**内存空间管理**

通过上文，我们知道了变量是按照一定的数据结构方式存储在内存中，那么，内存是怎么管理这些变量占用的内存，才能保证不使用的变量不浪费占用内存呢？这其实得益于内存自动的分配与回收机制。

在javascript中内存的生命周期如下：

1. 分配变量所需的内存
2. 读写分配到的内存
3. 释放不使用的内存

示例：

var a = 20 // 内存中给变量a分配了空间

console.log(a) // 读内存中的值

a = null // 标记内存使用完毕，可释放变量占用的内存空间

Javascript自动垃圾收集机制很简单，就是找出不继续使用的值，然后释放该值占用的内存。并且垃圾收集器会每隔固定的时间段就执行一次垃圾收集。

那么怎么找出不继续使用的值呢？在javascript中使用的【标记清除】的算法来找到哪些变量不再使用，所以上述示例中a=null就是一种标记，让a对应的值失去应用，使得这个值在下一次垃圾收集器执行时能被释放。

值的注意的是，在局部作用域中，比如函数执行完毕后，我们没必要给每个不使用的变量都null，因为垃圾收集器会自己做出判断和回收，但是对于全局比那里，什么时候需要释放内存空间就比较难判断了，因为在原则上开发中我们应该减少使用过多的全局变量。

对于垃圾收集机制的详细了解，可以阅读《JavaScript高级编程》中的4.3节。