

Python 安装及配置指南（Windows 版）

这篇指南将介绍基于 Anaconda3 的 Python（3.8）安装以及如何在 VS Code 上部署已经安装好的 Python。

一、基于 Anaconda 的 Python 安装

Python 是一种面向对象的解释型计算机程序设计语言，其使用具有跨平台的特点，可以在 Linux、macOS 以及 Windows 系统中搭建环境并使用，编写的代码可以在不同平台上运行。而 Anaconda 是 Python 的一个开源发行版本，主要面向科学计算，可以简单理解为 Anaconda 是一个预装了很多可能要用到的第三方库的 Python，而且在包管理、环境管理方面非常方便。

1. Anaconda 下载

安装地址包括 [Anaconda 官网安装地址](#) 或 [清华大学开源软件镜像站](#)（下载速度更快）。下载界面选择 Windows 版，根据电脑属性选择 64 位或 32 位版本（x86_64 是 64 位，x86 是 32 位），**操作系统位数**可右键“开始”查看“系统”。



HOME EVENTS BLOG RSS PODCAST MIRRORS

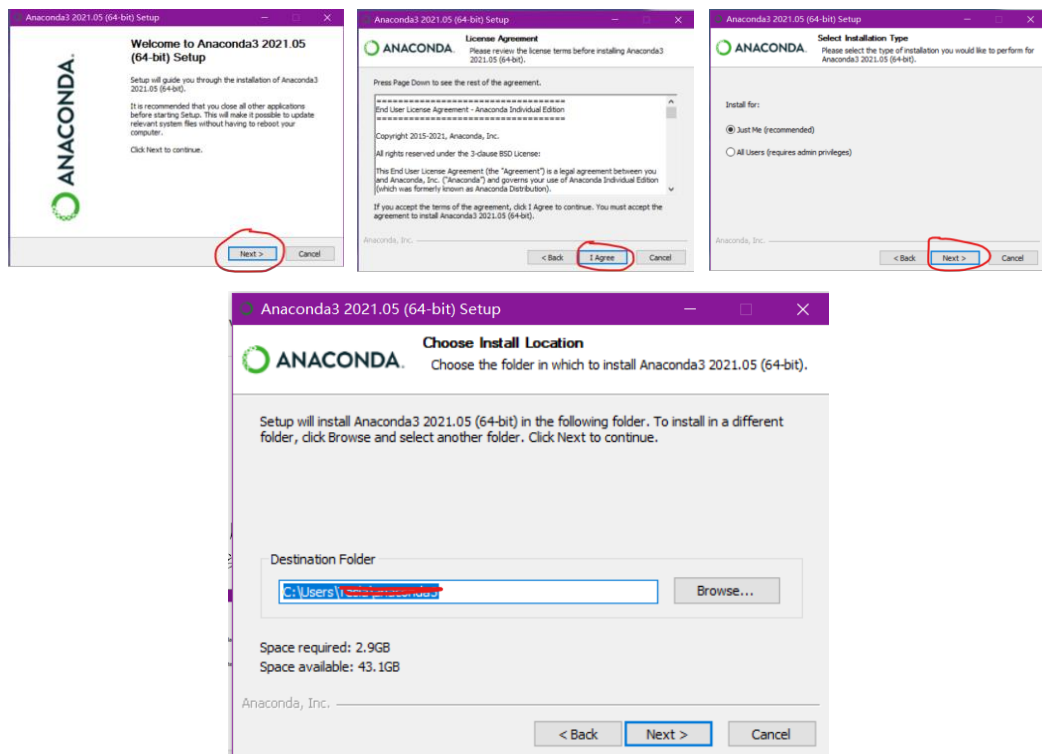
Index of /anaconda/archive/

Last Update: 2021-07-07 14:01

File Name ↓	File Size ↓	Date ↓
Parent directory/	-	-
Anaconda3-2021.05-Windows-x86_64.exe	477.2 MiB	2021-05-14 11:34
Anaconda3-2021.05-Windows-x86.exe	408.5 MiB	2021-05-14 11:34
Anaconda3-2021.05-MacOSX-x86_64.sh	432.7 MiB	2021-05-14 11:34
Anaconda3-2021.05-Linux-s390x.sh	291.7 MiB	2021-05-14 11:33
Anaconda3-2021.05-MacOSX-x86_64.pkg	440.3 MiB	2021-05-14 11:33
Anaconda3-2021.05-Linux-x86_64.sh	544.4 MiB	2021-05-14 11:33
Anaconda3-2021.05-Linux-aarch64.sh	412.6 MiB	2021-05-14 11:33
Anaconda3-2021.05-Linux-ppc64le.sh	285.3 MiB	2021-05-14 11:33

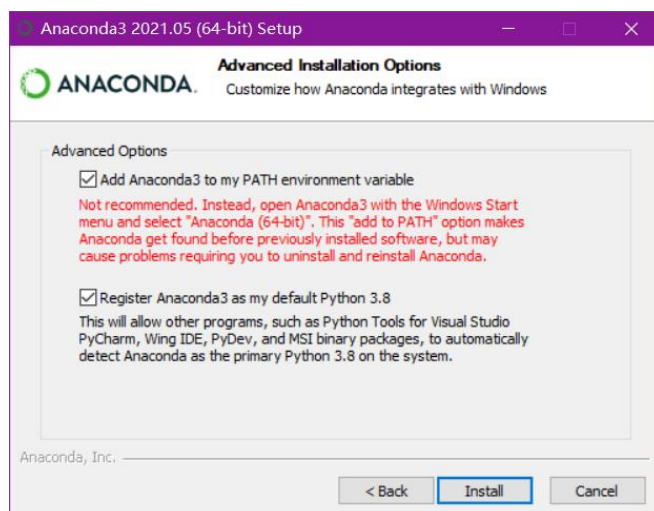
2. Anaconda 安装（以 64 位版为例）

下载完成后，开始安装：



C 盘空间充裕的同学可以直接“Next”，C 盘空间有限的同学**修改安装路径时切记安装路径中只能包含字母、数字和下划线，不要出现中文和空格等特殊字符……。**

选择好安装路径后的界面如下：



第一个勾表示是否把 Anaconda 加入环境变量，这涉及到能否直接在 cmd 命令提示符中使用 conda、jupyter、ipython 等命令，推荐打勾；**第二个**表示是否设置 Anaconda 所带的 Python 3.8 为系统默认的 Python 版本，这个选不选问题不大，后面我们可以在 VS Code 中重新选择 Python 解释器。

安装完成以后，可以按“Windows+R”组合键打开 cmd，分别输入 python、ipython、conda、jupyter notebook 等命令（输完记得按回车），**退出当前命令请输入 exit（）**。其中“python”是进入 python 交互命令行；“ipython”是进入 ipython 交互命令行；“conda”是 Anaconda 的配置命令；“jupyter notebook”则会启动 Web 端的 jupyter notebook。效果如下：

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.19042.1055]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\Tesla>python_

C:\Users\Tesla>python
Python 3.8.5 (default, Sep 3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32

Warning:
This Python interpreter is in a conda environment, but the environment has
not been activated. Libraries may fail to load. To activate this environment
please see https://conda.io/activation

Type "help", "copyright", "credits" or "license()" for more information.
Failed calling sys._interactivehook_
Traceback (most recent call last):
  File "D:\Anaconda\lib\site.py", line 440, in register_readline
    readline.read_history_file(history)
  File "D:\Anaconda\lib\site-packages\pyreadline\rlmain.py", line 165, in read_history_file
    self.mode_history.read_history_file(filename)
  File "D:\Anaconda\lib\site-packages\pyreadline\lineeditor\history.py", line 82, in read_history_file
    for line in open(filename, 'r'):
UnicodeDecodeError: 'gbk' codec can't decode byte 0xac in position 205: illegal multibyte sequence
>>>

>>> exit()

C:\Users\Tesla>python
Python 3.8.5 (default, Sep 3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)]
Type "help", "copyright", "credits" or "license()" for more information.
Python 7.19.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: exit()

C:\Users\Tesla>conda
usage: conda-script.py [-h] [-V] command ...

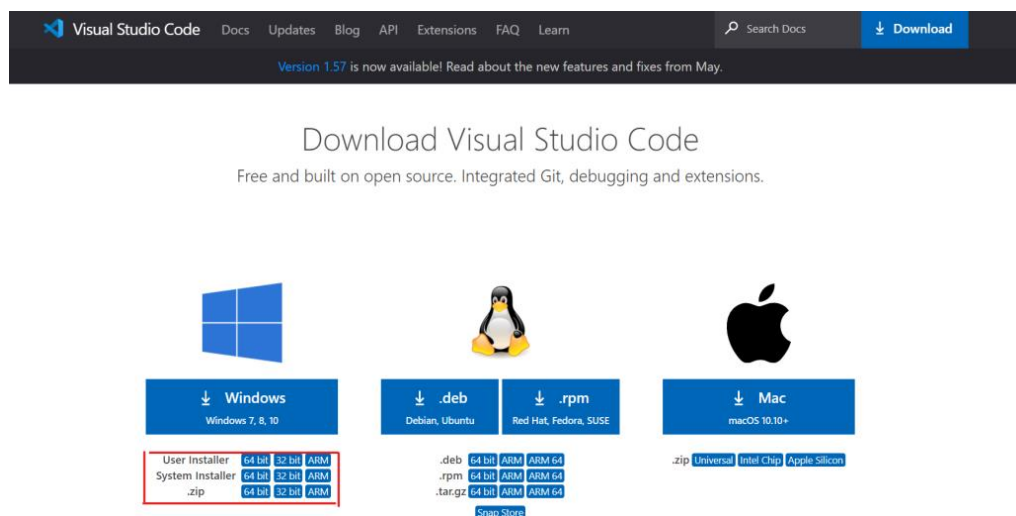
conda is a tool for managing and deploying applications, environments and packages.

Options:
positional arguments:
  command
  clean                Remove unused packages and caches.
```

二、在 VS Code 上配置 Python

1. 安装 VS Code: [VS Code 官方下载地址](#)

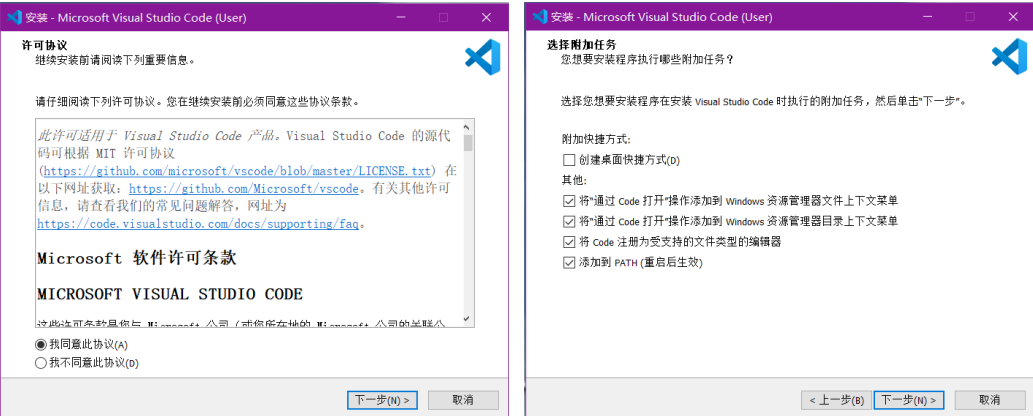
VSCode 是一款免费开源的现代化轻量级代码编辑器，下载界面如下：



user 版会安装在当前计算机帐户目录，这意味着如果使用另一个帐号登陆计算机将无法使用别人安装的 vscode。而 system 版本可以安装在非用户目录，例如 C 盘根目录，任何帐户都可以使用。一般个人电脑安装 User 版即可，这样会默认安装到 C 盘，如果使用 system

版则可以选择安装路径（注意路径命名规范）。.zip 版是一个 VS Code 的压缩包，下载后只需解压，不需要安装，也就是说，解压此压缩包之后，直接双击包含的 "code.exe" 文件，即可运行 VS Code。

下面以 user 版为例：



①将“通过 code 打开”操作添加到 windows 资源管理器文件上下文菜单

②将“通过 code 打开”操作添加到 windows 资源管理器目录上下文菜单

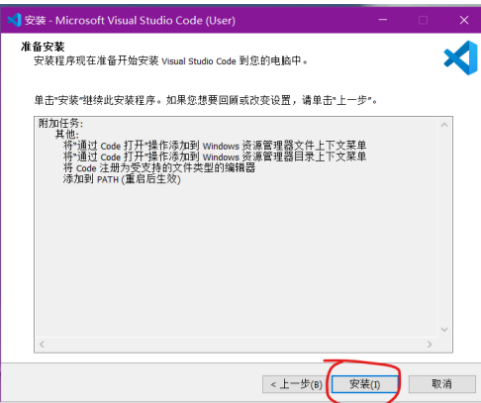
说明：①②建议勾选，以后可以对文件、目录点击鼠标右键，选择“使用 VScode 打开”。

③将 code 注册为受支持的文件类型的编辑器

说明：默认使用 VScode 打开诸如 .txt 等文本类型的文件。

④添加到 PATH


说明：建议勾选，这样以后不用配置环境变量，可以直接使用。



点击“安装”即可。

2. 配置 VS Code

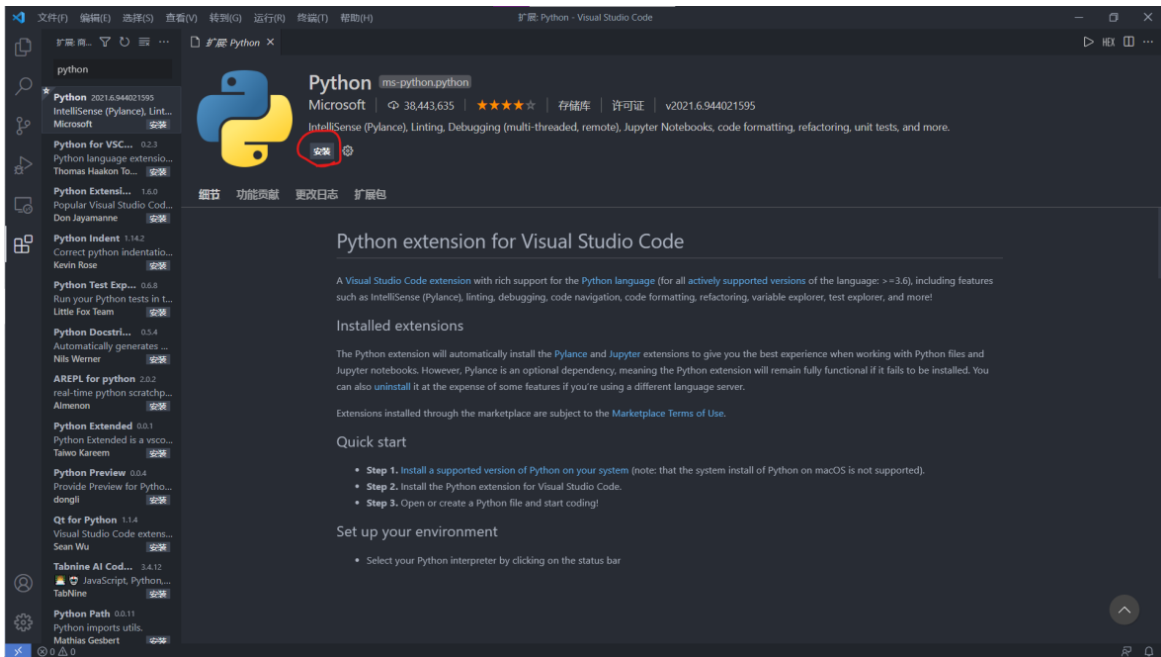
VS Code 本身只是一款文本编辑器而已，并不是一个集成开发环境(IDE)，几乎所有功能都是以**插件扩展（extension）**的形式所存在的。因此，我们想用它编程，不是把 vscode 下载下来就行，还需要安装对应编程语言的扩展以及相应的编译器。查找/安装插件的位置是

左侧竖列最下方的图标 

◆安装中文语言包（搜索 chinese，选择第一个插件安装，重启 VS Code 有效）



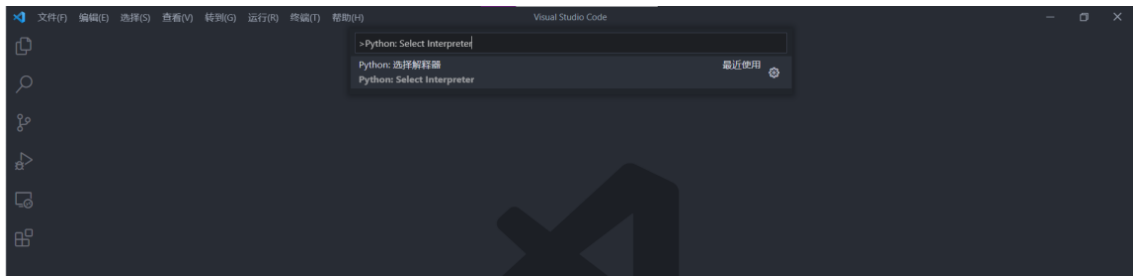
◆安装 Python 插件（搜索 python，选择第一个插件安装）



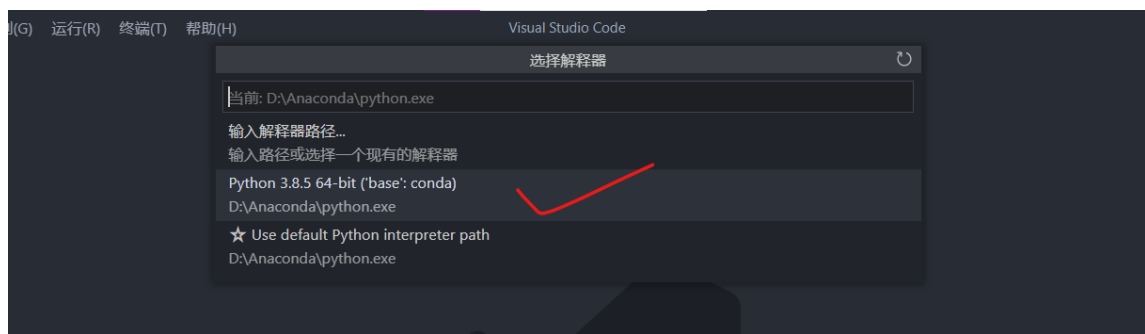
◆（*其它插件……

◆选择 python 解释器

在 VS Code 中按 “**Ctrl+Shift+P**” 打开控制面板，输入 “**Python: Select Interpreter**”:



选择之前安装好的 Anaconda 下的 python 路径：



如果没有打开具体的文件夹，那么选择的解释器将被写入全局用户设置，反之将被写入当前的工作空间设置：


Selecting an interpreter sets the `python.pythonPath` value in your workspace settings to the path of the interpreter. To see the setting, select **File > Preferences > Settings (Code > Preferences > Settings on macOS)**, then select the **Workspace Settings** tab.

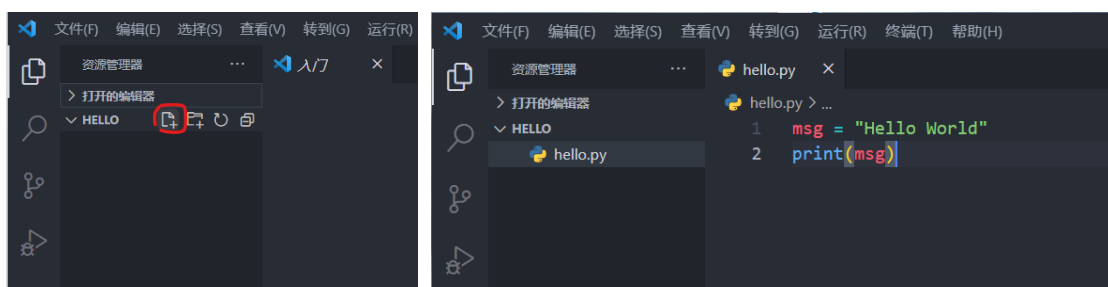
Note: If you select an interpreter without a workspace folder open, VS Code sets `python.pythonPath` in your user settings instead, which sets the default interpreter for VS Code in general. The user setting makes sure you always have a default interpreter for Python projects. The workspace settings lets you override the user setting.

3. *进一步熟悉 VS Code（可观看下方链接中的视频~

[VS Code Intro Videos](#)

4. 第一个 Python 程序

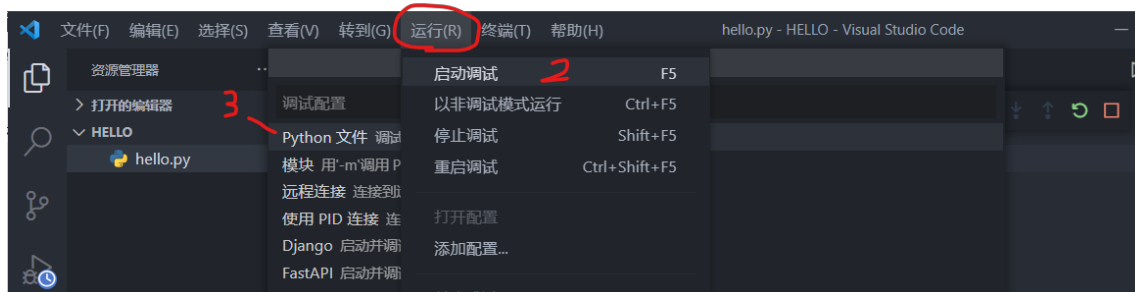
在 VS Code 上方的菜单栏中选择“文件”，“打开文件夹”（我们将在打开的文件夹内创建 python 程序），比如 HELLO 文件夹。从文件资源管理器工具栏中，选择文件夹 **HELLO** 上的“新建文件”按钮，命名文件 `hello.py`（.py 文件扩展名使 VS Code 将此文件解释为 Python 程序，以便它使用 Python 扩展名和选定的解释器），它会自动在编辑器中打开：



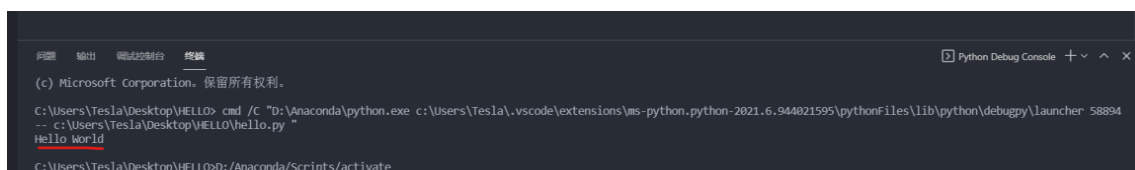
输入：

```
msg = "Hello World"
print("Hello World")
```

然后运行：



界面下方会打开一个终端，Python 解释器会在其中自动激活，然后运行 hello.py:



*程序调试方法见 [Getting Started with Python in VS Code](#) 中的“Configure and run the debugger”部分。

三、包（package）、环境（environment）管理

1. 为了方便地用 python 实现不同功能，我们经常要调用各种各样的包，这些包中涵盖了可能用到的变量和基本操作。包一般来自标准库（如 math、random 等）或者第三方库（如 numpy、pandas 等）。而为了解决开发过程中的版本兼容问题，可能需要创建虚拟环境，在不同环境中，我们可以只安装本项目需要的依赖包，各个环境互相隔离，互不影响。
2. conda 是一个开源的包、环境管理器，可以用于在同一个机器上安装不同版本的软件包及其依赖，并能够在不同的环境之间切换，我们之前安装的 Anaconda 中已经包含了这个工具。可以简单理解为，我们用 conda 创建包含不同版本的 Python 或其他软件包的隔离环境。

Conda 官网的解释：

Conda



Package, dependency and environment management for any language—Python, R, Ruby, Lua, Scala, Java, JavaScript, C/ C++, FORTRAN, and more.

Conda is an open source package management system and environment management system that runs on Windows, macOS and Linux. Conda quickly installs, runs and updates packages and their dependencies. Conda easily creates, saves, loads and switches between environments on your local computer. It was created for Python programs, but it can package and distribute software for any language.

Conda as a package manager helps you find and install packages. If you need a package that requires a different version of Python, you do not need to switch to a different environment manager, because conda is also an environment manager. With just a few commands, you can set up a totally separate environment to run that different version of Python, while continuing to run your usual version of Python in your normal environment.

3. 此外，pip 也是用于安装和管理软件包的包管理器，提供了对 Python 包的查找、下载、安装、卸载的功能。我们在安装 Python 时 PIP 已经捆绑安装了，不需要再单独安装。

类别	conda	pip
管理	二进制	wheel 或源码
需要编译器	no	yes
语言	any	Python
虚拟环境	支持	通过 virtualenv或venv等支持
依赖性检查	yes	屏幕提示用户选择
包来源	Anaconda repo和cloud	PyPi

Pip 和 conda 的区别详见文章 [understanding-conda-and-pip](#)，总结如下：

- [1] Pip 是 Python Packaging Authority 推荐的用于从 [Python Package Index](#) 安装包的工具。[Conda](#) 是跨平台的包和环境管理器，可以安装和管理来自 [Anaconda repository](#) 以 Anaconda Cloud 的 conda 包。conda 包不仅限于 Python 软件。它们还可能包含 C 或 C++ 库，R 包或任何其他软件。
- [2] **conda 能够创建可以包含不同版本的 Python 或其他软件包的隔离环境。**Pip 没有内置的环境支持，而是依赖于 [virtualenv](#) 或 [venv](#) 等其他工具来创建隔离环境。**pip 可以允许在任何环境中安装 python 包，conda 允许在 conda 环境中安装任何语言包（包括 C 语言或者 python）**
- [3] Pip 和 conda 在如何实现环境中的依赖关系方面也有所不同。安装包时，pip 指定安装哪个包就安装哪个包；而 conda 确保满足环境中安装的所有包的所有要求，即**安装所需的全部依赖**。
- [4] 在环境 A 中用 conda 或 pip 安装的 python 包都仅限于当前 A，不能用于 B 环境。鉴于 conda 的环境管理优势，一般采用 conda。但某些包可能 conda 上没有，这时需要用 pip 安装。

4. 常用命令汇总（关于命令的详细解释可以进入 [docs.conda](#) 进行搜索，比如搜索“info”可以找到查看“显示当前环境”的命令“conda info”）

首先在“开始”搜索“Anaconda Prompt”，打开下面的 anaconda 命令行窗口，输入对应的命令后按回车即可看到对应的信息。

♦查看当前系统下的环境：输入 **conda info -e** 或者 **conda env list**，当前已被激活的环境会显示有一个星号或者括号。

```
Anaconda Prompt (Anaconda)
(base) C:\Users\Tesla>conda info -e
# conda environments:
#
base * D:\Anaconda

(base) C:\Users\Tesla>_
```

```
Anaconda Prompt (Anaconda)
(base) C:\Users\Tesla>conda env list
# conda environments:
#
base * D:\Anaconda

(base) C:\Users\Tesla>_
```


- ◆创建新的环境：输入 `conda create -n env_name(环境的名字) (python=?) [packages...]`
其中 `python=?` 指定此环境中安装的 python 版本, `[packages]`指定此环境中要安装的包, 二者都是可选项。

```
Anaconda Prompt (Anaconda) - conda create -n nmos python=3.6 numpy

(base) C:\Users\Tesla>conda create -n nmos python=3.6 numpy
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
current version: 4.10.1
latest version: 4.10.3

Please update conda by running

$ conda update -n base conda

## Package Plan ##

environment location: D:\Anaconda\envs\nmos

added / updated specs:
- numpy
- python=3.6

The following packages will be downloaded:

package                                     build                                size  url
-----
blas-1.0                                     mkl                                6 KB  https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl-2017.0.3                                0                                126.3 MB https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free
numpy-1.13.1                                py36_0                             3.6 MB https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free
Total:                                     129.9 MB

The following NEW packages will be INSTALLED:

blas                anaconda/pkgs/main/win-64::blas-1.0-mkl
certifi             anaconda/pkgs/main/win-64::certifi-2021.5.30-py36h95532_0
mkl                 anaconda/pkgs/free/win-64::mkl-2017.0.3-0
numpy               anaconda/pkgs/free/win-64::numpy-1.13.1-py36_0
pip                 anaconda/pkgs/main/win-64::pip-21.1.3-py36h95532_0
```

注：安装过程中会出现[y/n]选择是否继续，输入 y 并回车即可。

再次查看当前环境，可见在 base 的基础上又多了一个 nmos 环境：

```
Anaconda Prompt (Anaconda)

(base) C:\Users\Tesla>conda info -e
# conda environments:
#
base                * D:\Anaconda
nmos                D:\Anaconda\envs\nmos

(base) C:\Users\Tesla>
```

- ◆激活/进入某一环境：输入 `conda activate env_name(环境名字)`

```
(base) C:\Users\Tesla>conda activate nmos
(nmos) C:\Users\Tesla>python
Python 3.6.13 |Anaconda, Inc.| (default, Mar 16 2021, 11:37:27) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import numpy;
>>> a = numpy.array([1, 2, 3])
>>> print(a)
[1 2 3]
>>>
```

- ◆退出当前激活的环境：输入 `conda deactivate`

```
(nmos) C:\Users\Tesla>conda deactivate  
(base) C:\Users\Tesla>
```

◆删除整个环境(包括环境内的所有包):

输入: `conda remove -n env_name(要删除的环境的名字) --all`

```
(base) C:\Users\Tesla>conda remove -n nmos --all  
Remove all packages in environment D:\Anaconda\envs\nmos:  
  
## Package Plan ##  
  
environment location: D:\Anaconda\envs\nmos  
  
The following packages will be REMOVED:  
  
blas-1.0-mkl  
certifi-2021.5.30-py36haa95532_0  
mkl-2017.0.3-0  
numpy-1.13.1-py36_0  
pip-21.1.3-py36haa95532_0  
python-3.6.13-h3758d61_0  
setuptools-52.0.0-py36haa95532_0  
sqlite-3.36.0-h2bbff1b_0  
vc-14.2-h21ff451_1  
vs2015_runtime-14.27.29016-h5e58377_2  
wheel-0.36.2-pyhd3eb1b0_0  
wincertstore-0.2-py36h7fe50ca_0  
  
Proceed ([y]/n)? y  
Preparing transaction: done  
Verifying transaction: done  
Executing transaction: done  
  
(base) C:\Users\Tesla>conda info -e  
# conda environments:  
#  
base * D:\Anaconda  
  
(base) C:\Users\Tesla>
```

◆包管理

查看当前环境下已安装的包

```
conda list
```

查看某个指定环境的已安装包

```
conda list -n nmos
```

查找包

```
conda search numpy
```

安装 package

```
conda install -n nmos numpy
```

```
# 如果不用-n 指定环境名称，则包将被安装在当前已激活环境
```

```
# 也可以通过-c 指定通过某个 channel 安装
```

```
# 更新 package
```

```
conda update -n nmos numpy
```

```
# 删除 package
```

```
conda remove -n nmos numpy
```

效果如下

```
(base) C:\Users\Tesla>conda install -n nmos numpy
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.10.1
  latest version: 4.10.3

Please update conda by running

  $ conda update -n base conda

## Package Plan ##

  environment location: D:\Anaconda\envs\nmos

  added / updated specs:
    - numpy

The following packages will be downloaded:
```

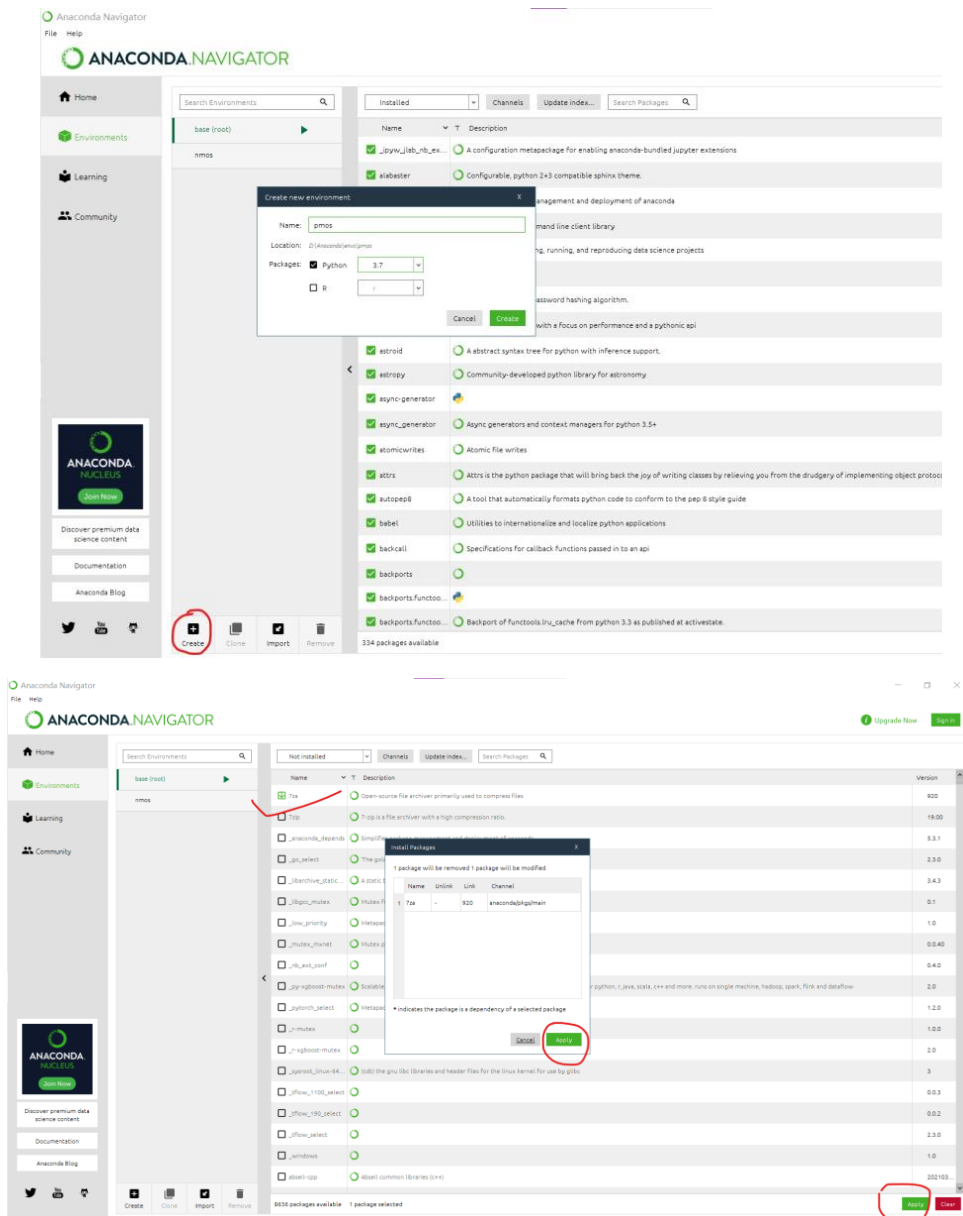
package	build		
ca-certificates-2021.7.5	haa95532_1	113 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
certifi-2021.5.30	py39haa95532_0	140 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
intel-openmp-2021.2.0	haa95532_616	1.8 MB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl-2021.2.0	haa95532_296	115.5 MB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl-service-2.3.0	py39h2bbff1b_1	49 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl_fft-1.3.0	py39h277e83a_2	137 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl_random-1.2.1	py39hf11a4ad_2	223 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
numpy-1.20.2	py39ha4e8547_0	23 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
numpy-base-1.20.2	py39hc2deb75_0	4.2 MB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
openssl-1.1.1k	h2bbff1b_0	4.8 MB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
pip-21.1.3	py39haa95532_0	1.8 MB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
python-3.9.5	h6244533_3	16.5 MB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
setuptools-52.0.0	py39haa95532_0	725 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
six-1.16.0	pyhd3eb1b0_0	18 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
tzdata-2021a	h52ac0ba_0	108 KB	https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main

```
Done
(base) C:\Users\Tesla>conda activate nmos
(nmos) C:\Users\Tesla>conda list
# packages in environment at D:\Anaconda\envs\nmos:
#
# Name                                Version           Build    Channel
blas                                  1.0               mkl      https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
ca-certificates                      2021.7.5          haa95532_1 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
certifi                              2021.5.30         py39haa95532_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
intel-openmp                         2021.2.0          haa95532_616 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl                                  2021.2.0          haa95532_296 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl-service                          2.3.0             py39h2bbff1b_1 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl_fft                             1.3.0             py39h277e83a_2 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl_random                          1.2.1             py39hf11a4ad_2 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
numpy                                1.20.2            py39ha4e8547_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
numpy-base                          1.20.2            py39hc2deb75_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
openssl                              1.1.1k            h2bbff1b_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
pip                                  21.1.3            py39haa95532_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
python                              3.9.5             h6244533_3 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
setuptools                          52.0.0            py39haa95532_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
six                                  1.16.0            pyhd3eb1b0_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
sqlite                               3.36.0            h2bbff1b_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
tzdata                              2021a             h52ac0ba_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
vc                                  14.2              h21ff451_1 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
vs2015_runtime                      14.27.29016       h5e58377_2 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
wheel                                0.36.2            pyhd3eb1b0_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
winertstore                          0.2               py39h2bbff1b_0 https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
(nmos) C:\Users\Tesla>
```

```
(base) C:\Users\Tesla>conda list -n nmos
# packages in environment at D:\Anaconda\envs\nmos:
#
# Name                    Version                   Build  Channel
blas                      1.0                     haa95532_1    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
certifi                   2021.5.30               py39haa95532_0    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
intel-openmp              2021.2.0               haa95532_616    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl                       2021.2.0               haa95532_296    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl-service               2.3.0                  py39h2bfbf1b_1    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl_fft                   1.3.0                  py39h277e83a_2    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
mkl_random                1.2.1                  py39h1f14ad_2     https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
numpy                     1.20.2                 py39ha4e8547_0    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
numpy-base                1.20.2                 py39h2bfbf1b_0    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
openssl                   1.1.1k                 py39haa95532_0    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
pip                       21.1.3                 h2b2bfbf_0       https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
python                    3.9.5                  h6244533_3       https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
setuptools                52.0.0                 py39haa95532_0    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
six                        1.16.0                 pyhd3eb1b0_0     https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
sqlite                    3.36.0                 h2b2bfbf_0       https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
tzdata                    2021a                   h52ac0ba_0       https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
vc                         14.2                   h21ff451_1       https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
vs2015_runtime            14.27.29016            h5e58377_2       https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
wheel                     0.36.2                 pyhd3eb1b0_0     https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
winertstore                0.2                    py39h2bfbf1b_0    https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
```

5. 使用 Anaconda navigator 进行[环境管理](#)、[Python 管理](#)、[包管理](#)

上述三个链接对应的是 Anaconda 官网对 Navigator 进行环境管理、Python 版本管理、包管理的说明（英文），解释得比较清楚。在 Navigator 进行操作的效果图如下：



四、Anaconda 修改镜像源方法

Python 包的默认服务器在国外，下载速度较慢，我们可以换源到国内镜像源

♦查看已有源：输入 `conda config --show channels`，回车。

```
Anaconda Prompt (Anaconda)

(base) C:\Users\Tesla>conda config --show channels
channels:
- defaults

(base) C:\Users\Tesla>
```

♦添加清华镜像源：

`conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgsg/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 --set show_channel_urls yes` # 设置搜索时显示通道地址

添加之后再次查看已有的源：

```
Anaconda Prompt (Anaconda)

(base) C:\Users\Tesla>conda config --show channels
channels:
- https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/msys2/
- https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgsg/free/
- defaults

(base) C:\Users\Tesla>
```

♦如果想删除单个源：输入 `conda config --remove channels your_url`（你想移除的指定镜像路径），回车。

```
(base) C:\Users\Tesla>conda config --remove channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgsg/free/

(base) C:\Users\Tesla>conda config --show channels
channels:
- https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/msys2/
- https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
- defaults

(base) C:\Users\Tesla>
```

♦如何恢复默认镜像源：输入 `conda config --remove-key channels`，回车

```
(base) C:\Users\Tesla>conda config --remove-key channels

(base) C:\Users\Tesla>conda config --show channels
channels:
- defaults

(base) C:\Users\Tesla>
```

五、总结

安装 Anaconda（python）→安装 VS Code→配置 python 环境→包和环境管理

提示：善用搜索引擎，查看官方文档 :-)