



链滴

Python 实战（二） | 基础语法

作者: [JavaFish](#)

原文链接: <https://ld246.com/article/1624087678947>

来源网站: [链滴](#)

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



01 前言

我是狗哥，一名程序猿。做过 Android、撸过 Java、目前在自学 Python。注册「一个优秀的废人」这个公号已有些日子，真正有心将它运营起来是这两天萌生的想法。注册这个号的初衷是分享我的 Python 学习笔记。一个知识，你自己懂，不算是真的懂，你能把他人说懂，才是真正掌握。分享，是最好的复习过程。

02 编码

默认情况下，Python 3 源码文件以 UTF-8 编码，所有字符串都是 unicode 字符串。当然你也可以源码文件指定不同的编码：

```
# -*- coding: cp-1252 -*-
```

上述定义允许在源文件中使用 Windows-1252 字符集中的字符编码，对应适合语言为保加利亚语、白罗斯语、马其顿语、俄语、塞尔维亚语。

03 标识符

Python 3 的标识符与 Java 中的一样：

首字母必须是字母表中的字母或者下划线

其他部分由字母，数字，下划线组成且对大小写敏感

04 关键字

Python 3 中的 keyword 模块，存储了 Python 3 中的所有关键字：

```
import keyword
```

```
print(keyword.kwlist)
['False', 'None', 'True', 'and', 'as', 'assert', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']
```

05 注释

Python 3 与 Java 的注释大同小异。

Python 3 中的单行注释以 # 开头 如:

```
print("Hello world!") # 输出语句, 输出 Hello world!
```

Python 3 中的单行注释以 """ 开头, """结尾 如:

```
"""
下面是输出语句
输出Hello world!
"""
print("Hello world!")
```

####5、行与缩进

Python 使用缩进来表示代码块, 不使用大括号, 缩进的空格数可变 (一般缩进 4 个空格), 但必须含相同的缩进空格数。正确格式如:

```
if True:
    print ("True")
else:
    print("False")
注意: 缩进的空格数与前面不一致, 会出错:
if True:
    print ("Answer")
    print ("True")
else:
    print ("Answer")
    print ("False") # 缩进不一致, 会导致运行错误
```

以上程序由于缩进不一致, 执行后会出现类似以下错误:

```
IndentationError: unindent does not match any outer indentation level
```

06 多行语句

Python 通常是一行写完一条语句, 但如果语句很长, 我们可以使用反斜杠()实现多行语句, 例如:

```
sum = x + \
      y + \
      z
```

但在 [], {}, 或 () 中的多行语句, 不需要使用反斜杠(\), 例如:

```
sum = ['x','y','z',
      'a','b']
```

07 数字(Number)类型

Python 3 中数字有四种类型：整数、布尔型、浮点数和复数。

int (整数), 如 10, 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

08 字符串 (String)

Python 3 中的单双引号使用完全相同

使用三引号 (""或"") 可指定一个多行字符串。

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, 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 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]

实例如下:

```
str='Nasus'
```

```
print(str)           # 输出字符串
print(str[0:-1])    # 输出第一个到倒数第二个的所有字符
print(str[0])       # 输出字符串第一个字符
print(str[2:4])     # 输出从第三个开始到第五个的字符
print(str[2:])      # 输出从第三个开始的后的所有字符
print(str * 2)      # 输出字符串两次
print(str + '你好') # 连接字符串
```

```
print('-----')
```

```
print('hello\nnasus') # 使用反斜杠(\)+n转义特殊字符
print(r'hello\nnasus') # 在字符串前面添加一个 r, 表示原始字符串, 不会发生转义
```

输出结果为:

```
Nasus
Nasu
N
sus
sus
NasusNasus
Nasus你好
```

```
-----  
hello  
nasus  
hello\nnasus
```

09 同一行显示多条语句

Python 可以在同一行中使用多条语句，语句之间使用分号(;)分割，以下是一个简单的实例：

```
import sys; x = 'nasus'; sys.stdout.write(x + '\n')
```

输出结果为：

```
nasus
```

10 多个语句构成代码组

缩进相同的一组语句构成一个代码块，我们称之为代码组。

像if、while、def和class这样的复合语句，首行以关键字开始，以冒号(:)结束，该行之后的一行或几行代码构成代码组。

将首行及后面的代码组称为一个子句(clause)。

如：

```
if expression :  
    suite  
elif expression :  
    suite  
else :  
    suite
```

11 Print 输出

print 默认输出是换行的，如果要实现不换行需要在变量末尾加上 end=""：

```
x="Hello"  
y="world"  
# 换行输出  
print( x )  
print( y )  
  
print('-----')  
# 不换行输出  
print( x, end=" " )  
print( y, end=" " )  
print()
```

以上实例执行结果为：

```
Hello  
world  
-----
```

Hello world

12 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 *
```

```
# 导入 sys 模块
```

```
import sys
```

```
print('====Python import mode=====');
```

```
print('命令行参数为:')
```

```
for i in sys.argv:
```

```
    print(i)
```

```
print('\n python 路径为',sys.path)
```

```
# 导入 sys 模块的 argv,path 成员
```

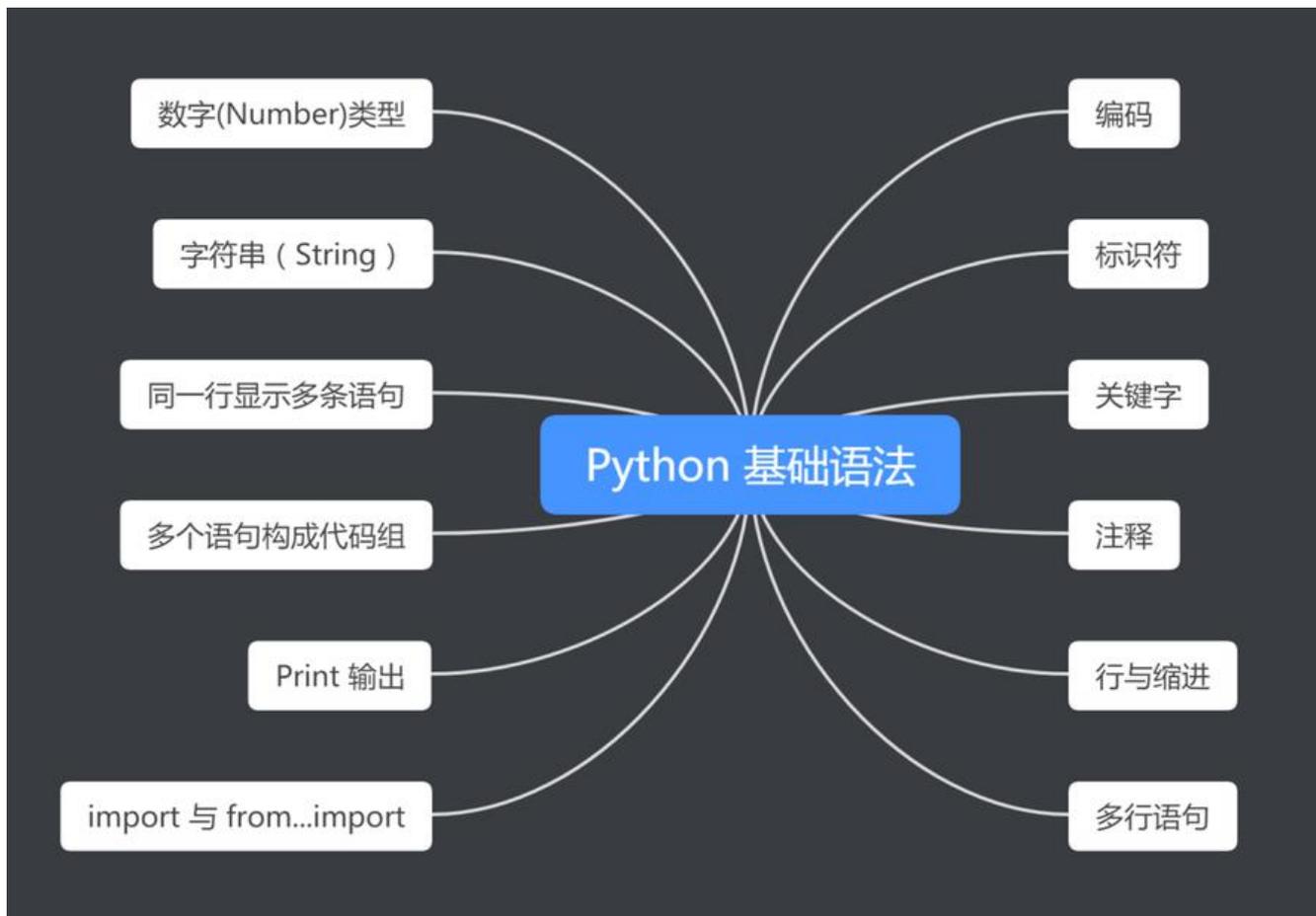
```
from sys import argv,path # 导入特定的成员
```

```
print('====python from import=====')
```

```
print('path:',path) # 因为已经导入 path 成员, 所以此处引用时不需要加 sys.path
```

```
####后语
```

按照惯例送上思维导图帮助你们理解并记忆, 但最重要的还是多动手自己实践, 虽然看上去很简单, 是你自己敲一遍可能会出现各种各样的问题。学习, 千万不能有所见即所得的想法, 实践才是检验真的唯一标准。



好啦，本节内容就到这里，下节内容：「自学 Python 之运算符」

13 大厂面试题 & 电子书

如果看到这里，喜欢这篇文章的话，请帮点个好看。

初次见面，也不知道送你们啥。干脆就送几百本电子书和2021最新面试资料吧。微信搜索JavaFish复电子书送你 1000+ 本编程电子书；回复面试送点面试题；回复1024送你一套完整的java 视频教程。

面试题都是有答案的，如下所示：有需要的就来拿吧，**绝对免费，无套路获取。**

电子书

- 大数据
- C 语言
- C++
- Java
- Git
- Python
- Go 语言
- Linux
- 经典必读
- 面试相关
- 前端
- 人工智能
- 设计模式

面试题

- > 快
- > Or
- > 此
- > 网

名称	修改E
7道消息队列ActiveMQ面试题! .pdf	2021,
10道Java高级必备的Netty面试题! .pdf	2021,
10道Java面试必备的设计模式面试题!	2021,
10个Java经典的List面试题! .pdf	2021,
10个Java经典的Main方法面试题! .pdf	2021,
10个Java经典的String面试题! .pdf	2021,
15道经典的Tomcat面试题! .pdf	2021,
15道面试常问的Java多线程面试题! .pdf	2021,
17道消息队列Kafka面试题! .pdf	2021,
18道非常牛逼的Nginx面试题! .pdf	2021,
20道顶尖的Spring Boot面试题! .pdf	2021,
20道面试官常问的JVM面试题! .pdf	2021,
22道面试常问的SpringMVC面试题! .pdf	2021,
24道经典的英语面试题! .pdf	2021,
24道消息队列RabbitMQ面试题! .pdf	2021,
27道顶尖的Java多线程、锁、内存模型...	2021,
29道常见的Spring面试题! .pdf	2021,
30个Java经典的集合面试题! .pdf	2021,
36道面试常问的MyBatis面试题! .pdf	2021,
40道常问的Java多线程面试题! .pdf	2021,
55道BAT精选的Mysql面试题! .pdf	2021,
60道必备的Java核心技术面试题! .pdf	2021,
70道阿里巴巴高级Java面试题! .pdf	2021,
Java 面试题经典 77 问! .pdf	2021,
分布式缓存 Redis + Memcached 经典...	2021,
搞定 HR 面试的 40 个必备问题! .pdf	2021,
精选7道Elastic Search面试题! .pdf	2021,
精选8道Dubbo面试题! .pdf	2021,
精选17道海量数量处理面试题! .pdf	2021,
史上最全40道Dubbo面试题! .pdf	2021,
史上最全50道Redis面试题! .pdf	2021,

22 个题目