

2.1 个人电脑的环境配置

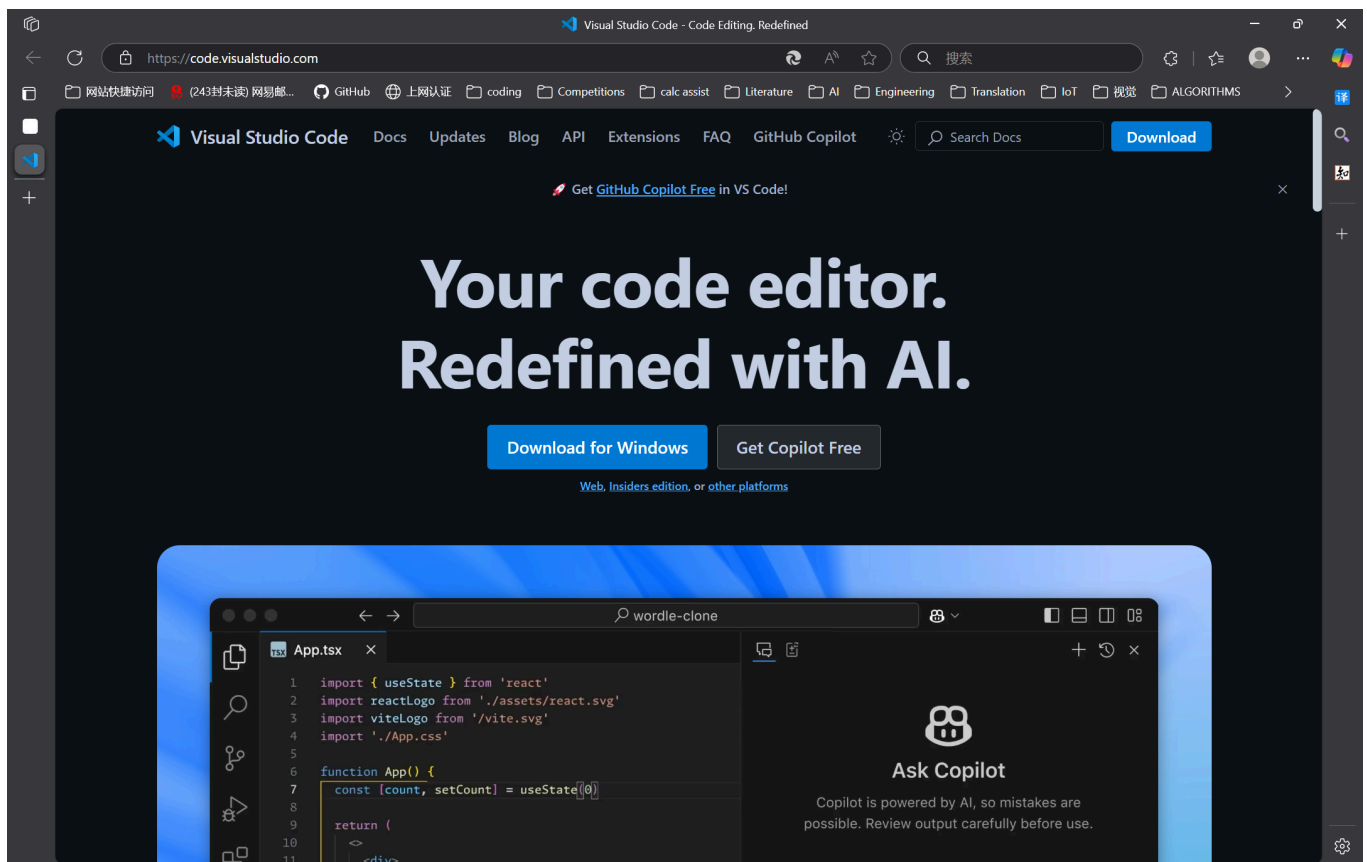
在Windows笔记本上配置VSCode，创建并启动虚拟环境

1 - 安装 VSCode + Anaconda

下载vscode

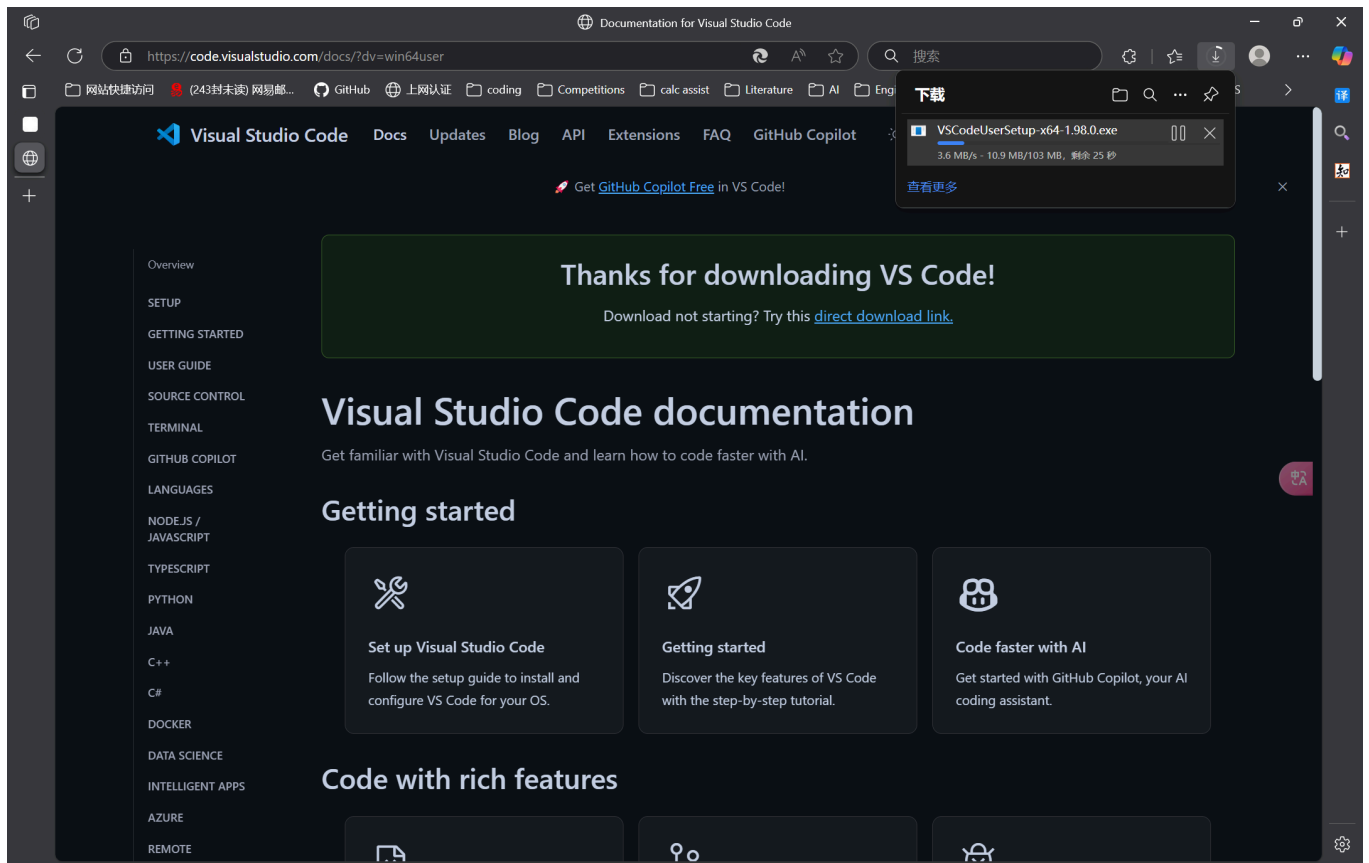
1. 打开VSCode官网链接:

[Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com)



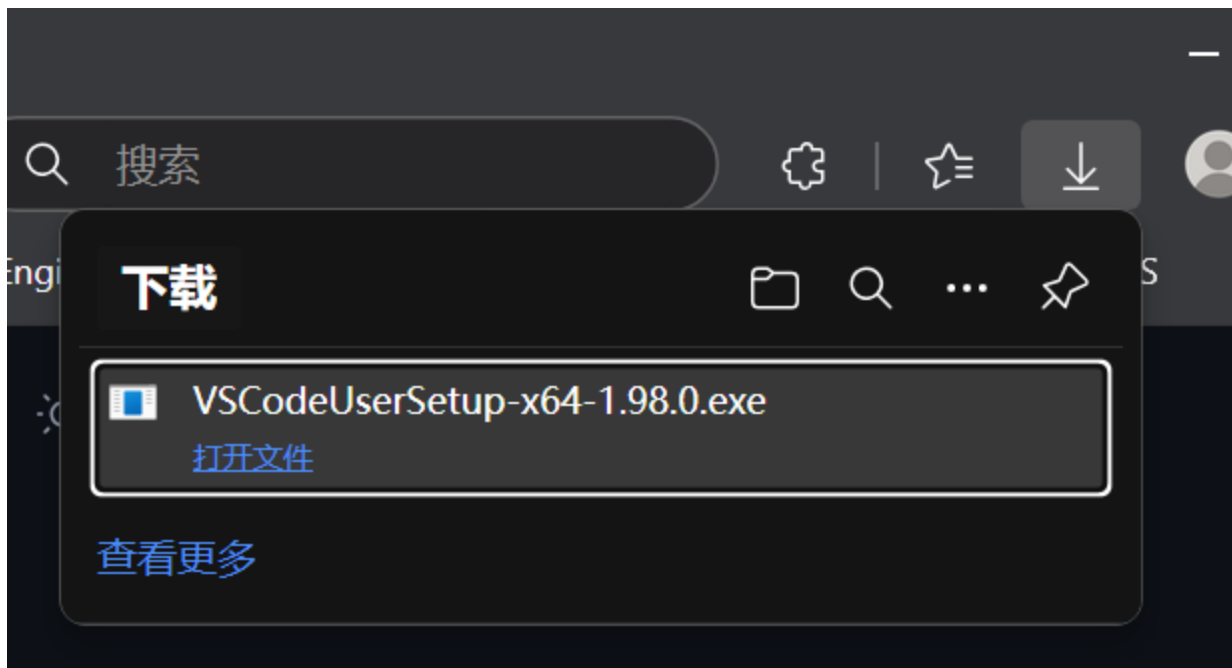
2. 点击Download for Windows

可以看到已经开始下载了

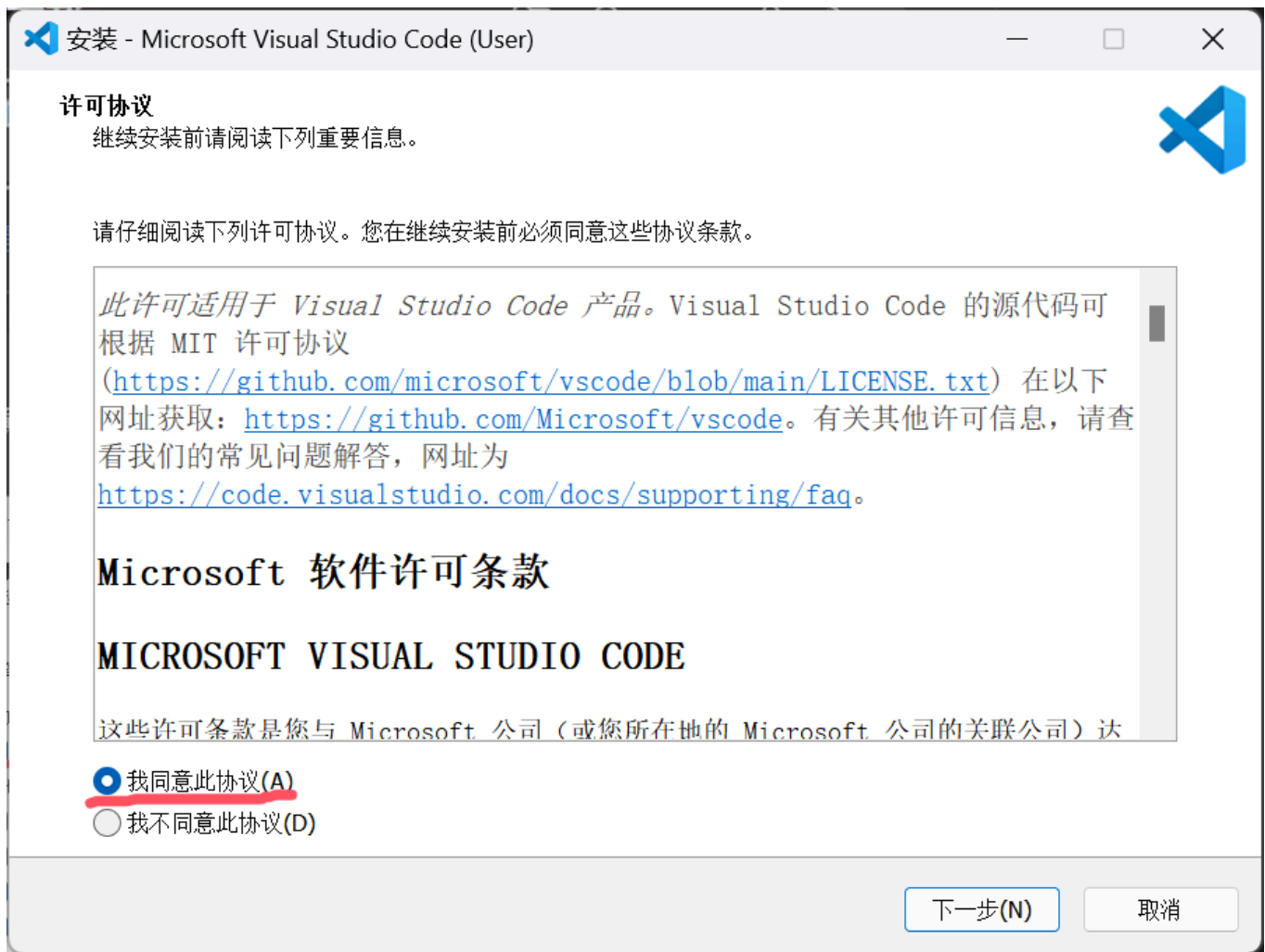


安装VSCode

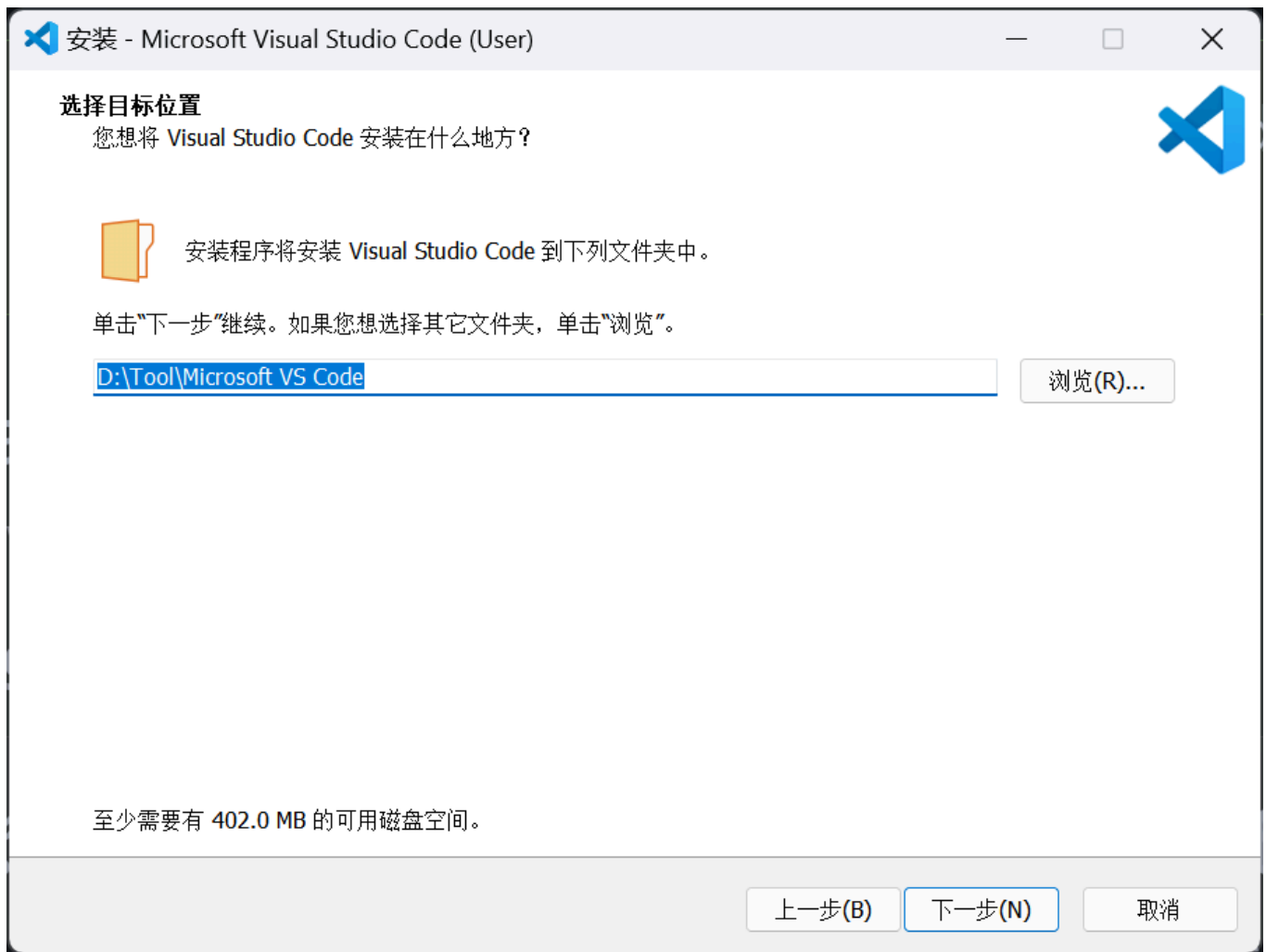
点击打开文件



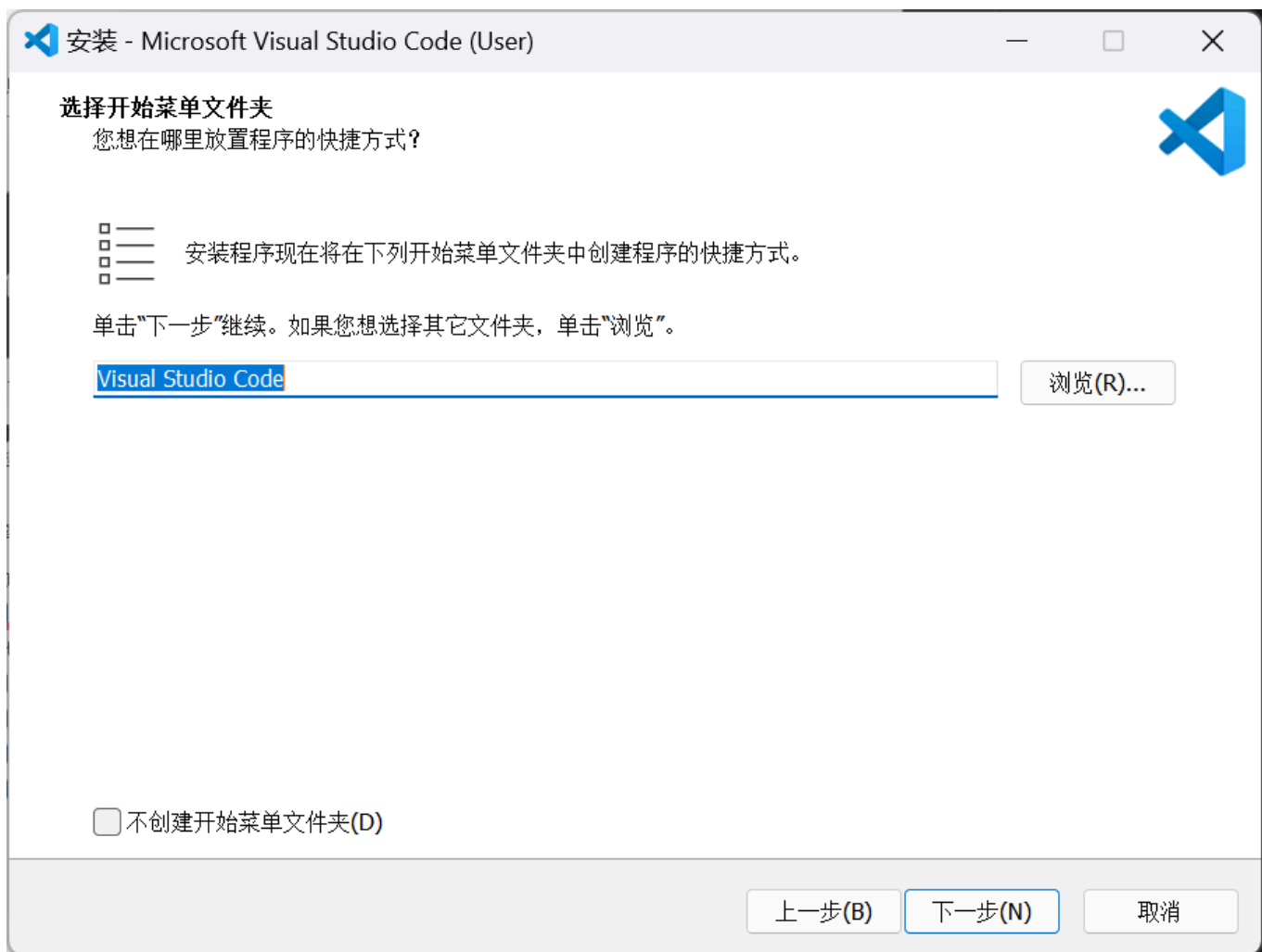
选择 "我同意此协议", 下一步



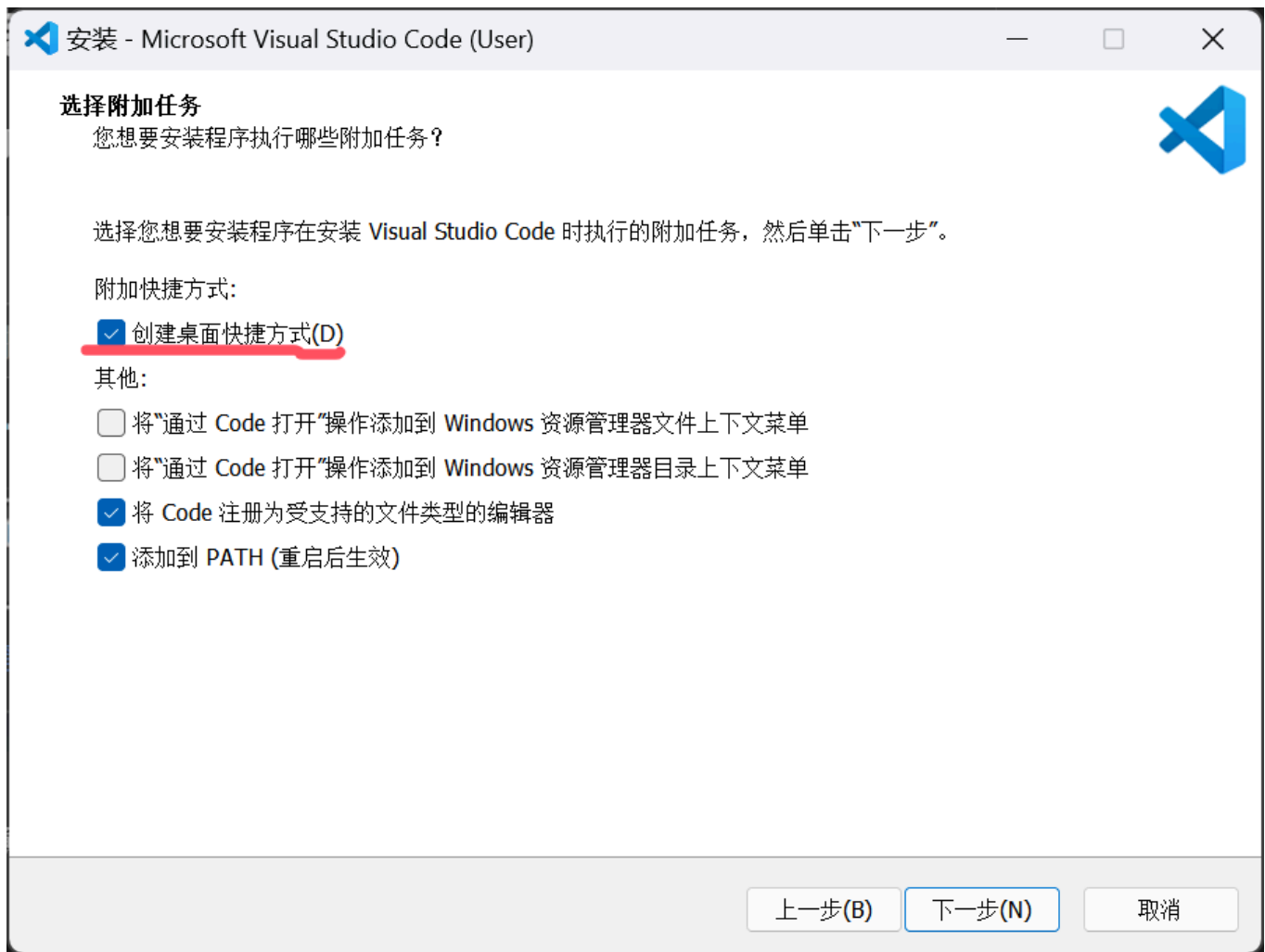
将路径替换到想放的地方, 下一步



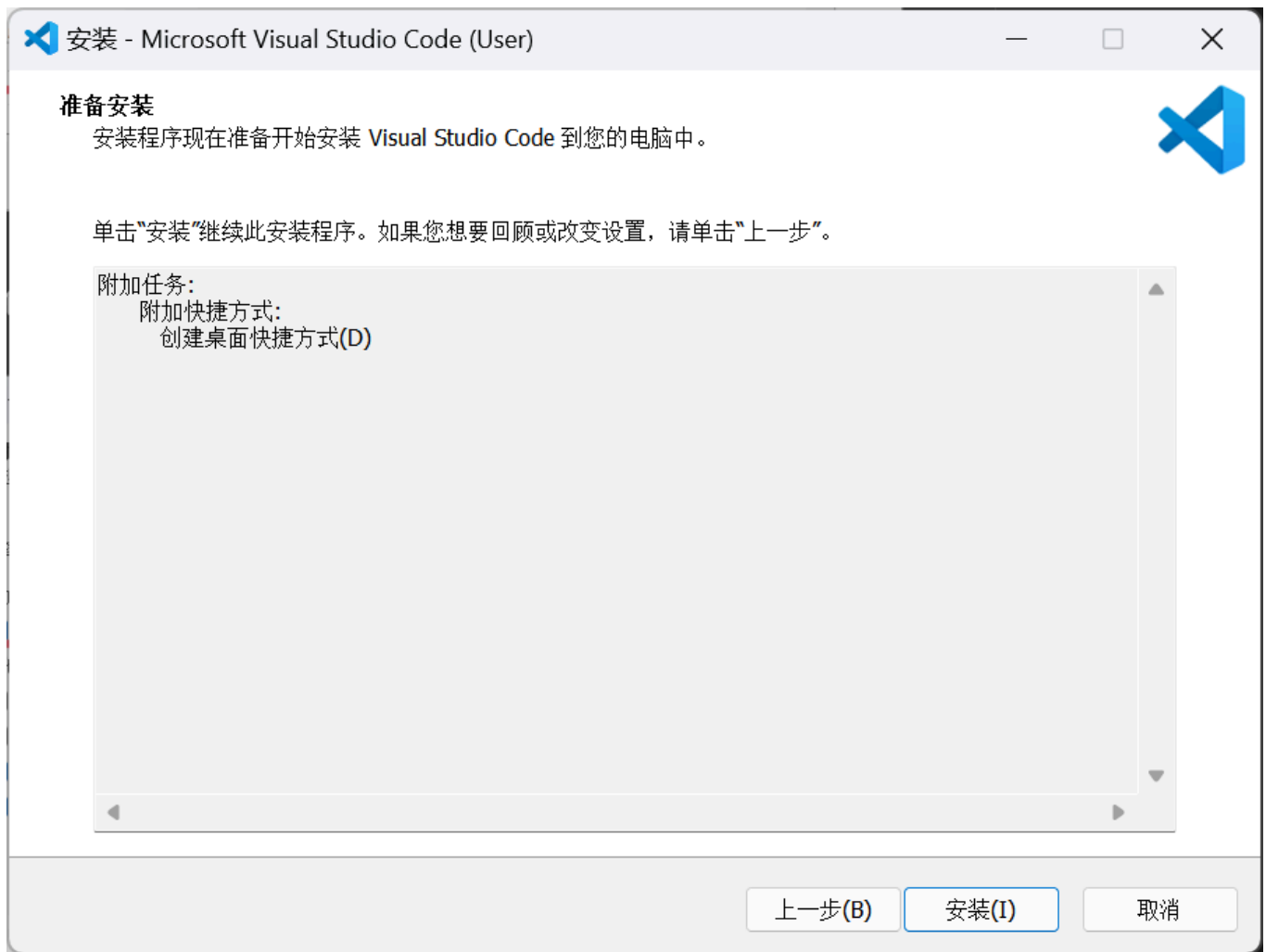
默认, 下一步



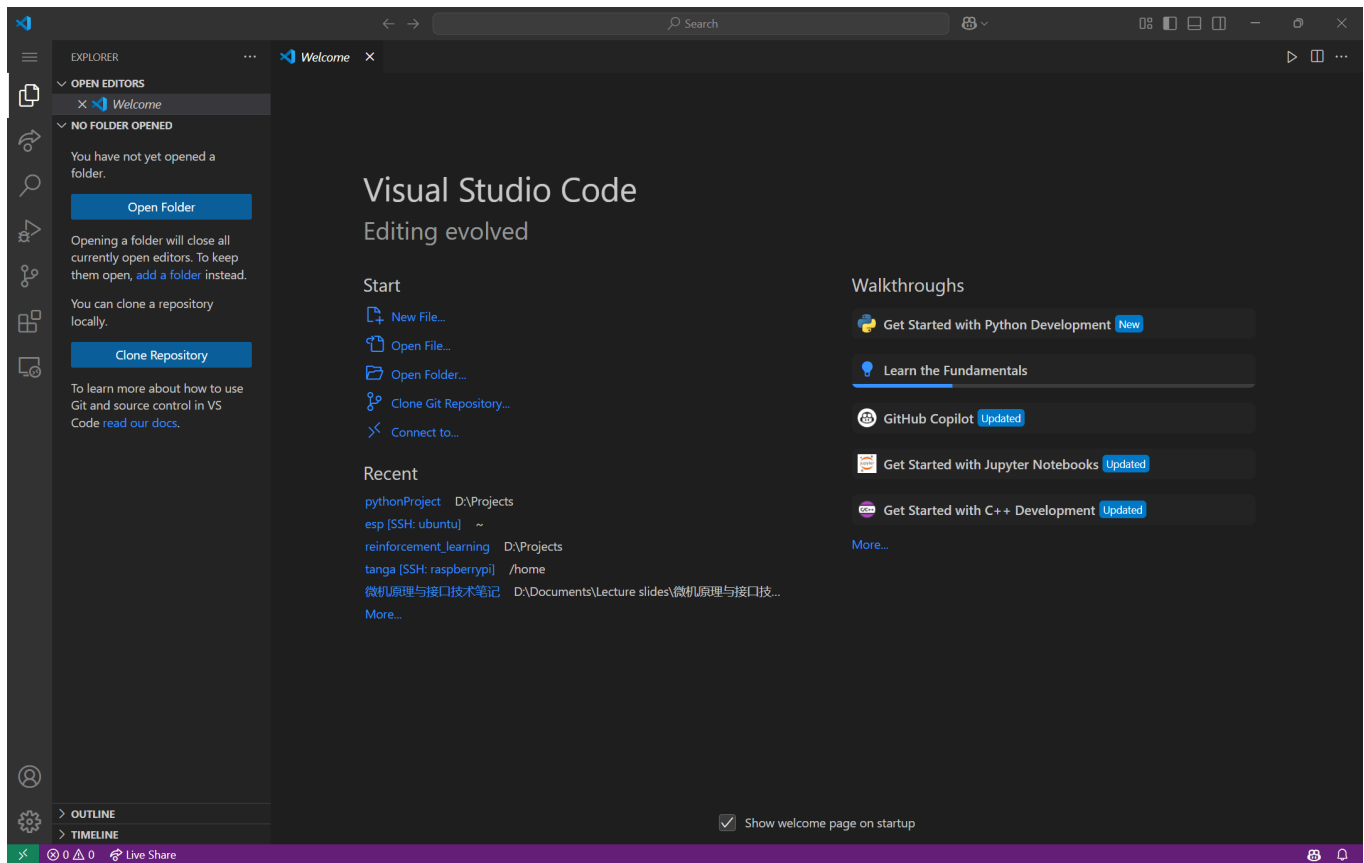
勾上 "创建桌面快捷方式", 下一步



安装



完成并打开vscode!



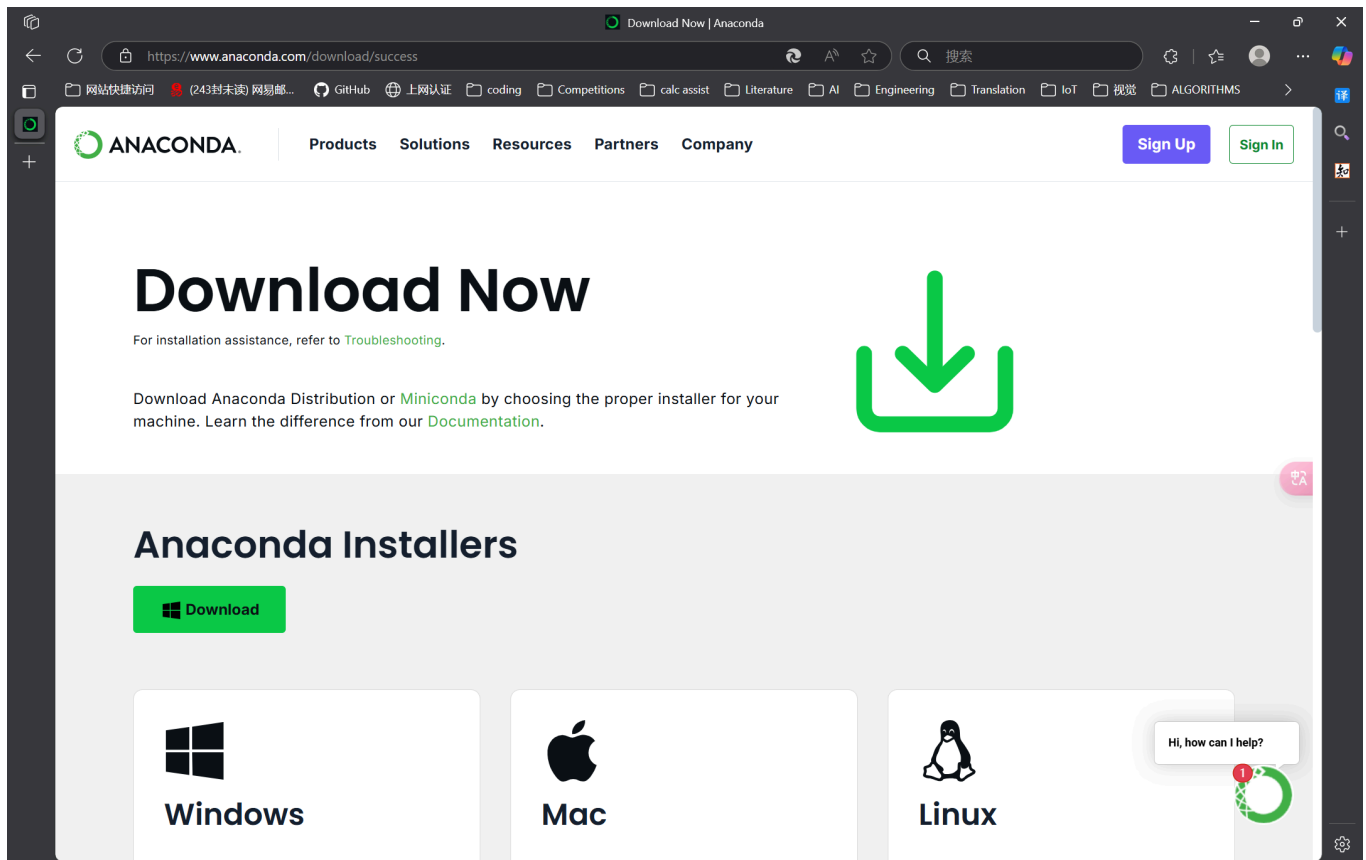
下载Anaconda

Anaconda 是一个流行的Python数据科学平台，Anaconda 可以看做Python的一个集成安装，安装它后就默认安装了python、IPython、集成开发环境Spyder和众多的包和模块，让你在管理环境和包时更加方便。

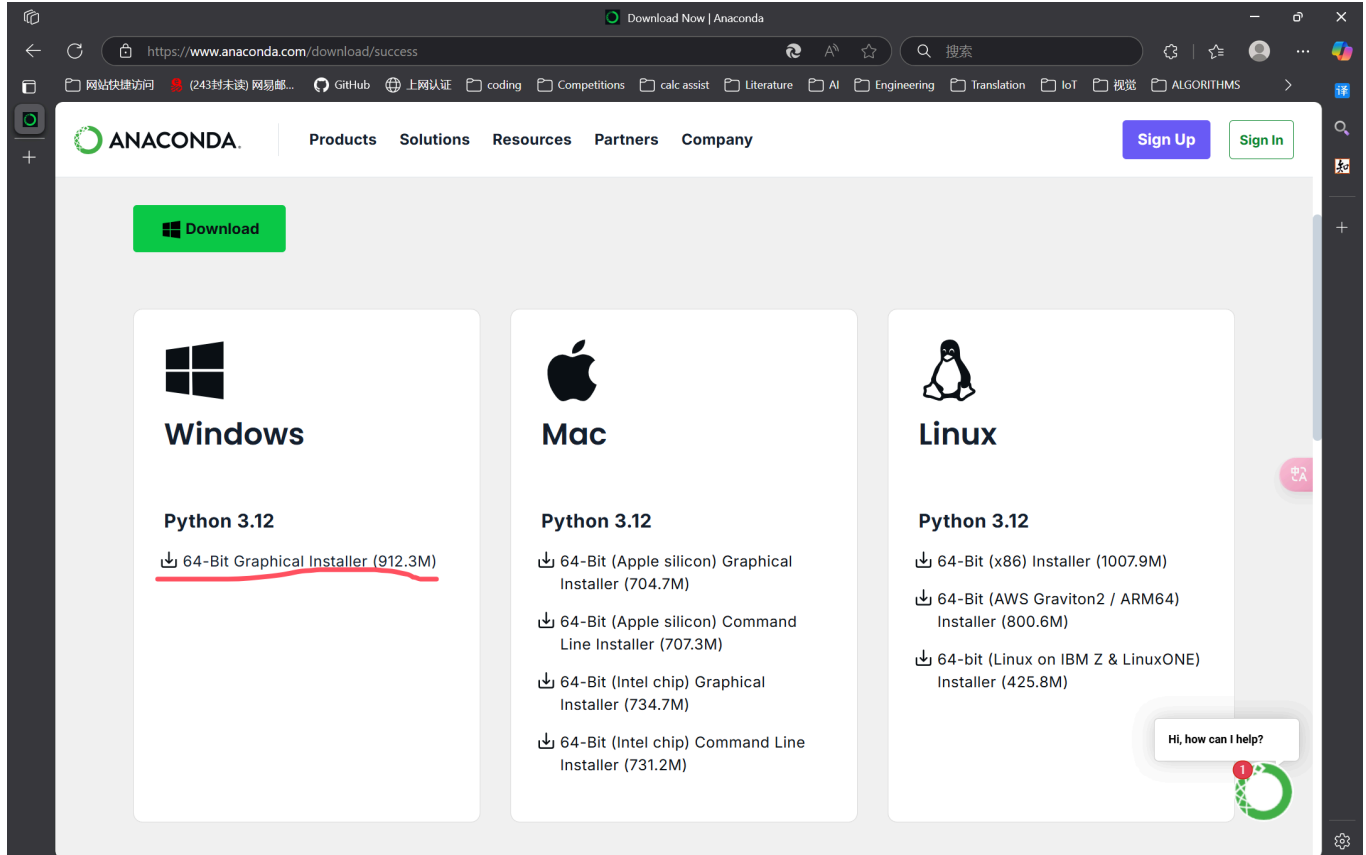
可以选择从官网或清华源下载 Anaconda 安装包，**清华源**下载速度相对较快。

1、从官网下载

打开官网[Download Now | Anaconda](https://www.anaconda.com/download/)

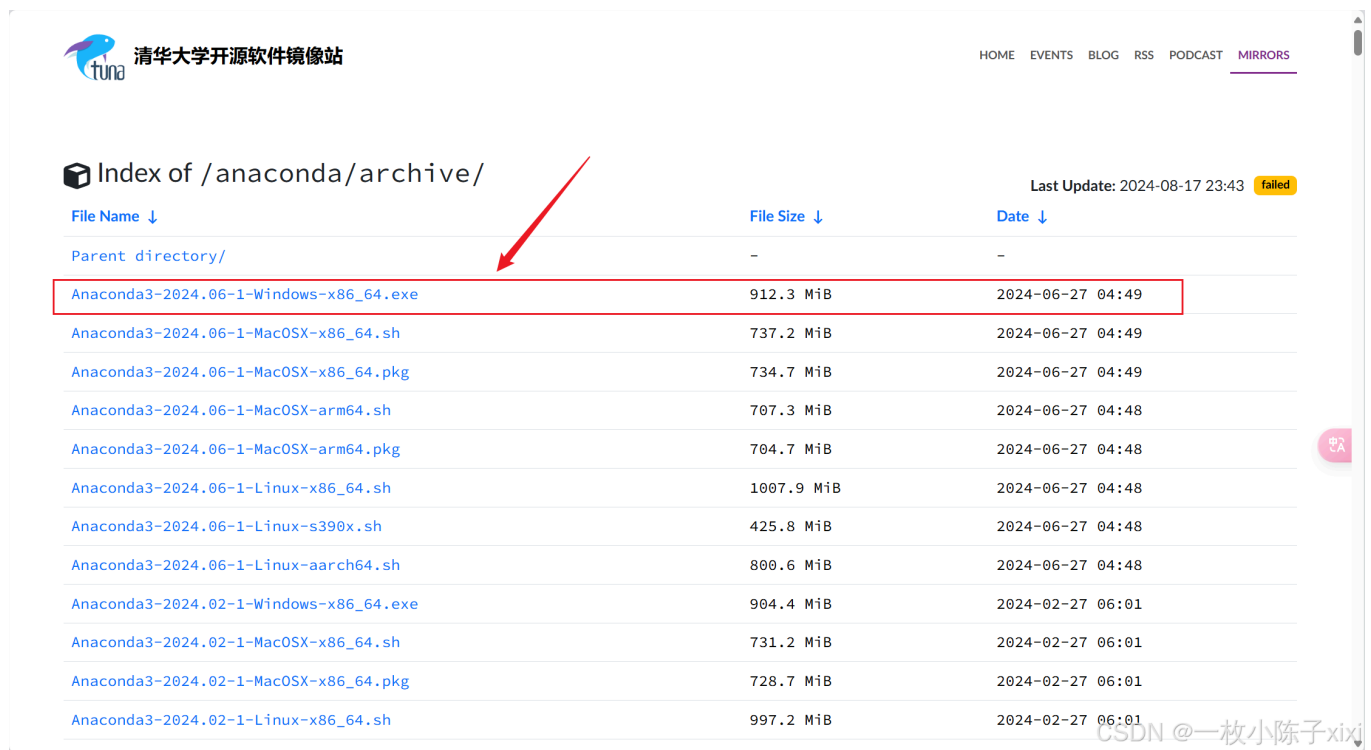


往下翻，选择windows安装包，点击下载



2、使用清华镜像源下载（下载速度快，推荐）

Anaconda清华镜像源下载



清华大学开源软件镜像站

HOME EVENTS BLOG RSS PODCAST MIRRORS

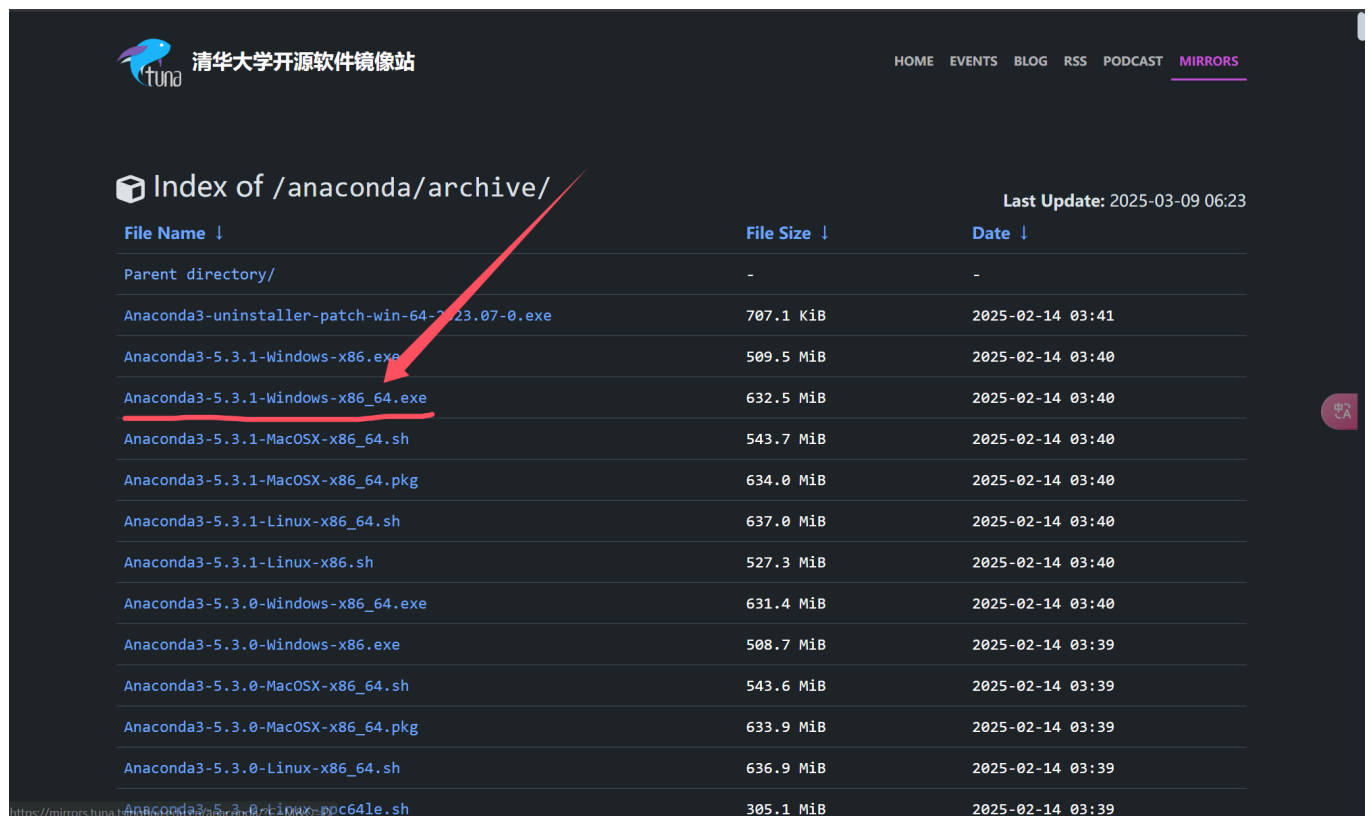
Index of /anaconda/archive/

Last Update: 2024-08-17 23:43 failed

File Name ↓	File Size ↓	Date ↓
Parent directory/	-	-
Anaconda3-2024.06-1-Windows-x86_64.exe	912.3 MiB	2024-06-27 04:49
Anaconda3-2024.06-1-MacOSX-x86_64.sh	737.2 MiB	2024-06-27 04:49
Anaconda3-2024.06-1-MacOSX-x86_64.pkg	734.7 MiB	2024-06-27 04:49
Anaconda3-2024.06-1-MacOSX-arm64.sh	707.3 MiB	2024-06-27 04:48
Anaconda3-2024.06-1-MacOSX-arm64.pkg	704.7 MiB	2024-06-27 04:48
Anaconda3-2024.06-1-Linux-x86_64.sh	1007.9 MiB	2024-06-27 04:48
Anaconda3-2024.06-1-Linux-s390x.sh	425.8 MiB	2024-06-27 04:48
Anaconda3-2024.06-1-Linux-aarch64.sh	800.6 MiB	2024-06-27 04:48
Anaconda3-2024.02-1-Windows-x86_64.exe	904.4 MiB	2024-02-27 06:01
Anaconda3-2024.02-1-MacOSX-x86_64.sh	731.2 MiB	2024-02-27 06:01
Anaconda3-2024.02-1-MacOSX-x86_64.pkg	728.7 MiB	2024-02-27 06:01
Anaconda3-2024.02-1-Linux-x86_64.sh	997.2 MiB	2024-02-27 06:01

CSDN @一枚小陈子xixi

这里选择下载最新版本的



清华大学开源软件镜像站

HOME EVENTS BLOG RSS PODCAST MIRRORS

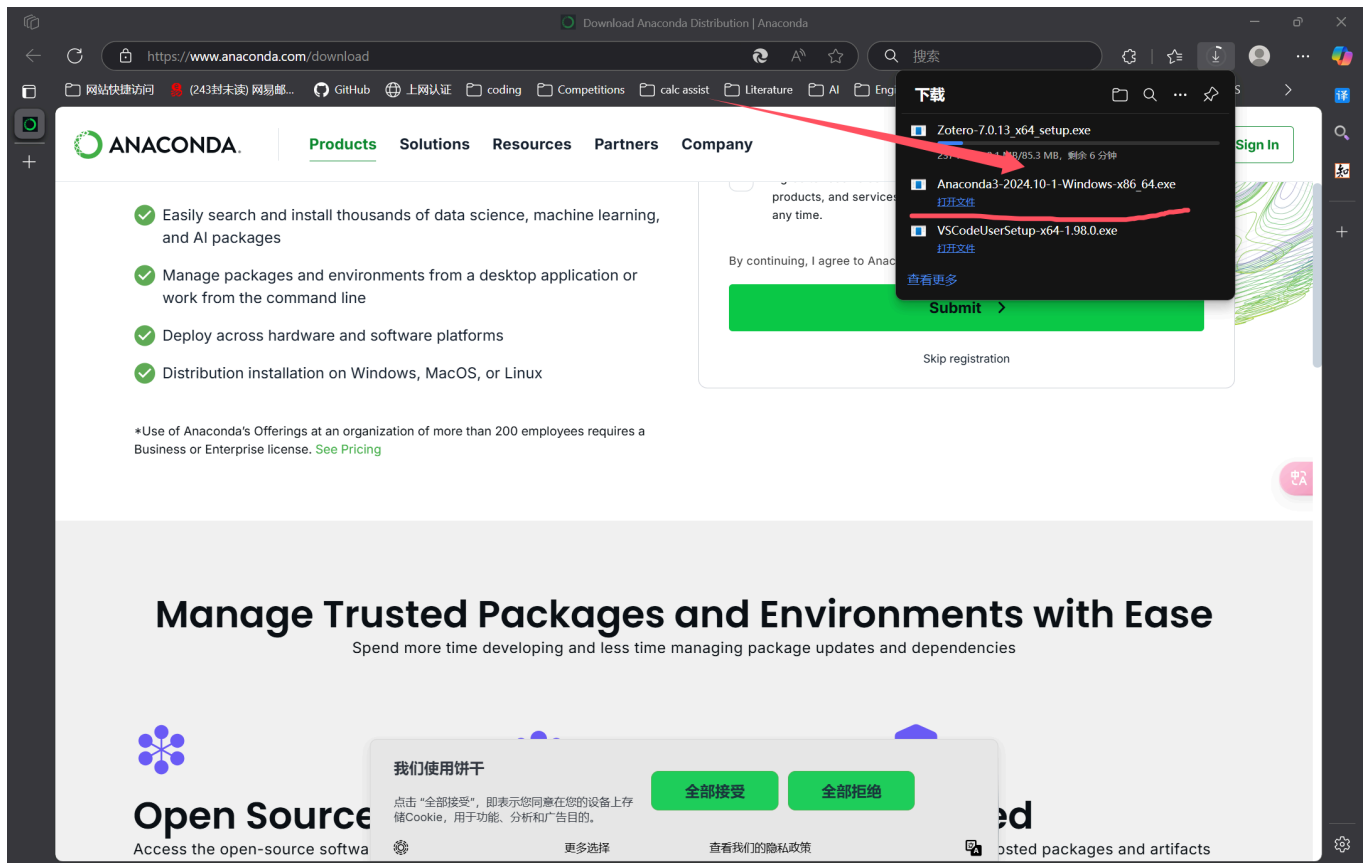
Index of /anaconda/archive/

Last Update: 2025-03-09 06:23

File Name ↓	File Size ↓	Date ↓
Parent directory/	-	-
Anaconda3-uninstaller-patch-win-64-2023.07-0.exe	707.1 KiB	2025-02-14 03:41
Anaconda3-5.3.1-Windows-x86.exe	509.5 MiB	2025-02-14 03:40
Anaconda3-5.3.1-Windows-x86_64.exe	632.5 MiB	2025-02-14 03:40
Anaconda3-5.3.1-MacOSX-x86_64.sh	543.7 MiB	2025-02-14 03:40
Anaconda3-5.3.1-MacOSX-x86_64.pkg	634.0 MiB	2025-02-14 03:40
Anaconda3-5.3.1-Linux-x86_64.sh	637.0 MiB	2025-02-14 03:40
Anaconda3-5.3.1-Linux-x86.sh	527.3 MiB	2025-02-14 03:40
Anaconda3-5.3.0-Windows-x86_64.exe	631.4 MiB	2025-02-14 03:40
Anaconda3-5.3.0-Windows-x86.exe	508.7 MiB	2025-02-14 03:39
Anaconda3-5.3.0-MacOSX-x86_64.sh	543.6 MiB	2025-02-14 03:39
Anaconda3-5.3.0-MacOSX-x86_64.pkg	633.9 MiB	2025-02-14 03:39
Anaconda3-5.3.0-Linux-x86_64.sh	636.9 MiB	2025-02-14 03:39
Anaconda3-5.3.0-Linux-x86_64.ppc64le.sh	305.1 MiB	2025-02-14 03:39

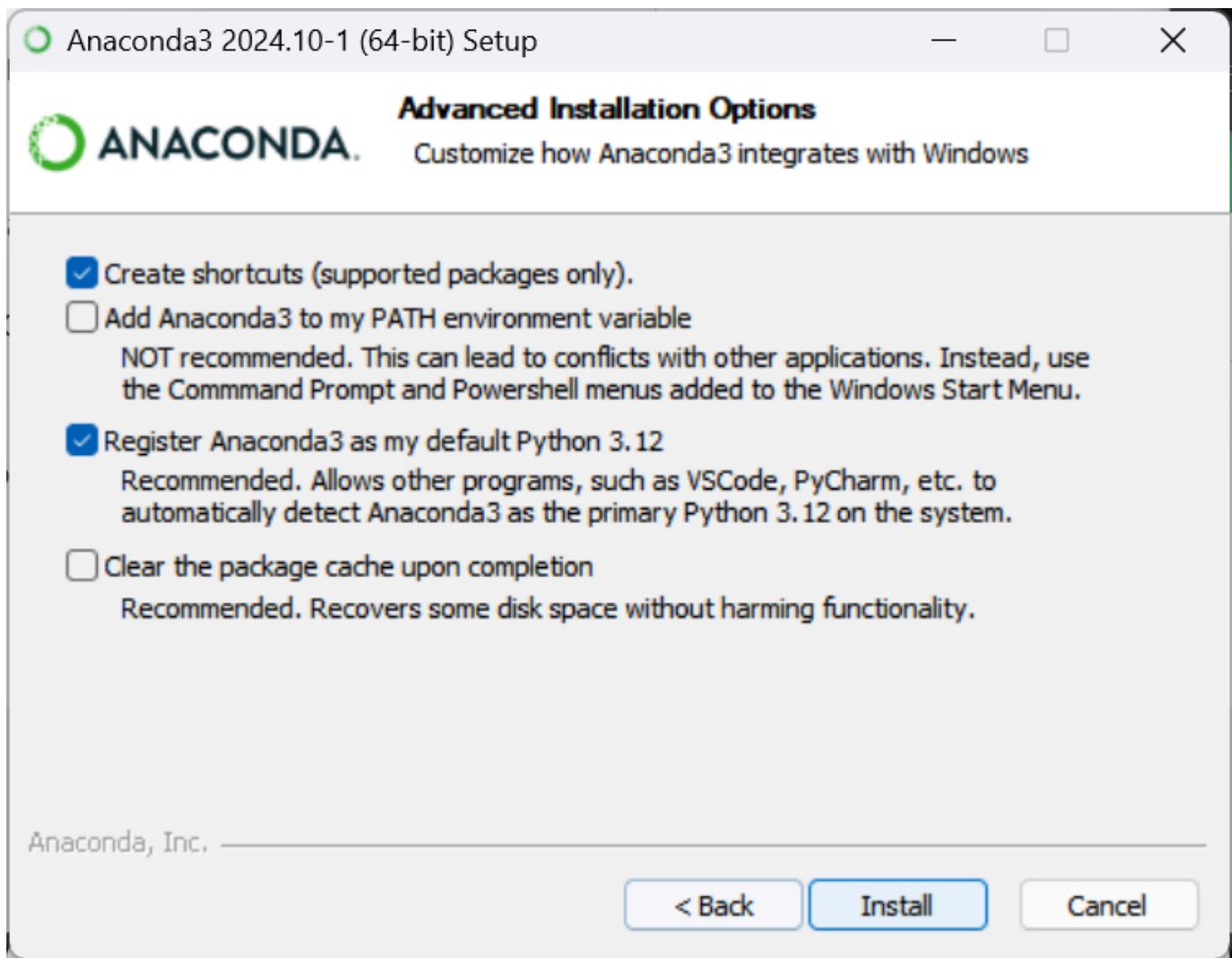
安装Anaconda

下载完成我们就可以得到一个exe文件，点击运行



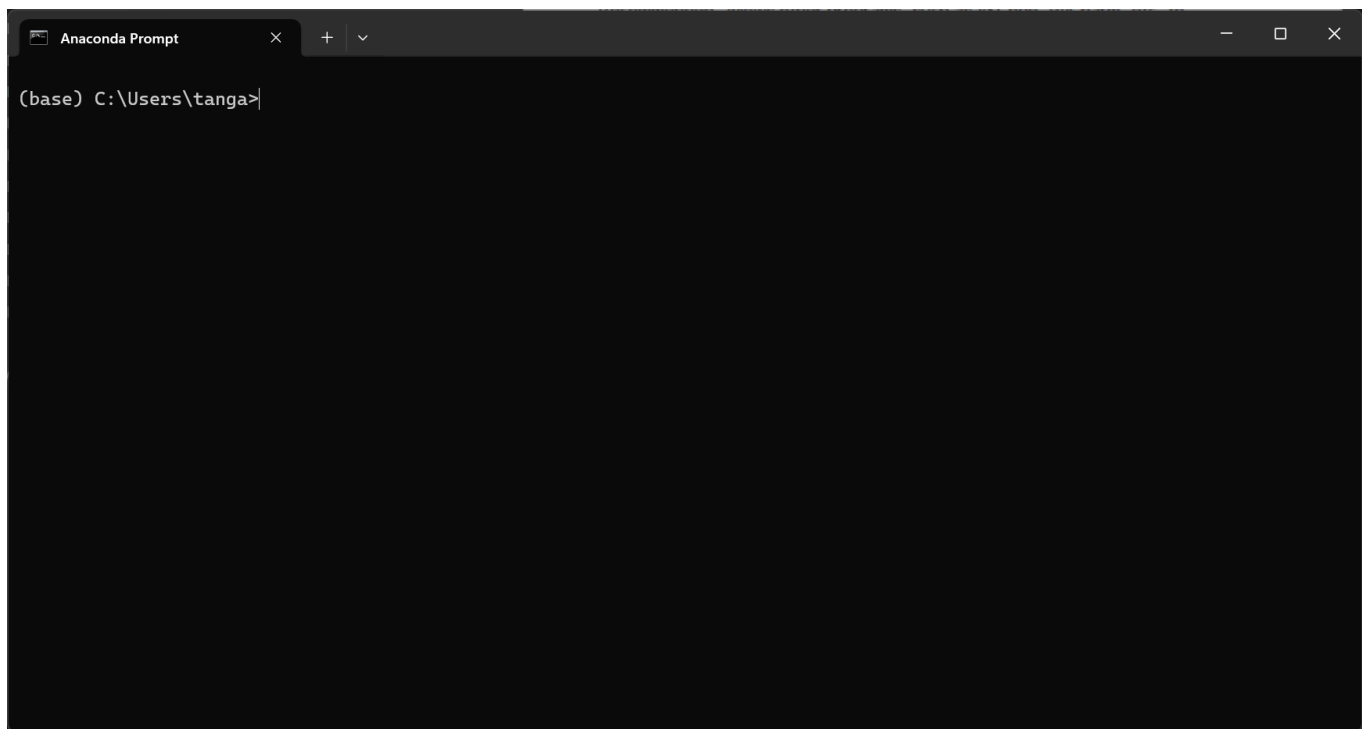
一直next，到这里，把anaconda安装目录改成自己的目录，next

默认，直接install



2 - 配置Anaconda

打开 Anaconda prompt



找到我们给的requirements.txt，复制requirements.txt的路径。

环境的搭建有两种方法，一种是使用conda源进行安装，一种是使用pip源进行安装，本次课程我们使用pip源安装作为教程

创建python虚拟环境

首先创建虚拟环境，复制粘贴下面的命令，回车。

这表示创建python版本为3.12、名字为env_name的虚拟环境。

```
conda create -n env_name python=3.12 # env_name替换成想要的名字
```

然后激活我们创建的conda环境

```
conda activate env_name # env_name替换成想要的名字
```

```
Anaconda Prompt
tk      anaconda/pkgs/main/win-64::tk-8.6.14-h0416ee5_0
tzdata  anaconda/pkgs/main/noarch::tzdata-2025a-h04d1e81_0
vc      anaconda/pkgs/main/win-64::vc-14.42-haa95532_4
vs2015_runtime anaconda/pkgs/main/win-64::vs2015_runtime-14.42.34433-he0abc0d_4
wheel   anaconda/pkgs/main/win-64::wheel-0.45.1-py312haa95532_0
xz      anaconda/pkgs/main/win-64::xz-5.6.4-h4754444_1
zlib    anaconda/pkgs/main/win-64::zlib-1.2.13-h8cc25b3_1

Proceed ([y]/n)? y

Downloading and Extracting Packages

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate env_name
#
# To deactivate an active environment, use
#
#     $ conda deactivate

(base) C:\Users\tanga>conda activate env_name

(env_name) C:\Users\tanga>
```

理论上讲, 这个时候我们就能使用

```
pip install -r "D:\workSpace\lyh\lyh_doc\requirements.txt"
```

pip源添加并安装requirements

有些软件包需要使用pip安装而不能使用conda安装, 但**Anaconda中的pip**没国内镜像源的配置, 会导致下载错误。

常用源 (清华源、阿里源、豆瓣源)

我们目前常用的三种源就是 **清华源**、**阿里源**以及**豆瓣源** 链接如下:

```
清华: https://pypi.tuna.tsinghua.edu.cn/simple
阿里: http://mirrors.aliyun.com/pypi/simple/
豆瓣: http://pypi.douban.com/simple/
```

方法一: 临时指定镜像源 (单次生效)

pip 临时换源十分简单只需要记住下面这个式子就行, 这里以清华源举例.

在 `pip install` 命令中直接添加 `-i` 参数指定镜像源:

```
pip install -i https://pypi.tuna.tsinghua.edu.cn/simple -r
"D:\workSpace\lyh\lyh_doc\requirements.txt" # 这里替换成你requirements.txt的实际
路径
```

```
Anaconda Prompt - pip insta x + v
5bb/anyio-4.9.0-py3-none-any.whl (100 kB)
Collecting distro<2,>=1.7.0 (from openai->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/12/b3/231ffd4ab1fc9d679809f356cebee130ac7daa00d6d6f3206dd4fd137e9e/distro-1.9.0-py3-none-any.whl (20 kB)
Collecting httpx<1,>=0.23.0 (from openai->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/2a/39/e50c7c3a983047577ee07d2a9e53faf5a69493943ec3f6a384bdc792deb2/httpx-0.28.1-py3-none-any.whl (73 kB)
Collecting jiter<1,>=0.4.0 (from openai->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/cd/9d/742b289016d155f49028fe1bfbef935c9bf0ffeedf77daf4a63a42bb72b/jiter-0.9.0-cp312-cp312-win_amd64.whl (207 kB)
Collecting pydantic<3,>=1.9.0 (from openai->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/f4/3c/8cc84deffa6e25d2d0c688ebb80635dfdbf1d3ea3e30c541c8cf4d860/pydantic-2.10.6-py3-none-any.whl (431 kB)
Collecting sniffio (from openai->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/e9/44/75a9c9421471a6c4805dbf2356f7c181a29c1879239abablea2cc8f38b40/sniffio-1.3.1-py3-none-any.whl (10 kB)
Collecting tqdm>4 (from openai->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/d0/30/dc54f88dd4a2b5dc8a0279bdd7270e735851848b762aeb1c1184ed1f6b14/tqdm-4.67.1-py3-none-any.whl (78 kB)
Collecting typing-extensions<5,>=4.11 (from openai->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/26/9f/ad63fc0248c5379346306f8668cda6e2e2e9c95e01216d2b8ffd9ff037d0/typing_extensions-4.12.2-py3-none-any.whl (37 kB)
Collecting notebook (from jupyter->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 2))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/f2/bf/5e5fcf79c559600b738d7577c8360bfd4cfa705400af06f23b3a049e44b6/notebook-7.3.3-py3-none-any.whl (13.1 MB)
13.1/13.1 MB 7.9 MB/s eta 0:00:00
Collecting jupyter-console (from jupyter->r D:\workSpace\lyh\lyh_doc\requirements.txt (line 2))
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/ca/77/71d78d58f15c22db16328a476426f7ac4a60d3a5a7ba3b9627ee2f7903d4/jupyter_console-6.6.3-py3-none-any.whl (24 kB)
```

方法二：永久修改 pip 源

创建或修改 pip 配置文件，设置默认镜像源。

• Windows 系统：

1. 打开文件资源管理器/此电脑，在地址栏中输入 %APPDATA% 按回车进行跳转到 Roaming 文件夹中
2. 新建文件夹 pip（若不存在），并在其中新建文件 pip.ini。
 - 如果有pip文件夹并且有pip.ini文件则在对应的 [] 中添加以下内容即可。
3. 编辑 pip.ini，添加以下内容：

```
[global]
timeout = 6000
index-url=http://mirrors.aliyun.com/pypi/simple/
[install]
trusted-host=mirrors.aliyun.com
```

复制粘贴下面的命令，回车

```
pip install -r D:\workSpace\pyProj\requirements.txt # 这里替换成你
requirements.txt的实际路径
```

可以看到我这里默认源已经改为阿里源了

```
Anaconda Prompt - conda de X + v
(env_name) C:\Users\tanga>pip install -r "D:\workSpace\lyh\lyh_doc\requirements.txt"
Looking in indexes: http://mirrors.aliyun.com/pypi/simple/
Collecting openai (from -r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
  Downloading http://mirrors.aliyun.com/pypi/packages/78/5a/e20182f7b6171642d759c548daa0ba20a1d3ac10d2bd0a13fd75704a9ac3/openai-1.66.3-py3-none-any.whl (567 kB)
    567.4/567.4 kB 6.5 MB/s eta 0:00:00
Collecting jupyter (from -r D:\workSpace\lyh\lyh_doc\requirements.txt (line 2))
  Downloading http://mirrors.aliyun.com/pypi/packages/38/64/285f20a31679bf547b75602702f7800e74dbabae36ef324f716c02804753/jupyter-1.1.1-py2.py3-none-any.whl (2.7 kB)
Collecting requests (from -r D:\workSpace\lyh\lyh_doc\requirements.txt (line 3))
  Downloading http://mirrors.aliyun.com/pypi/packages/f9/9b/335f9764261e915ed497fcdeb11df5dfd6f7bf257d4a6a2a686d80da4d54/requests-2.32.3-py3-none-any.whl (64 kB)
Collecting pymupdf (from -r D:\workSpace\lyh\lyh_doc\requirements.txt (line 4))
  Downloading http://mirrors.aliyun.com/pypi/packages/c3/9f/2872a29766dfe65fb97268233503cc9f7ba6e776eedbdb88ed47216aef84/pymupdf-1.25.4-cp39-abi3-win_amd64.whl (16.6 MB)
    16.6/16.6 MB 12.0 MB/s eta 0:00:00
Collecting anyio<5.0.0, >=3.5.0 (from openai->-r D:\workSpace\lyh\lyh_doc\requirements.txt (line 1))
```

查看已安装的环境信息

正常情况下环境的安装在这一步就已经**完成了**, 我们在 conda prompt 中输入, 回车

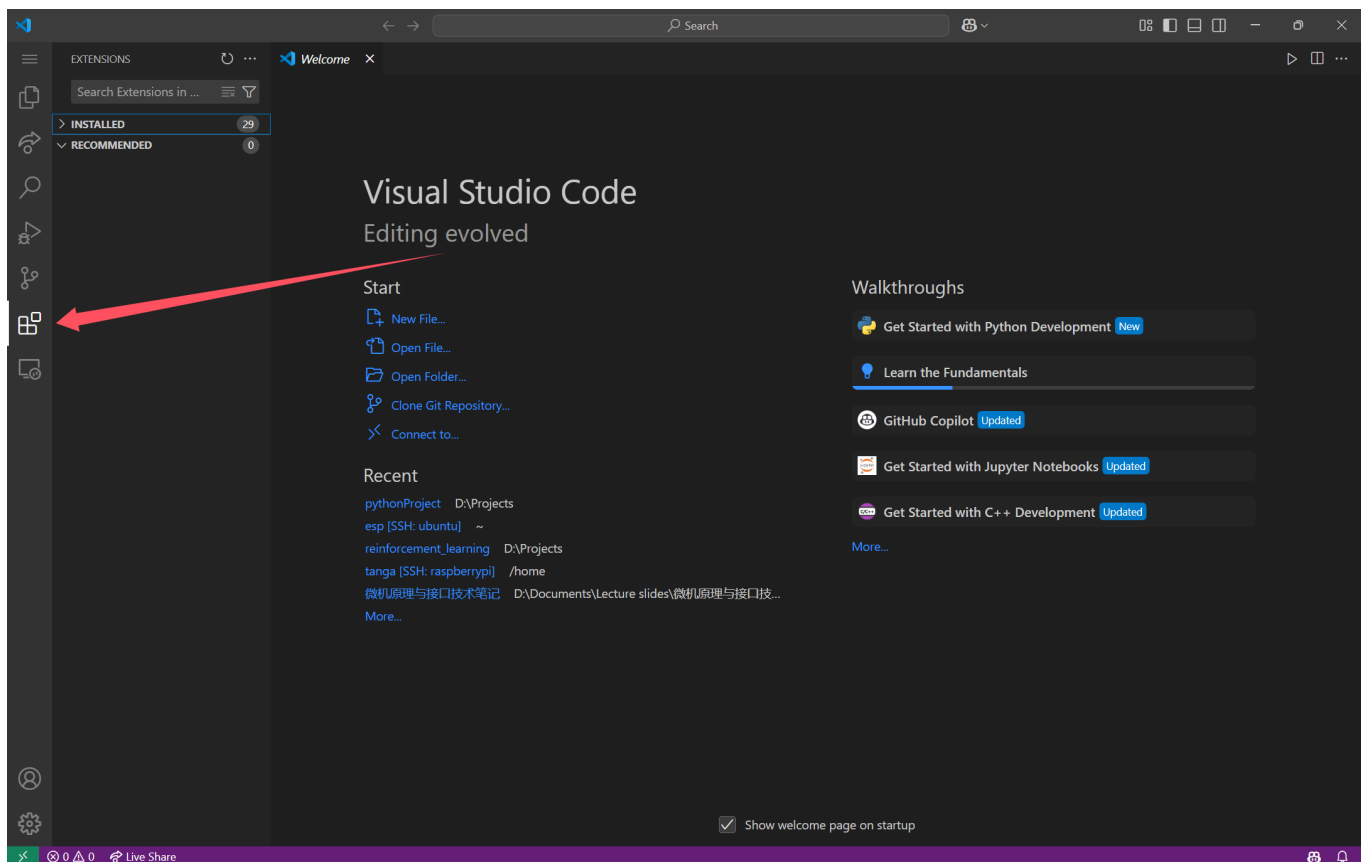
```
conda list
```

此处list中能看看到在**requirements.txt**中出现的包名比如说 openai, jupyter, pymupdf 就代表安装完成.

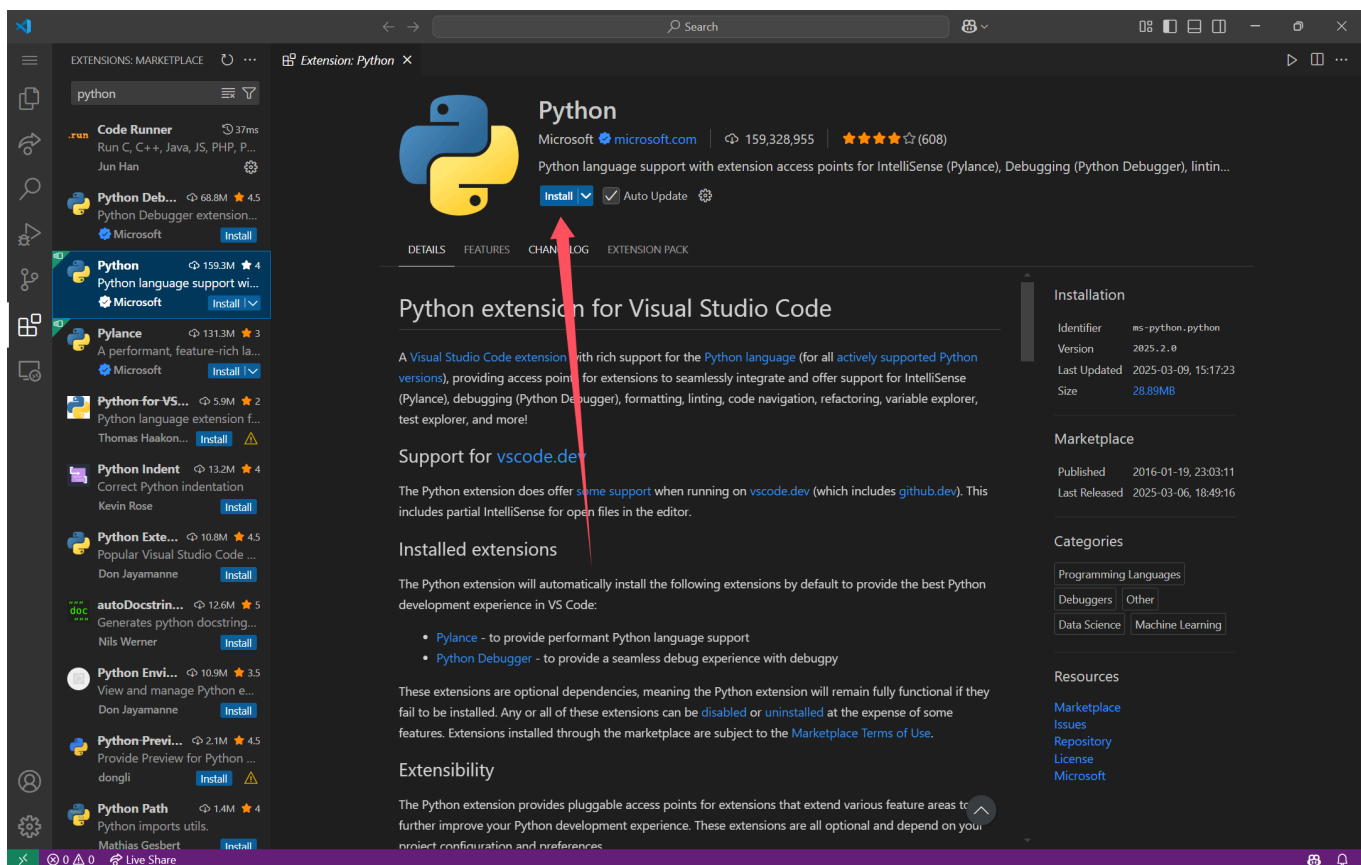
jupyter	1.1.1	pypi_0	pypi
jupyter-client	8.6.3	pypi_0	pypi
jupyter-console	6.6.3	pypi_0	pypi
jupyter-core	5.7.2	pypi_0	pypi
jupyter-events	0.12.0	pypi_0	pypi
jupyter-lsp	2.2.5	pypi_0	pypi
jupyter-server	2.15.0	pypi_0	pypi
jupyter-server-terminals	0.5.3	pypi_0	pypi
jupyterlab	4.3.6	pypi_0	pypi
jupyterlab-pygments	0.3.0	pypi_0	pypi
jupyterlab-server	2.27.3	pypi_0	pypi
jupyterlab-widgets	3.0.13	pypi_0	pypi
libffi	3.4.4	hd77b12b_1	https://mirrors.tuna
markupsafe	3.0.2	pypi_0	pypi
matplotlib-inline	0.1.7	pypi_0	pypi
mistune	3.1.2	pypi_0	pypi
nbclient	0.10.2	pypi_0	pypi
nbconvert	7.16.6	pypi_0	pypi
nbformat	5.10.4	pypi_0	pypi
nest-asyncio	1.6.0	pypi_0	pypi
notebook	7.3.3	pypi_0	pypi
notebook-shim	0.2.4	pypi_0	pypi
openai	1.66.3	pypi_0	pypi
openssl	3.0.16	h3f729d1_0	https://mirrors.tuna
overrides	7.7.0	pypi_0	pypi
packaging	24.2	pypi_0	pypi
pandocfilters	1.5.1	pypi_0	pypi
parso	0.8.4	pypi_0	pypi
pip	25.0	py312haa95532_0	https://mirrors.tuna
platformdirs	4.3.6	pypi_0	pypi
prometheus-client	0.21.1	pypi_0	pypi
prompt-toolkit	3.0.50	pypi_0	pypi
psutil	7.0.0	pypi_0	pypi
pure-eval	0.2.3	pypi_0	pypi
pyparsing	2.2.2	pypi_0	pypi
pydantic	2.10.6	pypi_0	pypi
pydantic-core	2.27.2	pypi_0	pypi
pygments	2.19.1	pypi_0	pypi
pymupdf	1.25.4	pypi_0	pypi
python	3.12.9	h11ff660_0	https://mirrors.tuna

配置VSCode

打开VSCode -> 左栏extension



搜索 **python**，安装 (install)

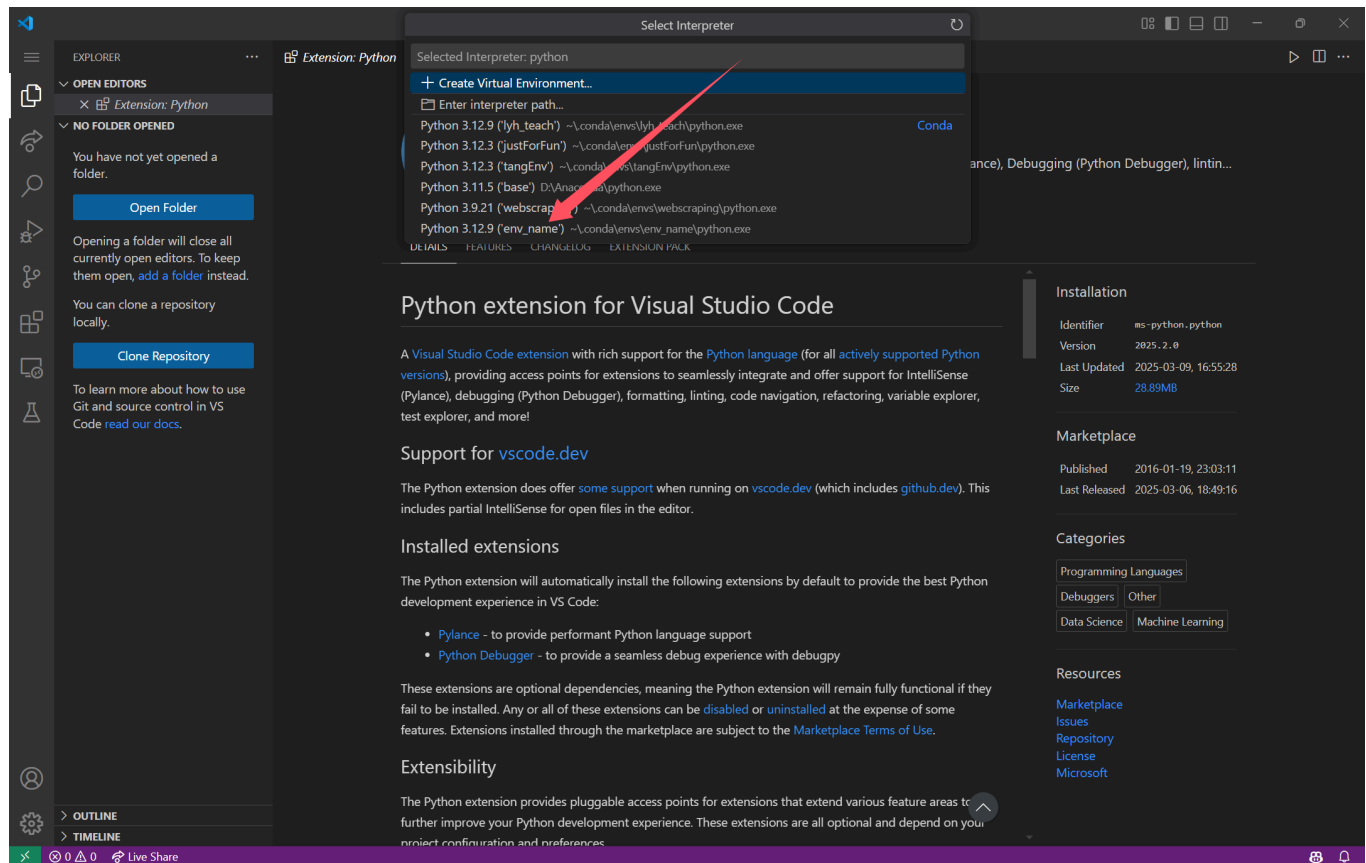


再次打开 **Anaconda prompt**

输入 `conda env list` , 找到之前安装的环境和它对应的路径地址, 看看自己新装的环境的地址在哪

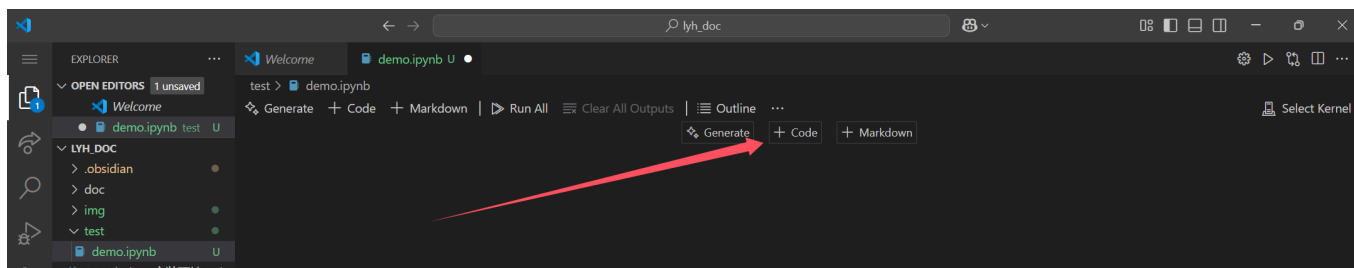
```
(base) C:\Users\tanga>conda env list
# conda environments:
#
env_name                C:\Users\tanga\.conda\envs\env_name
justForFun              C:\Users\tanga\.conda\envs\justForFun
lyh_teach               C:\Users\tanga\.conda\envs\lyh_teach
tangEnv                 C:\Users\tanga\.conda\envs\tangEnv
webscraping             C:\Users\tanga\.conda\envs\webscraping
base                    * D:\Anaconda
```

在 VSCode 中按 `Ctrl+Shift+P` , 输入 `Python: Select Interpreter` 并选择自己刚刚安装的 Conda环境:



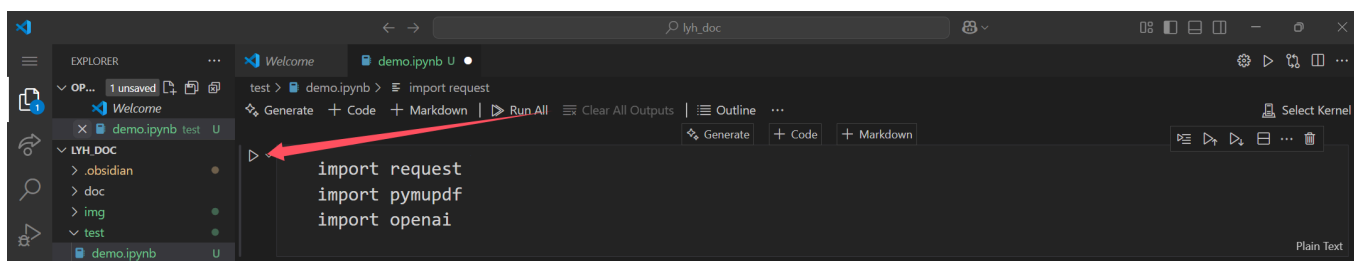
验证环境安装成功

打开我们的项目文件夹, 新建一个jupyter文件(文件后缀名为 `.ipynb`), 这里以 `demo.ipynb` 为例.

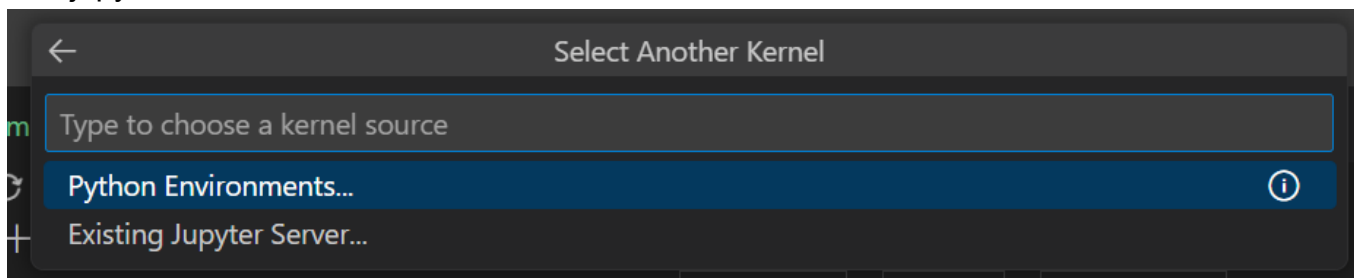


创建成功后添加一个Code块, 复制下面的代码进去, 然后 `ctrl+enter` / 点击左边RUN按钮 运行代码块

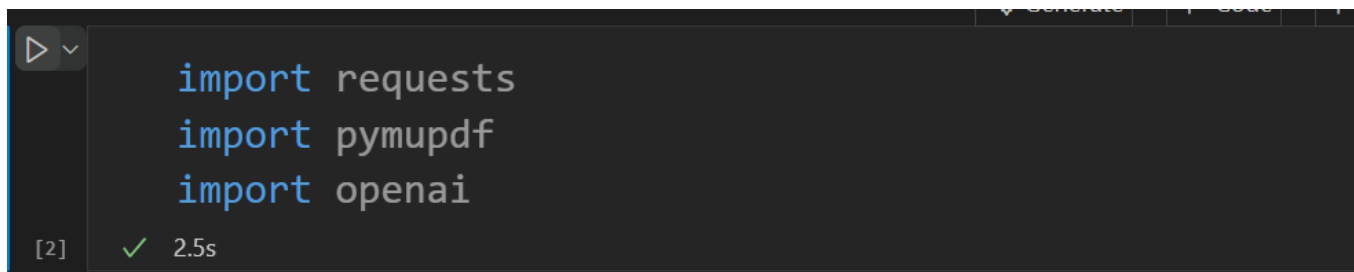
```
import request
import pymupdf
import openai
```



选择jupyter 核源



然后再选择我们刚刚创建的python环境. `ctrl+enter` / 点击左边RUN按钮 运行代码块, 成功执行



OK, 如果成功执行没有报错, 环境配置完成!