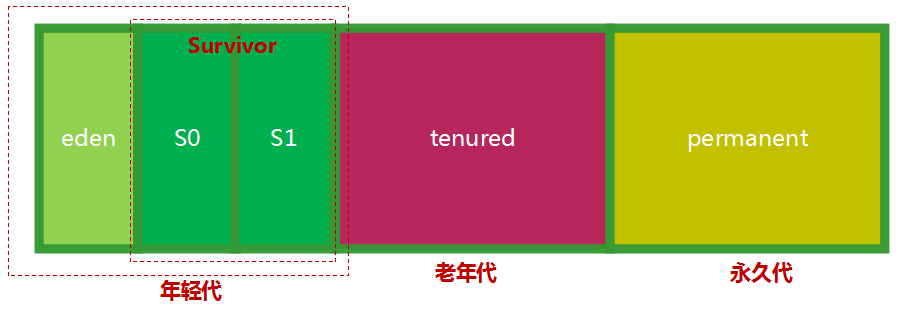
1. 资源回收机制
2. 什么是资源回收机制？

程序运行时，自动运行且管理程序使用的内存；清理堆内存中的无用对象，释放对象所占的内存资源。

1. Java堆内存中的对象分代存储



资源回收的过程，详细参照PPT。

1. 四种类型垃圾回收器

Serial Garbage Collector

Parallel Garbage Collector 默认

CMS Garbage Collector

G1 Garbage Collector

1. 判断对象是否是垃圾：

对象没有引用；

作用域发生未捕获异常；

程序在作用域正常执行完毕；

程序执行了System.exit()。

以上几种情况都会导致对象失去栈的直接或间接引用，没有引用的对象成为无用对象，即为垃圾对象。

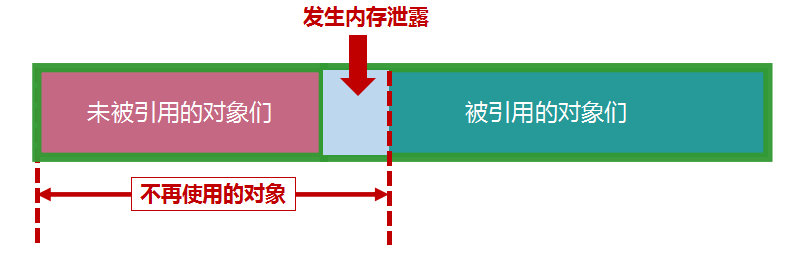
判断对象是否拥有引用：

引用计数器算法； jdk1.2前

根搜索算法， 判断引用路径是否可达。

1. 内存泄露

对象已经没有被应用程序使用，但是垃圾回收器没办法移除它们，因为还在被引用着，仍然占用内存资源。



如何有效防止内存泄露？

1. finalize()方法

Object中包含了一个叫做finalize()的方法，提供在对象被回收时调用，以释放资源，默认情况下其不执行任何动作。

当垃圾回收器确定了一个对象没有任何引用时，回收对象前，会调用finalize()方法。但是，finalize方法并不保证调用时机，因此也不建议重写finalize()方法。

如果必须要重写finalize()方法，请记住使用super.finalize()调用父类的清除方法，否则对象清理的过程可能不完整。

1. 强引用、软引用、弱引用、虚引用

JAVA把对象的引用分为四种级别，从而使程序能更加灵活的控制对象的生命周期。

