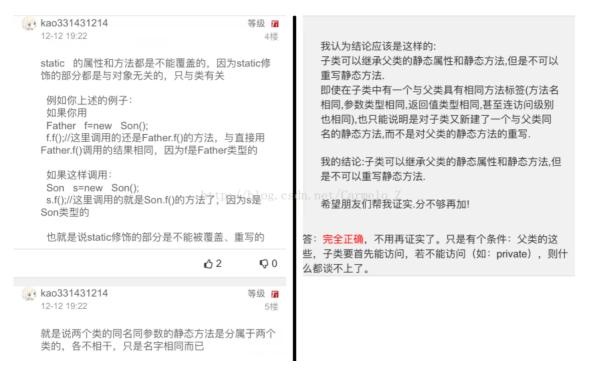
ıβ

2 2

Java静态属性与静态方法能否被继承的问题

2017年03月27日 23:25:21 Carmelo Z 阅读数:2788 更多

开始正文之前的一个参考:



原文链接:点击打开链接

结论:java中静态属性和静态方法可以被继承,但是没有被重写(overwrite)而是被隐藏.

原因:

- 2). 多态之所以能够实现依赖于继承、接口和重写、重载(继承和重写最为关键)。有了继承和重写就可以实现父类的引用指向不同子类的对象。重写的功能是:"重写"后子于父类的优先级,但是"隐藏"是没有这个优先级之分的。
- 3). 静态属性、静态方法和非静态的属性都可以被继承和隐藏而不能被重写,因此不能实现多态,不能实现父类的引用可以指向不同子类的对象。非静;继承和重写,因此可以实现多态。

例子:

```
1
   package com.study.test;
2
3
   public class A { //父类
           public static String staticStr = "A静态属性";
4
           public String nonStaticStr = "A非静态属性";
5
6
           public static void staticMethod(){
7
                   System.out.println("A静态方法");
8
9
           public void nonStaticMethod(){
                   System.out.println("A非静态方法");
10
11
12
```

```
package com.study.test;
   public class B extends A{//子类B
3
         public static String staticStr = "B改写后的静态属性";
4
5
         public String nonStaticStr = "B改写后的非静态属性";
6
         public static void staticMethod(){
7
                System.out.println("B改写后的静态方法");
8
9 }
package com.study.test;
2
   public class C extends A{//子类C继承A中的所有属性和方法
3
4
5 | }
1 package com.study.test;
2
3
   public class StaticExtendsTest {
4
5
          public static void main(String[] args) {
6
                 C c = new C();
7
                 System.out.println(c.nonStaticStr);
8
                System.out.println(c.staticStr);
                 c.staticMethod();//输出的结果都是父类中的非静态属性、静态属性和静态方法,推出静态属性和静态方法可以被继承
9
10
                 System.out.println("----");
11
12
13
                 A c1 = new C();
14
                 System.out.println(c1.nonStaticStr);
15
                 System.out.println(c1.staticStr);
                 c1.staticMethod();//结果同上,输出的结果都是父类中的非静态属性、静态属性和静态方法,推出静态属性和静态方法可以被继承
16
17
18
                 System.out.println("----");
19
                 B b = new B();
20
                 System.out.println(b.nonStaticStr);
21
                 System.out.println(b.staticStr);
                 b.staticMethod();
22
23
                 System.out.println("----");
24
25
                 A b1 = new B():
26
                 System.out.println(b1.nonStaticStr);
27
                 System.out.println(b1.staticStr);
28
                 b1.staticMethod();//结果都是父类的静态方法,说明静态方法不可以被重写,不能实现多态
29
          }
30
31
```

现在才知道,零基础学习高级lava后,年薪可以这么多!

零基础学IT选Java,易学、高薪、前景广,100万人才缺口,互联网必备人才

想对作者说点什么

★ weixin_40677431: 厉害了,解决了我多态和静态方法继承的疑难了 (3个月前 #2楼)

sili0816: 这相当于编译时多态和运行时多态的区别吗? (4个月前 #1楼)