

深度学习开发环境搭建:

anaconda + pycharm +pytorch +opencv

相关软件

- **anaconda**: (<https://www.anaconda.com>)

Python 环境管理器，集成了conda、Python等180多个科学包及其依赖项，可创建虚拟环境，便于第三方库的安装、删减和管理。

- **Pycharm**: (<https://www.jetbrains.com/pycharm/>)

代码编辑器。(只是记事本，具有运行代码的功能)，只有在配置好的python 解释器中，才可以运行代码。

- **PyTorch**: 基于 Python 深度学习框架，框架中包含人工智能编程常用的各类函数和 API 等。

PyTorch: <https://pytorch.org/get-started/locally/>

- **OpenCV**: (Open Source Computer Vision Library) 是一个开源的计算机视觉和机器学习软件库。

OpenCV: <https://apachecn.github.io/opencv-doc-zh/#/>

一. 安装 anaconda 软件

官网下载地址: <https://www.anaconda.com/download>

Download Now

For installation assistance, refer to [Troubleshooting](#).

Download Anaconda Distribution or [Miniconda](#) by choosing the proper installer for your machine. Learn the difference from our [Documentation](#).

Anaconda Installers

[Download](#)

Windows
Python 3.12
64-Bit Graphical Installer (912.3M)

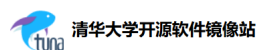
Mac
Python 3.12
64-Bit (Apple silicon) Graphical Installer (704.7M)
64-Bit (Apple silicon) Command Line Installer (707.3M)

Linux
Python 3.12
64-Bit (x86) Installer (1007.9M)
64-Bit (AWS Graviton2 / ARM64) Installer (800.6M)

国内镜像地址: <https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/>

清华大学开源软件镜像站

清华大学开源软件镜像站是由清华大学 TUNA 协会运行和维护的, 致力于为国内和校内用户提供高质量的开源软件镜像和Linux 镜像源服务



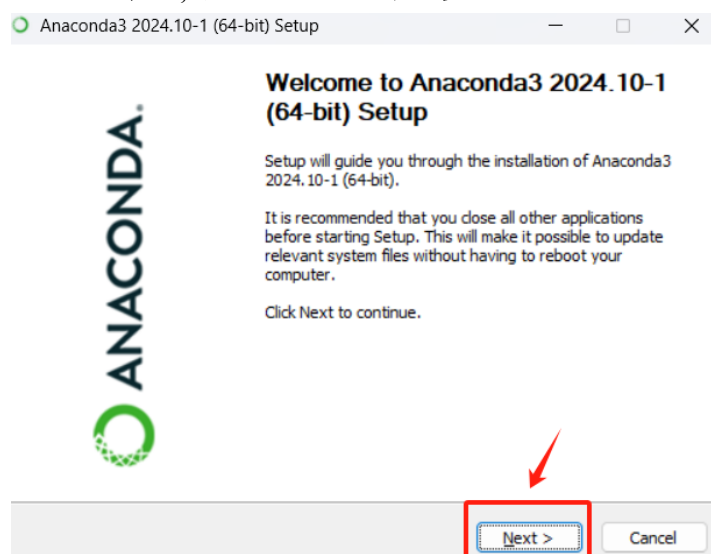
[HOME](#) [EVENTS](#) [BLOG](#) [RSS](#) [PODCAST](#) [MIRRORS](#)

Index of /anaconda/archive/

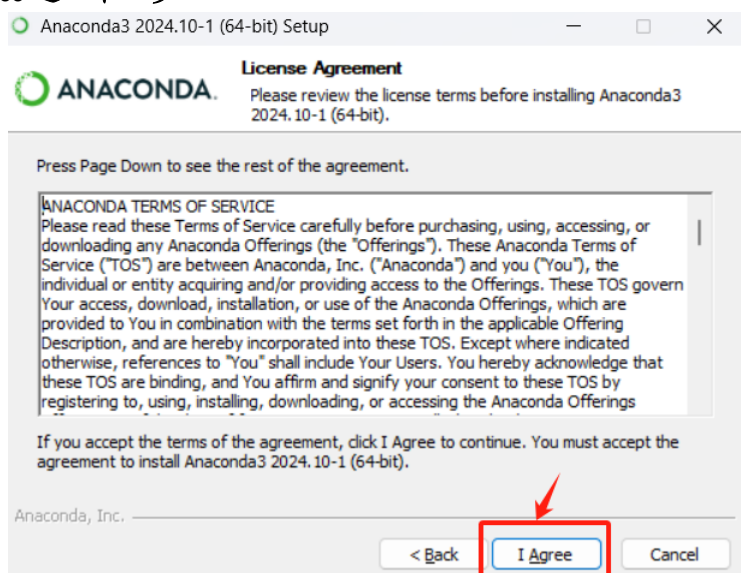
Last Update: 2024-10-25 16:15

File Name ↓	File Size ↓	Date ↓
Parent directory/	-	-
Anaconda3-2024.10-1-MacOSX-x86_64.sh	778.5 MiB	2024-10-24 01:52
Anaconda3-2024.10-1-MacOSX-x86_64.pkg	776.0 MiB	2024-10-24 01:52
Anaconda3-2024.10-1-Windows-x86_64.exe	950.5 MiB	2024-10-24 01:52
Anaconda3-2024.10-1-Linux-aarch64.sh	843.2 MiB	2024-10-24 01:52
Anaconda3-2024.10-1-MacOSX-arm64.pkg	744.6 MiB	2024-10-24 01:52
Anaconda3-2024.10-1-Linux-s390x.sh	441.2 MiB	2024-10-24 01:52
Anaconda3-2024.10-1-MacOSX-arm64.sh	747.2 MiB	2024-10-24 01:52
Anaconda3-2024.10-1-Linux-x86_64.sh	1.0 GiB	2024-10-24 01:52
Anaconda3-2024.06-1-Windows-x86_64.exe	912.3 MiB	2024-06-27 04:49

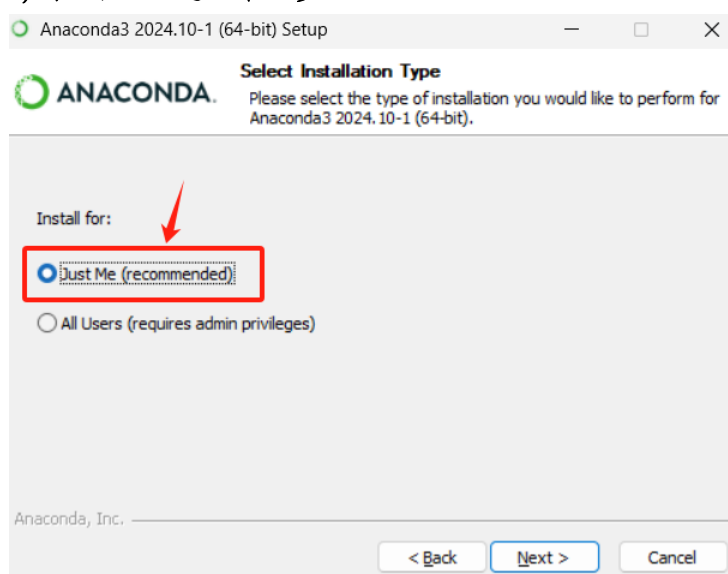
1. 双击打开 anaconda 安装包，点击 next 进入下一步



2. 点击 I Agree 进入下一步



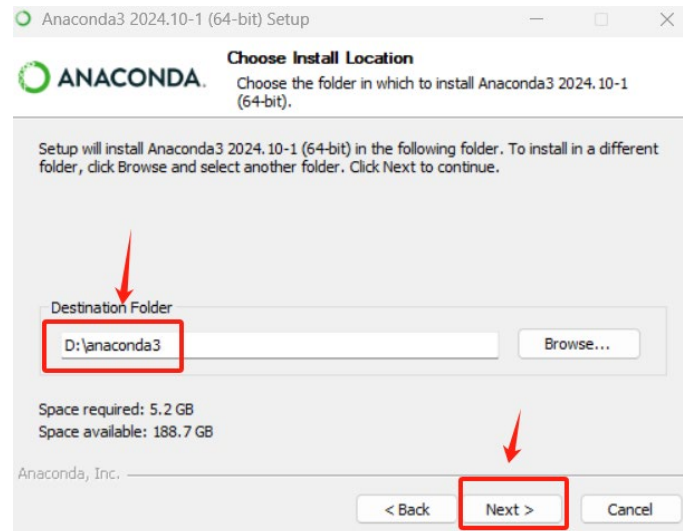
3. 选择 Just Me, 点击 next 进入下一步



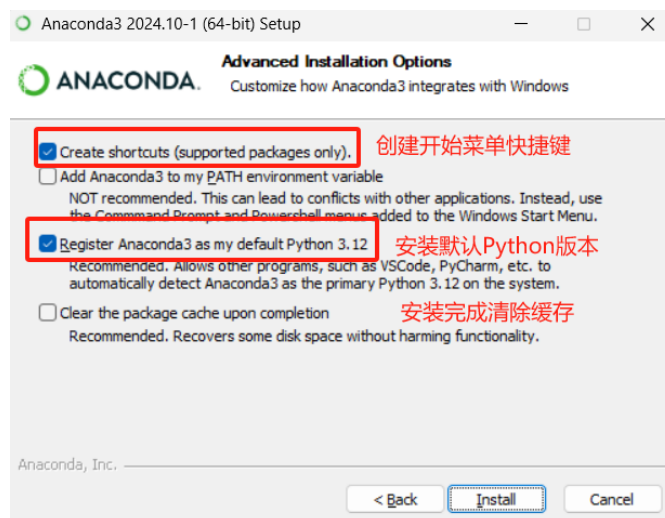
4. 安装地址，默认C盘，也可以安装在其他盘，建议安装在D盘，后续占用大量空间。点击next进入下一步

#需要记住这个安装地址，以便后续的一些操作需要。

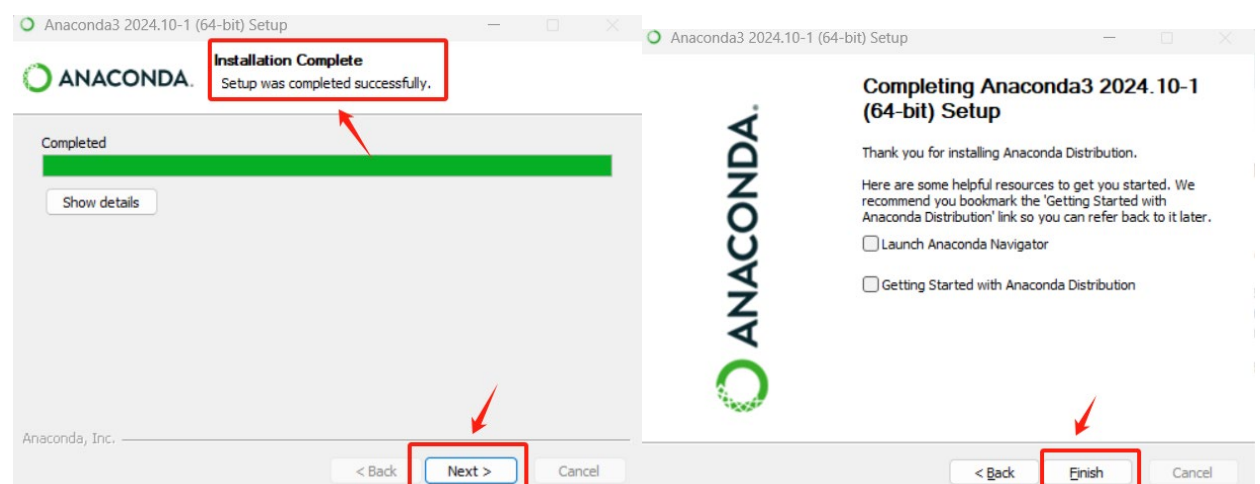
#注意尽量不要不改 **anaconda3** 这个文件夹名称。



5. 勾选选项，点击 install 执行安装过程



6. 安装成功。点击next 完成安装



7. 手动配置环境变量

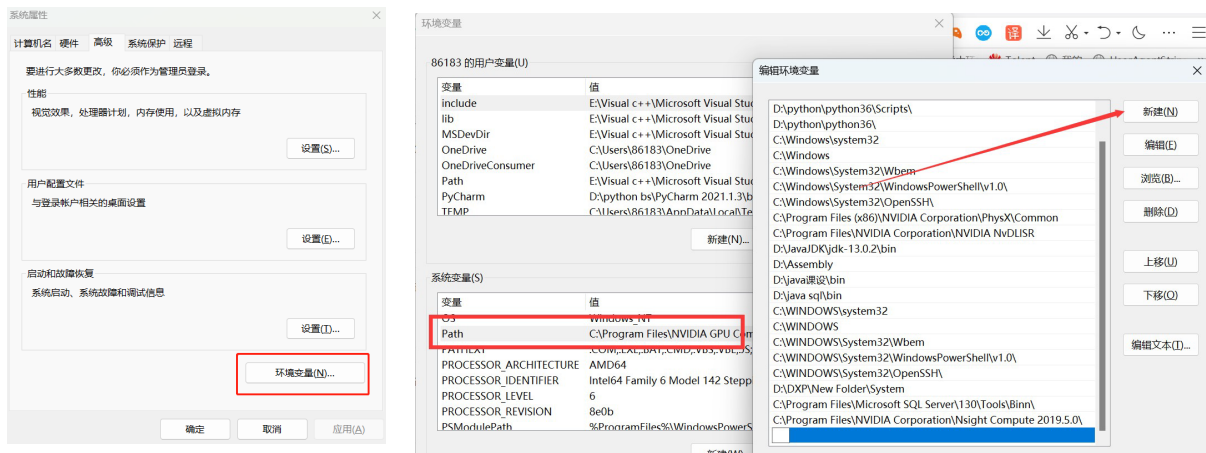
步骤：我的电脑 → 属性 → 高级系统设置 → 环境变量 → 系统变量 → path → 新建

输入如下的内容：

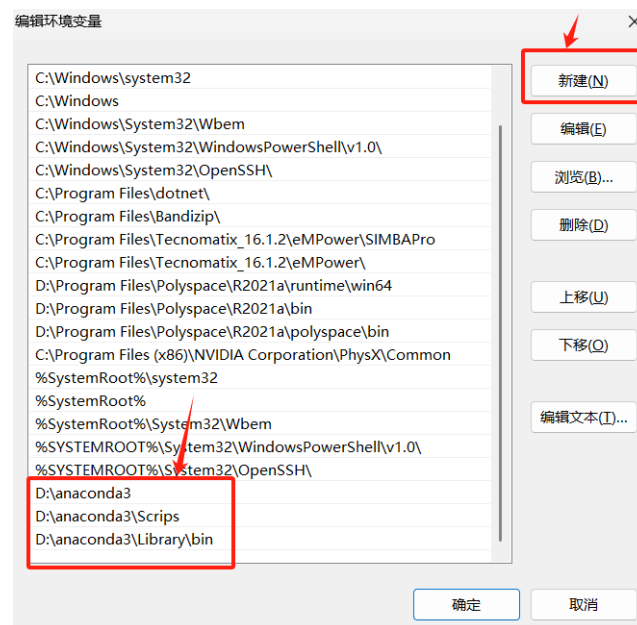
D:\anaconda3

D:\anaconda3\Scripts

D:\anaconda3\Library\bin



注意本次安装目录是 D:\anaconda3 根目录，如果不是，根据自己实际安装目录进行配置，



测试anaconda是否配置成功

Win+R → 输入cmd → 打开conda cmd (命令提示符) → 输入conda info, 查询conda 信息

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.26100.2161]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\Dell5>conda info

      active environment : None
      user config file : C:\Users\Dell5\.condarc
      populated config files : D:\anaconda3\.condarc
      conda version : 24.9.2
      conda-build version : 24.9.0
      python version : 3.12.7.final.0
      solver : libmamba (default)
      virtual packages : __archspec=1=skylake
                        __conda=24.9.2=0
                        __cuda=12.5=0
                        __win=0=0
      base environment : D:\anaconda3 (writable)
      conda av data dir : D:\anaconda3\etc\conda
      conda av metadata url : None
      channel URLs : https://repo.anaconda.com/pkgs/main/win-64
                    https://repo.anaconda.com/pkgs/main/noarch
                    https://repo.anaconda.com/pkgs/r/win-64
                    https://repo.anaconda.com/pkgs/r/noarch
                    https://repo.anaconda.com/pkgs/msys2/win-64
                    https://repo.anaconda.com/pkgs/msys2/noarch
      package cache : D:\anaconda3\pkgs
                    C:\Users\Dell5\.conda\pkgs
                    C:\Users\Dell5\AppData\Local\conda\conda\pkgs
      envs directories : D:\anaconda3\envs
                    C:\Users\Dell5\.conda\envs
                    C:\Users\Dell5\AppData\Local\conda\conda\envs
      platform : win-64
      user-agent : conda/24.9.2 requests/2.32.3 CPython/3.12.7 Windows/11 Windows/10.0.26100 solver/1.5.8 aau/0.4.4 c/. s/. e/.
      administrator : False
      netrc file : None
      offline mode : False
```

如果提示conda 不是内部或外部命令，那一般是 anaconda 的环境变量没配置好，需要重新检查一下，或者重启电脑，尝试解决。

二. 安装 Python + 第三方库等环境

1. 在安装的anaconda应用中，选择anaconda Prompt 命令提示窗，python 以及Python 相关的第三方库等都在这个命令窗口中，通过相关的命令来进行安装和管理。



2. 默认 anaconda 的源文件或者第三方库的下载地址是在官网服务器中，但是官网服务器的下载速度会很慢。为了改善下载速度，一般我们采用国内的清华大学或者同方大学的镜像下载地址，这样下载速度会大幅度提高。

更改 anaconda 源地址

在Anaconda prompt 中操作如下命令：

● 清华源地址：

```
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/main/
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/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/cloud/pytorch/
conda config --set show_channel_urls yes
```

查看是否修改好通道：

```
conda config --show channels
```

删除单个链接的命令：

```
conda config --remove channels your_url #你想移除的指定镜像路径
```

如果需要换回conda 的默认源，直接删除channels 即可

```
conda config --remove-key channels
```


3. 安装Python软件

Python 有多个版本，如果都放在一起，调用不同环境中的第三方的库或功能包，会出现版本不一致，导致版本冲突，管理混乱，运算出错。因此，为了便于管理，我们为不同的 python 版本创建一个独属于自己的环境房间（文件夹），管理自己的环境和第三方的库。

■ anaconda 中建立新环境命令：

`conda create --name <环境名称> python=版本号`

例如：`conda create --name python3.10 python=3.10`

```
(base) C:\Users\wuchu>conda create --name Python3.10 python=3.10
Retrieving notices: ...working... done
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: D:\anaconda3\envs\Python3.10

added / updated specs:
- python=3.10

The following packages will be downloaded:

package | build | size | url
-----|-----|-----|-----
python-3.10.9 | h4de0772_0_cpython | 15.1 MB | https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge/

Total: 15.1 MB

The following NEW packages will be INSTALLED:

bzip2          anaconda/cloud/conda-forge/win-64::bzip2-1.0.8-h8ffe710_4
ca-certificates anaconda/cloud/conda-forge/win-64::ca-certificates-2022.12.7-h5b45459_0
libffi         anaconda/cloud/conda-forge/win-64::libffi-3.4.2-h8ffe710_5
libsqlite      anaconda/cloud/conda-forge/win-64::libsqlite-3.40.0-hcfcfb64_0
libzlib        anaconda/cloud/conda-forge/win-64::libzlib-1.2.13-hcfcfb64_4
openssl        anaconda/cloud/conda-forge/win-64::openssl-3.0.8-hcfcfb64_0
pip            anaconda/cloud/conda-forge/noarch::pip-23.0-pyhd8ed1ab_0
python         anaconda/cloud/conda-forge/win-64::python-3.10.9-h4de0772_0_cpython
setuptools     anaconda/cloud/conda-forge/noarch::setuptools-67.1.0-pyhd8ed1ab_0
tk             anaconda/cloud/conda-forge/win-64::tk-8.6.12-h8ffe710_0
tzdata        anaconda/cloud/conda-forge/noarch::tzdata-2022g-h191b570_0
ucrt           anaconda/cloud/conda-forge/win-64::ucrt-10.0.22621.0-h57928b3_0
vc            anaconda/cloud/conda-forge/win-64::vc-14.3-hb6edc58_10
vs2015_runtime anaconda/cloud/conda-forge/win-64::vs2015_runtime-14.34.31931-h4c5c07a_10
wheel         anaconda/cloud/conda-forge/noarch::wheel-0.38.4-pyhd8ed1ab_0
xz            anaconda/cloud/conda-forge/win-64::xz-5.2.6-h8d14728_0

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

输入创建python3.10版本的代码，回车，搜索相关下载资源

python3.10的安装地址

输入 y, 执行安装

安装命令执行完成，Python 软件安装成功。

可以在 anaconda 安装目录下面的 envs 环境文件夹中看到我们创建的当前 python 版本的环境和各种依赖项。（比如 `D:\anaconda3\envs\Python3.10`）

如果需要安装 python 版本匹配的第三方库，同样在当前命令窗口中执行安装命令代码。

首先需要激活对应的环境，在当前环境中才能安装和管理需要的第三方库。

■ 激活当前环境：

`conda activate <环境名称>`

`conda activate python3.10`


```
Anaconda Prompt
(base) C:\Users\ DELL5>conda activate python3.10
(python3.10) C:\Users\ DELL5>
```

激活虚拟环境

查看当前环境中已安装的库包

conda list

```
Anaconda Prompt (anaconda)
(base) C:\Users\wuchu>conda activate Python3.11
(Python3.11) C:\Users\wuchu>conda list
# packages in environment at D:\anaconda3\envs\Python3.11:
#
# Name          Version      Build                                Channel
bzip2            1.0.8        h8ffe710_4                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
ca-certificates 2022.12.7    h5b45459_0                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
libffi           3.4.2        h8ffe710_5                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
libsqlite        3.40.0       hcfcfb64_0                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
libzlib          1.2.13       hcfcfb64_0                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
openssl          3.0.8        hcfcfb64_0                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
pip              23.0        pyhd8ed1ab_0                        https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
python           3.11.0       hf16a7b_0_cpython                  https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
setuptools       67.1.0       pyhd8ed1ab_0                        https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
tk               8.6.12       h8ffe710_0                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
tzdata          2022g        h191b570_0                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
ucrt             10.0.22621.0 h57928b3_0                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
vc               14.3         hb6edc58_10                         https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
vs2015_runtime  14.34.31931 h4c5c07a_10                         https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
wheel            0.38.4       pyhd8ed1ab_0                        https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
xz               5.2.6        h8d14728_0                          https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
(Python3.11) C:\Users\wuchu>
```

输入查看代码，回车

已安装的库和版本号

安装第三方库或者功能包

命令：conda install <第三方库的名称>

常用的第三方库：numpy, matplotlib, pandas, scikit-learn 等

conda install numpy (库的名称)

```
(Python3.11) C:\Users\wuchu>conda install numpy
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: D:\anaconda3\envs\Python3.11

added / updated specs:
- numpy

和Python3.11版本相匹配的numpy库的版本1.24.2

The following packages will be downloaded:

package | build | size | channel
--- | --- | --- | ---
numpy-1.24.2 | py311h0b4df5a_0 | 6.7 MB | https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
python_abi-3.11 | 3_cp311 | 6 KB | https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
Total: 6.7 MB

The following NEW packages will be INSTALLED:

intel-openmp | anaconda/cloud/conda-forge/win-64::intel-openmp-2023.0.0-h57928b3_25922
libblas | anaconda/cloud/conda-forge/win-64::libblas-3.9.0-16_win64_mkl
libblosc | anaconda/cloud/conda-forge/win-64::libblosc-2.8.0-h039e092_1
libiconv | anaconda/cloud/conda-forge/win-64::libiconv-1.17-h8ffe710_0
liblapack | anaconda/cloud/conda-forge/win-64::liblapack-3.9.0-16_win64_mkl
libxml2 | anaconda/cloud/conda-forge/win-64::libxml2-2.10.3-hc3477c8_0
mkl | anaconda/cloud/conda-forge/win-64::mkl-2022.1.0-h6a75c08_874
numpy | anaconda/cloud/conda-forge/win-64::numpy-1.24.2-py311h0b4df5a_0
pthreads-win32 | anaconda/cloud/conda-forge/win-64::pthreads-win32-2.9.1-hfa6e2cd_3
python_abi | anaconda/cloud/conda-forge/win-64::python_abi-3.11-3_cp311
tbb | anaconda/cloud/conda-forge/win-64::tbb-2021.7.0-h91493d7_1

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

输入安装库的代码，回车，搜索相关下载资源

输入 y, 回车，执行安装

■ 移除第三方库、移除环境

1) 移除某个第三方库

命令: `conda remove --name <环境名称> <第三方库名称>`

例如: 移除在python3.10环境中numpy这个库

```
conda remove --name python3.10 numpy
```

2) 移除环境

命令: `conda remove --name <环境名称> -all`

例如: 移除python3.10虚拟环境

```
conda remove --name python3.10 -all
```

■ 安装 PyTorch 人工智能框架

注意: 在安装Pytorch之前, 一定要确认是在anaconda已建好的我们需要的环境中安装, 否则就会安装在其他房间或者大厅(base)中, 导致无法使用Pytorch。

Pytorch 官网: <https://pytorch.org/>

选择我们需要的配置: 注意这里我们采用CPU模式。

当前Pytorch 只能采用pip模式安装

PyTorch Build	Stable (2.6.0)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
Compute Platform	CUDA 11.8	CUDA 12.4	CUDA 12.6	CPU
Run this Command:	<pre>pip3 install torch torchvision torchaudio</pre>			

复制代码

复制安装命令:

```
pip3 install torch torchvision torchaudio
```

```
Anaconda Prompt - conda ins × + v
(1) 激活建立的环境

(base) C:\Users\Dell5>conda activate python3.10

(python3.10) C:\Users\Dell5>conda install pytorch torchvision torchaudio cpuonly -c pytorch
Channels:
 - pytorch
 - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main (2) 复制pytorch安装命令
 - https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/pytorch
 - 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/pkgs/free
 - defaults
 - nvidia
Platform: win-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

environment location: D:\anaconda3\envs\python3.10

added / updated specs:
 - cpuonly
 - pytorch
 - torchaudio
 - torchvision

需要下载安装的文件

The following packages will be downloaded:

package | build | size | package
-----|-----|-----|-----
cpuonly-2.0 | 0 | 2 KB | pytorch
pytorch-2.5.1 | py3.10_cpu_0 | 147.4 MB | pytorch
pytorch-mutex-1.0 | cpu | 3 KB | pytorch
torchaudio-2.5.1 | py310_cpu | 5.8 MB | pytorch
torchvision-0.20.1 | py310_cpu | 6.5 MB | pytorch
-----|-----|-----|-----
Total: | | 159.7 MB |

The following NEW packages will be INSTALLED:
The following NEW packages will be INSTALLED:
cpuonly pytorch/noarch::cpuonly-2.0-0

The following packages will be SUPERSEDED by a higher-priority channel:
certifi anaconda/cloud/conda-forge/noarch::ce* --> anaconda/pkgs/main/win-64::certifi-2024.8.30-py310haa95

The following packages will be DOWNGRADED:
pytorch 2.5.1-py3.10_cuda12.4_cudnn9_0 --> 2.5.1-py3.10_cpu_0
pytorch-mutex 1.0-cuda --> 1.0-cpu
torchaudio 2.5.1-py310_cu124 --> 2.5.1-py310_cpu
torchvision 0.20.1-py310_cu124 --> 0.20.1-py310_cpu

Proceed ([y]/n)? | 输入Y, 回车, 安装pytorch
```

注意：安装过程中，会由于网络波动等各种问题，出现安装不成功。如果安装过程失败，出现报错，则重复上述过程，再次运行安装。

```
Anaconda Prompt

The following packages will be DOWNGRADED:

libblas                      3.9.0-25_win64_mkl --> 3.9.0-1_h8933c1f_netlib
libcblas                    3.9.0-25_win64_mkl --> 3.9.0-8_h719fc58_netlib
liblapack                   3.9.0-25_win64_mkl --> 3.9.0-8_h719fc58_netlib
mkl                         2024.2.2-h66d3029_14 --> 2023.1.0-h6a75c08_48682

Proceed ([y]/n)? y

Downloading and Extracting Packages:

CondaHTTPError: HTTP 000 CONNECTION FAILED for url <https://conda.anaconda.org/pytorch/win-64/pytorch-2.5.1-py3.10_cuda>
Elapsed: -

An HTTP error occurred when trying to retrieve this URL.
HTTP errors are often intermittent, and a simple retry will get you on your way.

CondaHTTPError: HTTP 000 CONNECTION FAILED for url <https://conda.anaconda.org/pytorch/win-64/pytorch-2.5.1-py3.10_cuda>
Elapsed: -

An HTTP error occurred when trying to retrieve this URL.
HTTP errors are often intermittent, and a simple retry will get you on your way.

(python3.10) C:\Users\Dell5>
```

网络连接超时间

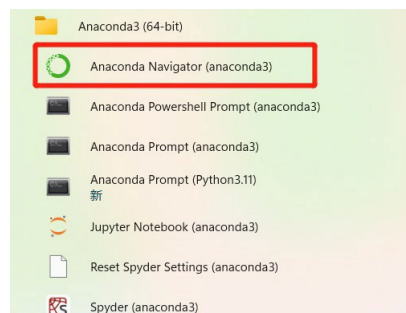
■ 安装 OpenCV

(1) 方法1:

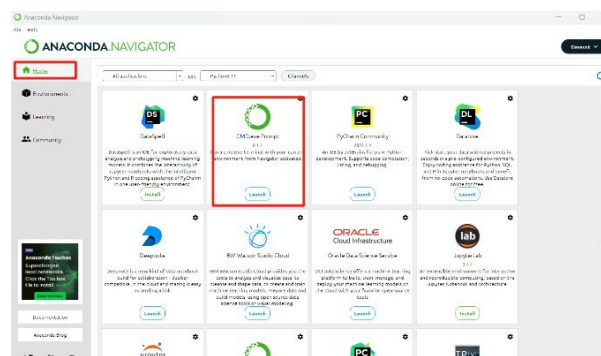
`conda install opencv`

(2) 方法2: 如果上述方法无法正常安装, 采用下面方法

打开 anaconda 环境编辑器: Anaconda Navigator



下载并打开 CMD.exe Prompt 或者 Powershell Prompt 命令窗口。



使用命令: `conda activate <环境名称>` 激活环境位置

```
C:\WINDOWS\system32\cmd. x + v
Microsoft Windows [版本 10.0.22621.2428]
(c) Microsoft Corporation。保留所有权利。

(Python3.11) C:\Users\wuchu>conda activate Pytorch
(Pytorch) C:\Users\wuchu>
```

使用安装命令 `pip install opencv-python` 安装 opencv 第三方库
下载，并安装成功，可以正常使用

```
(python3.10) C:\Users\wuchu>pip install opencv-python 安装命令
Collecting opencv-python
  Downloading opencv_python-4.8.1.78-cp37-abi3-win_amd64.whl.metadata (20 kB)
Requirement already satisfied: numpy>=1.21.2 in d:\anaconda3\envs\python3.10\lib\site-packages (from opencv-python) (1.26.0)
  Downloading opencv_python-4.8.1.78-cp37-abi3-win_amd64.whl (38.1 MB)
    38.1/38.1 MB 12.1 MB/s eta 0:00:00
Installing collected packages: opencv-python
Successfully installed opencv-python-4.8.1.78 安装成功
```

安装测试：运行opencv，测试是否正常安装

```
1 import cv2 as cv
2 # 读取图片
3 img = cv.imread('gear.png')
4 # 图像显示: cv.imshow('窗口名称', 图片目标)
5 cv.imshow('picture', img)
6 cv.waitKey() # 给定时间(ms)内等待用户按键触发
```

运行本次代码，看是否能够正常读取图片。

三. 安装 PyCharm 集成开发环境 (IDE)

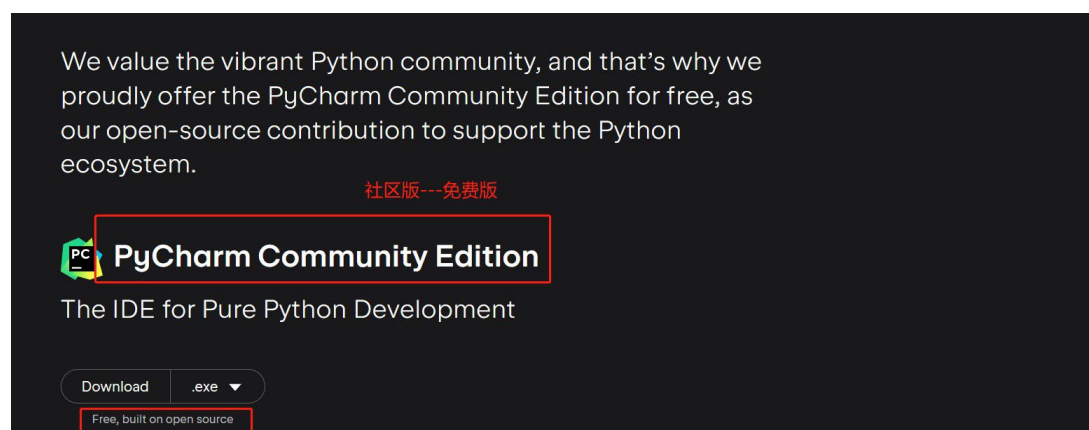
下载地址: <https://www.jetbrains.com/pycharm/>

Pycharm 有两个版本, 专业版(收费版)和社区版(免费版), 推荐使用社区版本, 这个版本免费。

专业版

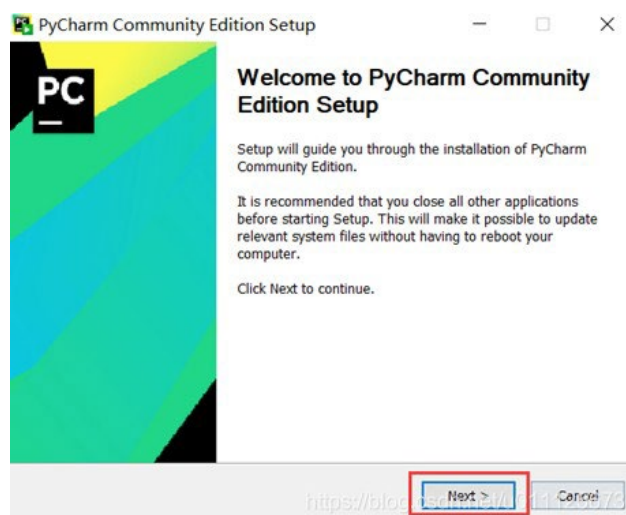


社区版

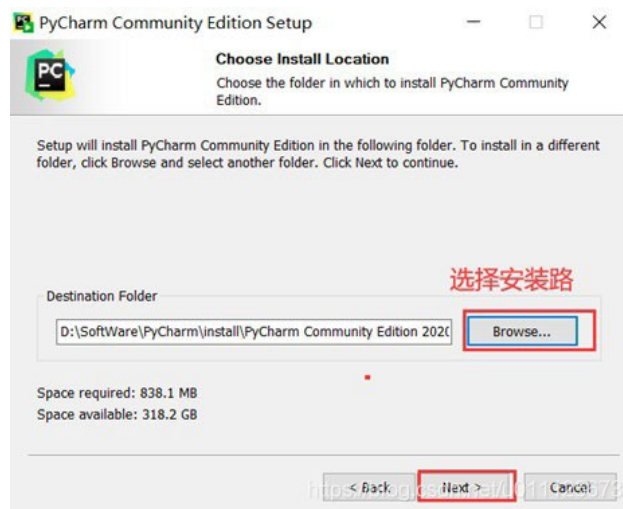


安装过程

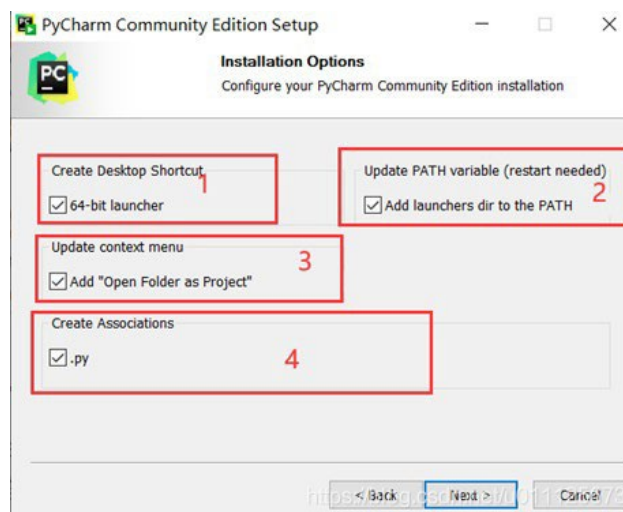
1、双击已下载的 PyCharm 安装包, 出现安装界面, 点击“next”



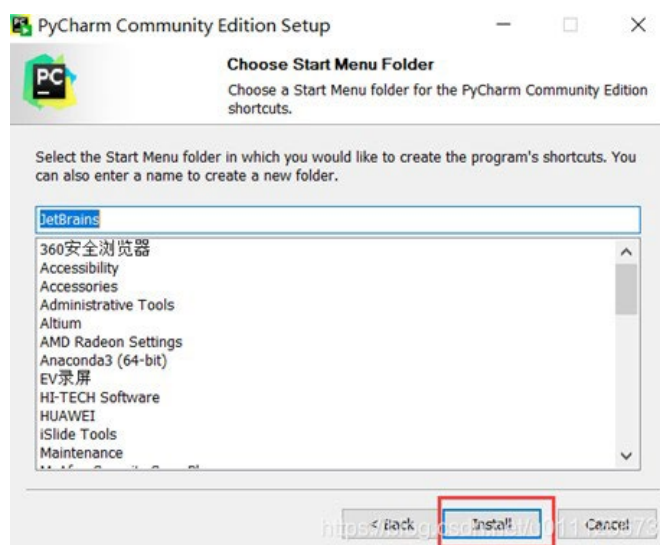
2、选择安装目录，Pycharm 需要的内存较多，建议将其安装在 D 盘或者 E 盘，不建议放在系统盘 C 盘。



3、进行相关设置，如果你无特殊需要按照图中勾选即可。



4、默认即可，点击 install。



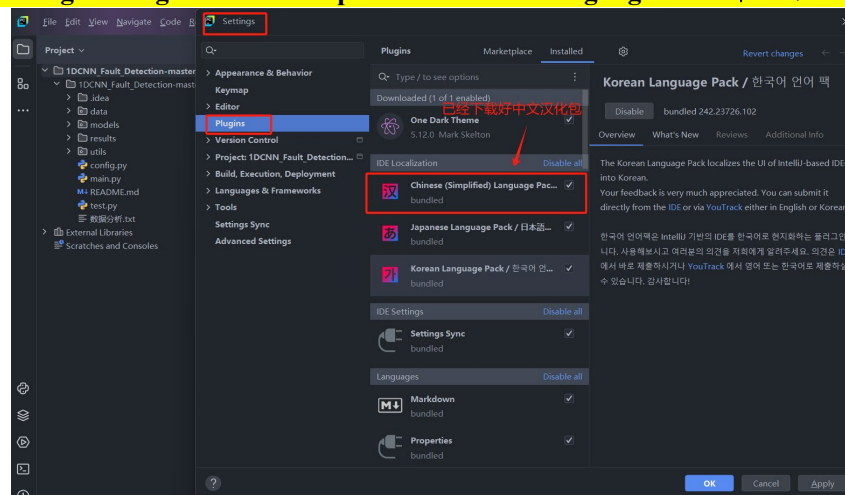
5、安装完成，需要重启电脑，即可正常使用 PyCharm。



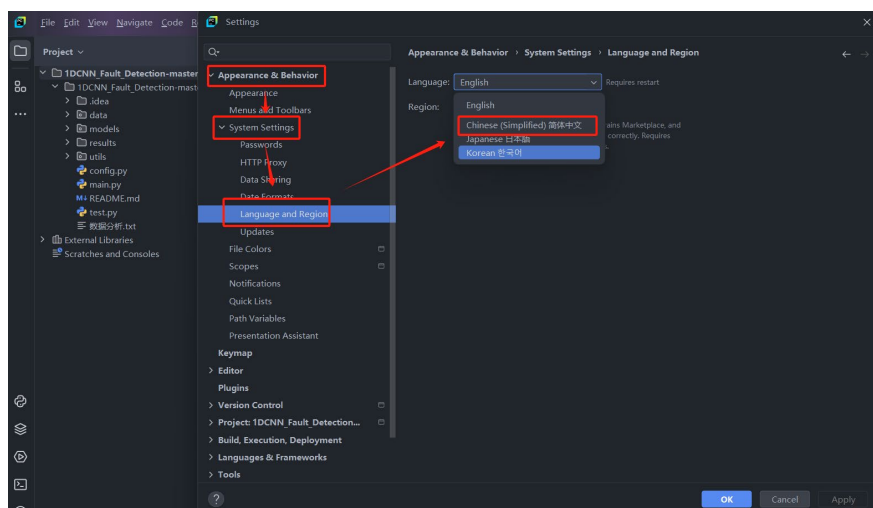
6、安装中文插件，是 PyCharm 中文界面，便于使用。

中文汉化过程：

File ---Setting --- Plugins ----Marketplace--- Chinese Language Pake/中文语言包---安装

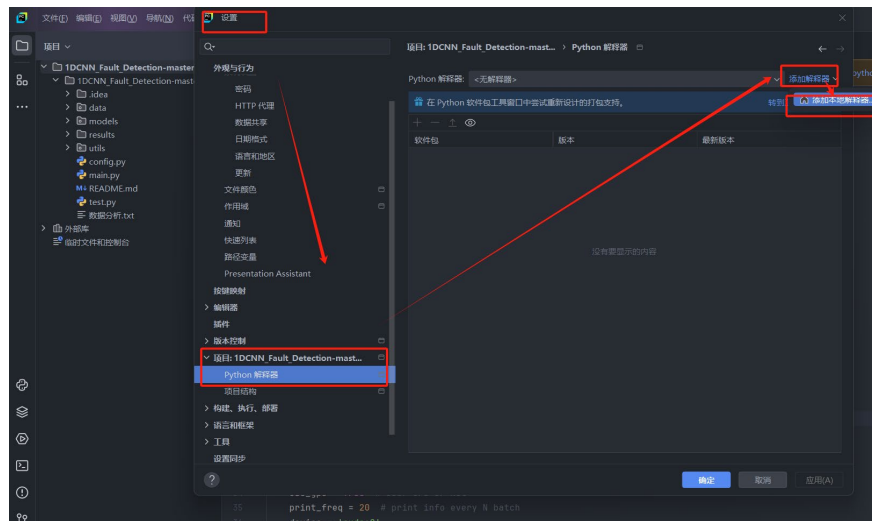


选择中文语言：File → Settings的弹框中选择Appearance&Behavior→System Settings→Language and Region中选择简体中文→apply，重启生效。



7、Python编译器的选择和配置：

【File】文件→【Settings】设置→【Project】项目→【Project Interpreter】解释器→添加解释器→添加本地解释器



选择<conda环境>方式→选择conda执行文件→ D:\anaconda3\Scripts\conda.exe →加载环境
→ 选择已经安装的python.exe执行文件

