

java认证辅导:Java新手进阶(细说引用类型)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\\_ti2020/559/2021\\_2022\\_java\\_E8\\_AE\\_A4\\_E8\\_AF\\_81\\_c104\\_559603.htm](https://www.100test.com/kao_ti2020/559/2021_2022_java_E8_AE_A4_E8_AF_81_c104_559603.htm)

引用类型的变量非常类似于C/C的指针。为了形象起见，也为了打字方便，本文后面的内容，都把“引用类型的变量”称为指针。所以，如果你原先有C/C背景，今天讲的内容对你来说应该很好理解；否则的话，可能要多琢磨琢磨了。 创建问题 假设我们在函数中写了如下这个简单的语句：`StringBuffer str = new StringBuffer("Hello world")`；别看这个语句简单，其实包含了如下三个步骤：首先，`new StringBuffer("Hello world")`在堆里申请了一坨内存，把创建好的StringBuffer对象放进去。其次，`StringBuffer str`声明了一个指针。这个指针本身是存储在栈上的（因为语句写在函数中），可以用来指向某个StringBuffer类型的对象。或者换一种说法，这个指针可以用来保存某个StringBuffer对象的地址。最后，当中这个等于号（赋值符号）把两者关联起来，也就是把刚申请的那一坨内存的地址保存成str的值。 /NewsFiles/2009-3/24/101057175.jpg

引用对象之间的赋值、判相等 通过上述的图解，大伙儿应该明白指针变量和该指针变量指向的对象是一个什么关系了吧。还是接着刚才的例子，再来看赋值的问题。对于如下语句：`StringBuffer str2 = str`；这个赋值语句是啥意思捏？实际上就是把str的地址复制给str2，记住，是地址的复制，StringBuffer对象本身并没有复制。所以两个指针指向的是同一个东东。再搞一张示意图，如下（今天画这些图把我累坏了）： /NewsFiles/2009-3/24/1010287192.jpg final常量的问

题 针对引用类型变量的final修饰符也是很多人搞混淆的地方。实际上final只是修饰指针的值（也就是限定指针保存的地址不能变）。至于该指针指向的对象，内容是否能变，那就管不着了。所以，对于如下语句：`final StringBuffer strConst = new StringBuffer（）`；你可以修改它指向的对象的内容，比如：`strConst.append（" "）`；但是不能修改它的值，比如：`strConst = null`；

传参的问题 引用类型（在函数调用中）的传参问题，是一个相当扯的问题。有些书上说是传值，有些书上说是传引用。搞得Java程序员都快成精神分裂了。所以，我们最后来谈一下“引用类型参数传递”的问题。v还是拿刚才的例子，假设现在要把刚才创建的那一坨字符串打印出来，我们会使用如下语句：`System.out.println（str）`；这个语句又是什么意思捏？这时候就两说了。第一种理解：可以认为传进函数的是str这个指针，指针说白了就是一个地址的值，说得再白一点，就是个整数。按照这种理解，就是传值的方式。也就是说，参数传递的是指针本身，所以是传值的。第二种理解：可以认为传进去的是StringBuffer对象，按照这种理解，就是传引用方式了。因为我们确实是把对象的地址（也就是引用）给传了进去。费了这么多口水，其实不论是传引用还是传值，都可以讲得通，关键取决于你是如何看待参数所传递的东西。这就好比量子力学中“光的波粒二象性”，如果你以粒子的方式去测量它，它看起来像粒子；如果你以波动的方式去观测它，它看起来像波动。假如你不太懂量子力学，前面这话当我没说：-）明白了赋值，判断相等的问题（就是==操作符）也就简单了。当我们写如下语句“`if（str2 == str）`”时，只是判断两个指针的值（也就是对象

的地址)是否相等,并不是判断被指向的对象是否内容相同。实际上两个指针的值相同,则肯定是指向同一个对象(所以对象内容必定相同)。但是两个内容相同的对象,它们的地址可能不一样(比如克隆出来的多个对象之间,地址就不同)。更多优质资料尽在百考试题论坛 百考试题在线题库 java认证更多详细资料 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](http://www.100test.com)