x1.匿名函数，进程、线程，

（1）匿名函数

什么是匿名函数：lambda函数，不用担心函数名冲突。

mylamb = lambda x, y : x+y # 声明一个lambda表达式

result = mylamb(2, 3) # 调用函数

print(result)

（2）进程、线程、协程的区别看github

①区别

一个程序至少有一个进程，一个进程至少有一个线程。

线程不能够独立执行，必须依存于进程中。

进程在执行过程中拥有独立的内存单元，而多个线程共享内存，从而极大地提高了程序的运行线路。

②优缺点

线程执行开销小，但不利于资源的管理和保护；而进程正相反。

**进程：**①操作系统进行资源分配和调度的基本单位，多个进程之间相互独立

②稳定性好，如果一个进程崩溃，不影响其他进程，但是进程消耗资源大，开启的进程数量有限制

**线程：**①CPU进行资源分配和调度的基本位，线程是进程的一部分，是比进程更小的能独立运行的基本单位，一个进程下的多个线程可以共享该进程的所有资源

协程（自带CPU上下文）是python中一种实现多任务的方式，比线程更小且占用更小执行单元 。把一个协程切换到另一个协程，只要这个过程中保存或恢复 CPU上下文，那么程序还是可以运行的。

（3）对if \_\_name\_\_ == ‘\_\_main\_\_’:的理解

\_\_name\_\_作为模块的内置属性，即‘.py’文件的调用方式。如果等于”\_\_main\_\_”就直接执行本文件。其它的则作为模块调用。

（4）什么是socket

socket是在应用层和传输层之间的一个抽象层，它把TCP/IP层复杂的操作抽象为几个简单的接口供应用层调用以实现进程在网络中通信。

（5）read()与readlines()

f = open('yes', 'r', encoding='utf-8')  
# res = f.read()  
# print(res)  
# print(type(res)) # read()返回的是字符串  
res1 = f.readlines()  
print(type(res1)) # readlines()返回的是列表

<class 'str'>

<class 'list'>

（6）python怎么合并两个列表

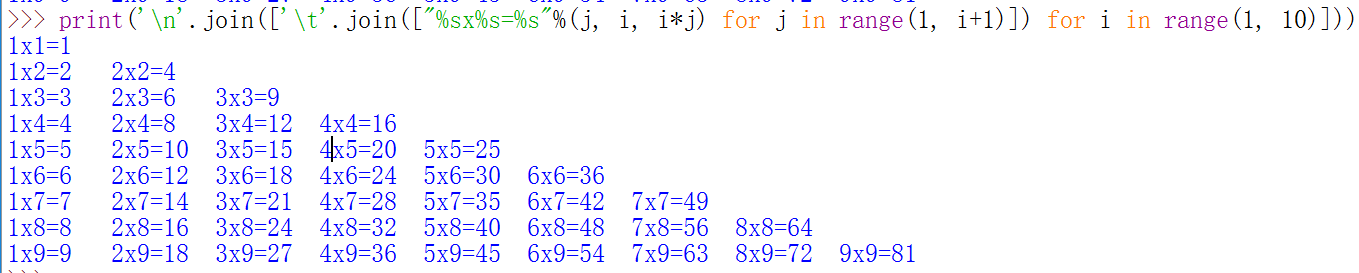
①append() 向列表尾部追加一个新元素，列表只占一个索引位，在原有列表上增加。  
②extend() 向列表尾部追加一个列表，将列表中的每个元素都追加进来，在原有列表上增加。  
③+ 直接用+号看上去与用extend()一样的效果，但是实际上是生成了一个新的列表存这两个列表的和，只能用在两个列表相加上。

# a = [1, 2, 3]  
# b = [5, 6, 7]  
# a.extend(b) # 输出结果为：[1, 2, 3, 5, 6, 7]  
# a = a + b # 输出结果为：[1, 2, 3, 5, 6, 7]  
# a.append(b) # 输出结果为：[1, 2, 3, [5, 6, 7]]  
# print(a)  
  
list1 = [3, 9, 5]  
list2 = [1, 2, 3]  
new\_list = []  
for item in list1:  
 new\_list.append(item)  
for item in list2:  
 new\_list.append(item)  
print(new\_list)

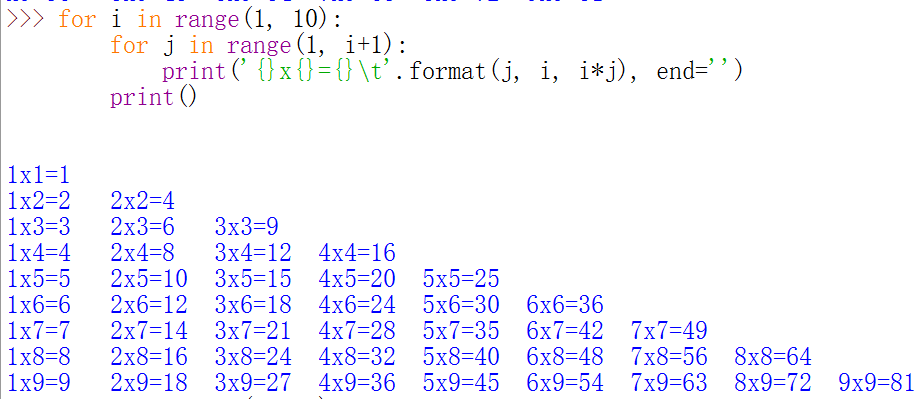
输出结果为：[3, 9, 5, 1, 2, 3]

2.编程练习

（1）实现九九乘法表



print('\n'.join(['\t'.join(["%sx%s=%s"%(j, i, i\*j) for j in range(1, i+1)]) for i in range(1, 10)]))



3.django中间件

4.csrf跨站请求伪造

5.django认证

6.redis垃圾回收机制

7.多线程、多进程、协程