**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*错误\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***



**自定义报错：try-except语句【在最容易出错的地方、用户交互处】**

**try :**

**指令**

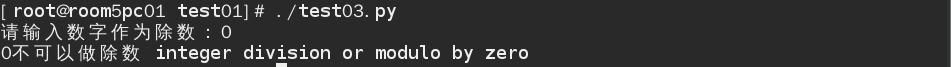
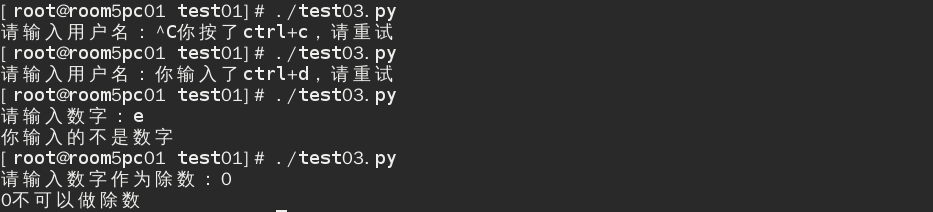
**except 异常原因：**

**Print “提示”**

**except 异常原因：**

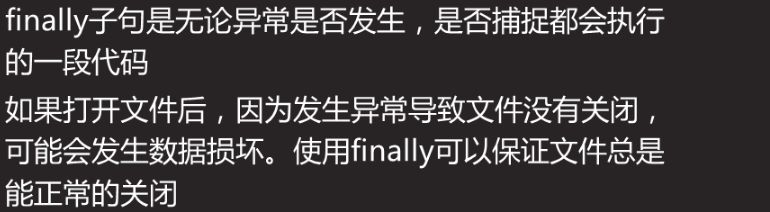
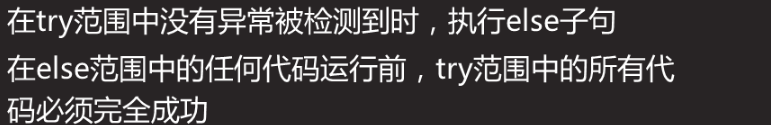
**Print “提示”**

**...**



**#把自带报错存入变量，打印自带报错信息**

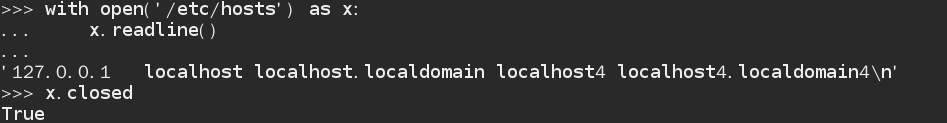
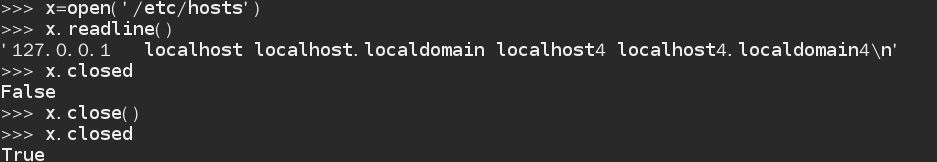
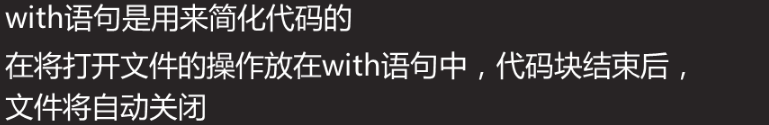
**try支持else、finally子句**



**#出不出错都执行**

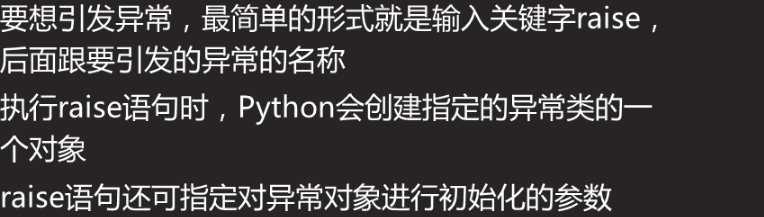
**#不出错则执行**

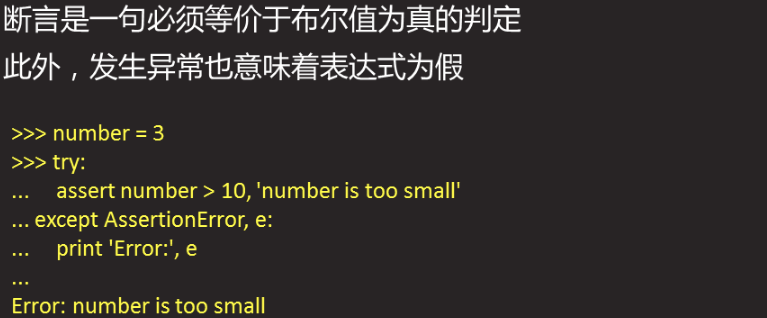
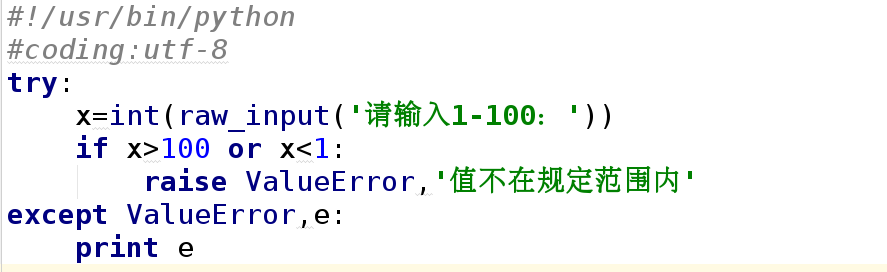
**with语句**



**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*触发异常\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**raise语句**



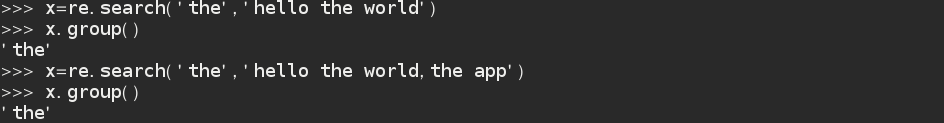
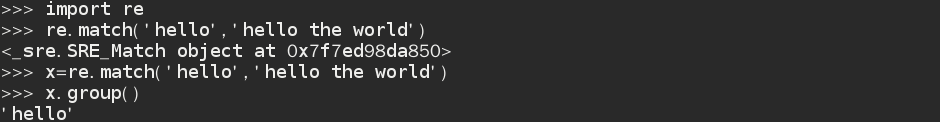


**#识别错误，提示错误信息**

**#制作错误，没错也报错**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*python正则\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**re模块：regular expression**

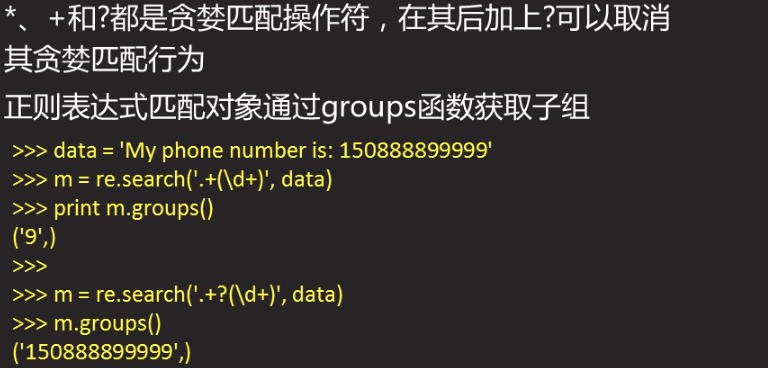
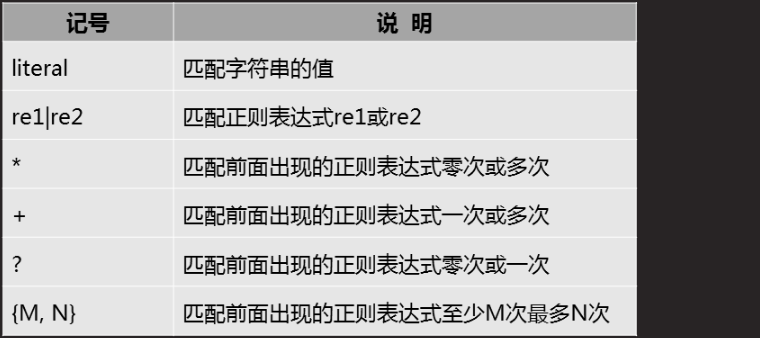
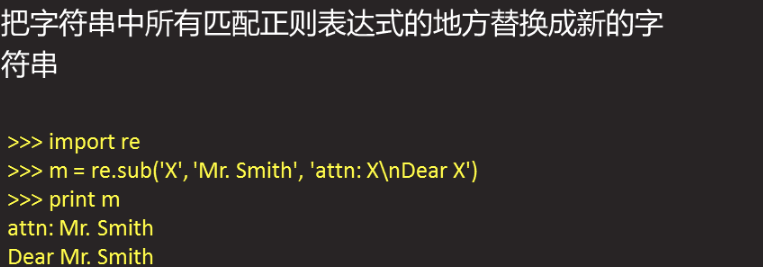
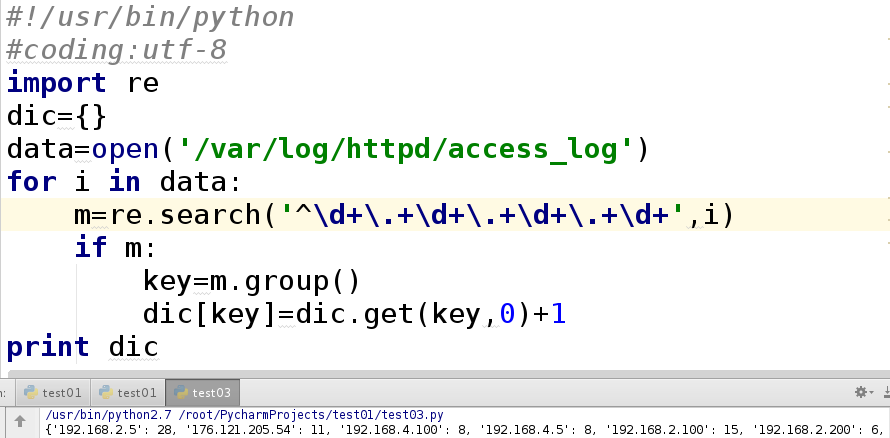
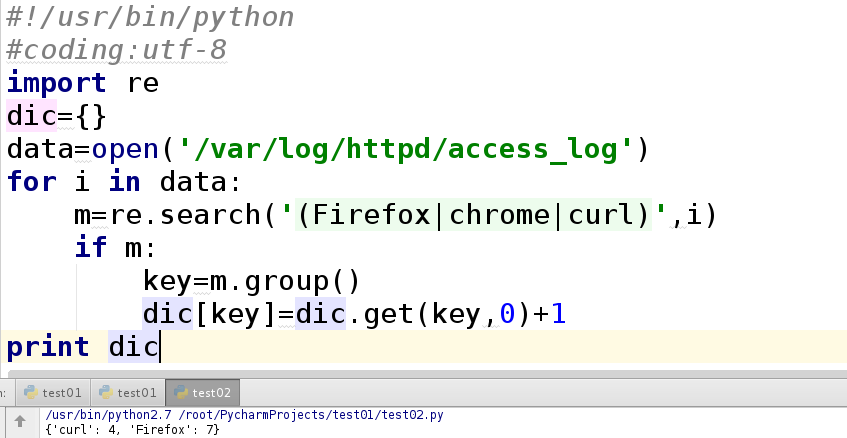
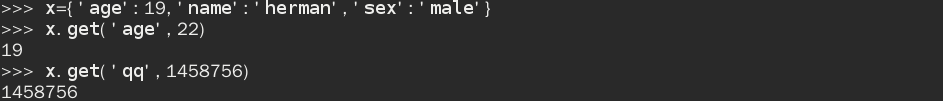
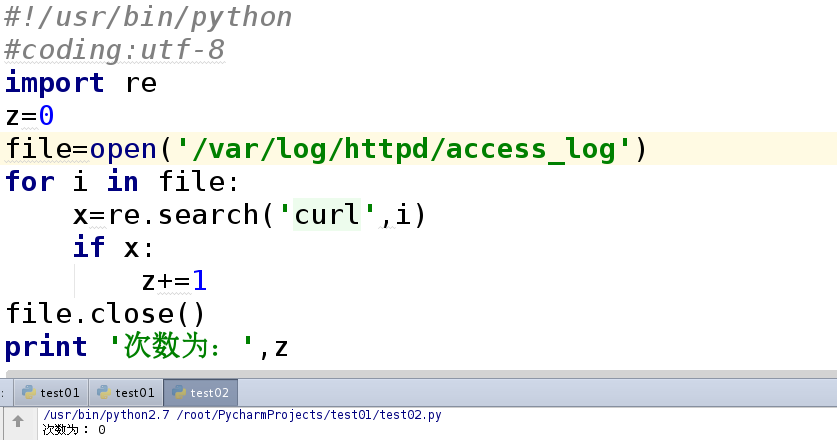
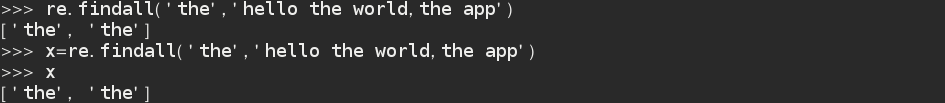


**re.match(‘key’,’内容’)**

**Search正则匹配，匹配全部位置，仅匹配第一个**

**re.group 查看**

**match正则匹配，仅匹配开头**



**findall正则匹配，匹配全部位置的全部内容，以列表形式直接显示**

**#等同于字典中的格式{‘key’:次数……}，添加key并且次数+1**

**#赋匹配的值**

**#创建一个空字典**

**x.get,如果字典中有，则直接显示，没有的话会把尝试的值显示**

**匹配一组字符**

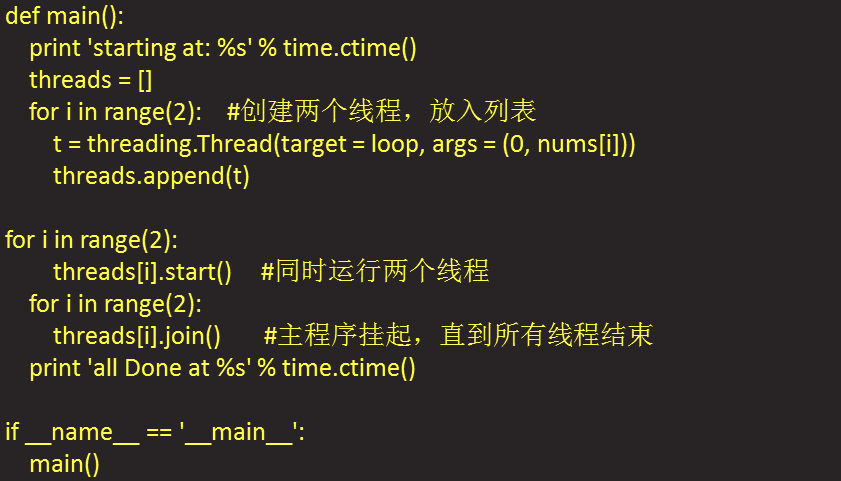
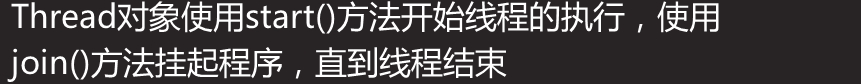
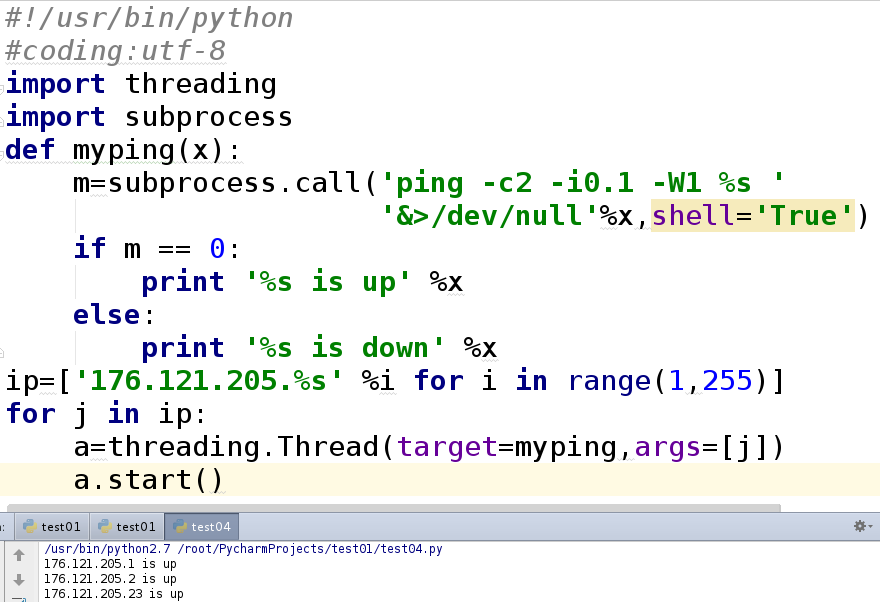
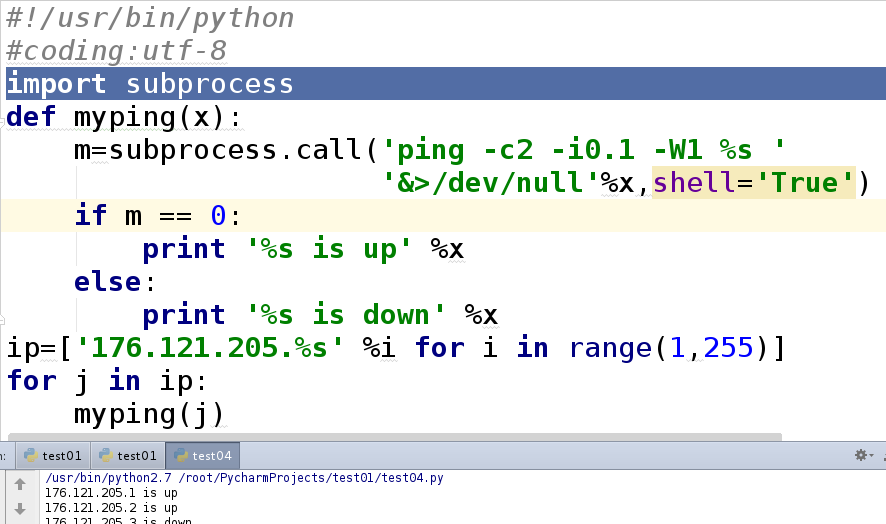
**匹配单个字符**

**Split方法:切割**

**sub方法：替换**

**贪婪匹配**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*多线程\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***



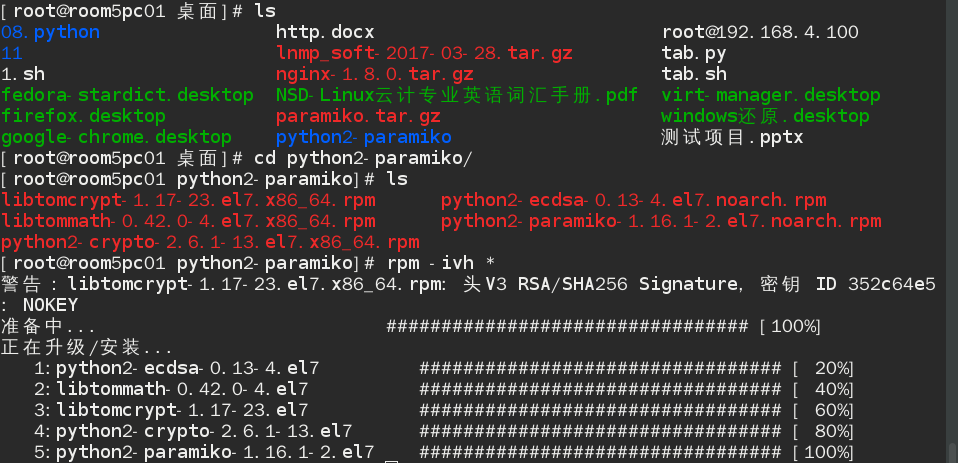
**多进程，每个进程都有自己的内存，较稳定**

**多进程方式测主机存活**

**多线程，占用同一个内存，速度快，稳定性较差**

**多线程方式测主机存活**

**Paramiko模块【远程】**



*#!/usr/bin/python  
#coding:utf-8***import** paramiko  
host=**'192.168.4.100'**ssh=paramiko.SSHClient()  
ssh.set\_missing\_host\_key\_policy(paramiko.AutoAddPolicy())  
ssh.connect(host,username=**'root'**,password=**'123456'**)  
ssh.exec\_command(**'yum -y install httpd'**)

