# 快速自查

1. 什么是闭包？

在函数内部再定义一个函数，并且这个函数用到了外边函数的变量，那么将这个函数以及用到的一些变量称之为闭包。

简单的说，如果在一个内部函数里，对在外部作用域（但不是在全局作用域）的变量进行引用，那么内部函数就被认为是闭包(closure)。来看一个简单的例子:

def addx(x):
 def adder(y):
 return x + y
 return adder
c = addx(8)
type(c) # <type 'function'>
c.\_\_name\_\_ # adder
c(10) # 18

1. 装饰器
2. 迭代器和生成器

可迭代对象：the iterable 生成器：the iterator 迭代器：generator

可迭代对象，如container，通过iter方法后就是迭代器 生成器是一种特殊的迭代器，不占用内存。 生成器通过函数yield，或者生成器表达式

1. 进程和线程

进程 进程是表示资源分配的基本单位，又是调度运行的基本单位。程序并不能单独运行，只有将程序装载到内存中，系统为它分配资源才能运行，而这种执行的程序就称之为进程。程序和进程的区别就在于：程序是指令的集合，它是进程运行的静态描述文本；进程是程序的一次执行活动，属于动态概念。

线程 线程是操作系统能够进行运算调度的最小单位。它被包含在进程之中，是进程中的实际运作单位。一条线程指的是进程中一个单一顺序的控制流，一个进程中可以并发多个线程，每条线程并行执行不同的任务。