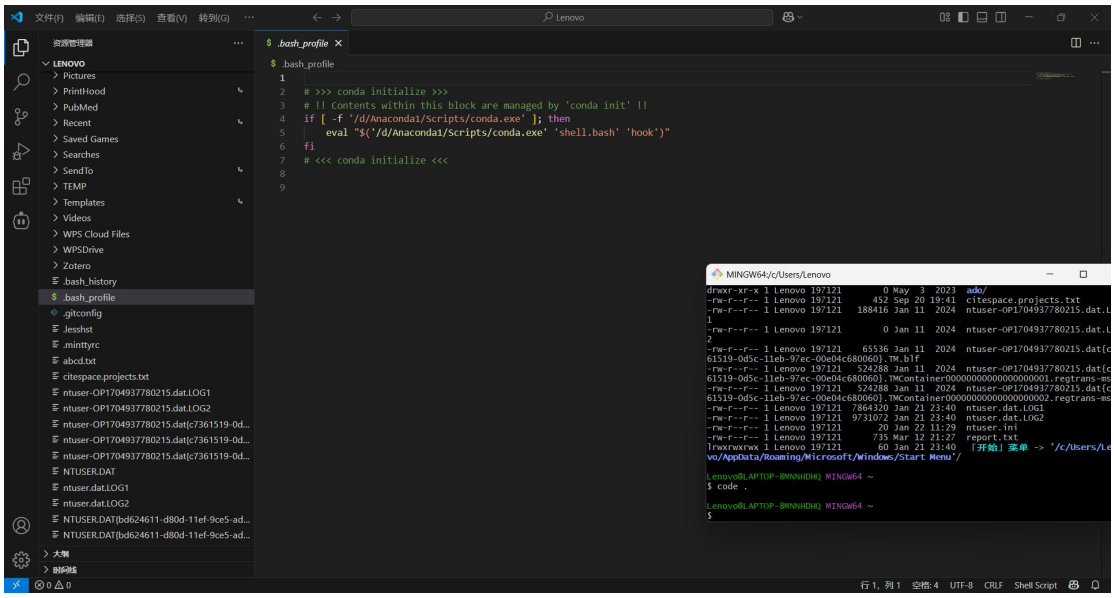
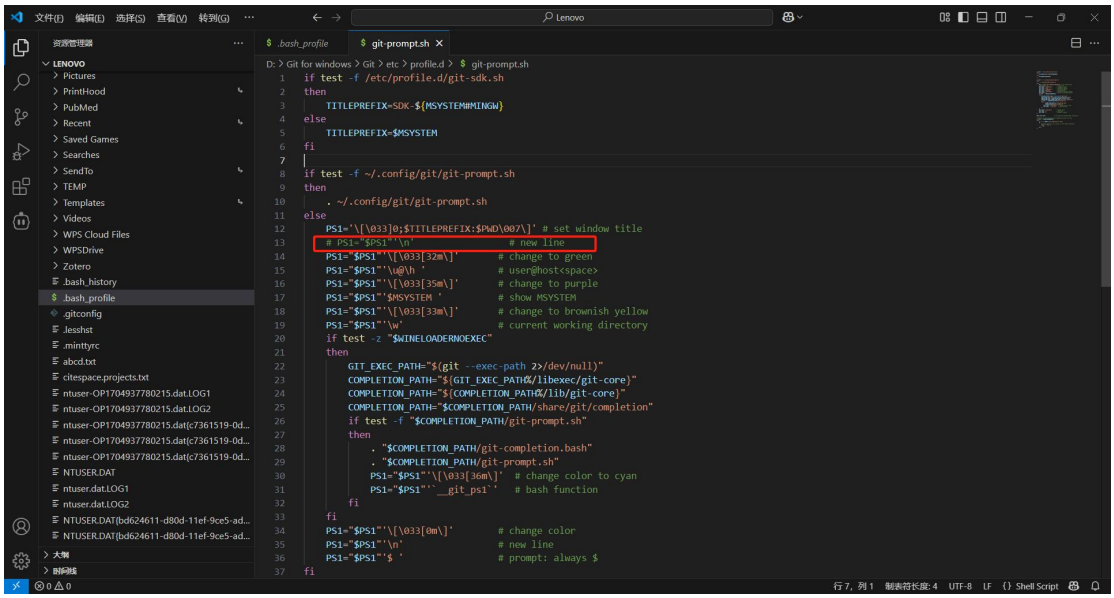
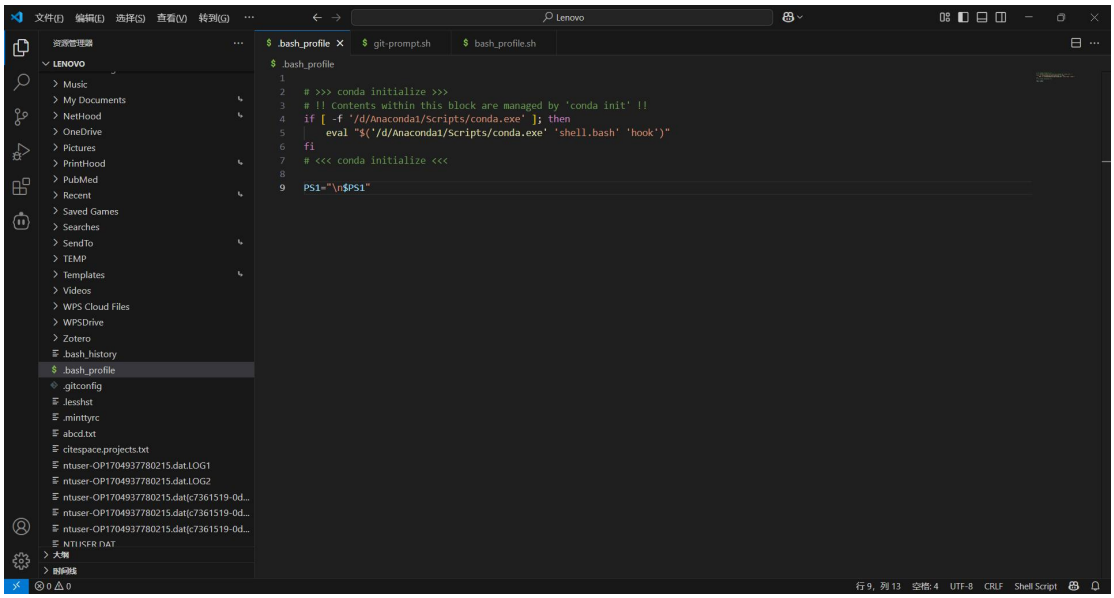
第三周 学习笔记——Python 项目 (初级)

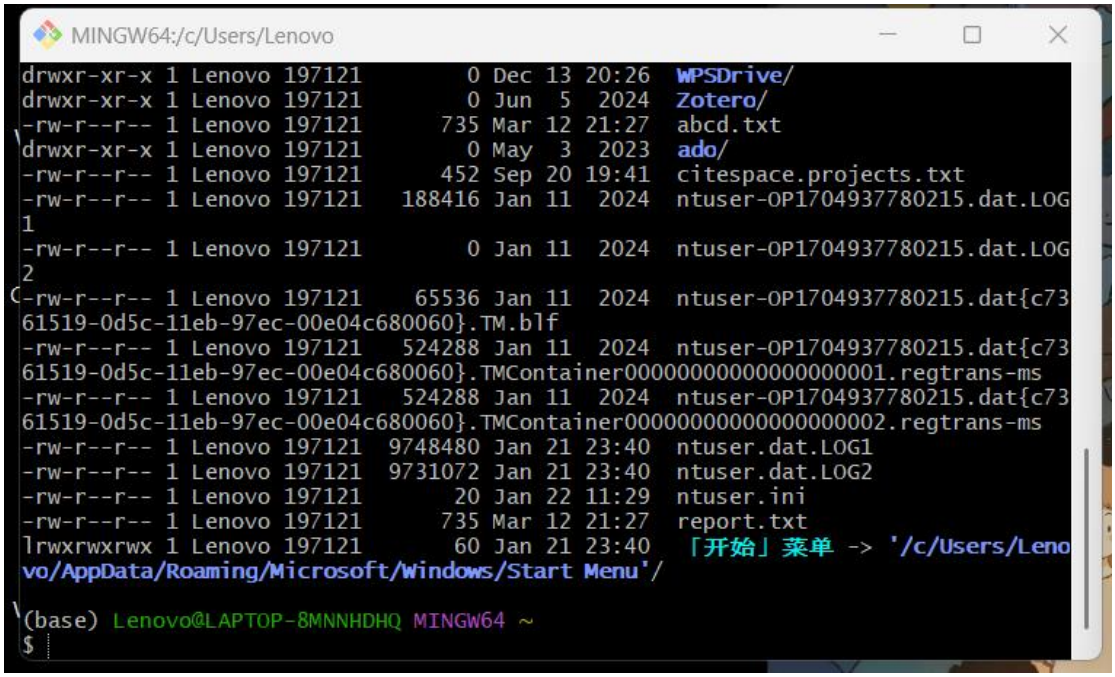
一、在终端配置 Conda 初始化

在自己的终端 Git Bash 中配置好 Conda Init，使得启动终端后，在提示符 $前能够看到 (base)。









二、创建 Conda 环境

2.1 使用 conda info 命令查看本机 Conda 的配置信息

```
(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ ^C

(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ conda info

     active environment : base
    active env location : D:\Anaconda1
            shell level : 1
       user config file : C:\Users\Lenovo\.condarc
 populated config files : D:\Anaconda1\.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=icelake
                          __conda=24.9.2=0
                          __cuda=11.7=0
                          __win=0=0
      base environment : D:\Anaconda1  (read only)
     conda av data dir : D:\Anaconda1\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:\Anaconda1\pkgs
                         C:\Users\Lenovo\.conda\pkgs
                         C:\Users\Lenovo\AppData\Local\conda\conda\pkgs
      envs directories : C:\Users\Lenovo\.conda\envs
                         D:\Anaconda1\envs
                         C:\Users\Lenovo\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 sol
ver/libmamba conda-libmamba-solver/24.9.0 libmambapy/1.5.8 aau/0.4.4 c/. s/.
         administrator : False
            netrc file : None
          offline mode : False
```

2.2 使用 conda env list 命令查看已有的 Conda 环境的名称和路径，理解 Conda 环境的概念

```
(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ conda env list
# conda environments:
#
base                  *  D:\Anaconda1

(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$
```

Conda 环境是一个由 Conda 创建和管理的独立的软件环境，它允许用户在同一台计算机上创建多个相互隔离的环境，每个环境可以有不同的 Python 版本、软件包及其依赖项，从而避免不同项目之间的依赖冲突。

2.3 使用 conda create 命令创建两个 Conda 环境，一个里面安装 Python 3.12 和 requests 软件包，另一个里面安装 Python 3.9、pandas 和 statsmodels 软件包，能够在终端里切换 Conda 环境，验证 Python 和软件包的版本

2.3.1 安装 Python 3.12 和 requests 软件包

```
The following NEW packages will be INSTALLED:

  brotli-python      pkgs/main/win-64::brotli-python-1.0.9-py312h5da7b33_9
  bzip2              pkgs/main/win-64::bzip2-1.0.8-h2bbff1b_6
  ca-certificates    pkgs/main/win-64::ca-certificates-2025.2.25-haa95532_0
  certifi            pkgs/main/win-64::certifi-2025.1.31-py312haa95532_0
  charset-normalizer pkgs/main/noarch::charset-normalizer-3.3.2-pyhd3eb1b0_0
  expat              pkgs/main/win-64::expat-2.6.4-h8ddb27b_0
  idna               pkgs/main/win-64::idna-3.7-py312haa95532_0
  libffi             pkgs/main/win-64::libffi-3.4.4-hd77b12b_1
  openssl            pkgs/main/win-64::openssl-3.0.16-h3f729d1_0
  pip                pkgs/main/win-64::pip-25.0-py312haa95532_0
done
#
# To activate this environment, use
#
#     $ conda activate prj1
#
# To deactivate an active environment, use
#
#     $ conda deactivate
```

2.3.2 安装 Python 3.9、pandas 和 statsmodels 软件包

在安装 python3.9 时，程序报错 CondaError: Failed to write to C:\Users\Lenovo\.conda\pkgs\mkl-2023.1.0-h6b88edl errno: 28，可能是因为 C 盘内存不够，在清理一些内存之后，再尝试就成功了！

```
The following NEW packages will be INSTALLED:

  blas               pkgs/main/win-64::blas-1.0-mkl
  bottleneck         pkgs/main/win-64::bottleneck-1.4.2-py39hc99e966_0
  ca-certificates    pkgs/main/win-64::ca-certificates-2025.2.25-haa95532_0
done
#
# To activate this environment, use
#
#     $ conda activate prj2
#
# To deactivate an active environment, use
#
#     $ conda deactivate


(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$
```

2.3.2 切换 Conda 环境，验证 Python 和软件包的版本

```
(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ conda activate prj1
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ which python
/c/Users/Lenovo/.conda/envs/prj1/python
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ python --version
Python 3.12.9
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ python
Python 3.12.9 | packaged by Anaconda, Inc. | (main, Feb  6 2025, 18:49:16) [MSC v.1929 64 bit (AMD64)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> ls -l
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'ls' is not defined
>>> quit()
(prj1)
```

```
$ conda activate prj2
(prj2)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ which python
/c/Users/Lenovo/.conda/envs/prj2/python
(prj2)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ python --version
Python 3.9.21
(prj2)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ python
Python 3.9.21 (main, Dec 11 2024, 16:35:24) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import requests
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'requests'
>>> import pandas
>>> pandas.__file__
'C:\\Users\\Lenovo\\.conda\\envs\\prj2\\lib\\site-packages\\pandas\\__init__.py'
>>> pandas.__version__
'2.2.3'
>>> import statsmodels
>>> statsmodels.__version__
'0.14.4'
>>>
```

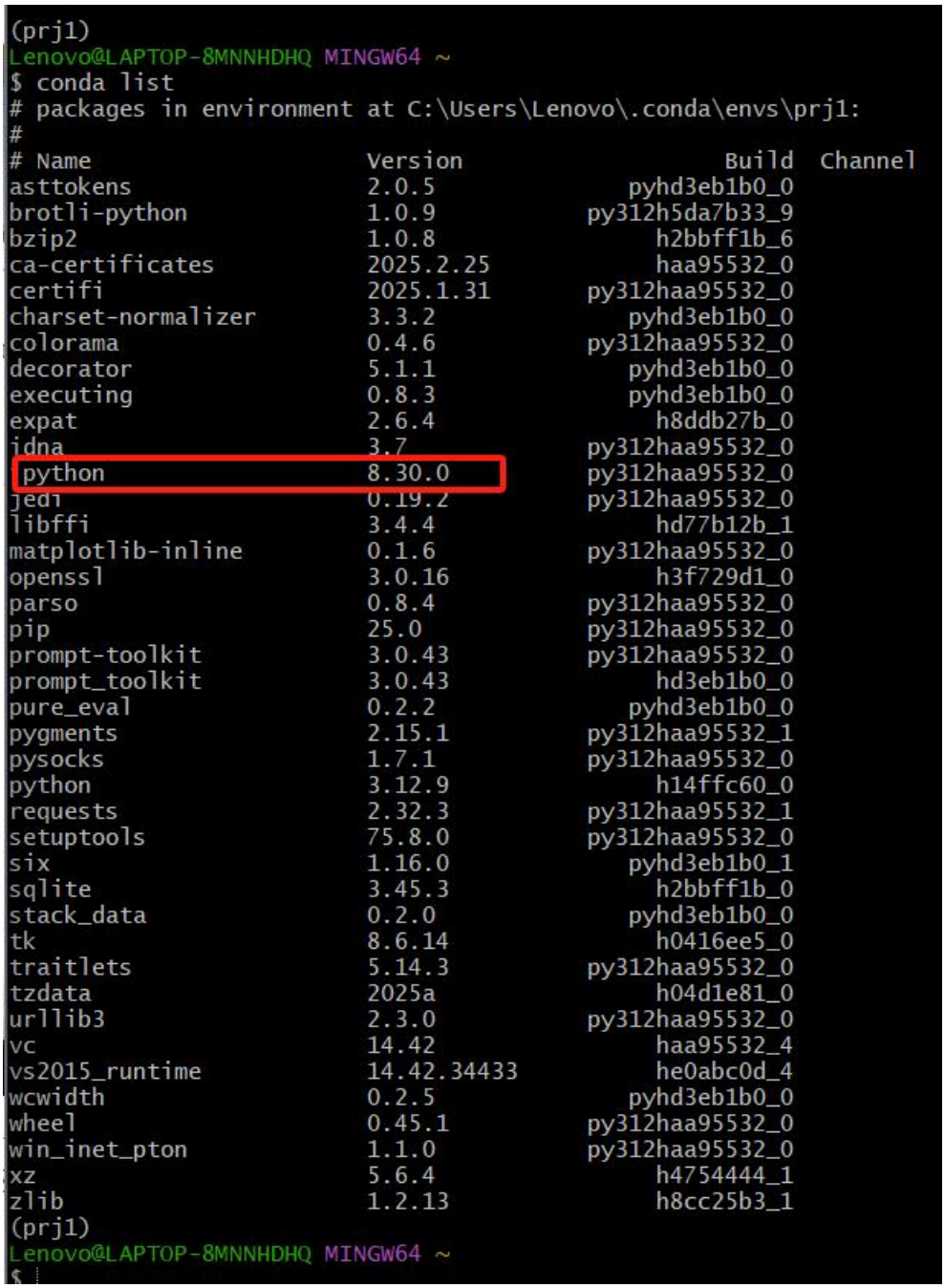
```
$ conda list
# packages in environment at C:\Users\Lenovo\.conda\envs\prj2:
#
# Name                    Version                   Build  Channel
blas                      1.0                         mkl
bottleneck                1.4.2           py39hc99e966_0
ca-certificates           2025.2.25            haa95532_0
icc_rt                    2022.1.0             h6049295_2
intel-openmp              2023.1.0        h59b6b97_46320
mkl                       2023.1.0        h6b88ed4_46358
mkl-service               2.4.0           py39h827c3e9_2
mkl_fft                   1.3.11          py39h827c3e9_0
mkl_random                1.2.8           py39hc64d2fc_0
numexpr                   2.10.1          py39h4cd664f_0
numpy                     2.0.2           py39h055cbcc_0
numpy-base                2.0.2           py39h65a83cf_0
openssl                   3.0.16               h3f729d1_0
packaging                 24.2            py39haa95532_0
pandas                    2.2.3           py39h5da7b33_0
patsy                     1.0.1           py39haa95532_0
pip                       25.0            py39haa95532_0
pybind11-abi              5                    hd3eb1b0_0
python                    3.9.21               h8205438_1
python-dateutil           2.9.0post0      py39haa95532_2
python-tzdata             2023.3             pyhd3eb1b0_0
pytz                      2024.1          py39haa95532_0
scipy                     1.13.1          py39h8640f81_1
setuptools                72.1.0          py39haa95532_0
six                       1.16.0             pyhd3eb1b0_1
sqlite                    3.45.3               h2bbff1b_0
statsmodels               0.14.4          py39h827c3e9_0
tbb                       2021.8.0             h59b6b97_0
tzdata                    2025a                h04d1e81_0
vc                        14.42                haa95532_4
vs2015_runtime            14.42.34433          he0abc0d_4
wheel                     0.45.1          py39haa95532_0
(prj2)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
```

2.4 使用 conda list 命令显示 Conda 环境里的软件包列表及其版本信息

```
$ conda activate prj1
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ conda list
# packages in environment at C:\Users\Lenovo\.conda\envs\prj1:
#
# Name                    Version                   Build  Channel
brotli-python             1.0.9           py312h5da7b33_9
bzip2                     1.0.8                h2bbff1b_6
ca-certificates           2025.2.25            haa95532_0
certifi                   2025.1.31       py312haa95532_0
charset-normalizer        3.3.2              pyhd3eb1b0_0
expat                     2.6.4                h8ddb27b_0
idna                      3.7             py312haa95532_0
libffi                    3.4.4                hd77b12b_1
openssl                   3.0.16               h3f729d1_0
pip                       25.0            py312haa95532_0
pysocks                   1.7.1           py312haa95532_0
python                    3.12.9               h14ffc60_0
requests                  2.32.3          py312haa95532_1
setuptools                75.8.0          py312haa95532_0
sqlite                    3.45.3               h2bbff1b_0
tk                        8.6.14               h0416ee5_0
tzdata                    2025a                h04d1e81_0
urllib3                   2.3.0           py312haa95532_0
vc                        14.42                haa95532_4
vs2015_runtime            14.42.34433          he0abc0d_4
wheel                     0.45.1          py312haa95532_0
win_inet_pton             1.1.0           py312haa95532_0
xz                        5.6.4                h4754444_1
zlib                      1.2.13               h8cc25b3_1
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ ^C
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
```

```
$ conda list
# packages in environment at C:\Users\Lenovo\.conda\envs\prj2:
#
# Name                    Version                   Build  Channel
blas                      1.0                         mkl
bottleneck                1.4.2            py39hc99e966_0
ca-certificates           2025.2.25            haa95532_0
icc_rt                    2022.1.0             h6049295_2
intel-openmp              2023.1.0         h59b6b97_46320
mkl                       2023.1.0         h6b88ed4_46358
mkl-service               2.4.0            py39h827c3e9_2
mkl_fft                   1.3.11           py39h827c3e9_0
mkl_random                1.2.8            py39hc64d2fc_0
numexpr                   2.10.1           py39h4cd664f_0
numpy                     2.0.2            py39h055cbcc_0
numpy-base                2.0.2            py39h65a83cf_0
openssl                   3.0.16               h3f729d1_0
packaging                 24.2             py39haa95532_0
pandas                    2.2.3            py39h5da7b33_0
patsy                     1.0.1            py39haa95532_0
pip                       25.0             py39haa95532_0
pybind11-abi              5                    hd3eb1b0_0
python                    3.9.21               h8205438_1
python-dateutil           2.9.0post0       py39haa95532_2
python-tzdata             2023.3             pyhd3eb1b0_0
pytz                      2024.1           py39haa95532_0
scipy                     1.13.1           py39h8640f81_1
setuptools                72.1.0           py39haa95532_0
six                       1.16.0             pyhd3eb1b0_1
sqlite                    3.45.3               h2bbff1b_0
statsmodels               0.14.4           py39h827c3e9_0
tbb                       2021.8.0             h59b6b97_0
tzdata                    2025a                h04d1e81_0
vc                        14.42                haa95532_4
vs2015_runtime            14.42.34433          he0abc0d_4
wheel                     0.45.1           py39haa95532_0
(prj2)
```

2.5 使用 conda install 命令往 Conda 环境里安装更多的软件包，并验证版本



```
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ conda list
# packages in environment at C:\Users\Lenovo\.conda\envs\prj1:
#
# Name                    Version                   Build  Channel
asttokens                 2.0.5              pyhd3eb1b0_0
brotli-python             1.0.9           py312h5da7b33_9
bzip2                     1.0.8                h2bbff1b_6
ca-certificates           2025.2.25            haa95532_0
certifi                   2025.1.31       py312haa95532_0
charset-normalizer        3.3.2              pyhd3eb1b0_0
colorama                  0.4.6           py312haa95532_0
decorator                 5.1.1              pyhd3eb1b0_0
executing                 0.8.3              pyhd3eb1b0_0
expat                     2.6.4                h8ddb27b_0
idna                      3.7             py312haa95532_0
python                    8.30.0          py312haa95532_0
jedi                      0.19.2          py312haa95532_0
libffi                    3.4.4                hd77b12b_1
matplotlib-inline         0.1.6           py312haa95532_0
openssl                   3.0.16               h3f729d1_0
parso                     0.8.4           py312haa95532_0
pip                       25.0            py312haa95532_0
prompt-toolkit            3.0.43          py312haa95532_0
prompt_toolkit            3.0.43               hd3eb1b0_0
pure_eval                 0.2.2              pyhd3eb1b0_0
pygments                  2.15.1          py312haa95532_1
pysocks                   1.7.1           py312haa95532_0
python                    3.12.9               h14ffc60_0
requests                  2.32.3          py312haa95532_1
setuptools                75.8.0          py312haa95532_0
six                       1.16.0             pyhd3eb1b0_1
sqlite                    3.45.3               h2bbff1b_0
stack_data                0.2.0              pyhd3eb1b0_0
tk                        8.6.14               h0416ee5_0
traitlets                 5.14.3          py312haa95532_0
tzdata                    2025a                h04d1e81_0
urllib3                   2.3.0           py312haa95532_0
vc                        14.42                haa95532_4
vs2015_runtime            14.42.34433          he0abc0d_4
wcwidth                   0.2.5              pyhd3eb1b0_0
wheel                     0.45.1          py312haa95532_0
win_inet_pton             1.1.0           py312haa95532_0
xz                        5.6.4                h4754444_1
zlib                      1.2.13               h8cc25b3_1
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$
```

三、Conda-Forge 清华镜像

根据文档，配置 Anaconda 清华镜像，加快 conda install 安装软件包的速度，将 conda-forge 设置为默认 Channel，让 conda install 能够安装更多的软件包。

.condarc

C: > Users > Lenovo > .condarc

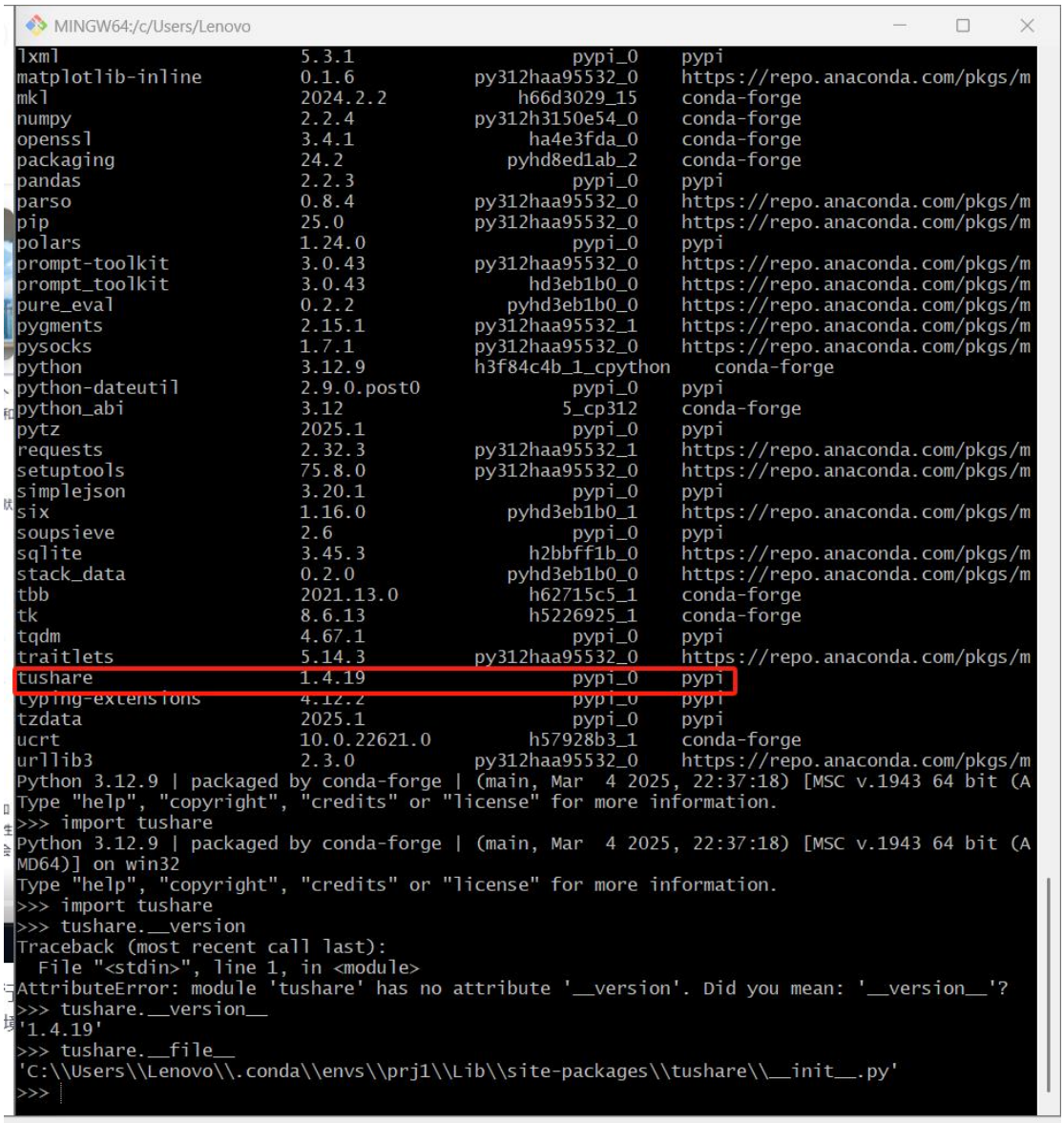
```
channels:
  - conda-forge
  - defaults
show_channel_urls: true
default_channels:
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/r
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/msys2
custom_channels:
  conda-forge: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
  pytorch: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
  channel_priority: strict

```

MINGW64:/c/Users/Lenovo

```
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~
$ conda list
# packages in environment at C:\Users\Lenovo\.conda\envs\prj1:
#
# Name                    Version                   Build  Channel
asttokens                 2.0.5              pyhd3eb1b0_0    https://repo.anaconda.com/pkgs/main
brotli-python             1.0.9           py312h5da7b33_9    https://repo.anaconda.com/pkgs/main
bzip2                     1.0.8                h2bbff1b_6    https://repo.anaconda.com/pkgs/main
ca-certificates           2025.2.25            haa95532_0    https://repo.anaconda.com/pkgs/main
certifi                   2025.1.31          pyhd8ed1ab_0    conda-forge
charset-normalizer        3.3.2              pyhd3eb1b0_0    https://repo.anaconda.com/pkgs/main
colorama                  0.4.6           py312haa95532_0    https://repo.anaconda.com/pkgs/main
decorator                 5.1.1              pyhd3eb1b0_0    https://repo.anaconda.com/pkgs/main
executing                 0.8.3              pyhd3eb1b0_0    https://repo.anaconda.com/pkgs/main
expat                     2.6.4                h8ddb27b_0    https://repo.anaconda.com/pkgs/main
idna                      3.7             py312haa95532_0    https://repo.anaconda.com/pkgs/main
intel-openmp              2024.2.1          h57928b3_1083    conda-forge
ipython                   8.30.0          py312haa95532_0    https://repo.anaconda.com/pkgs/main
jedi                      0.19.2          py312haa95532_0    https://repo.anaconda.com/pkgs/main
libblas                   3.9.0           31_h641d27c_mkl    conda-forge
libcblas                  3.9.0           31_h5e41251_mkl    conda-forge
libexpat                  2.6.4                he0c23c2_0    conda-forge
libffi                    3.4.4                hd77b12b_1    https://repo.anaconda.com/pkgs/main
libhwloc                  2.11.2          default_ha69328c_1001    conda-forge
libiconv                  1.18                 h135ad9c_1    conda-forge
liblapack                 3.9.0           31_h1aa476e_mkl    conda-forge
liblzma                   5.6.4                h2466b09_0    conda-forge
libsqlite                 3.49.1               h67fdade_2    conda-forge
libwinpthread             12.0.0.r4.gg4f2fc60ca      h57928b3_9    conda-forge
libxml2                   2.13.6               he286e8c_0    conda-forge
libzlib                   1.3.1                h2466b09_2    conda-forge
matplotlib-inline         0.1.6           py312haa95532_0    https://repo.anaconda.com/pkgs/main
mkl                       2024.2.2            h66d3029_15    conda-forge
numpy                     2.2.4           py312h3150e54_0    conda-forge
openssl                   3.4.1                ha4e3fda_0    conda-forge
packaging                 24.2               pyhd8ed1ab_2    conda-forge
parso                     0.8.4           py312haa95532_0    https://repo.anaconda.com/pkgs/main
pip                       25.0            py312haa95532_0    https://repo.anaconda.com/pkgs/main
polars                    1.24.0                   pypi_0    pypi
prompt-toolkit            3.0.43          py312haa95532_0    https://repo.anaconda.com/pkgs/main
prompt_toolkit            3.0.43               hd3eb1b0_0    https://repo.anaconda.com/pkgs/main
pure_eval                 0.2.2              pyhd3eb1b0_0    https://repo.anaconda.com/pkgs/main
pygments                  2.15.1          py312haa95532_1    https://repo.anaconda.com/pkgs/main
pysocks                   1.7.1           py312haa95532_0    https://repo.anaconda.com/pkgs/main
python                    3.12.9          h3f84c4b_1_cpython    conda-forge
python_abi                3.12                    5_cp312    conda-forge
requests                  2.32.3          py312haa95532_1    https://repo.anaconda.com/pkgs/main
setuptools                75.8.0          py312haa95532_0    https://repo.anaconda.com/pkgs/main
six                       1.16.0             pyhd3eb1b0_1    https://repo.anaconda.com/pkgs/main
sqlite                    3.45.3               h2bbff1b_0    https://repo.anaconda.com/pkgs/main
stack_data                0.2.0              pyhd3eb1b0_0    https://repo.anaconda.com/pkgs/main
tbb                       2021.13.0            h62715c5_1    conda-forge
tk                        8.6.13               h5226925_1    conda-forge
traitlets                 5.14.3          py312haa95532_0    https://repo.anaconda.com/pkgs/main
tzdata                    2025a                h04d1e81_0    https://repo.anaconda.com/pkgs/main
ucrt                      10.0.22621.0         h57928b3_1    conda-forge
```

四、环境配置的导出与重建

4.1 使用 pip install 命令往 Conda 环境里安装 Python 软件包，并验证版本



4.2 能够导出 environment.yml Conda 环境配置文件，能够删除 Conda 环境，能够用 environment.yml 配置文件重建 Conda 环境

```
$ cd twinkledehaha
(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha (main)
$ ls -l
total 25
-rw-r--r-- 1 Lenovo 197121    16 Mar 12 21:58 README.md
-rw-r--r-- 1 Lenovo 197121 22809 Mar 12 22:19 abcd.pdf
(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha (main)
$ nkdir prj1
bash: nkdir: command not found
(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha (main)
$ mkdir prj1
(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha (main)
$ cd prj1
(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/prj1 (main)
$ ls -l
total 0
(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/prj1 (main)
$ mv ~/environment.yml ./
(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/prj1 (main)
$ conda env list
# conda environments:
#
prj2                     C:\Users\Lenovo\.conda\envs\prj2
base                  *  D:\Anaconda1

(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/prj1 (main)
$ pwd
/c/Users/Lenovo/Desktop/twinkledehaha/prj1
(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/prj1 (main)
$ ls -l
total 4
-rw-r--r-- 1 Lenovo 197121 2535 Mar 19 11:52 environment.yml
(base)
```

```
#
# To activate this environment, use
#
#     $ conda activate prj1
#
# To deactivate an active environment, use
#
#     $ conda deactivate

(base)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/prj1 (main)
$ conda activate prj1
(prj1)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/prj1 (main)
$ conda list
# packages in environment at C:\Users\Lenovo\.conda\envs\prj1:
#
```

4.3 理解 Conda 与 Python 的关系，理解 Conda-Forge 与 Conda 的关系，理

解 Python 解释器、第三方软件包、PyPI 软件仓库、以及程序/软件包的路径问题

4.3.1 conda 与 python 的关系

Conda 是一个开源的包管理系统和环境管理系统，主要用于帮助用户更方便地管理 Python 项目中的依赖项和环境。

管理 Python 包：Conda 可以用于安装、更新和卸载 Python 包。

创建隔离的 Python 环境：Conda 允许用户创建多个相互隔离的 Python 环境。

跨平台兼容性：Conda 在不同操作系统（如 Windows、macOS 和 Linux）上都能很好地运行，并且能够处理不同平台上的包安装和环境配置问题。

4.3.2 Conda-Forge 与 Conda 的关系

Conda-Forge 是一个社区驱动的开源项目。

提供额外的软件包渠道：Conda 本身是一个包管理系统和环境管理系统，而 Conda-Forge 为 Conda 提供了一个丰富的软件包渠道。

社区协作维护：Conda-Forge 是由社区成员共同维护和管理的。

补充和扩展 Conda 功能：Conda-Forge 不仅提供软件包，还通过一些特殊的功能和工具来扩展 Conda 的能力。

4.3.3 Python 解释器

Python 解释器是 Python 语言的核心组件，其作用是将你编写的 Python 代码转化为计算机能够理解和执行的机器语言指令。

当你编写好 Python 代码后，Python 解释器会逐行读取代码，对代码进行词法分析、语法分析等一系列处理，然后将其转换为机器能够执行的指令，最终完成代码的执行。这种逐行解释执行的方式，使得 Python 代码具有很好的动态性和交互性。Python 解释器有 CPython、IPython、JPython、IronPython，在选择 Python 解释器时，需要根据具体的应用场景和需求来决定。

4.3.4 第三方软件包

第三方软件包是指由 Python 官方标准库以外的其他开发者、组织或社区创建和维护的软件包。这些软件包旨在扩展 Python 的功能，帮助开发者更高效地完成特定任务。通常使用 pip（Python 包管理工具）或 conda（包和环境管理系统）来安装和管理第三方软件包。

4.3.5 PyPI 软件仓库

PyPI 如同一个巨大的“软件宝库”，存储着大量由 Python 开发者和组织贡献的第三方软件包。当你使用 pip 命令安装 Python 软件包时，pip 默认会连接到 PyPI，在其中搜索你指定的软件包。若找到匹配的软件包，pip 会下载该软件包及其依赖项，并将它们安装到你的 Python 环境中。例如，在命令行输入 pip install requests，pip 就会从 PyPI 找到 requests 库并进行安装。

4.3.6 程序/软件包的路径问题

模块导入时的路径问题。当你使用 import 语句导入模块时，Python 会按照特定的路径顺序去查找该模块。如果模块不在 Python 解释器搜索的路径范围内，就会抛出 ModuleNotFoundError 异常。

文件读写时的路径问题。在进行文件读写操作时，如果指定的文件路径不正确，会导致 FileNotFoundError 异常。文件路径可以是相对路径或绝对路径，使用不当就会找不到文件。

4.3.7 Conda-Forge 和 PyPI 如何选择

软件包管理。PyPI：主要通过 pip 工具进行软件包的安装、升级和卸载。pip 专注于 Python 软件包的管理，对于纯 Python 的软件包，使用 pip 从 PyPI 安装是非常便捷的，且与 Python 的标准库集成紧密。Conda-Forge：使用 conda 命令进行软件包管理。conda 不仅能管理 Python 软件包，还能管理其他语言的软件包以及系统级的依赖，并且可以方便地创建和管理不同的虚拟环境，确保不同项目的依赖隔离。

软件包类型和兼容性。PyPI：包含大量的纯 Python 软件包，涵盖了各种领域和功能。但对于一些依赖于特定系统库或需要复杂编译过程的软件包，在不同操作系统和平台上可能会遇到兼容性问题。Conda-Forge：提供了丰富的软件包，包括 Python 软件包以及非 Python 的依赖项，如 C、C++ 等语言编写的库。Conda-Forge 注重跨平台兼容性，其软件包通常经过了预编译和优化，在不同操作系统（如 Windows、Linux、macOS）上更容易安装和运行，尤其适用于科学计算和数据科学领域中依赖多种底层库的软件包。

社区和生态系统。PyPI：作为 Python 官方的软件包索引，拥有庞大的社区和丰富的文档资源。Conda-Forge：是一个由社区驱动的开源项目，专注于为 conda

提供高质量的软件包。

五、编写和运行 Python 程序

```
(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha (main)
$ cd myproject/

(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$ ls -l
total 1
-rw-r--r-- 1 Lenovo 197121 75 Mar 19 12:47 environment.yml

(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$ cat environment.yml
name: myproject
channels:
  - conda-forge
dependencies:
  - python=3.12
(base) Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$
```

```
$ curl -O https://edg.epa.gov/EPADataCommons/public/OA/EPA_SmartLocationDatabase_V3_Jan_2021_Final.csv
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  192M  100  192M    0     0  79574      0  0:42:13  0:42:13 --:--:-- 1426k
(myproject)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$
(myproject)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$ python main.py
10.69% of U.S. residents live in highlywalkable neighborhoods.
(myproject)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$
```

```
conda list
packages in environment at C:\Users\Lenovo\.conda\envs\myproject:

Name                    Version                   Build  Channel
zip2                    1.0.8                h2466b09_7    conda-forge
a-certificates          2025.1.31            hb56e8100_0    conda-forge
ntel-openmp             2024.2.1          h57928b3_1083    conda-forge
ibblas                  3.9.0           31_h641d27c_mkl    conda-forge
ibcblas                 3.9.0           31_h5e41251_mkl    conda-forge
ibexpat                 2.6.4                he0c23c2_0    conda-forge
ibffi                   3.4.6                h537db12_0    conda-forge
ibhwloc                 2.11.2          default_ha69328c_1001    conda-fo
ibiconv                 1.18                 h135ad9c_1    conda-forge
iblapack                3.9.0           31_h1aa476e_mkl    conda-forge
iblzma                  5.6.4                h2466b09_0    conda-forge
ibsqlite                3.49.1               h67fdade_2    conda-forge
ibwinpthread            12.0.0.r4.gg4f2fc60ca      h57928b3_9    conda-fo
ibxml2                  2.13.6               he286e8c_0    conda-forge
ibzlib                  1.3.1                h2466b09_2    conda-forge
kl                      2024.2.2            h66d3029_15    conda-forge
umpy                    2.2.4           py312h3150e54_0    conda-forge
penssl                  3.4.1                ha4e3fda_0    conda-forge
andas                   2.2.3           py312h72972c8_1    conda-forge
ip                      25.0.1             pyh8b19718_0    conda-forge
ython                   3.12.9          h3f84c4b_1_cpython    conda-forge
ython-dateutil          2.9.0.post0        pyhff2d567_1    conda-forge
ython-tzdata            2025.1             pyhd8ed1ab_0    conda-forge
ython_abi               3.12                     5_cp312    conda-forge
ytz                     2024.1             pyhd8ed1ab_0    conda-forge
etuptools               75.8.2             pyhff2d567_0    conda-forge
ix                      1.17.0             pyhd8ed1ab_0    conda-forge
bb                      2021.13.0            h62715c5_1    conda-forge
k                       8.6.13               h5226925_1    conda-forge
zdata                   2025a                h78e105d_0    conda-forge
crt                     10.0.22621.0         h57928b3_1    conda-forge
c                       14.3                hbf610ac_24    conda-forge
c14_runtime             14.42.34438         hfd919c2_24    conda-forge
heel                    0.45.1             pyhd8ed1ab_1    conda-forge
(myproject)
enovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$ python main.py
ello, conda!
(myproject)
enovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
```

遇到的问题：AttributeError: module 'pandas' has no attribute '_version_'. Did you mean: '__version__'?

(myproject)

解决方法：将_换成__即可

```
$ python main.py
Hello, conda!
Traceback (most recent call last):
  File "C:\Users\Lenovo\Desktop\twinkledehaha\myproject\main.py", line 11, in <module>
    main()
  File "C:\Users\Lenovo\Desktop\twinkledehaha\myproject\main.py", line 6, in main
    print(pd._version_)
          ^^^^^^^^^^^
AttributeError: module 'pandas' has no attribute '_version_'. Did you mean: '__version__'?
(myproject)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$ python main.py
Hello, conda!
2.2.3
C:\Users\Lenovo\.conda\envs\myproject\Lib\site-packages\pandas\__init__.py
(myproject)
Lenovo@LAPTOP-8MNNHDHQ MINGW64 ~/Desktop/twinkledehaha/myproject (main)
$
```