吃井不忘挖水人，当然是先贴主要学习网站的网址啦

【学习网址：廖雪峰】

<https://www.liaoxuefeng.com/wiki/0014316089557264a6b348958f449949df42a6d3a2e542c000/0014316090478912dab2a3a9e8f4ed49d28854b292f85bb000>

特点：解释型语言、跨平台

优势：简单（除高级部分外）、有丰富的库（站在巨人的肩膀上）

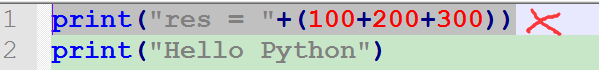
劣势：运行效率低、不能加密

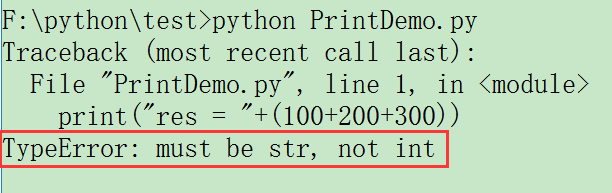
应用场景：网络应用、脚本

版本：3.6.x（向下2.x不兼容）

python官网：<https://www.python.org/>

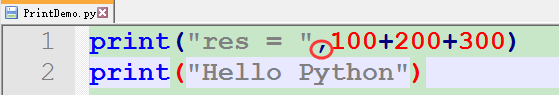
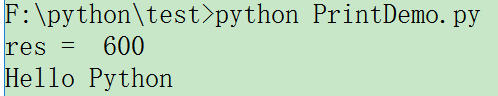
下载速度是真的慢，50-60k/s

Q: 



字符串拼接不行吗？

解决方案：英文逗号



编码格式设置为 UTF-8

#后面是注释部分，单行注释

大小写敏感

使用空格或Tab为缩进，而不是{}



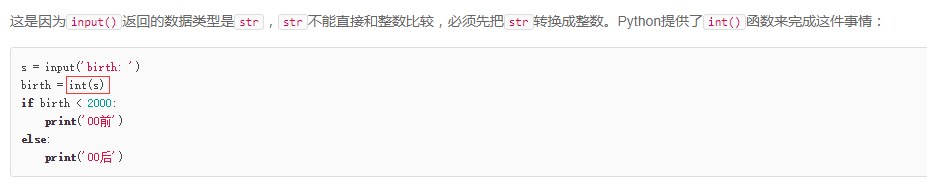
:结尾的时候，下面缩进的代码是{语句块}

Q:



num是字符串类型的……

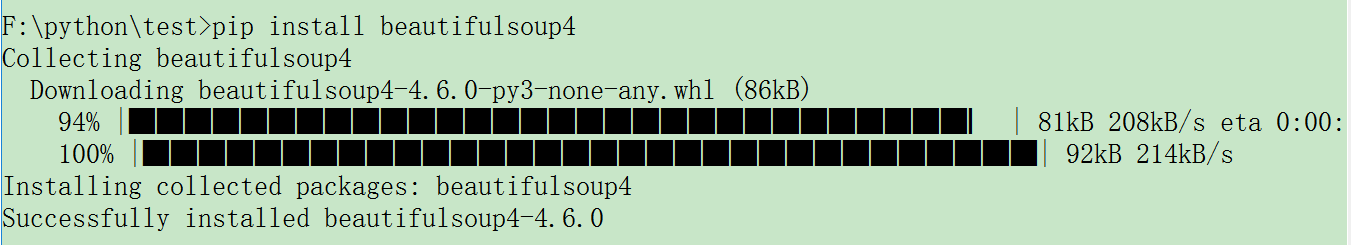
A：

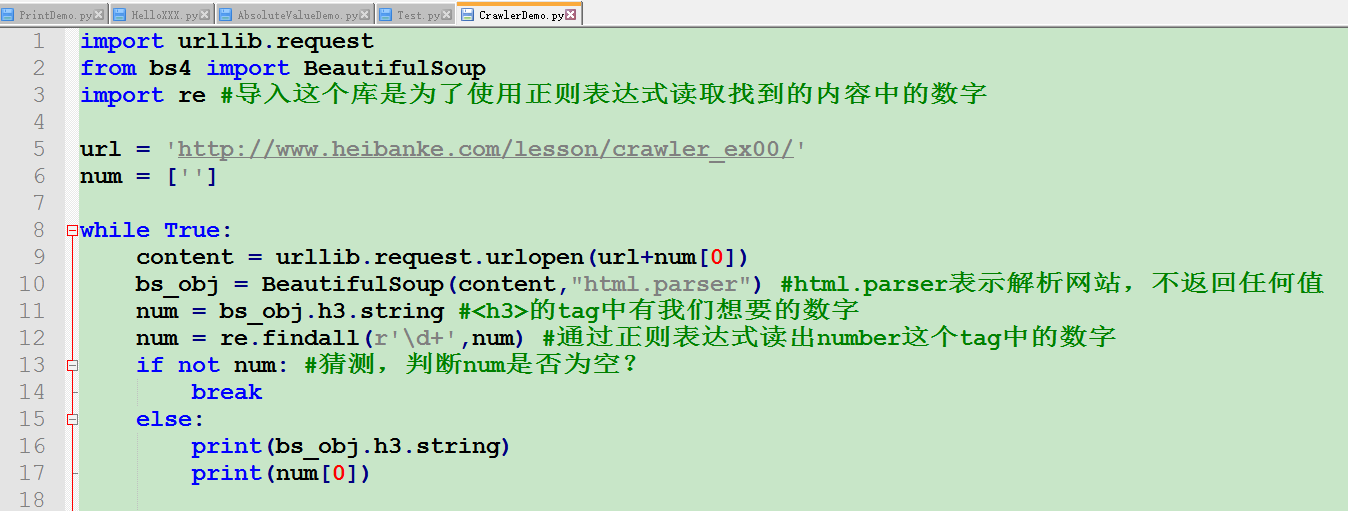


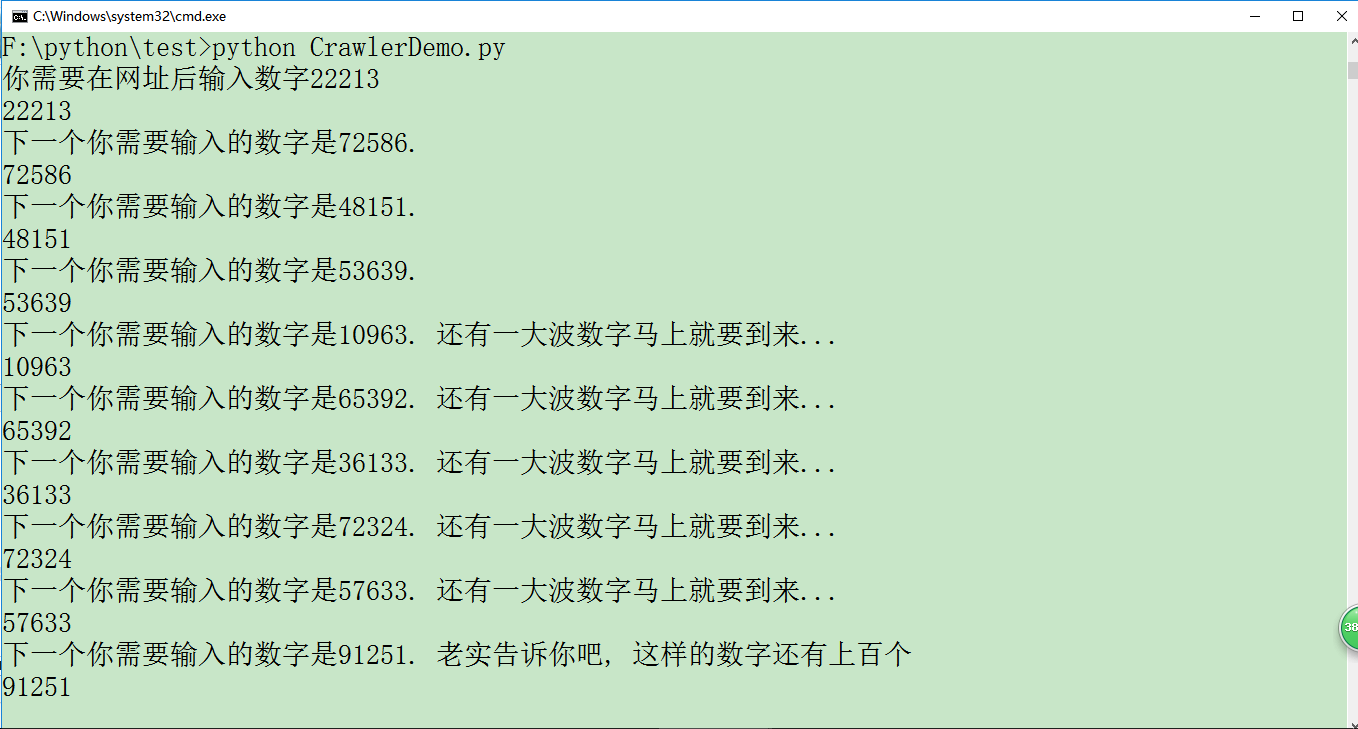
Python的整数类型是无限大的

爬虫初探

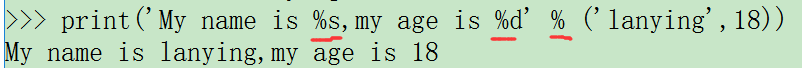
<https://www.jianshu.com/p/33f230c91bd7> （网站里的部分内容是2.x版本的，需要调整）

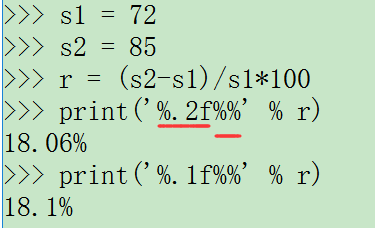






## 字符串和编码





## list和tuple

相同点：都是有序集合（存入和取出顺序一致）

不同点：

1. list可变，tuple不可变
2. list 是 [] 方括号

tuple 是 () 圆括号

<https://www.liaoxuefeng.com/wiki/0014316089557264a6b348958f449949df42a6d3a2e542c000/0014316724772904521142196b74a3f8abf93d8e97c6ee6000>

list 可变集合

>>> classmates = ['Michael', 'Bob', 'Tracy']

>>> classmates[-1] #倒数第一个元素

>>> classmates.append('Adam') #追加

>>> classmates.insert(1, 'Jack') #插入

>>> classmates.pop() #删除末尾元素

>>> classmates.pop(1) #删除指定下标元素

>>> L = ['Apple', 123, True] #集合中的元素类型可以不一致

>>> L = [] #长度为0的集合 ，此时len(L) == 0

>>> s = ['python', 'java', ['asp', 'php'], 'scheme'] #集合中存放集合

tuple 元组，只读集合（final修饰的集合，集合中元素(引用变量)指向的数据仍可改变）

>>> classmates = ('Michael', 'Bob', 'Tracy')

>>> t = (1,) #逗号 表示此处的小括号是**tuple**，而不是**运算符**



## dict

>>> d = {'Michael': 95, 'Bob': 75, 'Tracy': 85}

>>> d['Adam'] = 67

>>> 'Thomas' **in** d #判断字典d中是否有‘Thomas’这个Key

>>> d.get('Thomas', -1) #从d中查找Key‘Thomas’对应的值，找不到则-1

>>> d.pop('Bob') #删除键为‘Bob’的Key-Value

请务必注意，dict内部存放的顺序和key放入的顺序是没有关系的。

# 函数

【python函数官方文档】<https://docs.python.org/3/library/functions.html#abs>

## 调用函数

>>> abs(100)

>>> max(2, 3, 1, -5)

>>> int('123')

>>> int(12.34)

12

>>> float('12.34')

12.34

>>> str(1.23)

'1.23'

>>> str(100)

'100'

>>> bool(1)

True

>>> bool('')

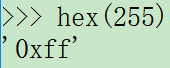
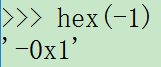
False

>>> a = abs *# 变量a指向abs函数*

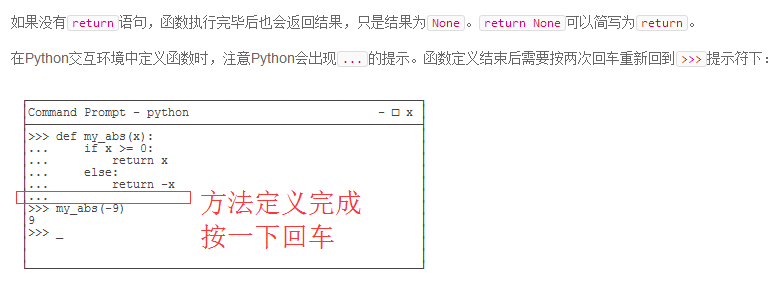
>>> a(-1) *# 所以也可以通过a调用abs函数*

1

hex(xxx)函数将整数转为十六进制数

 好调皮

## 定义函数



在Python交互模式中，可以通过

from 文件名 import 函数名

导入函数

### 空函数

**def** **nop**():

**pass**

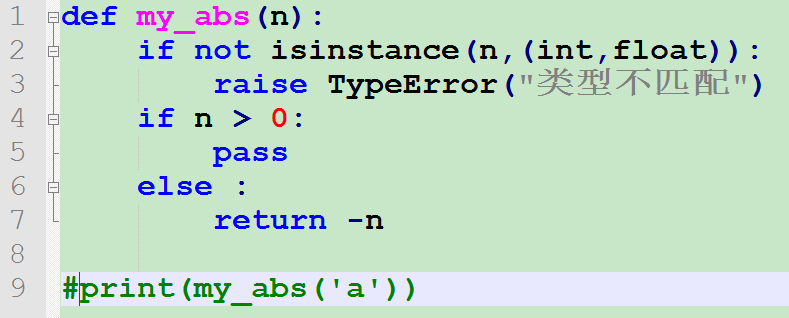
pass用来**占位**，什么都不做（如果什么都不写，会报错）

也可以用于其它语句

**if** age >= 18:

**pass**

数据类型检查可以用内置函数isinstance()实现



可以return多个对象？！

其实返回的是一个tuple （元组）

## 函数的参数

位置参数：根据第n个参数的位置决定是哪个变量

**def** **power**(x, n):

s = 1

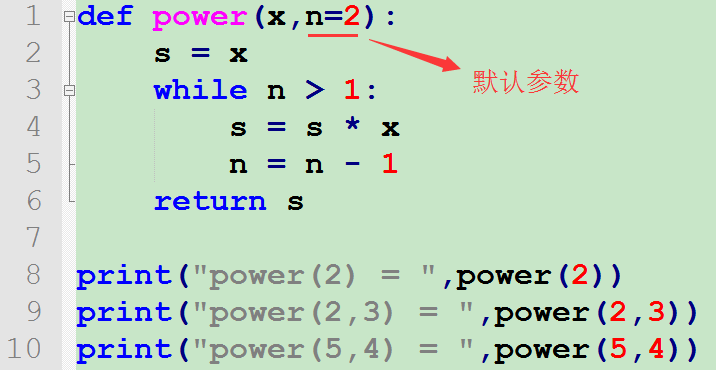
**while** n > 0:

n = n - 1

s = s \* x

**return** s

默认参数



多个默认参数

**def** **enroll**(name, gender, age=6, city='Beijing'):

print('name:', name)

print('gender:', gender)

print('age:', age)

print('city:', city)

给指定的某个参数赋值：

enroll('Bob', 'M', 7) # city为默认值‘Beijing’

enroll('Adam', 'M', city='Tianjin') # age为默认值6

将默认值设置为不可变对象None以解决“记住”上一次的数据的问题。

**def** **add\_end**(L=None):

**if** L **is** None:

L = []

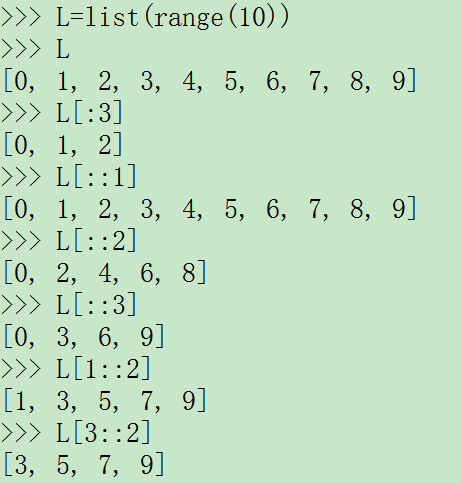
L.append('END')

**return** L

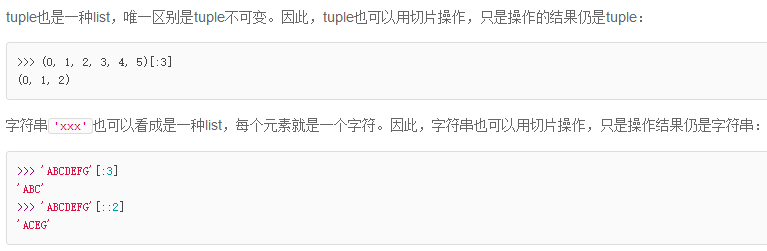
# 高级特性

## 切片（Slice）

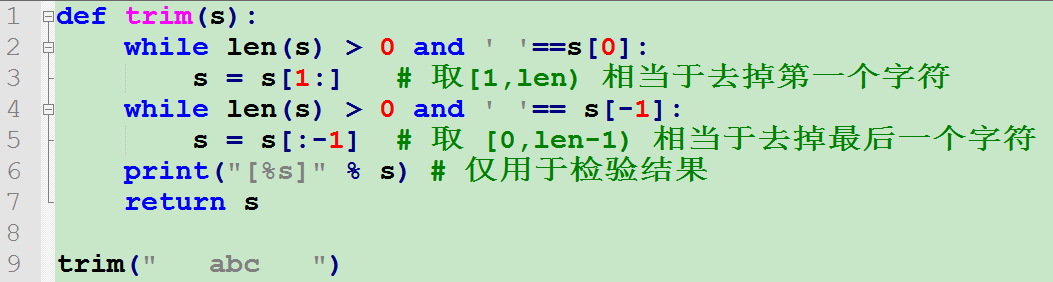
作用：取指定索引范围的数据（左闭右开）



list,tuple,字符串均可以使用 *切片*



【案例】trim()方法

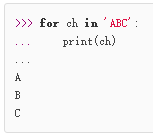


## 迭代

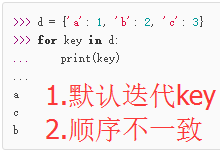
形式：for … in … :

应用：list , tuple , 字符串 , dict

### 迭代字符串如下：



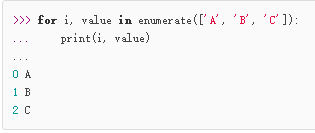
### 迭代dict如下：



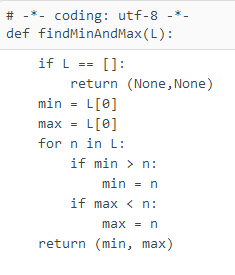
### 如果要迭代value，可以用for value in d.values();

### 如果要同时迭代key和value，可以用for k, v in d.items()

### list --> 索引-元素对

如果要对list实现类似Java那样的下标循环怎么办？Python内置的enumerate函数可以把一个list变成索引-元素对，这样就可以在for循环中同时迭代索引和元素本身：

【案例】请使用迭代查找一个list中最小和最大值，并返回一个tuple



## 列表生成式

