



链滴

Python 语句进阶 学习笔记

作者: [angels](#)

原文链接: <https://ld246.com/article/1502428919342>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

序列解包(可选解包)

- 多个赋值操作同时进行（注意，解包的序列中的元素必须和赋值号左边的变量数量一致）

```
x,y,z = (1,2,3)
print(x) : 1
print(y) : 2
print(z) : 3
```

```
x = 1;
y = 2;
x,y = y,x;
print(x) : 2
print(y) : 1
```

```
values = 1,2,3
print(values) : (1,2,3)
x,y,z = values;
print(x) : 1
print(y) : 2
print(z) : 3
```

链式赋值

- 将同一个值给多个变量

```
x=y=somefunction();
等价于： y = somefuncation(); x = y
不等价： x = somefuncation();y=somefuncation()
```

语句块

- 语句块并非一种语句
- 语句块是一组语言，在代码前放置空格来缩进语句即可创建语句块
- 在python中用：来标识语句块的开始

```
if True : print("hello");
```

条件

- 描述

```
if 条件:
    语句块
elif:
    语句块
```

else:
 语句块

elif 表示 else if

and 表示且

or 表示 或

while 条件:
 语句块

for something in XXXX:

 语句块

- 即表示对XXXX中的每一个元素，执行某些语句块，XXXX可以是列表，字典，元组，迭代器等等。

断言 assert

- 后面语句为真，否则出现 AssertionError
- 用来检查一个条件，如果它为真，就不做任何事。如果它为假，则会抛出AssertionError并且包含错误信息。

pass

- pass 表示这里什么都没有，不执行任何操作

如果你的程序还有未完成的函数和类等，你可以先添加一些注释，然后代码部分仅仅写一个 pass，样程序可以运行不会报错，而后期你可以继续完善你的程序

del

- del 删除的只是引用和名称，并不删除值，也就是说，Python 会自动管理内存，负责内存的回收这也是 Python 运行效率较低的一个原因吧

range语句 range(start,end)

- Range函数的工作方式类似于分片。
- 包含start，不包含end。
- 默认start = 0;

```
for i in range(10):  
    print(i)  
0
```

```
1
2
3
4
5
6
7
8
9
```

- xrange类似于range，range是一次性创建整个序列，xrange一次性创建一个，迭代大量数据时首xrange

zip：压缩函数，将两个序列压缩成一个

```
x= ['a','b'];
y = ['1','2','3'];
z = zip(x,y)
for v in z:
    print(v)
('a', '1')
('b', '2')
```

可以看到，z只存最短的序列的数量值

编号迭代 enumerate

```
s1 = "hello"
for index , s in enumerate(s1):
    print("index is %d and value is %s"%(index,s))
index is 0 and value is h
index is 1 and value is e
index is 2 and value is l
index is 3 and value is l
index is 4 and value is o
```

exec :动态地执行语句中的代码

```
exec("print('hello')")
hello
```

eval 计算python表达式