**迭代**

**# 使用for循环迭代的对象： 字符串、列表、元组、字典、集合、range  
# 使用for循环遍历取值的过程叫做迭代  
for i, value in enumerate(["苹果", "香蕉"]):  
 print(i, value)  
 # 0  
 # 苹果  
 # 1  
 # 香蕉  
for key, value in {"name":"罗义松", "age": 22}.items():  
 print(key, value)  
 # age 22  
 # name 罗义松  
# 整型,自定义的类不是可迭代对象。  
from collections import Iterable  
# 判断对象是否是可迭代对象  
result = isinstance("name", Iterable)  
print("字符串是否是可迭代对象:", result)  
#字符串是否是可迭代对象: True**

**# 在类里面定义\_\_iter\_\_和\_\_next\_\_的这两个方法表示就是迭代器(Iterator)**

**，迭代器就是记录当前遍历位置及获取下一个位置的值**

**提供\_\_iter\_\_的方法，该对象就是一个可迭代对象**

**iter函数表示： 获取可迭代对象的迭代器  
next函数表示: 获取迭代器中的下一个值**

**生成器(generator)。生成器是一类特殊的迭代器**。**可以使用next函数获取下一个值**

创建一个生成器，只要把一个列表生成式的 [ ] 改成 ( )

 L = [ x\*2 for x in range(5)]是一个列表，而 G = ( x\*2 for x in range(5))

是一个生成器。可以直接打印出列表L的每一个元素，而对于生成器G，按照迭代器的使用方法来使用，通过next()函数、for循环、list()等方法使用。

**协程，又称微线程，纤程。英文名Coroutine。比线程更小的执行单元，也可以称为用户级线程，程序员可以控制顺序**

**标志：有yield没有return**

**在单线程的基础上完成多任务，多任务交替执行**

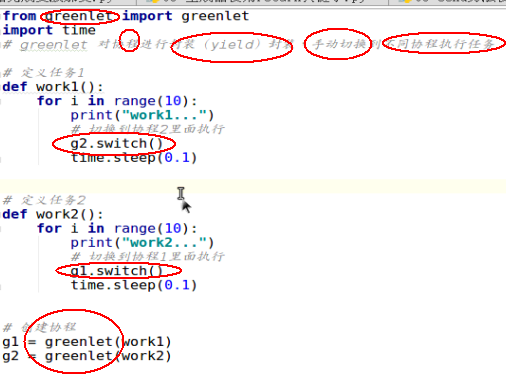
创建一个协程使用4-5K 创建一个线程512k-1M 进程更大

协程：主要应用网络请求和爬虫

**greenlet 对协程进行封装（yield）封装，手动切换到不同协程执行任务**

***gevent是封装的是greenlet，根据耗时操作自动进行协程间切换执行*import gevent  
import time  
from gevent import monkey  
*# 打补丁， 让gevent能够识别系统的耗时操作，提示： 这句代码一定要先执行*monkey.patch\_all()**

**# 创建协程指派任务  
# 第一个参数是函数名  
g1 = gevent.spawn(work1)  
g2 = gevent.spawn(work2)  
# 主线程等待两个协程任务执行完成以后程序在退出  
g1.join()  
g2.join()**



**Gevent**根据耗时自动切换任务

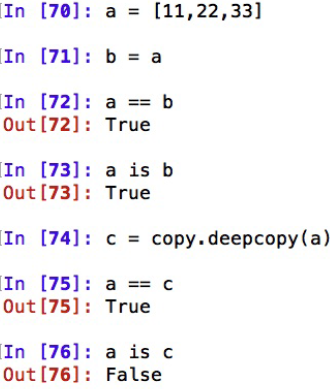
import sys

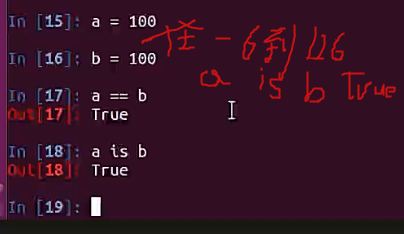
sys.path

sys.path.append(“”) #没有的情况下，添加路径

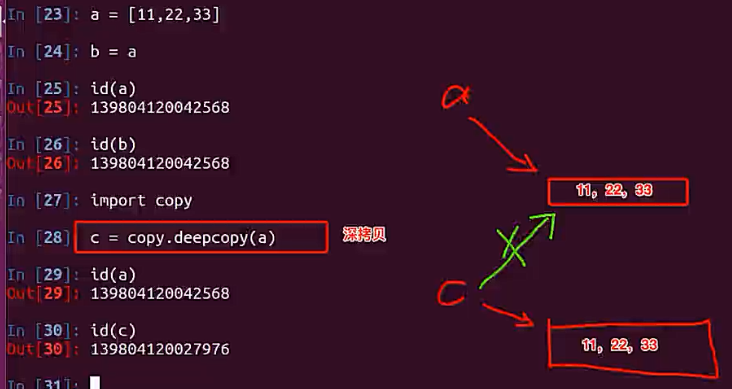
**is** **是⽐较两个引⽤是否指向了同⼀个对象（引⽤⽐较）。**

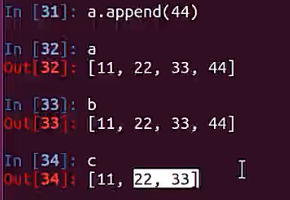
== 是⽐较两个对象是否相等。**(判断内容)**

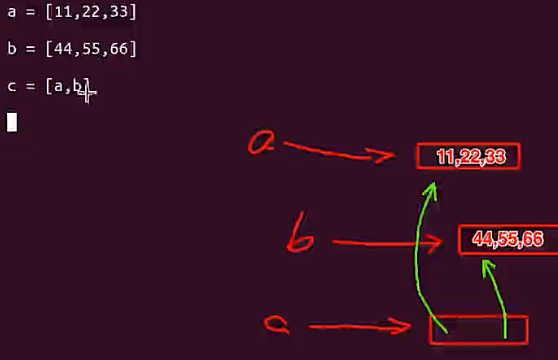
****

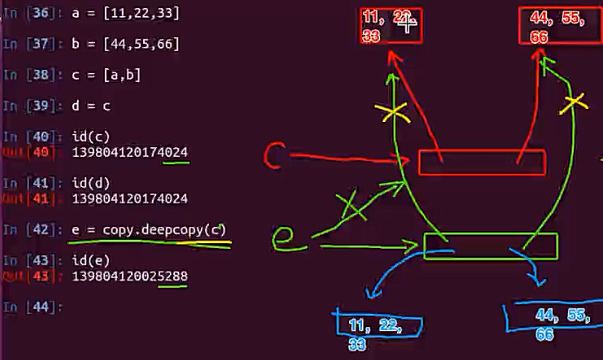
****

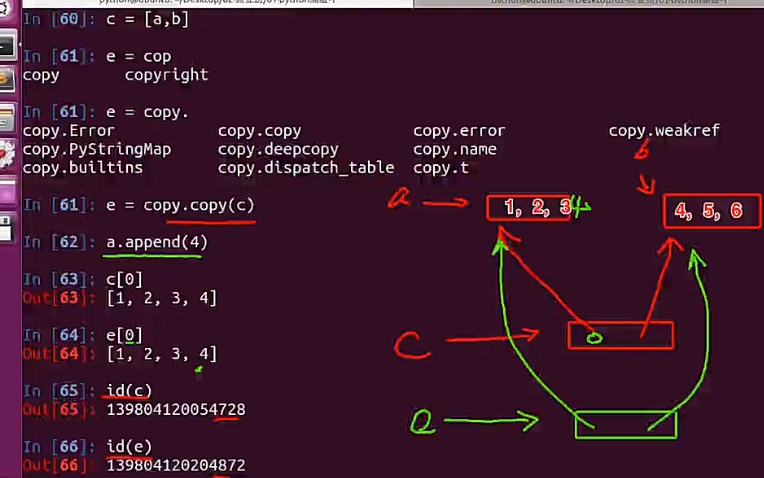












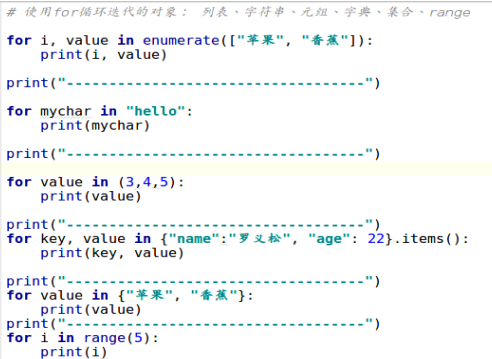
**迭代器**

**迭代是访问集合元素的一种方式。迭代器是一个可以记住遍历的位置的对象。迭代器对象从集合的第一个元素开始访问，直到所有的元素被访问完结束。迭代器只能往前不会后退。**

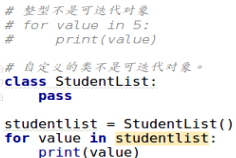
**使用for可迭代的对象：字符串/列表/元组/字典/集合/range**

# 使用for循环遍历取值的过程叫做迭代

#整数和自定义的类不可迭代







除了for循环能接收可迭代对象，list、tuple等也能接收。

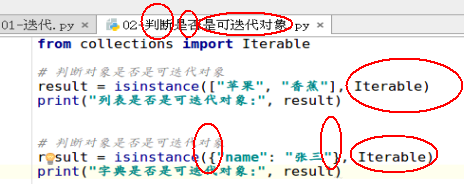
li = list(FibIterator(15))

print(li)

tp = tuple(FibIterator(6))

print(tp)

递归数列最多998次 迭代器没有上限控制



from collections import Iterable

自定义可迭代对象



