

Pycharm 及其 anaconda 安装教程

安装之前我需要强调一点，pycharm 和 anaconda 这两个软件每年都在不断的更新，这就会导致新的版本和旧的版本会有一点差异，可能会导致你用别的版本的软件安装过程中和我的版本安装过程中会有差异，因此如果你是新手，请和我视频中的 pycharm 和 anaconda 版本严格保持一致。课程资料中也会提供对应版本的安装包。具体软件版本如下：



Anaconda3-2023.07-2-Windows-x86_64



pycharm-community-2023.2.1

1、Pycharm 安装教程

1.1、Pycharm 下载

输入如下 pycharm 下载网址，里面有不同历史版本的 pycharm：

<https://www.jetbrains.com/pycharm/download/other.html>

注意两点，1、需要下载社区版本，该版本免费；2、需要下载对应的版本，例如本课程用的版本为 2023.2.1 版本。

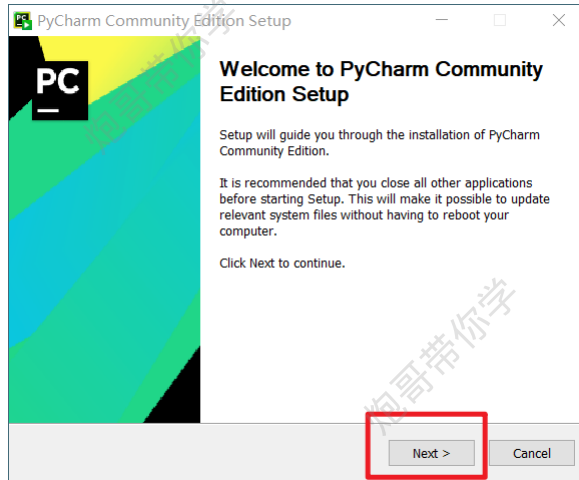
Version 2023.2 2023.2.1 **版本选择**

一定是社区版本

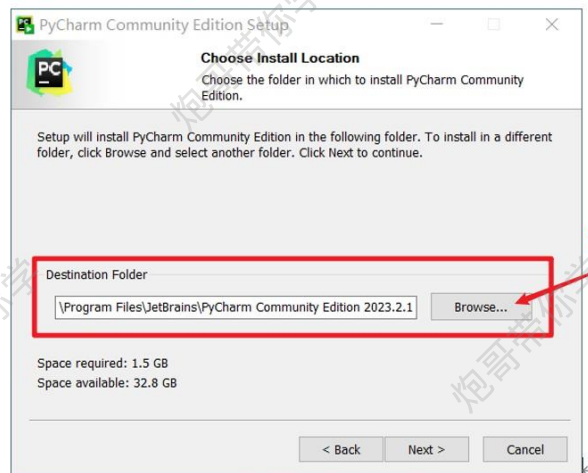
PyCharm Professional Edition	PyCharm Community Edition	Version: 2023.2.1 (Release notes)
2023.2.1 - Linux (tar.gz)	2023.2.1 - Linux (tar.gz)	Build: 232.9559.58
2023.2.1 - Linux ARM64 (tar.gz)	2023.2.1 - Linux ARM64 (tar.gz)	Released: 24 August 2023
2023.2.1 - Windows (exe)	2023.2.1 - Windows (exe) 对应版本	Major version: 2023.2
2023.2.1 - Windows ARM64 (exe)	2023.2.1 - Windows ARM64 (exe)	Released: 26 July 2023
2023.2.1 - macOS (dmg)	2023.2.1 - macOS (dmg)	PyCharm Professional Edition third-party software
2023.2.1 - macOS Apple Silicon (dmg)	2023.2.1 - macOS Apple Silicon (dmg)	PyCharm Community Edition third-party software

1.2、Pycahrm 安装

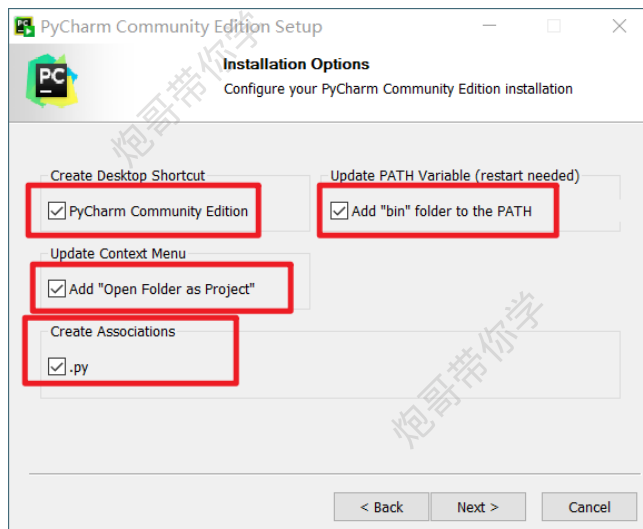
如果你下载好了本课程对应的版本 pycharm 安装包，该安装包是一个 **exe 文件**。双击 exe 进行安装，按如下步骤：



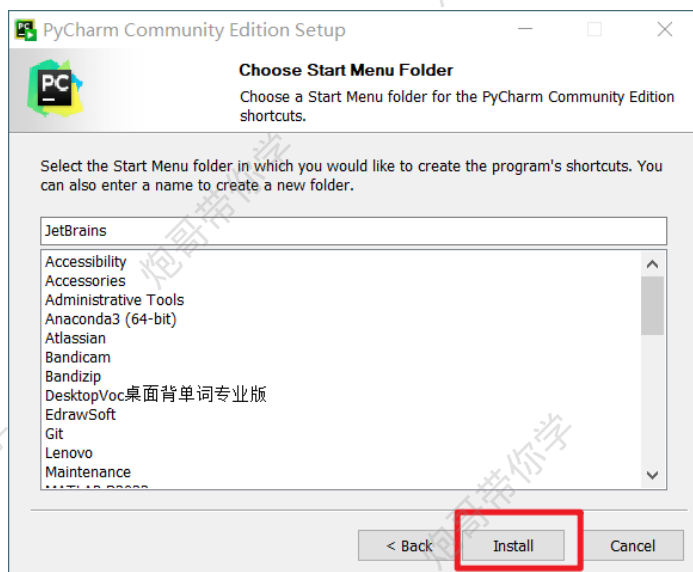
上述步骤会跳转到如图的步骤，该步骤是安装路径，默认路径是安装在 c 盘，但是不建议放到 c 盘，我是安装在 D 盘，同时路径中不要有中文，以免后续有各种奇怪的问题。



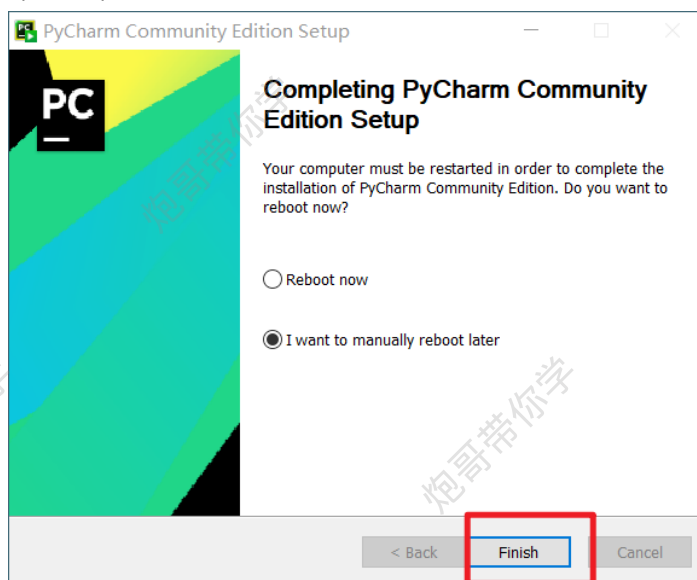
上述步骤后，跳转到如下步骤，**将所有的框全部勾选上**



点击安装，开始安装即可



安装完毕，点击完成即可



此时桌面会有一个 pycharm 的图标，表格该软件安装完成。



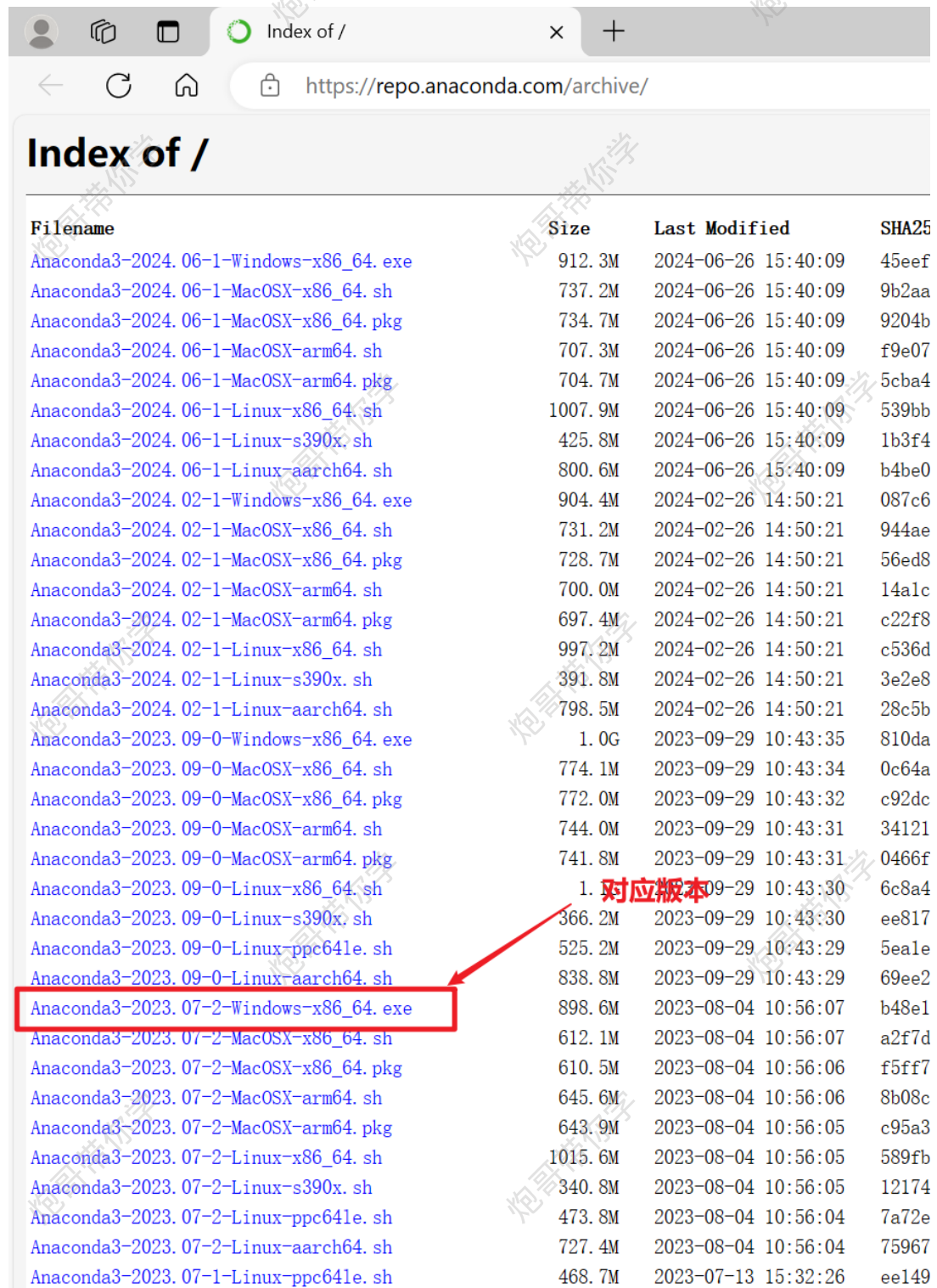
2、anaconda 安装

2.1、anaconda 下载

输入如下 anaconda 下载网址，里面有不同历史版本的 anaconda：

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

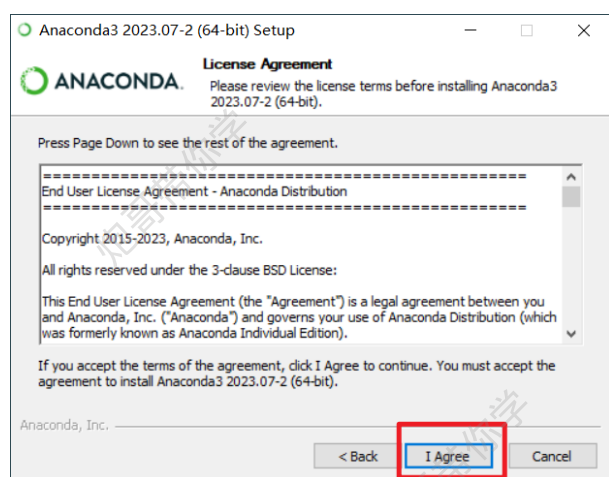
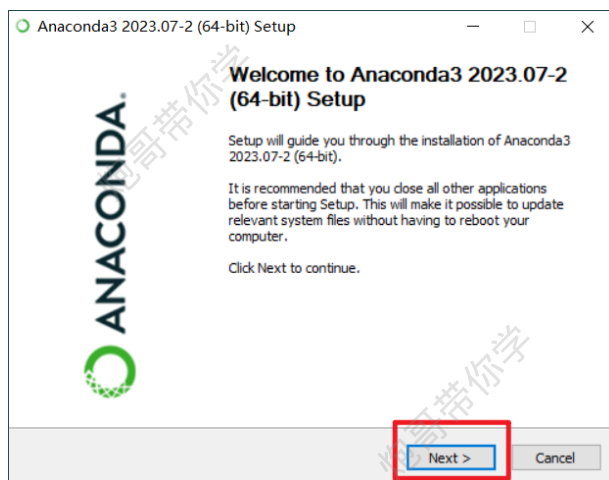
注意：需要下载对应的版本，例如本课程用的版本为 2023.07-2 版本。



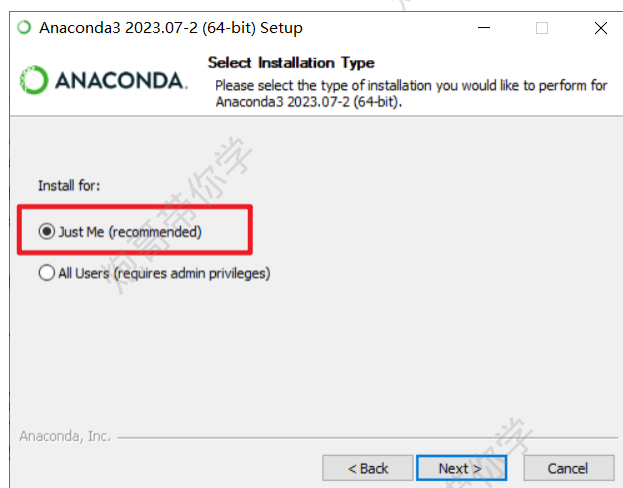
Filename	Size	Last Modified	SHA256
Anaconda3-2024.06-1-Windows-x86_64.exe	912.3M	2024-06-26 15:40:09	45eef
Anaconda3-2024.06-1-MacOSX-x86_64.sh	737.2M	2024-06-26 15:40:09	9b2aa
Anaconda3-2024.06-1-MacOSX-x86_64.pkg	734.7M	2024-06-26 15:40:09	9204b
Anaconda3-2024.06-1-MacOSX-arm64.sh	707.3M	2024-06-26 15:40:09	f9e07
Anaconda3-2024.06-1-MacOSX-arm64.pkg	704.7M	2024-06-26 15:40:09	5cba4
Anaconda3-2024.06-1-Linux-x86_64.sh	1007.9M	2024-06-26 15:40:09	539bb
Anaconda3-2024.06-1-Linux-s390x.sh	425.8M	2024-06-26 15:40:09	1b3f4
Anaconda3-2024.06-1-Linux-aarch64.sh	800.6M	2024-06-26 15:40:09	b4be0
Anaconda3-2024.02-1-Windows-x86_64.exe	904.4M	2024-02-26 14:50:21	087c6
Anaconda3-2024.02-1-MacOSX-x86_64.sh	731.2M	2024-02-26 14:50:21	944ae
Anaconda3-2024.02-1-MacOSX-x86_64.pkg	728.7M	2024-02-26 14:50:21	56ed8
Anaconda3-2024.02-1-MacOSX-arm64.sh	700.0M	2024-02-26 14:50:21	14a1c
Anaconda3-2024.02-1-MacOSX-arm64.pkg	697.4M	2024-02-26 14:50:21	c22f8
Anaconda3-2024.02-1-Linux-x86_64.sh	997.2M	2024-02-26 14:50:21	c536d
Anaconda3-2024.02-1-Linux-s390x.sh	391.8M	2024-02-26 14:50:21	3e2e8
Anaconda3-2024.02-1-Linux-aarch64.sh	798.5M	2024-02-26 14:50:21	28c5b
Anaconda3-2023.09-0-Windows-x86_64.exe	1.0G	2023-09-29 10:43:35	810da
Anaconda3-2023.09-0-MacOSX-x86_64.sh	774.1M	2023-09-29 10:43:34	0c64a
Anaconda3-2023.09-0-MacOSX-x86_64.pkg	772.0M	2023-09-29 10:43:32	c92dc
Anaconda3-2023.09-0-MacOSX-arm64.sh	744.0M	2023-09-29 10:43:31	34121
Anaconda3-2023.09-0-MacOSX-arm64.pkg	741.8M	2023-09-29 10:43:31	0466f
Anaconda3-2023.09-0-Linux-x86_64.sh	1.0G	2023-09-29 10:43:30	6c8a4
Anaconda3-2023.09-0-Linux-s390x.sh	366.2M	2023-09-29 10:43:30	ee817
Anaconda3-2023.09-0-Linux-ppc64le.sh	525.2M	2023-09-29 10:43:29	5eale
Anaconda3-2023.09-0-Linux-aarch64.sh	838.8M	2023-09-29 10:43:29	69ee2
Anaconda3-2023.07-2-Windows-x86_64.exe	898.6M	2023-08-04 10:56:07	b48e1
Anaconda3-2023.07-2-MacOSX-x86_64.sh	612.1M	2023-08-04 10:56:07	a2f7d
Anaconda3-2023.07-2-MacOSX-x86_64.pkg	610.5M	2023-08-04 10:56:06	f5ff7
Anaconda3-2023.07-2-MacOSX-arm64.sh	645.6M	2023-08-04 10:56:06	8b08c
Anaconda3-2023.07-2-MacOSX-arm64.pkg	643.9M	2023-08-04 10:56:05	c95a3
Anaconda3-2023.07-2-Linux-x86_64.sh	1015.6M	2023-08-04 10:56:05	589fb
Anaconda3-2023.07-2-Linux-s390x.sh	340.8M	2023-08-04 10:56:05	12174
Anaconda3-2023.07-2-Linux-ppc64le.sh	473.8M	2023-08-04 10:56:04	7a72e
Anaconda3-2023.07-2-Linux-aarch64.sh	727.4M	2023-08-04 10:56:04	75967
Anaconda3-2023.07-1-Linux-ppc64le.sh	468.7M	2023-07-13 15:32:26	ee149

2.2、anaconda 安装

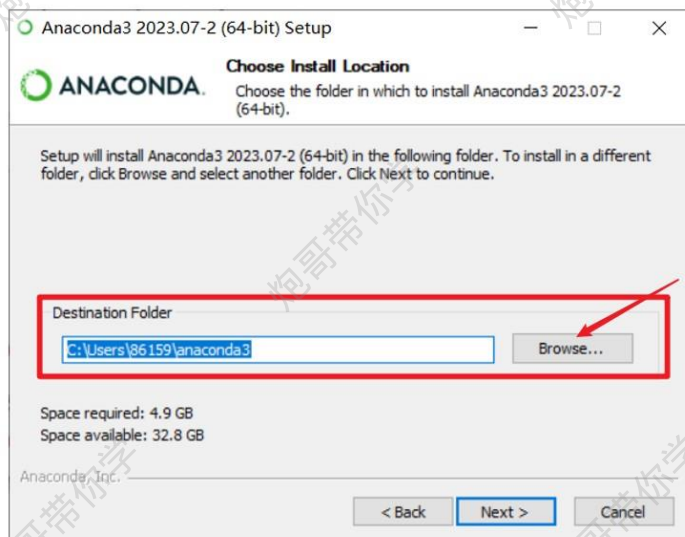
下载完成，会有一个安装包的 exe，双击 exe 进行安装，按如下步骤



这里一定要选择 just me 否则在后续创建环境的时候会出各种各样的问题

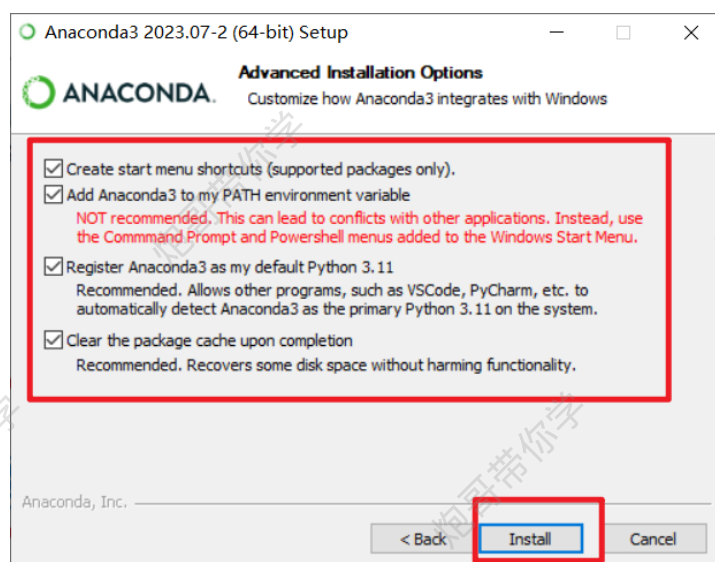


这里安装路径的选择，一定最好不要放在默认路径，默认路径在 C 盘，最好在其他盘中创建路径，并记住该路径。因为后续在安装各种环境的时候环境占用空间很大，放在 c 盘空间会不够。

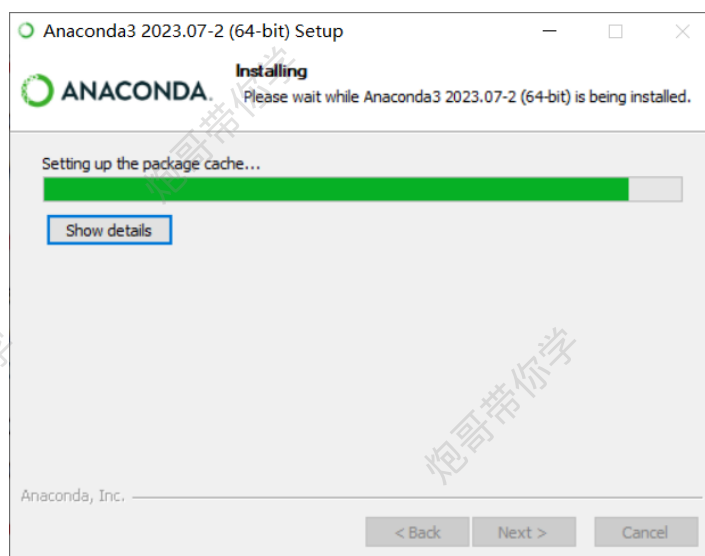


选择安装的路径，
最好在非C盘的路径下
同时路径不要有中文

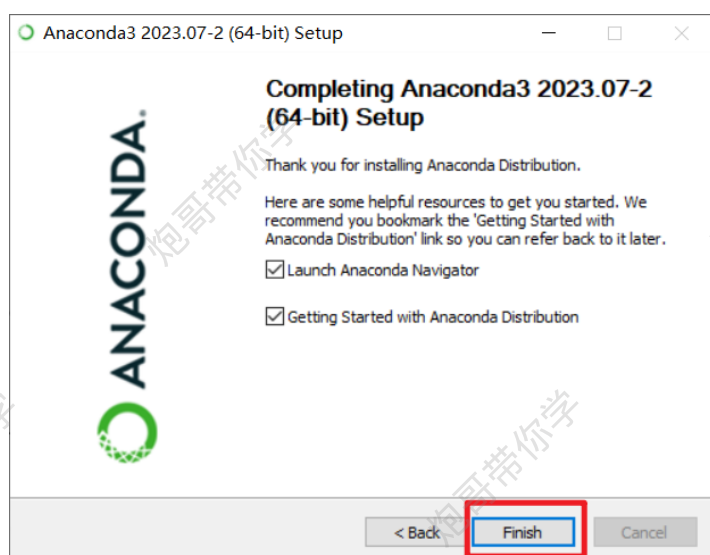
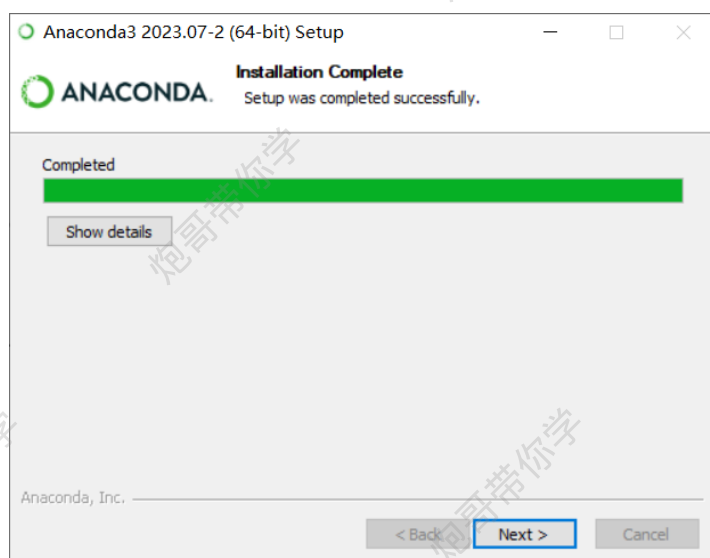
和我一样将这里的√全部框上，点击安装



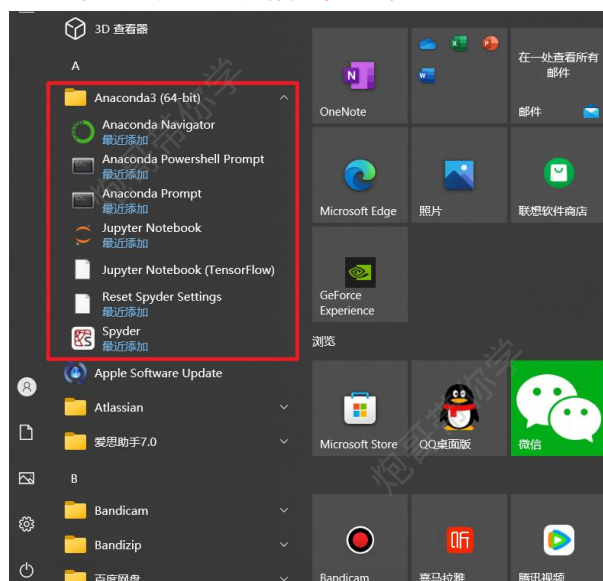
这里下载安装在最后一段可能需要一些时间，请安心等待



安装完毕点击完成即可

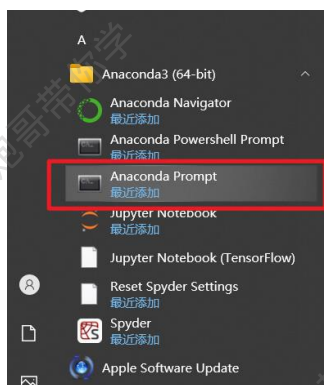


安装完毕以后，在电脑的开始键里面就有安装好的 anaconda



2.3、Anaconda 中 Python 环境的创建

点击图标 anaconda prompt, 会弹出一个黑框



输入如下命令：

```
conda create -n pytorch python==3.8
```

上述的命令的意思是，创建一个名字为 pytorch 的环境，其中 Python 版本为 3.8，其中名字可以指定，只要不是中文都行，不一定是 pytorch，Python 的版本也可以，3.5、3.6、3.7 都可以。



创建环境的时候 Solving environment 一直卡着转圈圈

```
DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/r/win-64/current_repodata.json HTTP/1.1" 304 0
DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/main/win-64/current_repodata.json HTTP/1.1" 304 0
DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/msys2/win-64/current_repodata.json HTTP/1.1" 304 0
/ DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/r/noarch/current_repodata.json HTTP/1.1" 304 0
- DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/main/noarch/current_repodata.json HTTP/1.1" 304
0
done
Solving environment: unsuccessful initial attempt using frozen solve. Retrying with flexible solve.
Solving environment: done
Collecting package metadata (repodata.json): / DEBUG:urllib3.connectionpool:Starting new HTTPS connection (1): repo.anaconda.com:443
DEBUG:urllib3.connectionpool:Starting new HTTPS connection (1): repo.anaconda.com:443
DEBUG:urllib3.connectionpool:Starting new HTTPS connection (1): repo.anaconda.com:443
DEBUG:urllib3.connectionpool:Starting new HTTPS connection (1): repo.anaconda.com:443
DEBUG:urllib3.connectionpool:Starting new HTTPS connection (1): repo.anaconda.com:443
\ DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/r/win-64/repodata.json HTTP/1.1" 304 0
DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/r/noarch/repodata.json HTTP/1.1" 304 0
| DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/main/win-64/repodata.json HTTP/1.1" 304 0
| DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/msys2/win-64/repodata.json HTTP/1.1" 304 0
DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/main/noarch/repodata.json HTTP/1.1" 304 0
/ DEBUG:urllib3.connectionpool:https://repo.anaconda.com:443 "GET /pkgs/msys2/noarch/repodata.json HTTP/1.1" 304 0
done
Solving environment: /
```

可以去网上搜对应的问题解决方案

会跳出如下界面，输入 y，表示下载对应的环境依赖包

```
package                                     build
-----
python-3.8.0                               hff0d562_2      15.9 MB defaults
Total:                                     15.9 MB

The following NEW packages will be INSTALLED:

ca-certificates    pkgs/main/win-64::ca-certificates-2023.08.22-haa95532_0
openssl            pkgs/main/win-64::openssl-1.1.1v-h2bbff1b_0
pip                pkgs/main/win-64::pip-23.2.1-py38haa95532_0
python             pkgs/main/win-64::python-3.8.0-hff0d562_2
setuptools         pkgs/main/win-64::setuptools-68.0.0-py38haa95532_0
sqlite             pkgs/main/win-64::sqlite-3.41.2-h2bbff1b_0
vc                 pkgs/main/win-64::vc-14.2-h21ff451_1
vs2015_runtime     pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2
wheel              pkgs/main/win-64::wheel-0.38.4-py38haa95532_0

Proceed ([y]/n)? y
```

所有的包安装完毕输入如下命令查看环境数量：

conda env list，可以看到现在环境中除了基础的 base 环境，还有刚刚创建的名字为 pytorch 的环境

```
(base) C:\Users\S6159>conda env list
# conda environments:
#
base                * D:\code\anaconda
pytorch             D:\code\anaconda\envs\pytorch

(base) C:\Users\S6159>
```

输入命令，activate pytorch，进入到 pytorch 环境中

```
(base) C:\Users\86159>conda env list
# conda environments:
#
base                * D:\code\anaconda
pytorch             D:\code\anaconda\envs\pytorch

(base) C:\Users\86159>activate pytorch
(pytorch) C:\Users\86159>_
```

输入 conda list 命令，查看该环境下的所有依赖包，可以看到有一个 Python 版本为 3.8，到此 anaconda 中的 Python 环境就安装好了，我们有了对应版本的 Python 解释器可以进行 Python 代码的撰写了。

```
(base) C:\Users\86159>activate pytorch
(pytorch) C:\Users\86159>conda list
# packages in environment at D:\code\anaconda\envs\pytorch:
#
# Name                   Version             Build             Channel
ca-certificates         2023.08.22          haa95532_0        defaults
openssl                 1.1.1v              h2bbff1b_0        defaults
pip                     23.2.1              py38haa95532_0    defaults
python                  3.8.0               hff0d562_2        defaults
setuptools               68.0.0              py38haa95532_0    defaults
sqlite                  3.41.2              h2bbff1b_0        defaults
vc                      14.2                h21ff451_1        defaults
vs2015_runtime          14.27.29016         h5e58377_2        defaults
wheel                   0.38.4              py38haa95532_0    defaults

(pytorch) C:\Users\86159>
```