1 python垃圾回收机制主要以引用计数为主，分代回收为辅的方式处理

2 引用计数增加的方式有4种

（1）对象被创建

（2）对象被引用

（3）对象被作为参数传入函数中使用

（4）对象被作为元素，存储在容器中

引用计数减少的方式有4种

（1）当对象的别名被显示销毁

（2）当对象的引用别名被赋予新的对象

（3）当对象离开他的作用域

（4）当元素从存储的容器中删除

引用计数不为0，则不被回收，引用计数为0，则被回收

3 为解决循环引用的情况，需要使用分代回收机制

4 标记-清除机制

（1）将所有容器对象放到一个双向链表中（链表为了方便插入删除），这些对象为0代 （2）循环遍历链表，如果被本链表内的对象引入，自身的被引用数-1，如果被引用数 为0，则触发引用计数回收条件，被回收掉

（3）未被回收的对象，升级为1代

因为循环引用的原因，并且因为你的程序使用了一些比其他对象存在时间更长的对象，从而被分配对象的计数值与被释放对象的计数值之间的差异在逐渐增长。一旦这个差异累计超过某个阈值，则Python的收集机制就启动了，并且触发上边所说到的零代算法，释放“浮动的垃圾”，并且将剩下的对象移动到一代链表。

随着时间的推移，一代链表越来越多，多到触发gc阈值，同样会对一代链表进行标记清除操作，然后将剩下活跃对象升为二代