

java认证辅导:构造方法的继承Java认证考试 PDF转换可能丢失
图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022_java_E8_AE_A4_E8_AF_81_c104_644785.htm 构造方法是一种特殊的方法，

它的继承规则比较简单，而且与普通方法有较大的区别 1 无

参数构造方法的构造方法示例 class hasConstructor{ protected
int x=100. public void showMsg(){ System.out.println("this is a
method in ancesstor"). } public hasConstructor(){

System.out.println("this is a constructor in ancestor without
parameter"). } public hasConstructor(int ix){

System.out.println("this is a constructor in ancestor with
parameter"). } }

下面这个子类会自动调用不带参数的构造方法
public class inheritConstruct_1 extends hasConstructor{ public
static void main(String args[]){ inheritConstruct_1 oa=new
inheritConstruct_1 (). } }

在类inheritConstruct_1 中并没有定义不带参数的构造方法，但调用的是父类的不带参数的构造方法，这看上去是一种继承，实际上系统是为子类添加了一个不带参数的构造方法，而子类这个构造方法又会自动调用父类无参的构造方法。

2 带参数的构造方法的继承 带参数的构造方法不会被子类继承，也不会自动调用 public class
inheritConstruct_1 extends hasConstructor{ public static void
main(String args[]){ inheritConstruct_2 oa=new
inheritConstruct_1(100). //错误，试图调用不带参数的构造方法，它没有被继承下来 } }

3 无参数构造方法的覆盖 首先明确一点带参数的构造方法不会被继承，也不存在覆盖的问题，因此只有无参的构造方法才存在覆盖这个问题，但是由于构

造方法不会被继承，也不会自动调用 public class
inheritConstruct_1 extends hasConstructor{ public static void
main(String args[]){ inheritConstruct_2 oa=new
inheritConstruct_1(100). //错误，试图调用不带参数的构造方法，它没有被继承下来 } }

3 无参数构造方法的覆盖 首先明确一点带参数的构造方法不会被继承，也不存在覆盖的问题，因此只有无参的构造方法才存在覆盖这个问题，但是由于构

造方法不会被继承，也不会自动调用 public class
inheritConstruct_1 extends hasConstructor{ public static void
main(String args[]){ inheritConstruct_2 oa=new
inheritConstruct_1(100). //错误，试图调用不带参数的构造方法，它没有被继承下来 } }

3 无参数构造方法的覆盖 首先明确一点带参数的构造方法不会被继承，也不存在覆盖的问题，因此只有无参的构造方法才存在覆盖这个问题，但是由于构

造方法不会被继承，也不会自动调用 public class
inheritConstruct_1 extends hasConstructor{ public static void
main(String args[]){ inheritConstruct_2 oa=new
inheritConstruct_1(100). //错误，试图调用不带参数的构造方法，它没有被继承下来 } }

造方法必须与所在的类同名，而子类的名称和父类不同，因此构造方法名字也显然不同，所以这种覆盖和普通方法的覆盖相比，无论从形式上还是从执行方法上都有很大的区别。

编辑特别推荐: 一个有趣的编程:程序员的爱情故事 Java认证权威问答精华集 Java核心API需要掌握的程度 100Test 下载频道 开通，各类考试题目直接下载。详细请访问 www.100test.com