



谈谈js中的内存机制——垃圾回收机制



兔子先生家的煎蛋君 (/u/7b40a1344fff) [+ 关注](#)

0.1 2018.02.05 10:47 字数 661 阅读 975 评论 0 喜欢 5

(/u/7b40a1344fff)

内存管理机制就是分配内存管理，每种编程语言都有它的内存管理机制，JavaScript的内存管理机制是：内存基元在变量（对象，字符串等等）创建时分配，然后在他们不再被使用时“自动”释放。后者被称为垃圾回收。这个“自动”是混淆并给JavaScript（和其他高级语言）开发者一个错觉：他们可以不用考虑内存管理，JS内存空间分为栈(stack)、堆(heap)、池(一般也会归类为栈中)。

其中栈存放变量，堆存放复杂对象，池存放常量，在js中的分配的内存一般有如下生命周期

- 内存分配（当我们声明变量，函数，对象时系统自动为他们分配内存）
- 内存使用（使用变量，函数等）
- 内存回收（使用完毕，由垃圾回收机制自动回收不再使用的内存）

当内存走到最后一步的时候就开始内存回收，js中使用的是垃圾回收机制

垃圾回收有2种基本方式——

2. 标记清除——

垃圾回收器会在运行时给存储在内存中的所有变量加一个标记，然后去除环境中的变量以及被环境中的变量所引用的变量（闭包）在这些完成后仍存在标记的就是要删除的变量了，因为环境中的变量已经无法访问到这些变量了

2. 引用计数——

引用计数的策略是跟踪记录每个值被使用的次数。当声明了一个变量并将一个引用类型赋值给该变量时，这个值得引用次数就加一，如果该变量的值变成了另一个，则这个值得引用次数就减一，当这个值的引用次数为0的时候，说明没有变量在使用，这个值无法访问。由此可以将其占用的空间回收，这些垃圾回收器就会在运行时清理掉引用次数为0的值占用的空间，但这种方法容易引起内存泄漏，因为这种方式没有解决循环引用的问题，所以不建议使用！

以上就是我对js的内存机制的理解，如有错误请联系我进行更改！！！！

小礼物走一走，来简书关注我

赞赏支持

