

## python2和python3

### Python 2

Python 2 发布于 2000 年年底，这是一种更加清晰和更具包容性的语言开发过程。Python 2 包括了更多的程序性功能，包括能自动化地管理内存的循环检测垃圾收集器，增加了对 Unicode 的支持以实现字符的标准化，并采用列表综合的方式以在现有列表基础上创建列表。

随着 Python 2 的不断发展，更多的功能被添加进来，包括将 Python 的类型和类在 Python 2.2 版本中统一为一层。

### Python 3

Python 3 开发的重点是清理代码库并删除冗余，清晰地表明只能用一种方式来执行给定的任务。对 Python 3.0 的主要修改包括将 print 语句更改为内置函数，改进整数分割的方式，并对 Unicode 提供更多的支持。

此外，许多封装库只适用于 Python 2，但是由于 Python 3 背后的开发团队重申了终止对 Python 2 的支持，促使更多的库被移植到 Python 3 上来。

按照当前时间点（Python 2.7 和 Python3.6），从宏观上介绍下Python 3和Python 2的区别，并举一些对应常见的例子：

1. **统一了字符编码支持。**我特意把它拿出来放在第一条...
  2. **增加了新的语法。**print/exec等成为了函数，[格式化字符串变量](#)，[类型标注](#)，添加了nonlocal、yield from、async/await、yield for关键词和\_\_annotations\_\_、\_\_context\_\_、\_\_traceback\_\_、\_\_qualname\_\_等dunder方法。
  3. **修改了一些语法。**metaclass，raise、[map](#)、[filter](#)以及dict的items/keys/values方法返回迭代对象而不是列表，描述符协议，[保存类属性定义顺序](#)，[保存关键字参数顺序](#)
  4. **去掉了一些语法。**cmp、<>(也就是!=)、xrange（其实就是range）、不再有经典类
  5. **增加一些新的模块。**concurrent.futures、venv、[unittest.mock](#)、[asyncio](#)、[selectors](#)、[typing](#)等
  6. **修改了一些模块。**主要是对模块添加函数/类/方法（如functools.lru\_cache、threading.Barrier）或者参数。
  7. **模块改名。**把一些相关的模块放进同一个包里面（如httpplib, BaseHTTPServer, CGIHTTPServer, SimpleHTTPServer, Cookie, cookielib放进了http里面，[urllib](#), urllib2, urlparse, robotparse放进了[urllib](#)里面），个例如SocketServer改成了socketserver，Queue改成queue等
  8. **去掉了一些模块或者函数。**gopherlib、md5、contextlib.nested、inspect.getmoduleinfo等。去掉的内容的原因主要是2点：1. 过时的技术产物，已经没什么人在用了；2. 出现了新的替代产物后者被证明存在意义不大。理论上对于开发者影响很小。
  9. **优化。**重新实现了dict可以减少20%-25%的内存使用；提升[pickle](#)序列化和反序列化的效率；collections.OrderedDict改用C实现；通过os.scandir对glob模块中的glob()及iglob()进行优化，使得它们现在大概快了3-6倍等.. 这些都是喜大普奔的好消息，同样开发者不需要感知，默默的就会让结果变得更好。
  10. **其他。**构建过程、C的API、安全性等方面的修改，通常对于开发者不需要关心。
- 最后，还是那句话：

Python 2/3的思想基本是共通的，只有少量的语法有差别甚至不兼容。当对Python熟悉到一定程度时，即使只会Python 2也可以在很短的时间就能写Python 3的代码。

如果有的项目用py2，而另一些用了py3，我该怎么办？

两个方案：

1、virtualEnv

2、Anaconda

virtualenv 用来创建最小环境. 只带标准库. 不同环境是不同容器. 不同环境安装的软件互相版本独立.

pyenv 能用 virtualenv 创建环境, 并指定解释器版本.

**conda 同 pyenv, 能指定解释器. 是用 virtualenv 创建的环境, 并在此环境安装好科学计算库.**

**Anaconda官网**

<https://www.anaconda.com/download/>

[Anaconda](#) 是一个包含数据科学常用包的 Python 发行版本。它基于 conda ——一个包和环境管理器——衍生而来。Anaconda 实际上是一个软件发行版，它附带了 conda、Python 和 150 多个科学包及其依赖项。conda是包和环境的管理器。Anaconda 的下载文件比较大（约 500 MB），因为它附带了 Python 中最常用的数据科学包。如果只需要某些包，或者需要节省带宽或存储空间，也可以使用 Miniconda 这个较小的发行版（仅包含 conda 和 Python）。但你仍可以使用 conda 来安装任何可用的包，它只是自身没有附带这些包而已。

**各版本的conda**

<https://repo.anaconda.com/archive/>

**不同版本conda和python对应关系，常用**

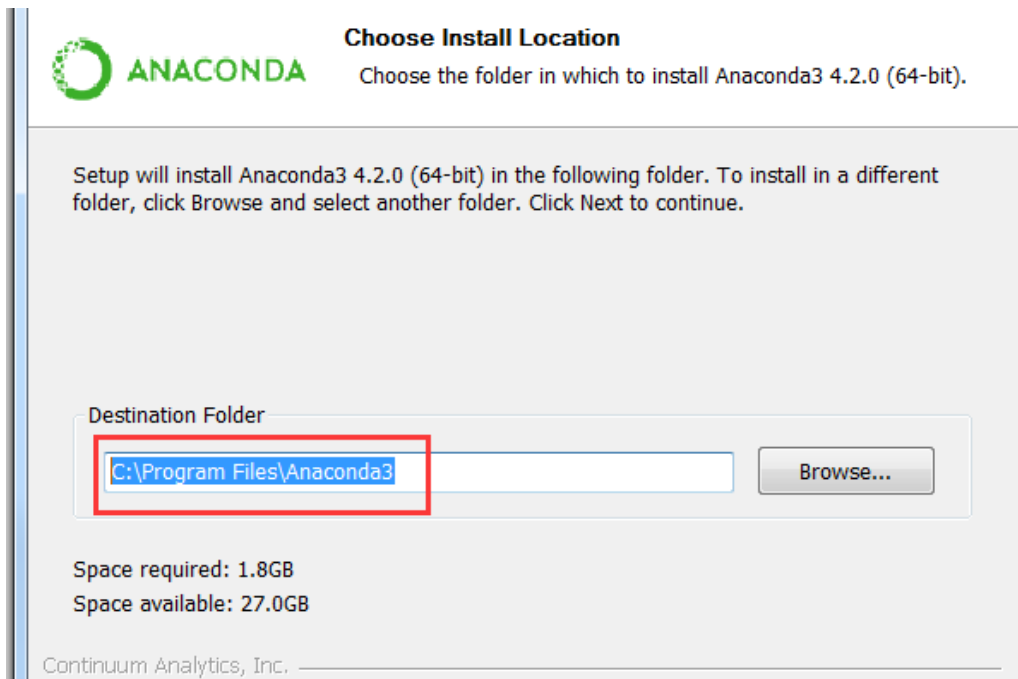
	A	B	C
1	python2	python3	anaconda2/3
2	2.7.14	3.6.5	5.2.0
3	2.7.14	3.6.4	5.1.0
4	2.7.14	3.6.3	5.0.1
5	2.7.13	3.6.2	5.0.0
6	2.7.13	3.6.1	4.4.0
7	2.7.13	3.6.0	4.3.1
8	2.7.13	3.6.0	4.3.0
9	2.7.12	3.5.2	4.2.0
10	2.7.12	3.5.2	4.1.1
11	2.7.11	3.5.1	4.1.0
12	2.7.11	3.5.1	4.0.0

下载 [https://repo.anaconda.com/archive/Anaconda3-4.2.0-Windows-x86\\_64.exe](https://repo.anaconda.com/archive/Anaconda3-4.2.0-Windows-x86_64.exe)

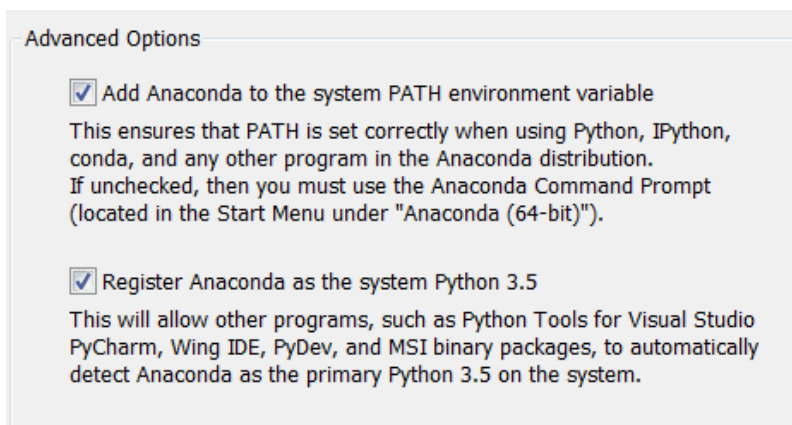
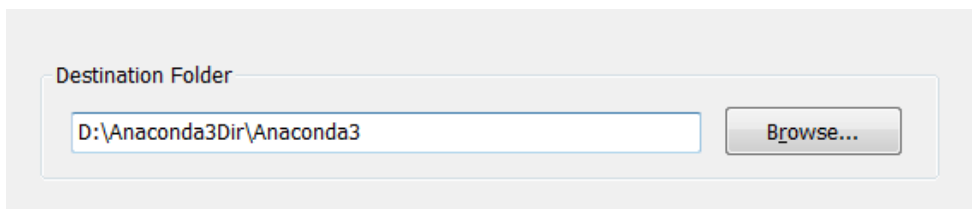
**安装过程注意问题：**

1、安装路径可以改变，但一定要是非汉语命名的文件夹路径，而且一定不能有空格；

2、一定要选择 “Add PATH TO ....” (增加环境变量)，如果首次安装，这个一定要选择，虽然软件不建议，但也要选择；不然在win命令提示符窗口，输入一些更新或下载命令，会显示 “...不是内部或外部命令”；例如，输入命令：conda list 或 conda --version 或 python



改成



安装完成后，使用conda install 包名 安装需要的Python非常方便，但是官方的服务器在国外，因此下载速度很慢，国内清华大学提供了Anaconda的仓库镜像，我们只需要配置Anaconda的配置文件，添加清华的镜像源，然后将其设置为第一搜索渠道即可

```
conda config --set show_channel_urls yes
```

```
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/
```

```
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge/
```

```
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/msys2/
```

```
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/
```

配置完成后，运行 `conda config` 查看是否生效

## Anaconda常用命令大全

<https://blog.csdn.net/sunxinyu/article/details/78801534>

## 创建py2，py3的环境

### 创建py2

`conda create --name python27 python=2.7`，默认会在`{Anaconda_DIR}/envs`目录下创建

下创建

```
C:\Users\Administrator>conda create --name python27 python=2.7
Fetching package metadata .....
```

进一步验证我们上面的镜像配置正确

```
s. tuna. tsinghua. edu. cn/anaconda/pkgs/main
-----
Total: 23.9 MB

The following NEW packages will be INSTALLED:

certifi: 2018.10.15-py27_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
pip: 10.0.1-py27_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
python: 2.7.15-h2880e7c_2 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
setuptools: 40.4.3-py27_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
vc: 9-h7299396_1 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
vs2008_runtime: 9.00.30729.1-hfaea7d5_1 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
wheel: 0.32.2-py27_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
wincertstore: 0.2-py27hf04cefb_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
```

安装完成后

onda3Dir ▶ Anaconda3 ▶ envs ▶ python27 ▶

文件夹

名称	修改日期	类型	大小
conda-meta	2018/10/30 11:24	文件夹	
DLLs	2018/10/30 11:23	文件夹	
include	2018/10/30 11:23	文件夹	
Lib	2018/10/30 11:23	文件夹	
Library	2018/10/30 11:23	文件夹	
libs	2018/10/30 11:23	文件夹	
Scripts	2018/10/30 11:24	文件夹	
tcl	2018/10/30 11:23	文件夹	
Tools	2018/10/30 11:23	文件夹	
msvcm90.dll	2010/8/20 17:06	应用程序扩展	240 KB
msvcp90.dll	2010/8/20 17:06	应用程序扩展	834 KB
msvcr90.dll	2010/8/20 17:06	应用程序扩展	610 KB
python.exe	2018/10/24 2:33	应用程序	28 KB
python.pdb	2018/10/24 2:33	Program Debug ...	187 KB
python27.dll	2018/10/24 2:33	应用程序扩展	3,349 KB
python27.pdb	2018/10/24 2:33	Program Debug ...	4,859 KB
pythonw.exe	2018/10/24 2:33	应用程序	28 KB
pythonw.pdb	2018/10/24 2:33	Program Debug ...	195 KB
vcomp90.dll	2010/8/20 17:06	应用程序扩展	56 KB
w9xpopen.exe	2018/10/24 2:33	应用程序	115 KB

```

vs2008_runtime 100% |#####| Time: 0:00:02 399.45 kB/
vc-9-h7299396_ 100% |#####| Time: 0:00:00 406.27 kB/
python-2.7.15- 100% |#####| Time: 0:00:54 391.37 kB/
certifi-2018.1 100% |#####| Time: 0:00:00 440.48 kB/
wincertstore-0 100% |#####| Time: 0:00:00 2.72 MB/
setuptools-40. 100% |#####| Time: 0:00:01 403.34 kB/
wheel-0.32.2-p 100% |#####| Time: 0:00:00 554.77 kB/
pip-10.0.1-py2 100% |#####| Time: 0:00:04 396.92 kB/
Extracting packages ...
[ COMPLETE ] |#####| 100
Linking packages ...
[ COMPLETE ] |#####| 100
#
# To activate this environment, use:
# > activate python27
#
# To deactivate this environment, use:
# > deactivate python27
#
# * for power-users using bash, you must source
#

```

激活环境

```
C:\Users\Administrator>activate python27

(python27) C:\Users\Administrator>python --version
Python 2.7.15 :: Anaconda, Inc.
```

Python27环境下，完成工作后，切回原来的Python环境，执行命令：deactivate

```
(python27) C:\Users\Administrator>deactivate python27

C:\Users\Administrator>python --version
Python 3.5.3

C:\Users\Administrator>_
```

## 创建py3

conda create --name python354 python=3.5.4

```
Fetching packages ...
vs2015_runtime 100% |#####| Time: 0:00:05 394.35 kB/s
vc-14.1-h0510f 100% |#####| Time: 0:00:00 119.00 kB/s
python-3.5.4-h 100% |#####| Time: 0:00:49 392.25 kB/s
certifi-2018.8 100% |#####| Time: 0:00:00 435.84 kB/s
wincertstore-0 100% |#####| Time: 0:00:00 2.30 MB/s
setuptools-40. 100% |#####| Time: 0:00:01 392.89 kB/s
wheel-0.31.1-p 100% |#####| Time: 0:00:00 487.04 kB/s
pip-10.0.1-py3 100% |#####| Time: 0:00:04 396.59 kB/s
Extracting packages ...
[python          ]|#####| 25%
半:
```

## 激活环境

```
#

C:\Users\Administrator>activate python354

(python354) C:\Users\Administrator>python --version
Python 3.5.4 :: Anaconda, Inc.

(python354) C:\Users\Administrator>_
```

这样，系统中就同时有了py2和py3的环境了

Anaconda3Dir ▸ Anaconda3 ▸ envs ▸			
)			
新建文件夹			
名称	修改日期	类型	
python27	2018/10/30 11:24	文件夹	
python354	2018/10/30 11:39	文件夹	

更新后验证安装包：

```
conda list
```

常用命令：

# 安装包

```
conda install package_name
```

# 更新包

```
conda update package_name
```

# 卸载包

```
conda remove package_name
```

## jupyter notebook 是什么

[Jupyter notebook](#) 是一种 Web 文档。写过项目的都知道，我们在编译器写代码，然后又去打开word或者其他的文本编辑工具去写开发文档，而且调试也不是非常的方便，是不是感觉特麻烦。Jupyter的出现就解决我们的各种麻烦，能够让我们把文本，图像和代码全部组合在一个文档中，而且，调试也特别的方便，大大的提高我们开发的效率——简直就是一个神奇  
安装完成anaconda后，jupyter notebook就已经安装成功了；

notebook关联conda环境

【1】首先激活对应的conda环境

```
activate python27
```

【2】需要安装

```
conda install ipykernel
```

【3】将环境写入Notebook的kernel中

```
python -m ipykernel install --user --name python27 --display-name "Python  
python27"
```

【4】再次运行Jupyter Notebook打开ipynb文件时，就会让你选择conda环境啦

## 解决Anaconda navigator闪退问题

[https://blog.csdn.net/u012318074/article/details/78844789?  
utm\\_source=blogxgwz2](https://blog.csdn.net/u012318074/article/details/78844789?utm_source=blogxgwz2)

**推荐参考:**

<https://zhuanlan.zhihu.com/p/25198543>

<https://zhuanlan.zhihu.com/p/32805175>