

销毁对象的三把利剑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__E9_94_80_E6_AF_81_E5_AF_B9_E8_c104_644569.htm

在任何一个面向对象的语言中，这个对象销毁机制都是很重要的。因为如果对象不能够被及时的回收，那么很多没用的对象就仍然会占用内存地址。久而久之，就会导致内存耗竭，从而导致服务器当机。在这个对象的销毁上，Java语言比其他面向对象的语言更胜一筹。因为其他面向对象的语言，需要在程序代码中手工回收废弃的对象。不过Java语言则不同。在Java语言中拥有一套完整的对象垃圾回收机制。即程序开发人员不需要手工的回收废弃的对象。垃圾回收器将回收无用对象所占用的内存资源。不过这个垃圾回收器并不是万能的，需要结合其他的对象回收方式，才能够最终解决这个对象回收的问题。

在Java中，总共有三种方式可以用来解决对象回收的问题，笔者将他们称为三把利剑。第一把剑：垃圾回收器 垃圾回收器是Java平台中用的最频繁的一种对象销毁方法。垃圾回收器会全程侦测Java应用程序的运行情况。当反先有些对象成为垃圾时，垃圾回收器就会销毁这些对象，并释放这些对象所占用的内存空间。在这里，程序开发人员需要知道，在哪些情况下垃圾回收器会认为这些对象是垃圾对象。通常情况下，如果发生以下两种情况时，系统会认为这些对象是垃圾对象，需要销毁。一是将一个NULL值赋值给对象。如用户先建立了一个对象str1。对象用完了之后，再利用赋值语句，将NULL值赋值给这个对象str1，即使用str1=NULL的方式将NULL值赋值给这个对象。此时这个对象与内存中对象的存

储地址之间就失去了联系。此时内存中的对象就好像成为了一个无主的对象，就会被垃圾回收器销毁。不过这也有例外。如现在同一个对象有两个名字，分别为str1与str2。此时若只是将NULL值赋值给str1，那么内存的这个对象仍然有一个主人，即str2。此时这个对象还暂时不会被垃圾回收器回收。除非在代码中讲这个对象所关联的所有对象名字都赋值为NULL。此时这个对象才变为无主的对象，才会被垃圾回收器回收。在Java中定义自己的工具库

二是对象其超出了作用范围，这个这个对象就被认为是垃圾对象，被垃圾回收器回收并释放内存。其实对象跟变量一样，其也有作用域。当超过这个作用域的话，跟变量一样，这个对象也就无效了，其他地方不能够再引用这个对象。此时这个对象就是名存实亡了。为此垃圾回收器也把这种超过作用范围的对象当作垃圾对象来处理。所以说，如果程序员要让对象中的某些成员变量或者成员方法一直有效的話，就需要采用static关键字，让其在变量销毁时仍然有效，或则仍然可以调用这些成员变量与成员方法。来源：www.100test.com 另外笔者在这里各位读者一个小的思考题。大家想想看，如果遇到如下这种情况，Java垃圾回收器是否会回收对象?现在笔者先定义两个对象str1，str2。此时在内存中已经为这个两个对象分配了存储空间。然后笔者再利用str2=str1语句，将str1指向的内存地址赋值给str2。此时对象str1、str2就指向同一个内存地址，即代表同一个对象。而原先对象str2指向的对象就变成了一个无主的对象了，对象名str2已经与这个对象失去了联系。在这种情况下，这个对象是否会被当作垃圾对象被销毁呢?给为读者回去可以好好想想这个问题。若能够得出正确的答案，则说明

大家对于垃圾回收器已经有了比较透彻的了解。JDK7更新为Java带来更多模块 第二把利剑：finalize方法 虽然说垃圾回收器已经是一个比较完善的对象销毁机制，但是其并不能够解决Java语言中所有对象的销毁问题。如Java语言平台中，创建对象主要有两种方式，分别为利用NEW关键字来创建(这是对象创建的主要方式)与不适利用NEW关键字来创建。如果Java语言中某些对象不是利用new关键字为对象在内存中分配一块存储区域，那么这种类型的对象就不能够被垃圾回收站回收。即使在代码中将NULL值赋值给这个对象，仍然不能够被垃圾回收器回收。虽然这种对象比较少，但是毕竟存在。为了解决这类对象的回收与内存释放问题，在Java语言中提供了一个finalize的方法。通过这个方法可以显示的让系统回收这个对象。这是一个Object类的方法。通常情况下，这个方法是被声明为protected。程序开发人员在必要的时候，可以在自定的类中定义这个方法。假设现在程序开发人员定义了这个方法，那么在对象销毁时，垃圾回收器会先采用这个方法来销毁对象。并且在下一次垃圾回收动作发生时，最终释放对象的内存。可见，采用了finalize方法，虽然可以让这个对象被垃圾回收器回收，但是其原理是不同的。垃圾回收器是先调用这个方法，然后在下次运行这个垃圾回收作业时再释放其占用的内存。另外需要注意的是，这个方法并不一定会保证发生。也就是说，相对来讲，其安全性并没有垃圾回收器那么的高。故在实际创建对象时，最好还是采用new关键字来创建比较安全，也更加的方便。因为在利用New关键字来创建对象时，在代码中不需要加入另外额外的代码。所以从安全性与便利性上来说，笔者都建议采用new方法为对象在

内存中分配存储区域，而不建议采用其他方式。在采用垃圾回收器的时候，其会自动检测对象的状态，而不需要程序员告知。其发现对象超出了作业域范围或者对象被重置为空时，其就会被自动销毁并释放对象所占用的内存，这种方式处理起来相对来说更加的安全与便利。www.Examda.CoM考试就到百考试题 第三把利剑：利用System.gc方法强制启动垃圾回收器 垃圾回收器其实自动启动的。也就是说，垃圾回收机制起会自动监测垃圾对象，并在适当的时候启动垃圾回收器来销毁对象，释放内存。但是这个垃圾回收器也会有不合作的时候。也就是说，这个垃圾回收器不受程序代码的控制，其具体执行的时间也会不确定，从而导致上面的finalize方法无法执行，某些对象无法及时销毁。为此有时候需要利用代码来强制启动垃圾回收器，来销毁对象。百考试题论坛 笔者平时在给一些学员培训时，往往将这个垃圾回收器比喻成医院里的护士。平时的时候，护士会每隔一段时间来进行查房，看看病人有没有什么问题。不过在必要的时候，病人也可以通过警铃来呼叫护士。如当盐水挂完了就可以要求护士来更换盐水。其实这个护士定期查房就好像是垃圾回收器会根据应用程序的运行情况来定时的或者不定时的启动。而这个警铃就好像是这个System.gc方法。在有需要的时候，程序开发人员可以在代码中调用这个方法，来强制让护士来病房，即强制启动这个垃圾回收器。采用这个方法，给程序开发人员多了一种主动权，而不需要被动的等着垃圾回收器的启用。有时候，在代码种类利用这个方法不定期或者定期的强制启动垃圾回收器，对及时销毁垃圾对象能够起到一个辅助与保障作用。 编辑特别推荐: 指点一下：到底该不该去

考JAVA认证? Java面试题：第一锦 Java面试题：第二锦 Java面试题：第三锦 Java面试题：第四锦 Java面试题：第五锦 一个有趣的编程:程序员的爱情故事 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