

关于JAVA内存泄漏问题注意事项 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/290/2021_2022__E5_85_B3_E4_BA_8EJAVA_c104_290406.htm

Java的一个重要优点就是通过垃圾收集器（Garbage Collection，GC）自动管理内存的回收，程序员不需要通过调用函数来释放内存。因此，很多程序员认为Java不存在内存泄漏问题，或者认为即使有内存泄漏也不是程序的责任，而是GC或JVM的问题。其实，这种想法是不正确的，因为Java也存在内存泄露，但它的表现与C不同。在Java中，内存泄漏就是存在一些被分配的对象，这些对象有下面两个特点，首先，这些对象是可达的，即在有向图中，存在通路可以与其相连；其次，这些对象是无用的，即程序以后不会再使用这些对象。如果对象满足这两个条件，这些对象就可以判定为Java中的内存泄漏，这些对象不会被GC所回收，然而它却占用内存。在C中，内存泄漏的范围更大一些。有些对象被分配了内存空间，然后却不可达，由于C中没有GC，这些内存将永远收不回来。在Java中，这些不可达的对象都由GC负责回收，因此不需要考虑这部分的内存泄露。

注意事项：1.最基本的建议是尽早释放无用对象的引用。如：`..... A a = new A () ; //应用a对象 a = null ; //当使用对象a之后主动将其设置为空 ...`。注：如果a是方法的返回值，不要做这样的处理，否则你从该方法中得到的返回值永远为空，而且这种错误不易被发现、排除

2.尽量少用finalize函数。它会加大GC的工作量。

3.如果需要经常使用经常用到的图片，可以使用soft应用类型。它尽可能把图片保存在内存中

4.注意集合数据类型，包括数组、树、图、链表等数据结构，这些

数据结构对GC来说，回收更为复杂。

5. 尽量避免在类的默认构造器中创建、初始化大量的对象，防止在调用其自类的构造器时造成不必要的内存资源浪费
6. 尽量避免强制系统做垃圾内存的回收，增长系统做垃圾回收的最终时间
7. 尽量避免显式申请数组空间
8. 尽量做远程方法调用类应用开发时使用瞬间值变量，除非远程调用端需要获取该瞬间值变量的值。
9. 尽量在合适的场景下使用对象池技术以提高系统性能。

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