

有空青年——python学习



Copyright@铁板豆腐(hellogitty) • 2019/06/18



contents

千里之堤的一片砖

- 数据类型
- 运算符
- 条件判断与循环语句（基于菜鸟demo）

不要做罚抄100遍的小学生——学会使用函数

- 普通函数
- 入口函数——让脚本更优雅
- 外部引入——让逻辑更清晰

Python强大的理由——Package(包)的管理与pip

- Demo wxpy如何运行自己的机器人

数据中自有黄金屋，挖掘要靠好工具

- numpy
- pandas
- scipy
- matplotlib/pyqtgraph

要有对象

- 类与对象

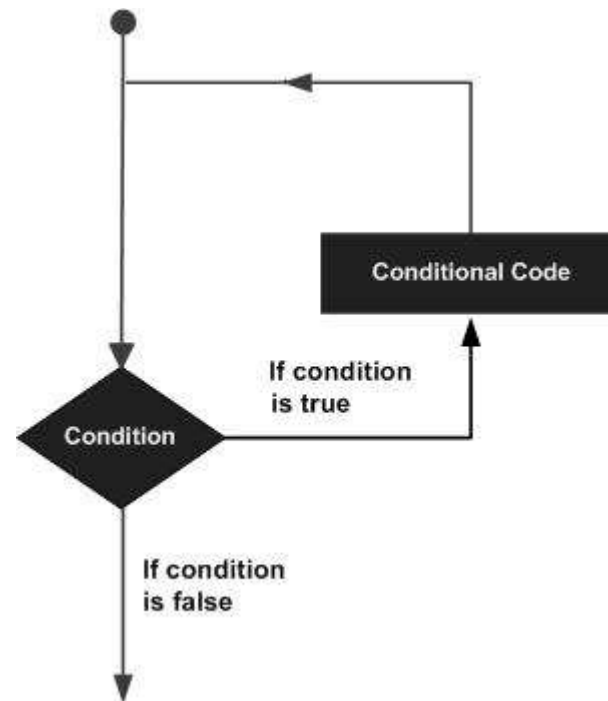
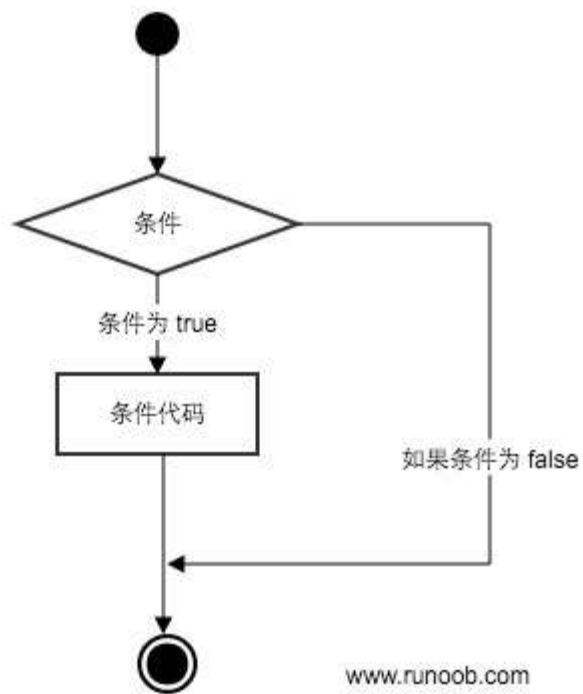




千里之堤的一片砖

条件判断: <https://www.runoob.com/python/python-if-statement.html>

循环语句: <https://www.runoob.com/python/python-loops.html>



- While
- For
- 嵌套
- Continue
- Break
- Pass

不要做罚抄100遍的小学生——学会使用函数

你可以定义一个由自己想要功能的函数，以下是简单的规则：

- 函数代码块以 `def` 关键词开头，后接函数标识符名称和圆括号`()`。
- 任何传入参数和自变量必须放在圆括号中间。圆括号之间可以用于定义参数。
- 函数的第一行语句可以选择性地使用文档字符串—用于存放函数说明。
- 函数内容以冒号起始，并且缩进。
- `return` [表达式] 结束函数，选择性地返回一个值给调用方。不带表达式的`return`相当于返回 `None`。

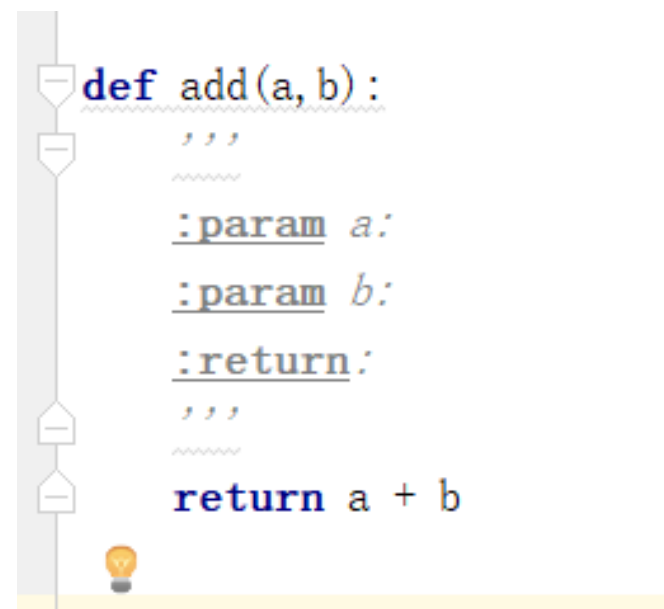
普通函数

```
def functionname( parameters ):
    "函数_文档字符串"
    function_suite
    return [expression]
```

- 必备参数
- 关键字参数
- 可选参数
- 变量作用域
- 返回值
- 不定长参数

不定长函数

```
def functionname([formal_args,] *var_args_tuple ):
    "函数_文档字符串"
    function_suite
    return [expression]
```



TIPS：注意良好编程习惯的培养

不要做罚抄100遍的小学生——学会使用函数

匿名函数——python 使用 lambda 来创建匿名函数。

- lambda只是一个表达式，函数体比def简单很多。
- lambda的主体是一个表达式，而不是一个代码块。仅仅能在lambda表达式中封装有限的逻辑进去。
- lambda函数拥有自己的命名空间，且不能访问自有参数列表之外或全局命名空间里的参数。
- lambda函数只能写一行。

```
lambda [arg1 [,arg2,.....argn]]:expression
```

可写函数说明

```
sum = lambda arg1, arg2: arg1 + arg2;
```

调用sum函数

```
print "相加后的值为 :", sum( 10, 20 )
```

```
print "相加后的值为 :", sum( 20, 20 )
```


不要做罚抄100遍的小学生——学会使用函数

入口函数——一个python脚本的大门，让脚本更优雅

瀑布式执行

```
# coding=utf-8

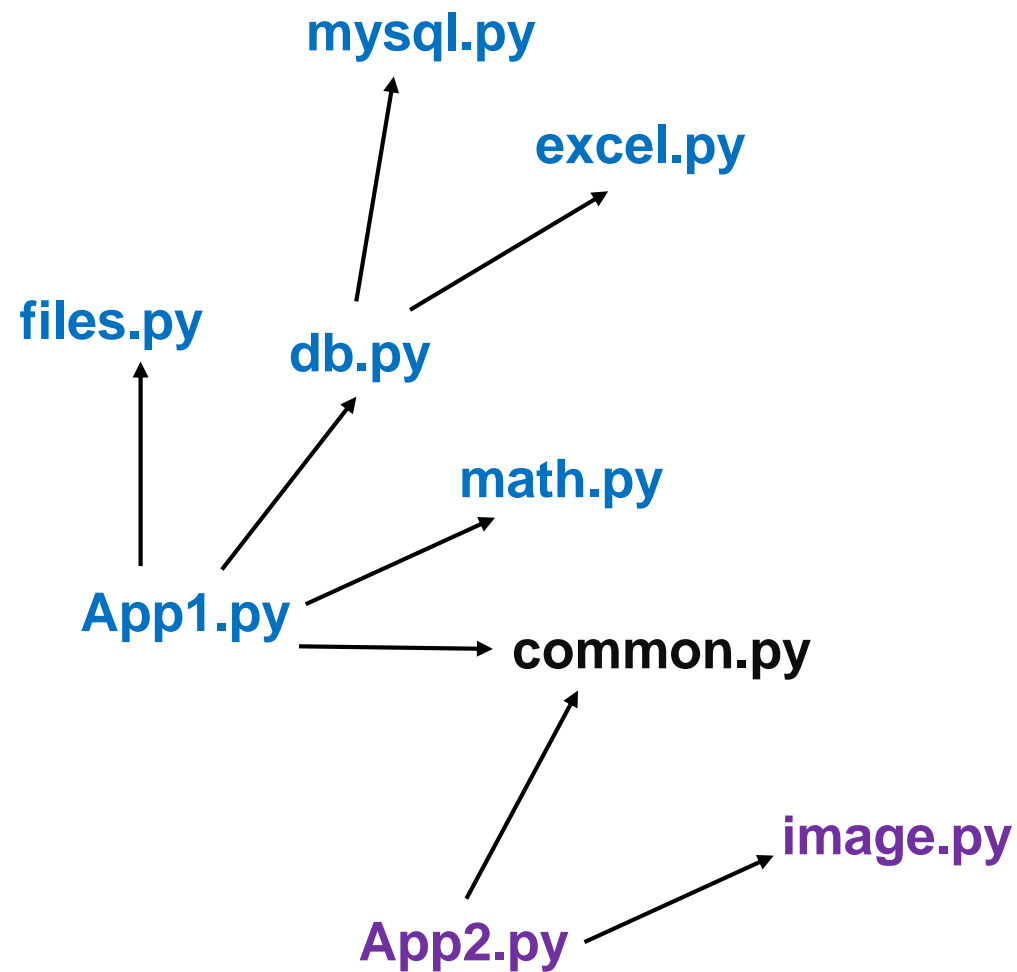
def add(arg1, arg2):
    return arg1 + arg2

if __name__ == "__main__":
    print(add(1,2))
```

__main__
入口大门

不要做罚抄100遍的小学生——学会使用函数

外部引入——各司其职分工明确，让逻辑更清晰



demo: 文件读写与求和



Python强大的理由——Package(包)的管理与pip

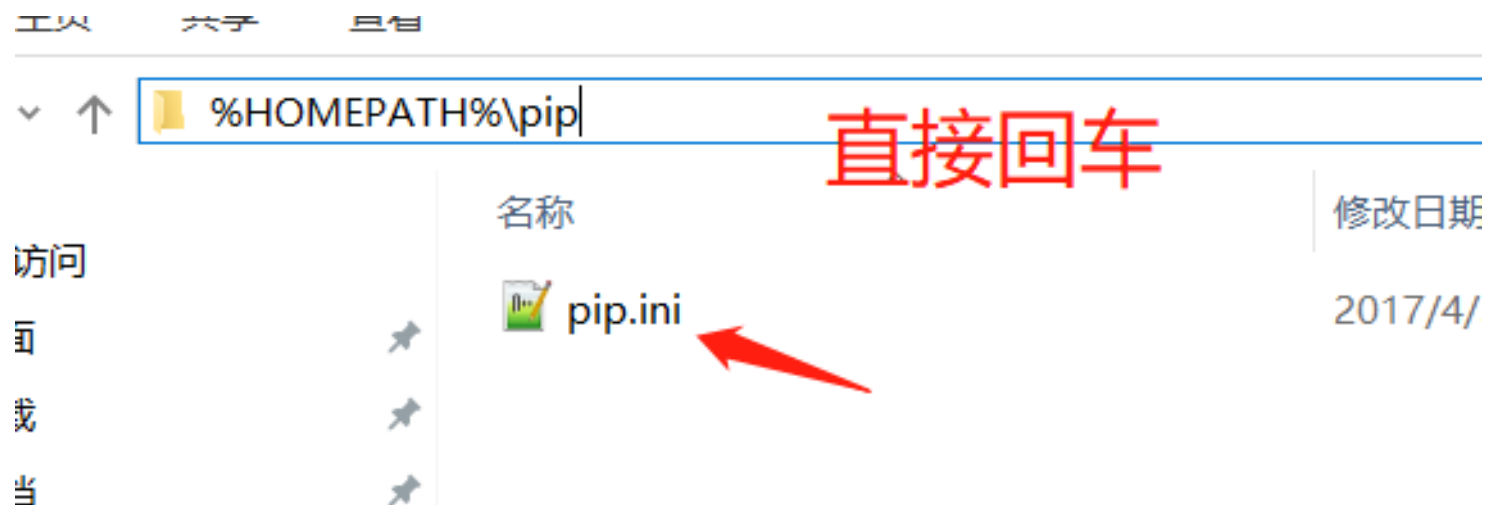
包path: <YOUR_PATH_TO_PYTHON>\Lib\site-packages

台电脑 > OS (C:) > ProgramData > Anaconda3 > Lib > site-packages

- | | | |
|---|--------------------------------------|--------------------------------------|
| nbconvert-5.3.1-py3.6.egg-info | past | pylint-1.7.4-py3.6.egg-info |
| nbformat | path.py-10.3.1-py3.6.egg-info | pyodbc-4.0.17-py3.6.egg-info |
| nbformat-4.4.0-py3.6.egg-info | pathlib2-2.3.0-py3.6.egg-info | pyOpenSSL-17.2.0-py3.6.egg-info |
| networkx | patsy | pyparsing-2.2.0-py3.6.egg-info |
| networkx-2.0-py3.6.egg-info | patsy-0.4.1-py3.6.egg-info | pypng-0.0.19.dist-info |
| nltk | pep8-1.7.0-py3.6.egg-info | pyqrcode |
| nltk-3.2.4-py3.6.egg-info | pickleshare-0.7.4-py3.6.egg-info | PyQRCode-1.2.1.dist-info |
| nose | PIL | PyQt5 |
| nose-1.3.7-py3.6.egg-info | Pillow-4.2.1-py3.6.egg-info | pyqtgraph |
| notebook | pip | pyqtgraph-0.10.0.dist-info |
| numba | pip-9.0.1-py3.6.egg-info | PySocks-1.6.7-py3.6.egg-info |
| numba-0.35.0+10.g143f70e-py3.6-win-amd64.egg-info | pkg_resources | pytest-3.2.1-py3.6.egg-info |
| numexpr | pkginfo | python-dateutil-2.6.1-py3.6.egg-info |
| numexpr-2.6.2-py3.6.egg-info | pkginfo-1.4.1-py3.6.egg-info | pythonwin |
| numpy | ply | pytz |
| numpy-1.13.3-py3.6.egg-info | ply-3.10-py3.6.egg-info | pytz-2017.2-py3.6.egg-info |
| numpydoc | progress | PyWavelets-0.5.2-py3.6.egg-info |
| numpydoc-0.7.0-py3.6.egg-info | progress-1.3-py3.6.egg-info | pywin32_system32 |
| odo | prompt_toolkit | pywt |
| odo-0.5.1-py3.6.egg-info | prompt_toolkit-1.0.15-py3.6.egg-info | pyximport |
| olefile | psutil | pyzmq-16.0.2-py3.6.egg-info |
| opencv_python-3.4.1.15.dist-info | psutil-5.4.0-py3.6.egg-info | qtawesome |
| openpyxl | py | QtAwesome-0.4.4-py3.6.egg-info |

Python强大的理由——Package(包)的管理与pip

新建 pip 配置文件夹，直接在user用户目录中创建一个名为 pip 的文件夹(即%HOMEPATH%\pip), 如下图所示:



其中内容如下:

```
[global]
trusted-host=mirrors.aliyun.com
index-url=http://mirrors.aliyun.com/pypi/simple/
```

```
Windows PowerShell
PS C:\ProgramData\Anaconda3> .\python.exe -m pip list
DEPRECATION: The default format will switch to columns in the future.
format=(legacy|columns) in your pip.conf under the [list] section) t
alabaster (0.7.10)
anaconda-client (1.6.5)
anaconda-navigator (1.6.9)
anaconda-project (0.8.0)
asn1crypto (0.22.0)
astroid (1.5.3)
astropy (2.0.2)
babel (2.5.0)
```

Python强大的理由——Package(包)的管理与pip

Demo 暖场狗狗v0版本

- 运行wxpy失败，提示依赖不满足
- 安装包，再次运行
- 代码走读补充注释



数据中自有黄金屋，挖掘要靠好工具 之 numpy

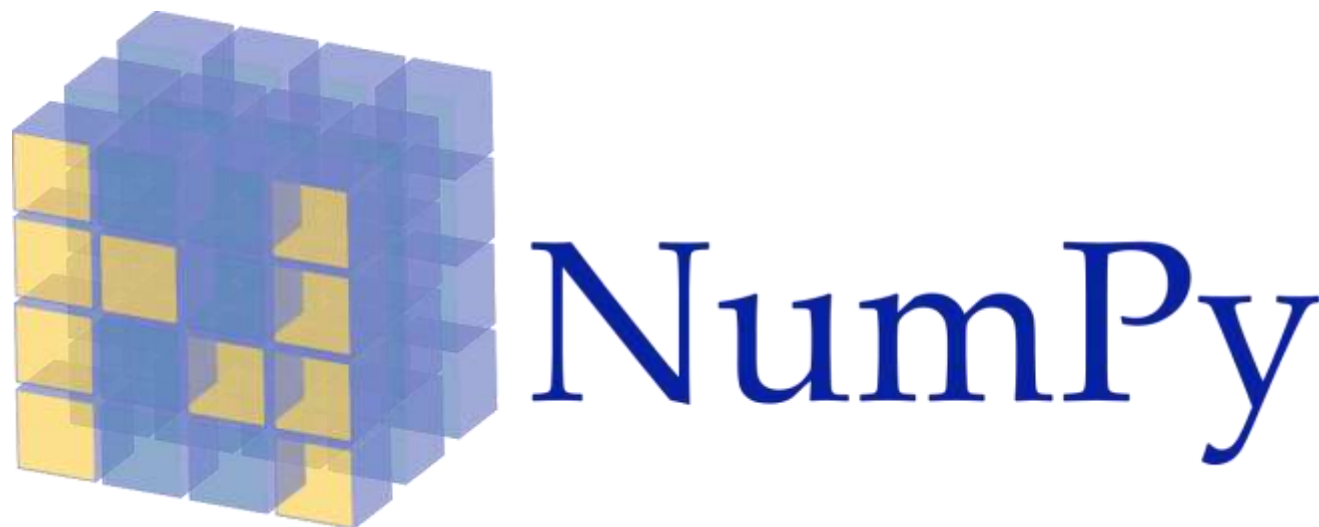
NumPy(Numerical Python) 是 Python 语言的一个扩展程序库，支持大量的**维度数组与矩阵运算**，此外也针对数组运算提供大量的数学函数库。

NumPy 的前身 Numeric 最早是由 Jim Hugunin 与其它协作者共同开发，2005 年，Travis Oliphant 在 Numeric 中结合了另一个同性质的程序库 Numarray 的特色，并加入了其它扩展而开发了 NumPy。NumPy 为开放源代码并且由许多协作者共同维护开发。

NumPy 是一个运行速度非常快的数学库，主要用于数组计算，包含：

- 一个强大的N维数组对象 ndarray
- 线性代数、傅里叶变换、随机数生成等功能

NumPy 官网 <http://www.numpy.org/>



数据中自有黄金屋，挖掘要靠好工具 之 numpy

安装成功验证

```
from numpy import *  
print(eye(4))
```

```
C:\ProgramData\Anaconda3\python.exe D:
```

```
[[ 1.  0.  0.  0.]  
 [ 0.  1.  0.  0.]  
 [ 0.  0.  1.  0.]  
 [ 0.  0.  0.  1.]]
```




content



thanks