<http://www.runoob.com/python3/python3-list.html>

# 基础语法

## 标识符（变量声明）

* 第一个字符必须是字母表中字母或下划线 **\_** 。
* 标识符的其他的部分由字母、数字和下划线组成。
* 标识符对大小写敏感。

## python关键字

我们不能把它们用作任何标识符名称。Python 的标准库提供了一个 keyword 模块，可以输出当前版本的所有关键字

## 注释

单行注释以 # 开头

多行注释可以用多个 **#** 号，还有 **'''** 和 **"""**：

## 数字(Number)

python中数字有四种类型：整数、布尔型、浮点数和复数。

* **int** (整数), 如 1, 只有一种整数类型 int，表示为长整型，没有 python2 中的 Long。
* **bool** (布尔), 如 True。
* **float** (浮点数), 如 1.23、3E-2
* **complex** (复数), 如 1 + 2j、 1.1 + 2.2j

## 字符串(String)

* python中单引号和双引号使用完全相同。
* 使用三引号('''或""")可以指定一个多行字符串。
* 转义符 '\'
* 反斜杠可以用来转义，使用r可以让反斜杠不发生转义。。 如 r"this is a line with \n" 则\n会显示，并不是换行。
* 按字面意义级联字符串，如"this " "is " "string"会被自动转换为this is string。
* 字符串可以用 + 运算符连接在一起，用 \* 运算符重复。
* Python 中的字符串有两种索引方式，从左往右以 0 开始，从右往左以 -1 开始。
* *Python中的字符串不能改变。*

***猜测与 Java String类相似***

* *Python 没有单独的字符类型，一个字符就是长度为 1 的字符串。*
* 字符串的截取的语法格式如下：**变量[头下标:尾下标]**

## 等待用户输入

input("\n\n按下 enter 键后退出。")

input() 函数返回的是字符串

n = input("输入一个正整数n:")

print( n )

for i in range(1,*int(n)*) :

print( i )

#若无 int(n) 将出错

## 输出

不换行输出：print( “”,end=”” )

## import 与 from...import

在 python 用 **import** 或者 **from...import** 来导入相应的模块。

将整个模块(somemodule)导入，格式为： **import somemodule**

从某个模块中导入某个函数,格式为： **from somemodule import somefunction**

从某个模块中导入多个函数,格式为： **from somemodule import firstfunc, secondfunc, thirdfunc**

将某个模块中的全部函数导入，格式为： **from somemodule import \***

## 循环

### while

while 判断条件：

语句

#### while 循环使用 else 语句

count = 0

while count < 5:

print (count, " 小于 5")

count = count + 1

else:

print (count, " 大于或等于 5")

### For:

for <variable> in <sequence>:

<statements>

else:

<statements>

#### range()函数

>>>for i in range(5): ...

print(i)

...

0

1

2

3

4

## 方法声明Def：

Expamle:

#打印列表

#如果列表为 range()函数构成 打印结果为 range(...）

def print\_list( \_list\_ ) :

for i in \_list\_ :

print( i ,end=' ' )

print() #换行

关于 形参 我也不知道为什么python喜欢这种风格 \_list\_

# 序列

## 列表

列表是最常用的Python数据类型，它可以作为一个方括号内的逗号分隔值出现。

列表的数据项不需要具有相同的类型

使用range创建列表

range( n )

range( start , end )

range( start , end , step )

参数均为number类型

list = range(1,int(n))

# 元组

# **[\_\_new\_\_和\_\_init\_\_的区别](https://zhuanlan.zhihu.com/p/35943253" \t "https://www.zhihu.com/people/hou-pu-yuan/_blank)**

https://zhuanlan.zhihu.com/p/35943253

## \_\_init\_\_的用法

Example

class Person(object):

def \_\_init\_\_(self, name, age):

print("in \_\_init\_\_")

self.\_name = name

self.\_age = age

p = Person("Wang", 33)

执行效果：

in \_\_init\_\_

<\_\_main\_\_.Person object at 0x7fb2e0936450>

实际上，\_\_init\_\_函数并不是真正意义上的构造函数，\_\_init\_\_方法做的事情是在对象创建好之后初始化变量。真正创建实例的是\_\_new\_\_方法。

## Singleton（单例模式）

class Singleton(object):

\_instance = None

def \_\_new\_\_(cls, \*args, \*\*kwargs):

if cls.\_instance is None:

cls.\_instance = object.\_\_new\_\_(cls, \*args, \*\*kwargs)

return cls.\_instance s1 = Singleton()

s2 = Singleton()

print(s1)

print(s2)

# Range

TypeError: ‘range’ object does not support item assignment

Range()函数返回的是Range Object 并非 List 对象

用list( Range range ) 将range 对象转化为 list对象