파이썬활용 데이터 분석과정

강사: 황성민

Contents

1. Python이란?

1. 1989년 귀도 반 로썸(Guido van Rossum)에 의해 개발된 고급 프로그래밍 언어.
2. 비교적 쉽고 간단한 문법으로 배우고 사용하기 쉬운 언어.
3. 머신러닝(Machine Learning), 딥러닝(Deep Learning)등 4차 산업의 핵심적인 기술을 구현하는데 필요한 라이브러리가 많음.

Aug 2021	Aug 2020	Change	Programming Language	Ratings	Change
1	1		C	12.57%	-4.41%
2	3	▲	Python	11.86%	+2.17%
3	2	▼	Java	10.43%	-4.00%
4	4		C++	7.36%	+0.52%
5	5		C#	5.14%	+0.46%

세계 프로그래밍 언어 순위

출처: https://www.tiobe.com/tiobe-index/

2. Python 언어의 특징

1. 인터프리티드 언어로 컴파일이 필요 없다.
   C 언어처럼 컴파일을 하지 않기 때문에

2. 읽고 쓰기 편하다. (공백 4칸 들여쓰기 = 코드를 적으면서 자동적으로 정리가 됨)

```
def bigger_than_five(x):
    if x > 5:    print("X is bigger than five")
    else:
        print("x is 5 or smaller")
```

2. Python 언어의 특징

3. 변수 타입을 자동으로 지정

자바

```
String myName = "Erik";
int myAge = 37;
float mySalary = 1250.70;
```

파이썬

```
my_name = "Erik"
my_age = 37
my_salary = 1250.70
```

4. Garbage Collection (메모리 관리)을 자동으로 해줌

5. 간결한 코드로 쉽고 빠르게 프로그래밍 가능


ANACONDA®

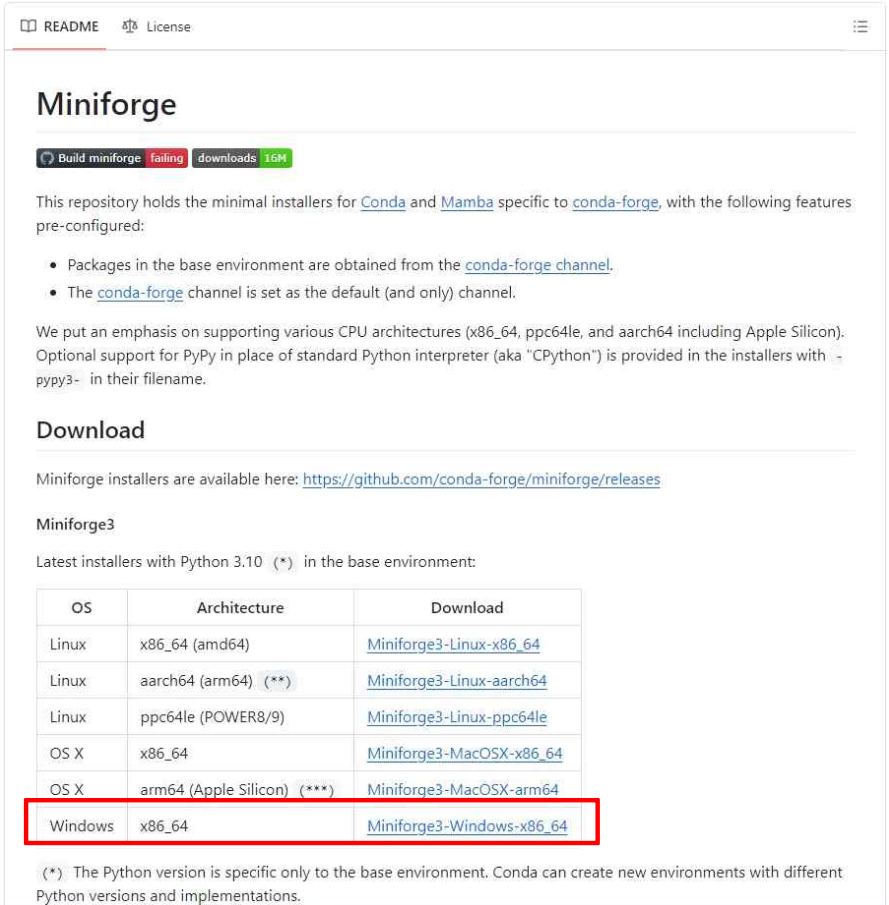
3. Anaconda?

파이썬의 여러 패키지 간 호환성을 관리해 주는 배포판

4.Mini-forge 설치

1. google에서 mini-forge 검색 후 mini-forge github 접속
   https://github.com/conda-forge/miniforge

4.Mini-forge 설치



README License

Miniforge

Build miniforge failing downloads 16M

This repository holds the minimal installers for Conda and Mamba specific to conda-forge, with the following features pre-configured:

- Packages in the base environment are obtained from the conda-forge channel.
- The conda-forge channel is set as the default (and only) channel.

We put an emphasis on supporting various CPU architectures (x86_64, ppc64le, and aarch64 including Apple Silicon). Optional support for PyPy in place of standard Python interpreter (aka "CPython") is provided in the installers with `-pypy3-` in their filename.

Download

Miniforge installers are available here: https://github.com/conda-forge/miniforge/releases

Miniforge3

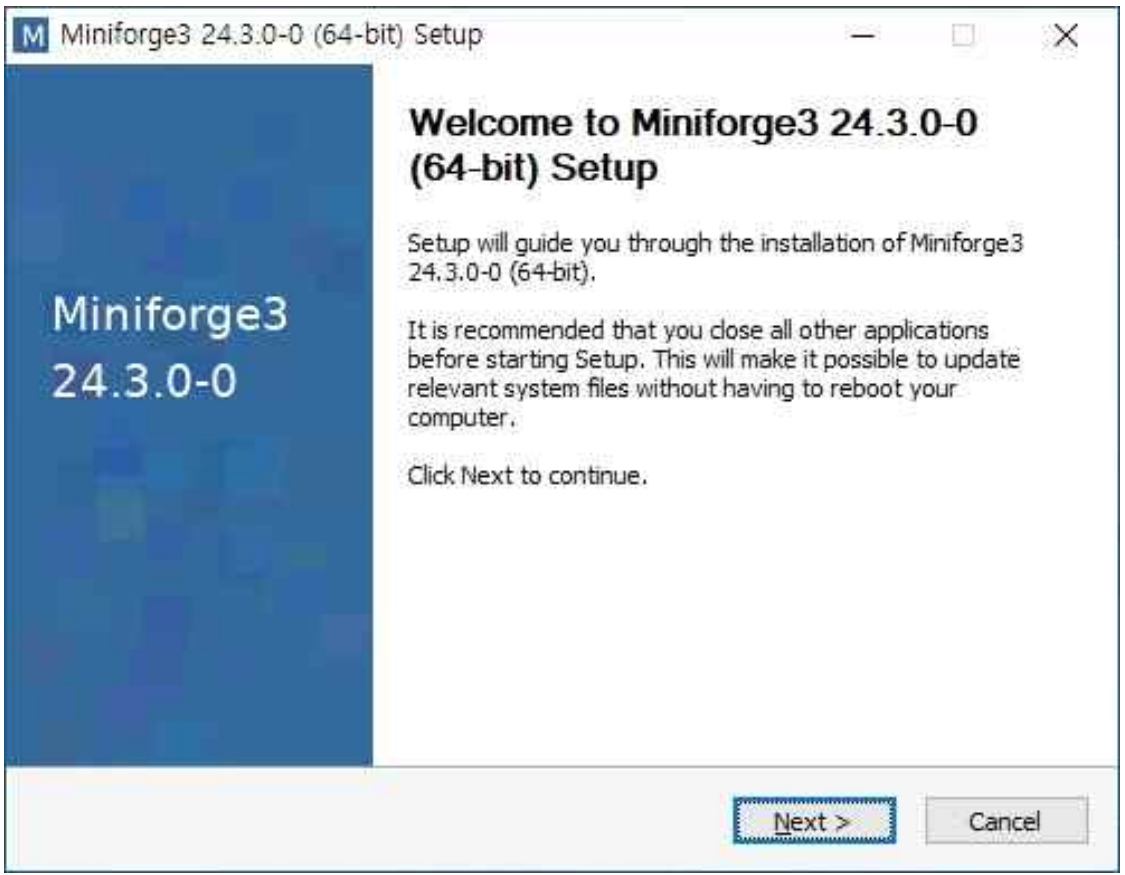
Latest installers with Python 3.10 `(*)` in the base environment:

OS	Architecture	Download
Linux	x86_64 (amd64)	Miniforge3-Linux-x86_64
Linux	aarch64 (arm64) `(**)`	Miniforge3-Linux-aarch64
Linux	ppc64le (POWER8/9)	Miniforge3-Linux-ppc64le
OS X	x86_64	Miniforge3-MacOSX-x86_64
OS X	arm64 (Apple Silicon) `(***)`	Miniforge3-MacOSX-arm64
Windows	x86_64	Miniforge3-Windows-x86_64

`(*)` The Python version is specific only to the base environment. Conda can create new environments with different Python versions and implementations.

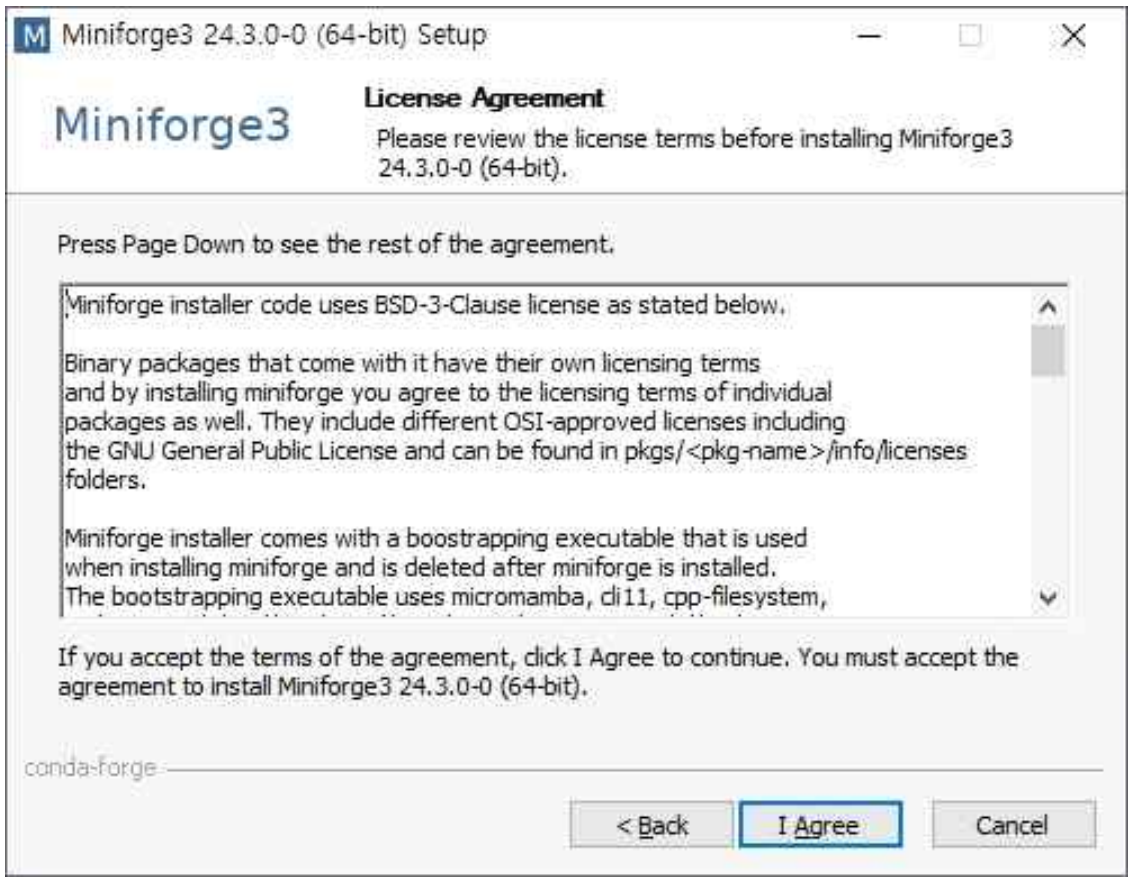
운영체제에 맞는 버전
다운 후 설치

4.Mini-forge 설치



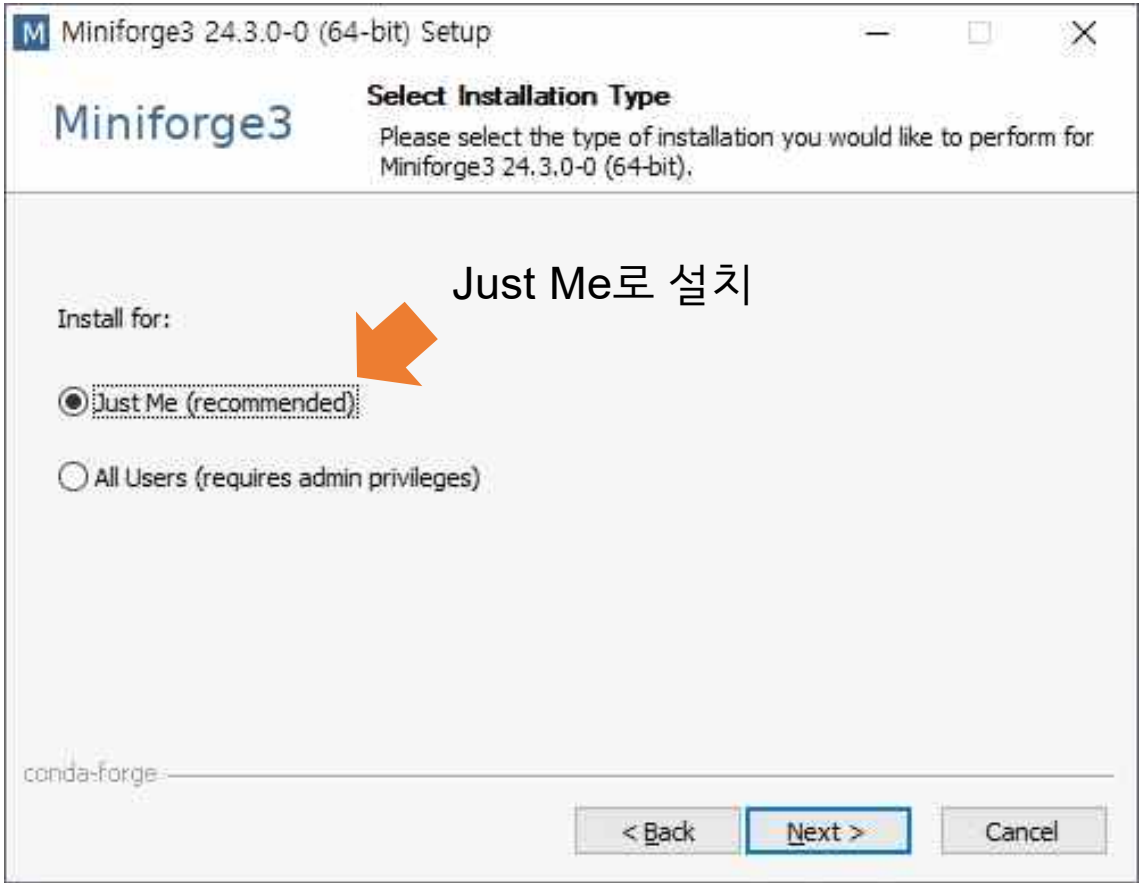
next 클릭

4.Mini-forge 설치



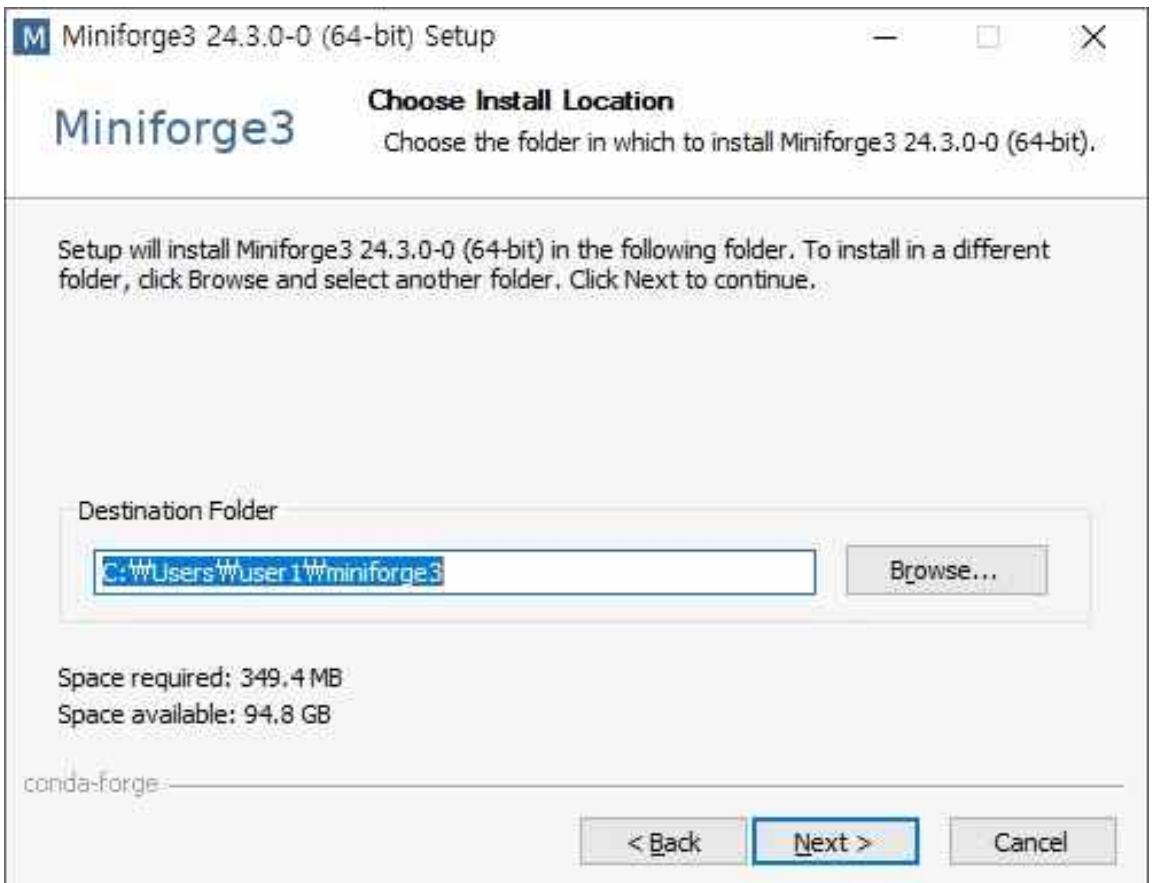
I Agree 클릭

4.Mini-forge 설치



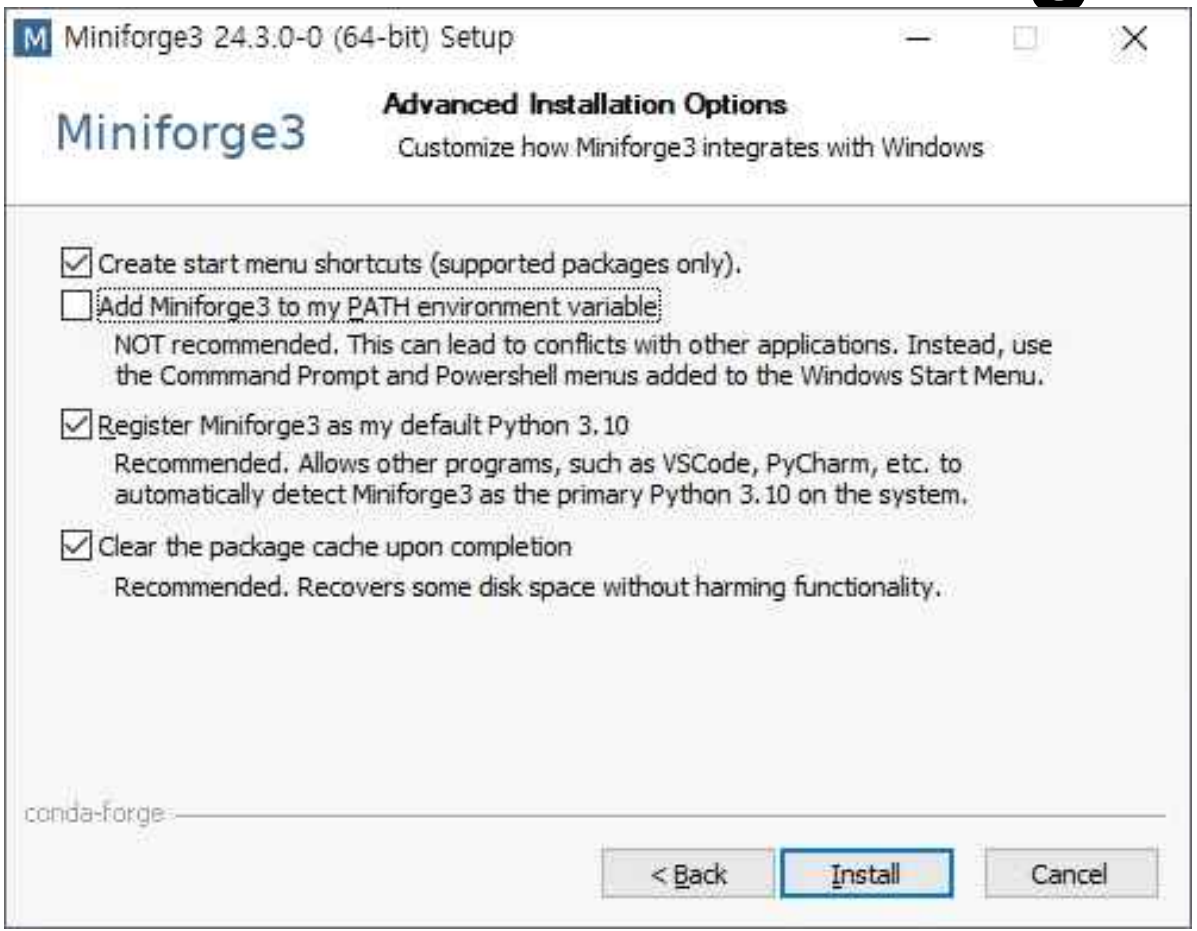
Just Me로 설치

4.Mini-forge 설치



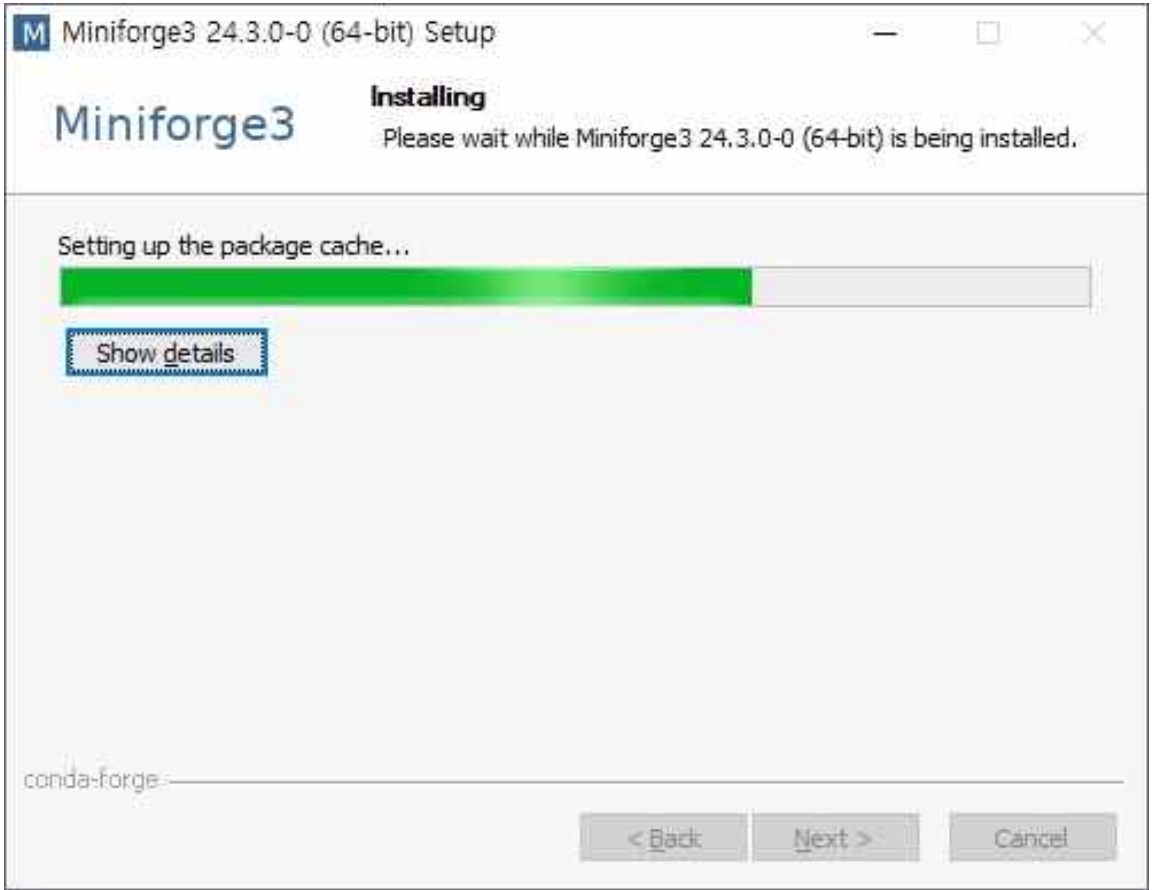
기본 경로로 설치
설치 용량 약 350MB

4.Mini-forge 설치

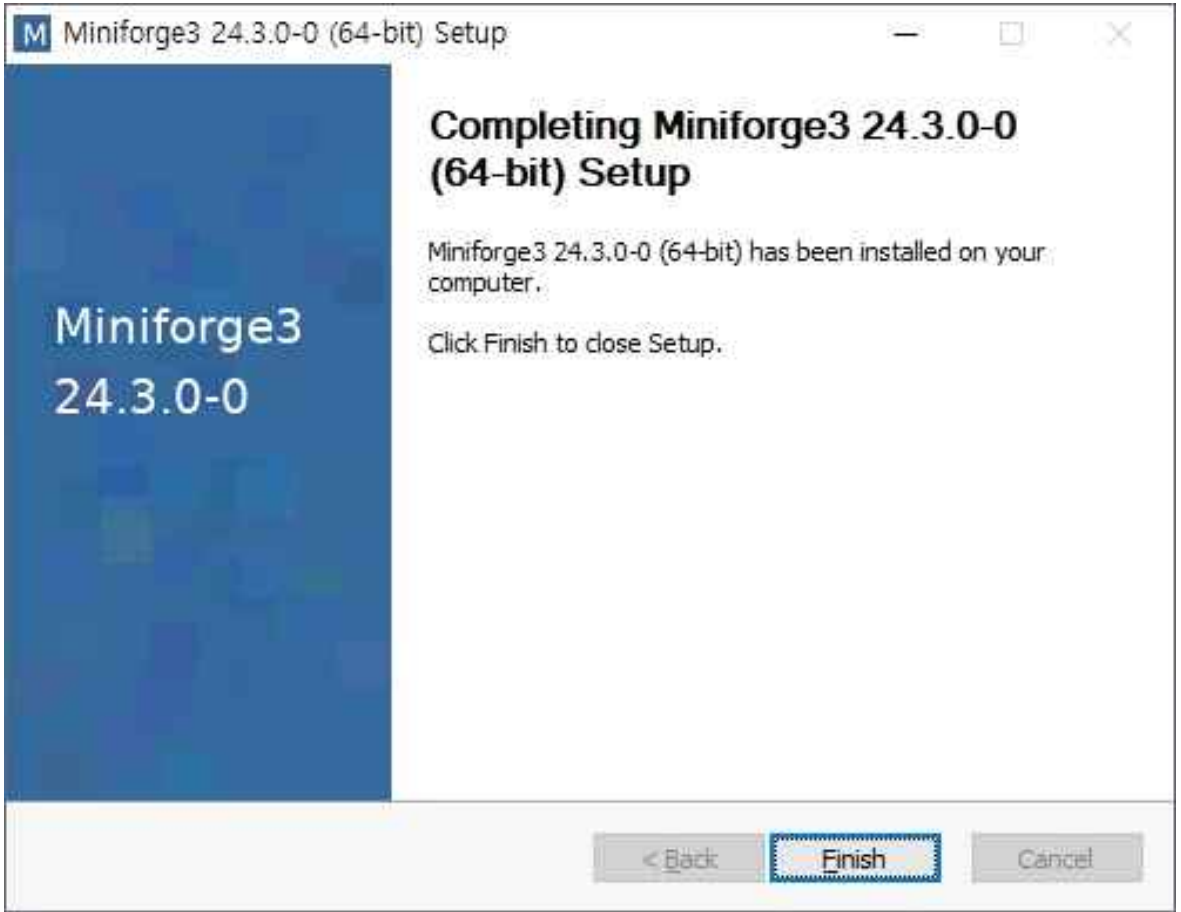


왼쪽과 같이
Add Miniforge3 to my Path만
제외하고 체크 후
Install 누름

4.Mini-forge 설치

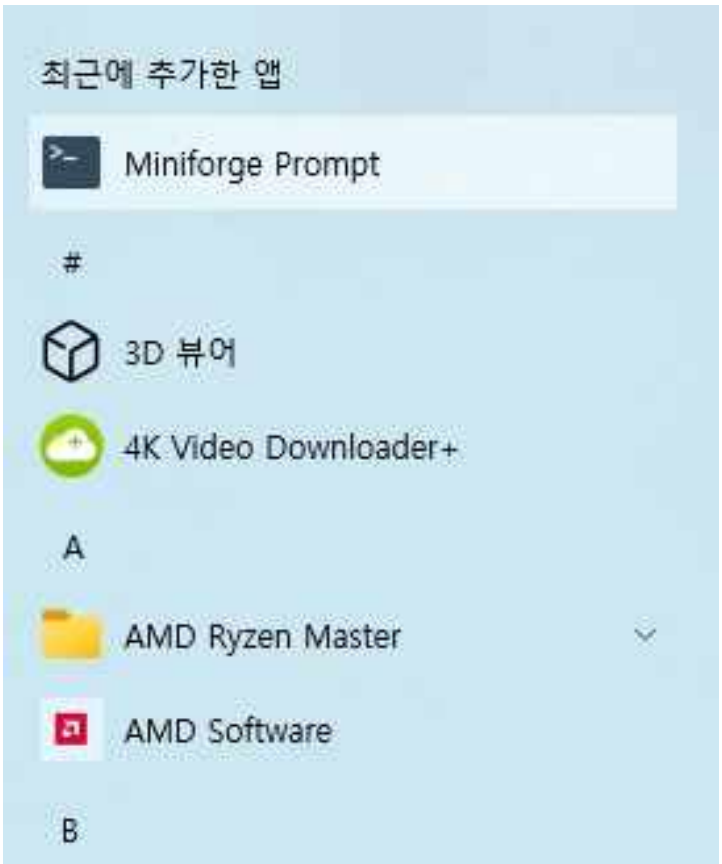


4.Mini-forge 설치



설치가 완료되면
Finish를 누르고 닫음

5.conda 가상환경 생성

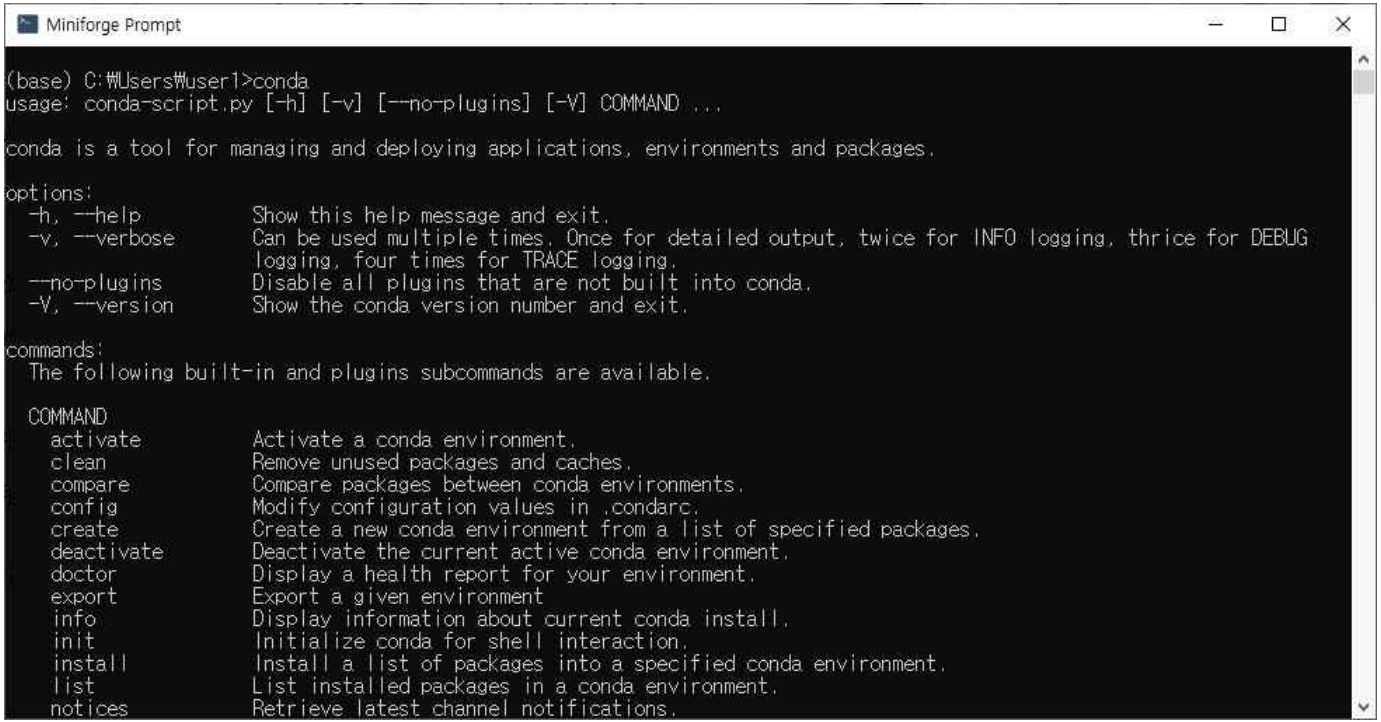


윈도우 시작버튼

▼

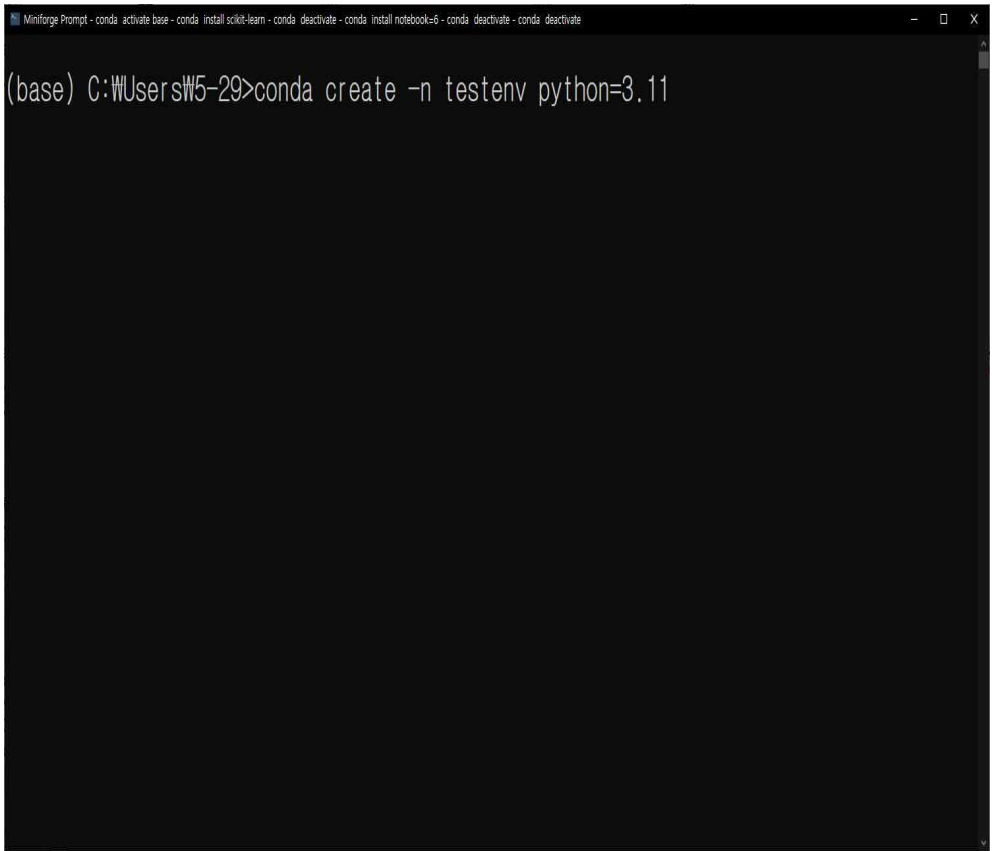
MIniforge Prompt 클릭

5.conda 가상환경 생성



경로에 (base)가 보이고 conda 명령어를 입력했을 때 실행이 되면 성공

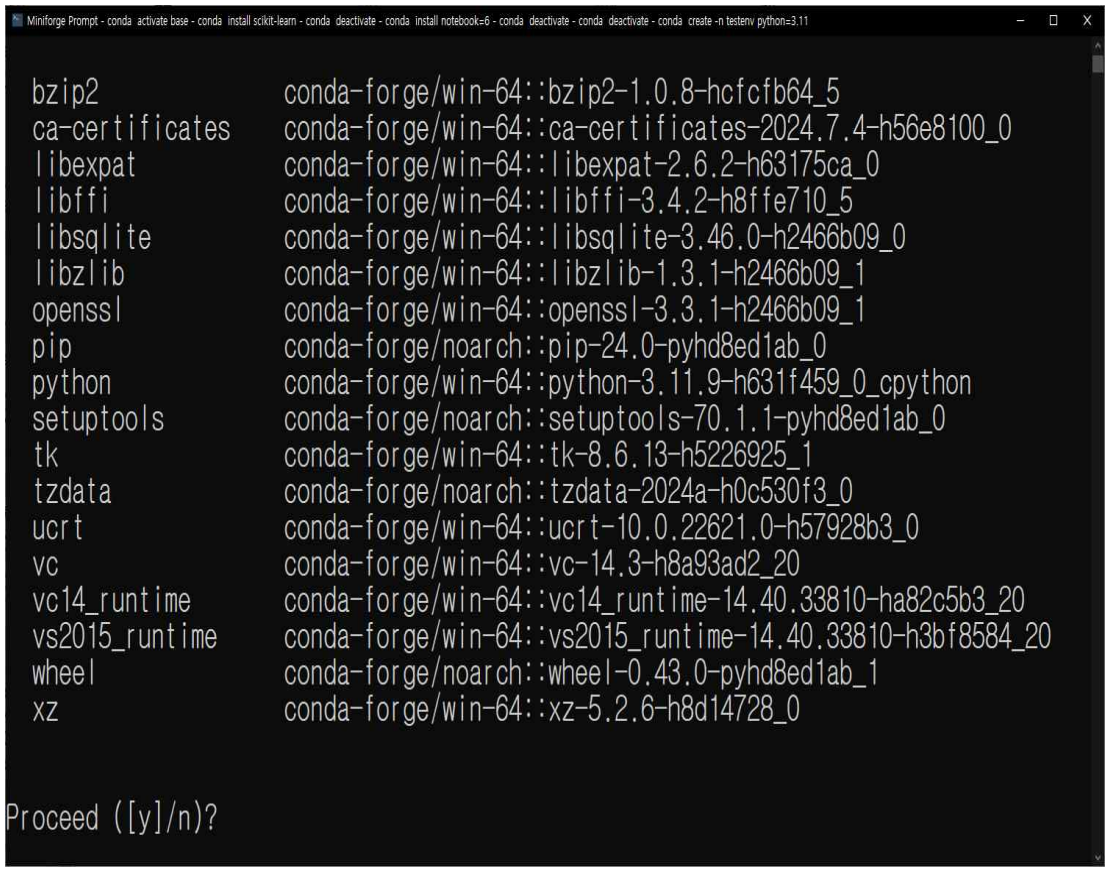
5.conda 가상환경 생성



conda create -n 가상환경명 python=버전

conda create -n testenv python=3.11

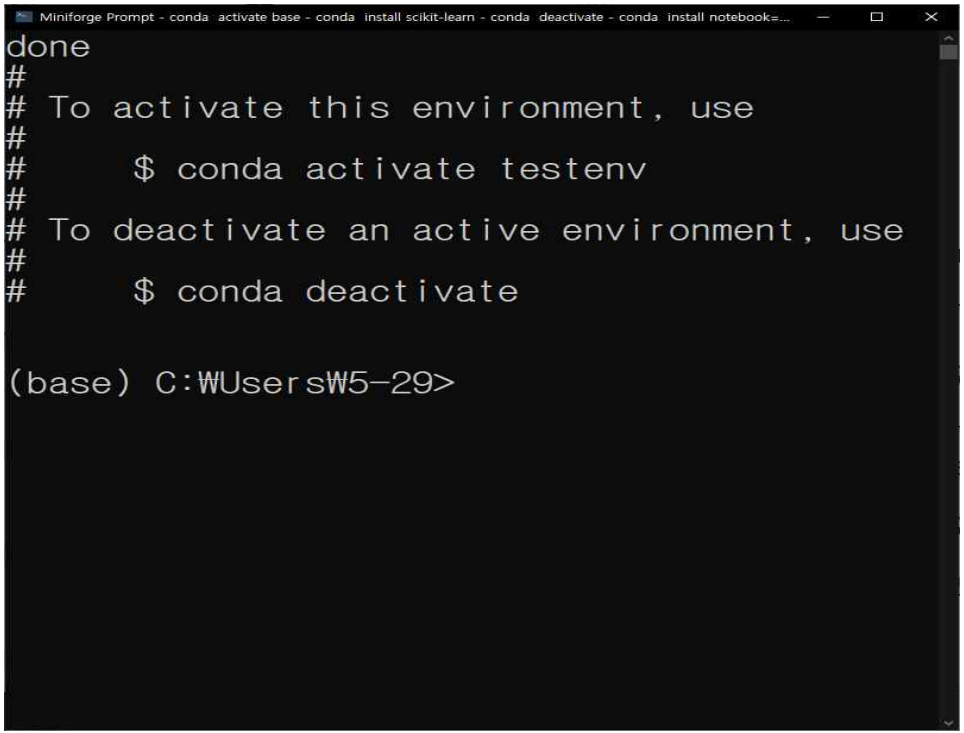
5.conda 가상환경 생성



Proceed([y]/n)? 이 나오면

y 입력 후 엔터

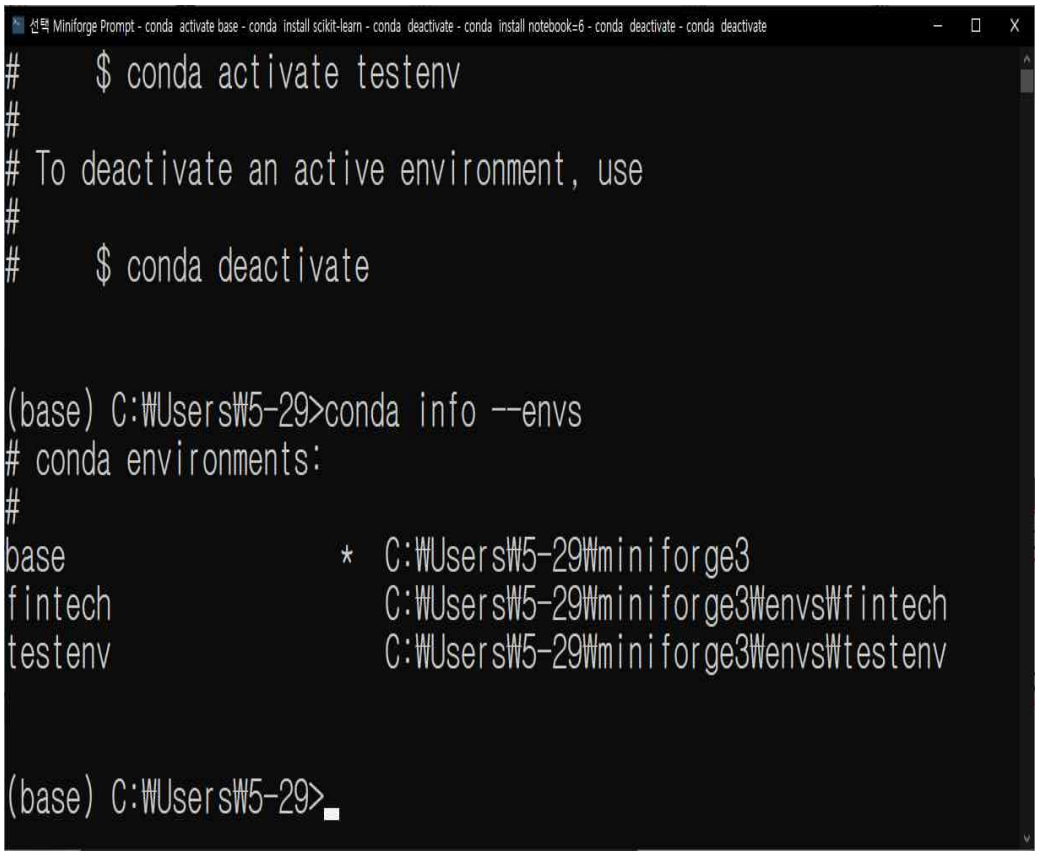
5.conda 가상환경 생성



done
#
To activate this environment, use
#
$ conda activate testenv
#
To deactivate an active environment, use
#
$ conda deactivate

가 뜨면 가상환경 생성 완료

5.conda 가상 환경 생성 확인



conda info -–envs

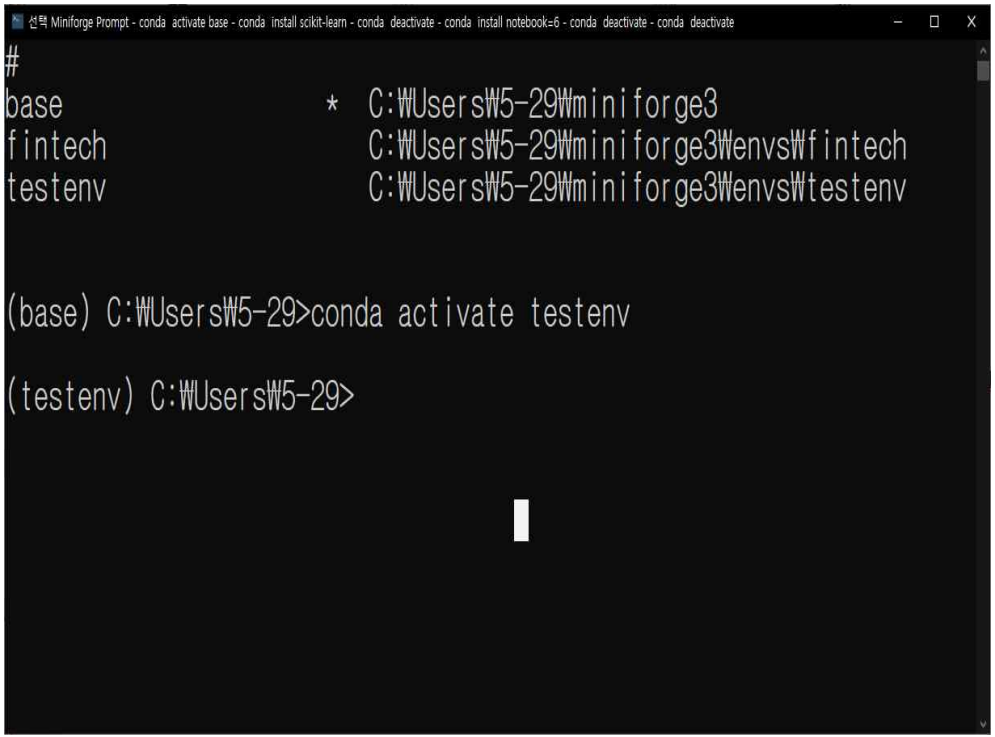
아나콘다의 가상 환경 확인 가능
* 이 있는 곳이 현재의 가상 환경

가상 환경은 여러 개를 만들고 삭제 가능

가상 환경을 만드는 이유

프로젝트 별로 다른 버전의 python이나 패키지를 설치해서 의존성 충돌 문제를 방지

5.conda 가상 환경 활성화



가상 환경 활성화

(base) C:\Users\5-29>conda activate testenv

가상 환경이 활성화 되면 base가 활성화된 가상 환경 이름으로 변경됨

(base) C:\Users\5-29>

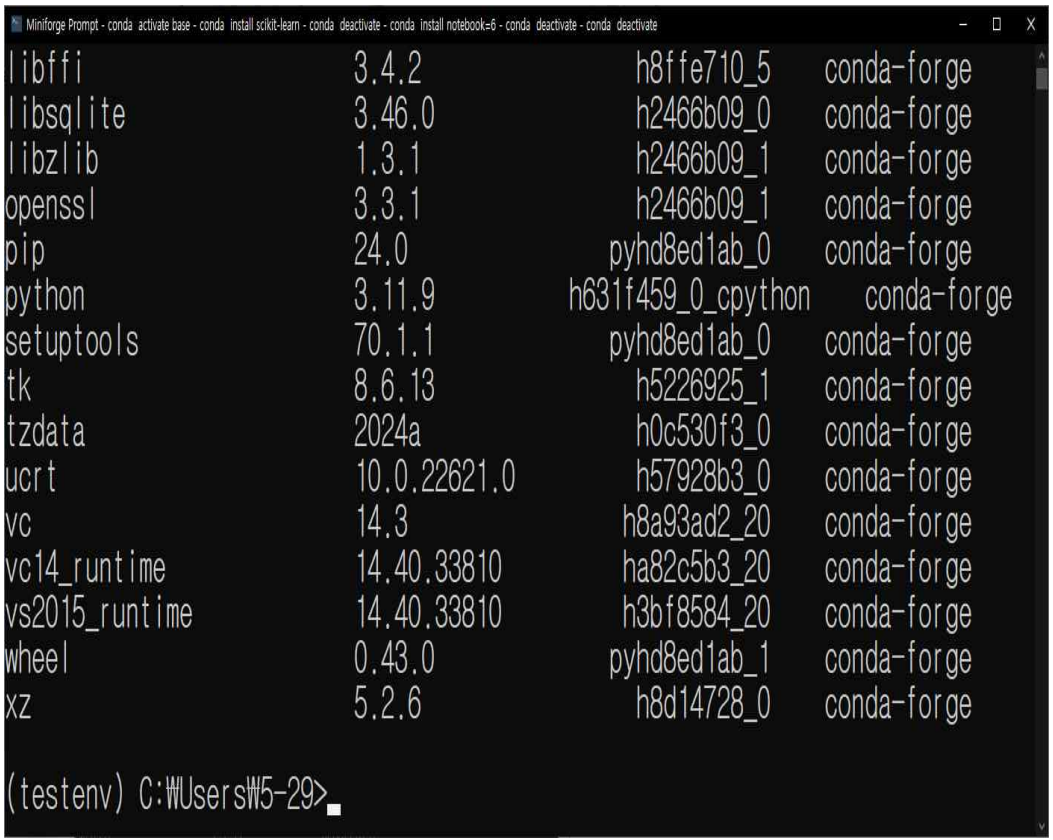
▼

(testenv) C:\Users\5-29>

가상 환경 비활성화

(testenv) C:\Users\5-29>conda deactivate

5.conda 가상 환경 패키지 목록



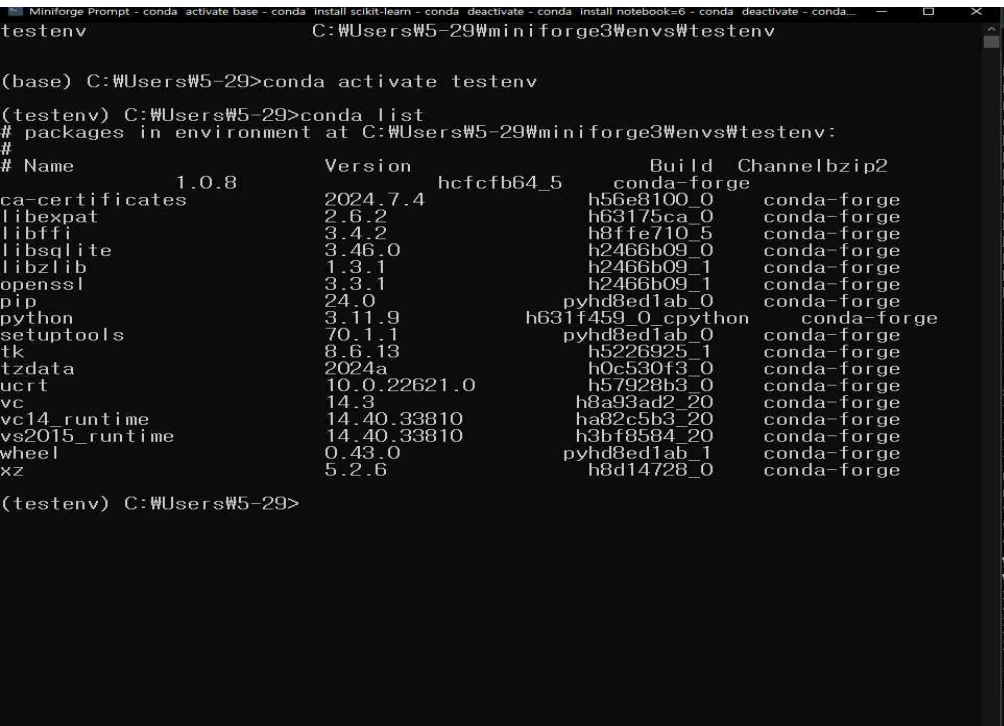
conda activate testenv

▼

conda list

가상환경 testenv에 설치된
패키지 목록 확인

5.conda 가상 환경 패키지 목록

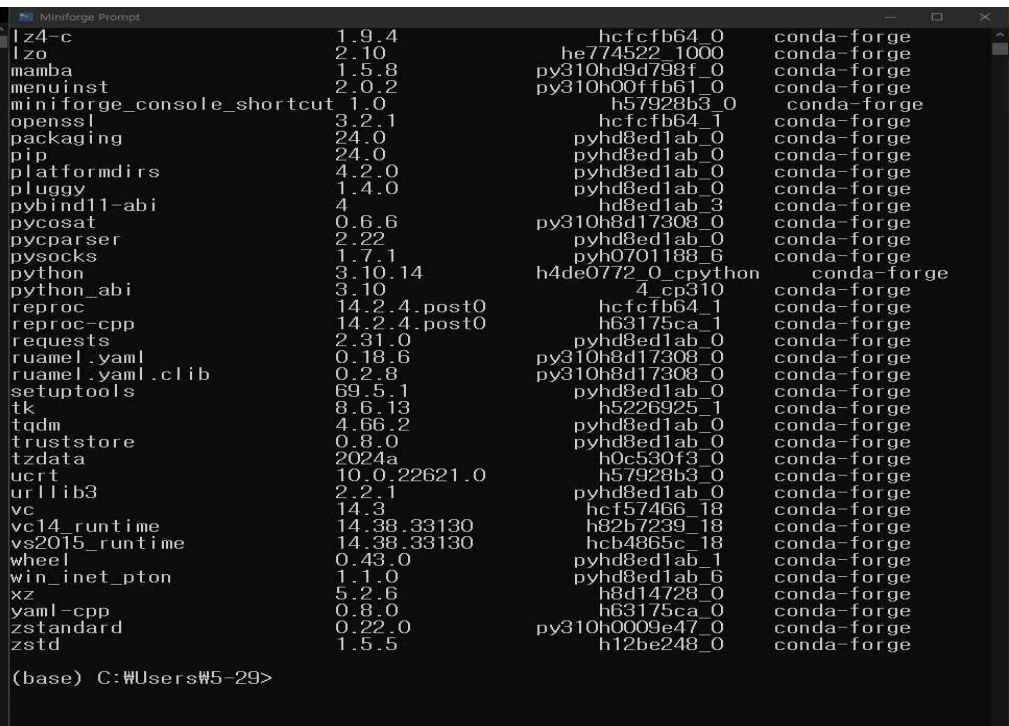


```
Miniforge Prompt - conda  activate base - conda  install scikit-learn - conda  deactivate - conda  install notebook=6 - conda  deactivate - conda...
testenv                  C:₩Users₩5-29₩miniforge3₩envs₩testenv

(base) C:₩Users₩5-29>conda activate testenv

(testenv) C:₩Users₩5-29>conda list
# packages in environment at C:₩Users₩5-29₩miniforge3₩envs₩testenv:
#
# Name                    Version                   Build  Channelbzip2
                1.0.8                hcfcfb64_5    conda-forge
ca-certificates           2024.7.4                h56e8100_0    conda-forge
libexpat                  2.6.2                   h63175ca_0    conda-forge
libffi                    3.4.2                   h8ffe710_5    conda-forge
libsqlite                 3.46.0                  h2466b09_0    conda-forge
libzlib                   1.3.1                   h2466b09_1    conda-forge
openssl                   3.3.1                   h2466b09_1    conda-forge
pip                       24.0                  pyhd8ed1ab_0    conda-forge
python                    3.11.9          h631f459_0_cpython    conda-forge
setuptools                70.1.1                pyhd8ed1ab_0    conda-forge
tk                        8.6.13                  h5226925_1    conda-forge
tzdata                    2024a                   h0c530f3_0    conda-forge
ucrt                      10.0.22621.0            h57928b3_0    conda-forge
vc                        14.3                   h8a93ad2_20    conda-forge
vc14_runtime              14.40.33810            ha82c5b3_20    conda-forge
vs2015_runtime            14.40.33810            h3bf8584_20    conda-forge
wheel                     0.43.0                pyhd8ed1ab_1    conda-forge
xz                        5.2.6                   h8d14728_0    conda-forge

(testenv) C:₩Users₩5-29>
```



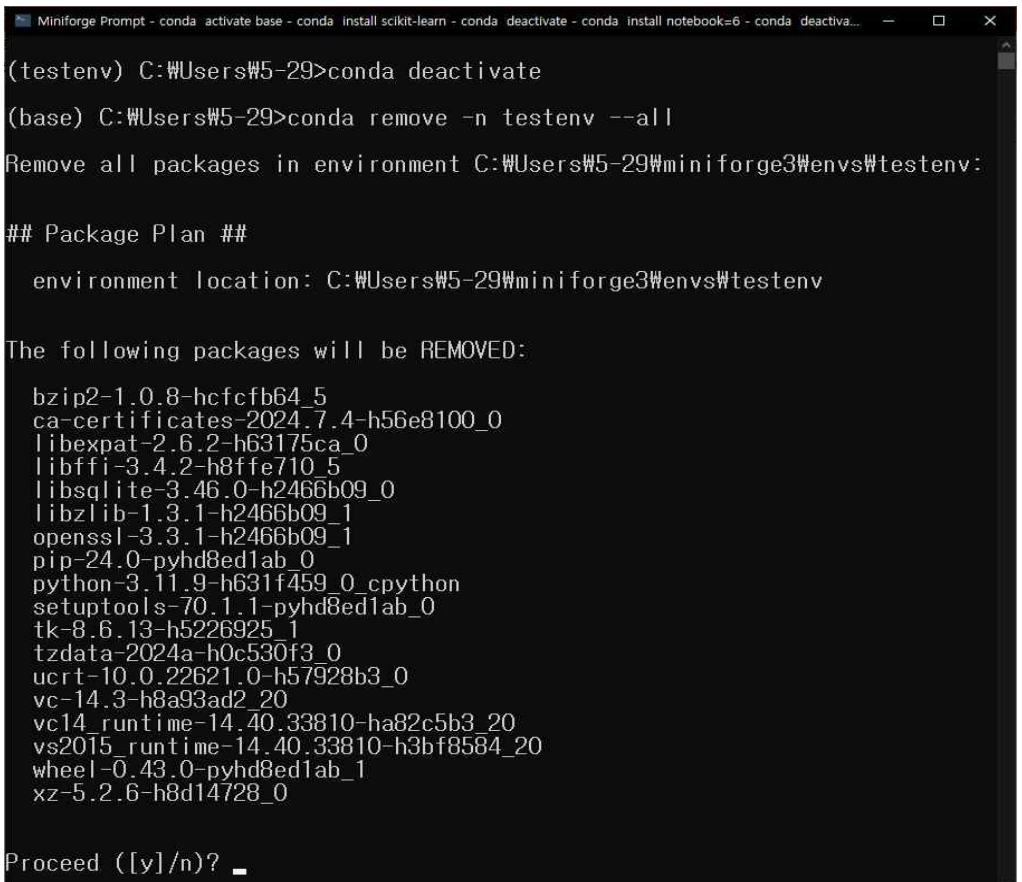
```
Miniforge Prompt
lz4-c                     1.9.4                hcfcfb64_0    conda-forge
lzo                       2.10              he774522_1000    conda-forge
mamba                     1.5.8            py310hd9d798f_0    conda-forge
menuinst                  2.0.2            py310h0ffb61_0    conda-forge
miniforge_console_shortcut 1.0                 h57928b3_0    conda-forge
openssl                   3.2.1                hcfcfb64_1    conda-forge
packaging                 24.0                 pyhd8ed1ab_0    conda-forge
pip                       24.0                 pyhd8ed1ab_0    conda-forge
platformdirs              4.2.0                pyhd8ed1ab_0    conda-forge
pluggy                    1.4.0                pyhd8ed1ab_0    conda-forge
pybind11-abi              4                     hd8ed1ab_3    conda-forge
pycosat                   0.6.6            py310h8d17308_0    conda-forge
pycparser                 2.22                 pyhd8ed1ab_0    conda-forge
pysocks                   1.7.1                pyh0701188_6    conda-forge
python                    3.10.14          h4de0772_0_cpython    conda-forge
python_abi                3.10                      4_cp310    conda-forge
reproc                    14.2.4.post0         hcfcfb64_1    conda-forge
reproc-cpp                14.2.4.post0         h63175ca_1    conda-forge
requests                  2.31.0               pyhd8ed1ab_0    conda-forge
ruamel.yaml               0.18.6           py310h8d17308_0    conda-forge
ruamel.yaml.clib          0.2.8            py310h8d17308_0    conda-forge
setuptools                69.5.1               pyhd8ed1ab_0    conda-forge
tk                        8.6.13                  h5226925_1    conda-forge
tqdm                      4.66.2               pyhd8ed1ab_0    conda-forge
truststore                0.8.0                pyhd8ed1ab_0    conda-forge
tzdata                    2024a                   h0c530f3_0    conda-forge
ucrt                      10.0.22621.0            h57928b3_0    conda-forge
urllib3                   2.2.1                pyhd8ed1ab_0    conda-forge
vc                        14.3                  hcf57466_18    conda-forge
vc14_runtime              14.38.33130           h82b7239_18    conda-forge
vs2015_runtime            14.38.33130           hcb4865c_18    conda-forge
wheel                     0.43.0               pyhd8ed1ab_1    conda-forge
win_inet_pton             1.1.0                pyhd8ed1ab_6    conda-forge
xz                        5.2.6                   h8d14728_0    conda-forge
yaml-cpp                  0.8.0                   h63175ca_0    conda-forge
zstandard                 0.22.0           py310h0009e47_0    conda-forge
zstd                      1.5.5                   h12be248_0    conda-forge

(base) C:₩Users₩5-29>
```

testenv가상 환경의 패키지 목록
Python 버전 3.11.9

base 가상 환경의 패키지 목록
Python 버전 3.10.14

5.conda 가상 환경 삭제



conda remove -n testenv --all

testenv 가상 환경에 있는 모든 것을 삭제

Proceed([y]/n)?가 나오면 Y


jupyter

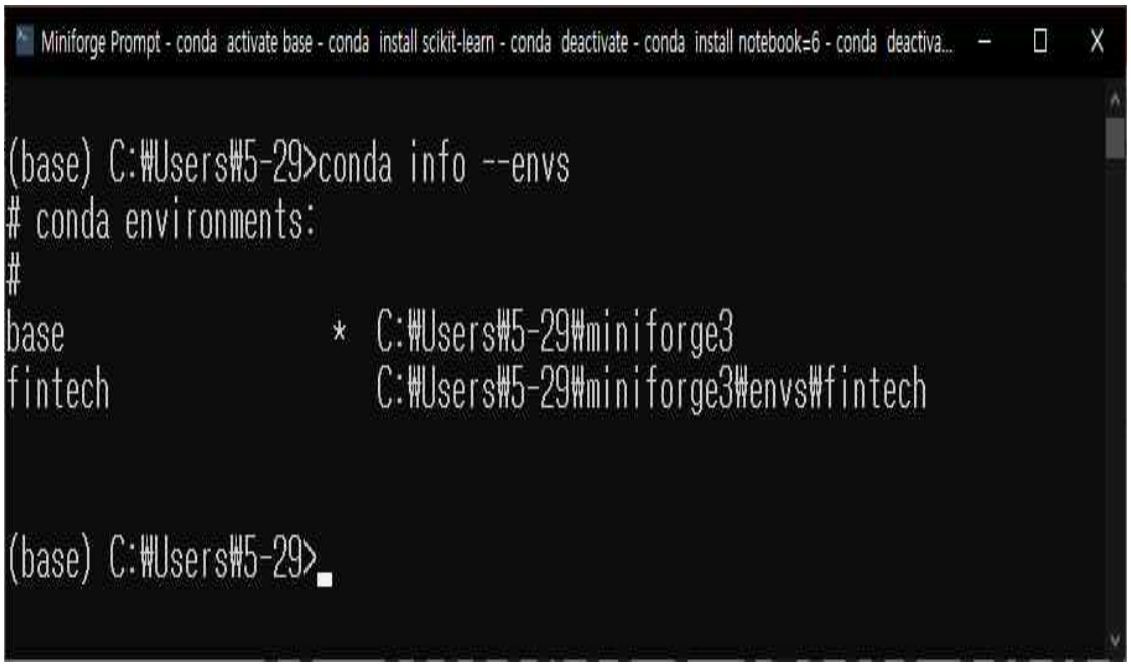
6. Jupyter Notebook 설치

가상 환경 재설치

conda create -n 가상환경명 python=버전

conda create -n fintech python=3.10

6. Jupyter Notebook 설치

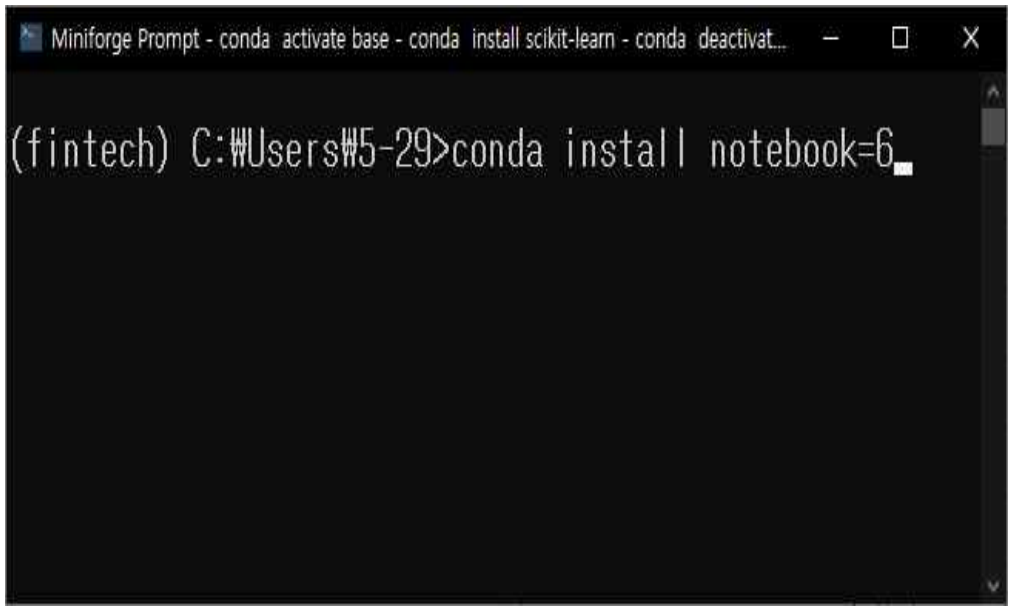


가상 환경 생성 확인

conda info -–envs

아나콘다의 가상 환경 확인 가능
* 이 있는 곳이 현재의 가상 환경

6. Jupyter Notebook 설치



가상 환경 활성화

(base) C:\Users\5-29>conda activate fintech

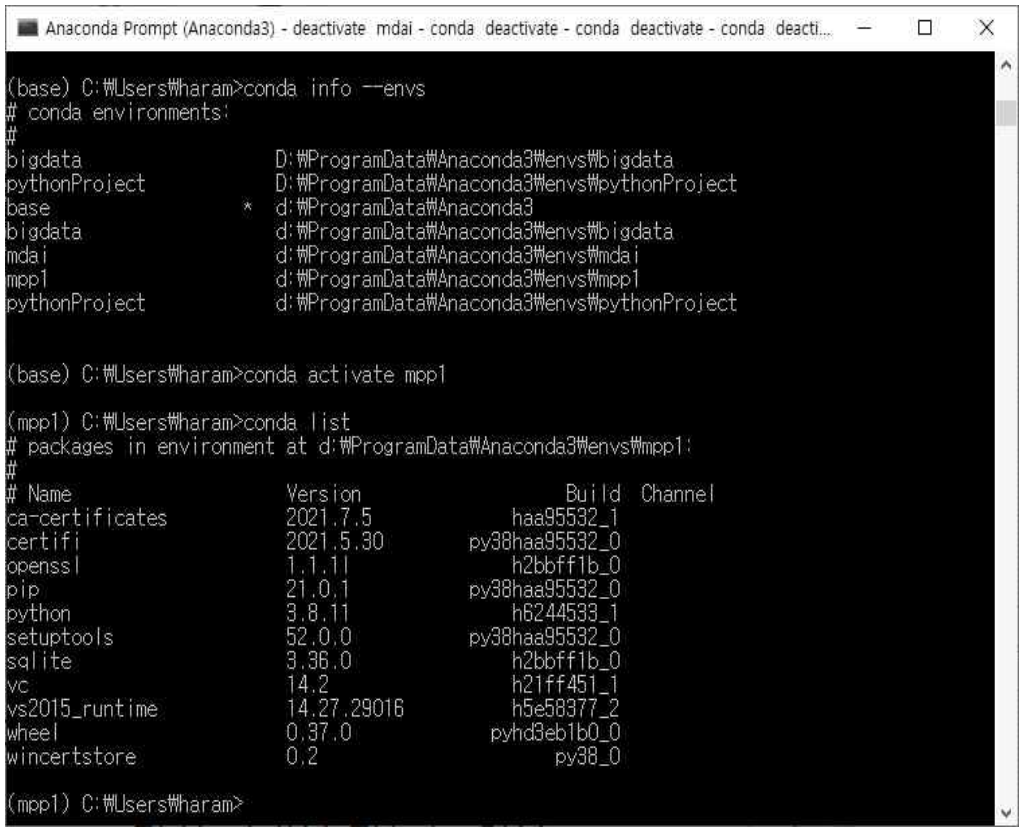
가상 환경이 활성화 되면 base가 활성화된 가상 환경 이름으로 변경되는 것 확인

(base) C:\Users\5-29>

▼

(fintech) C:\Users\ 5-29 >

6. Jupyter Notebook 설치



가상환경 fintech에 설치된 패키지 목록 확인

conda activate fintech

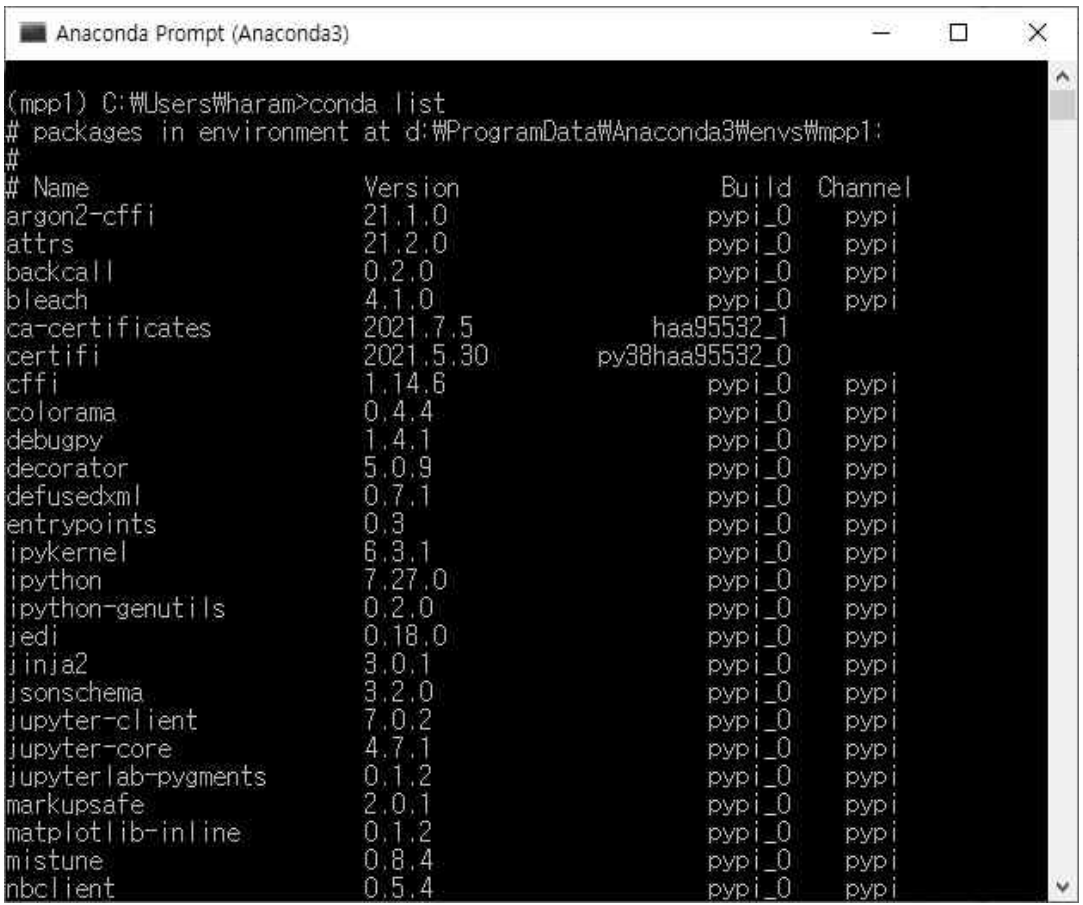
▼

conda list

Jupyter notebook이 없으므로 설치

conda install notebook=6

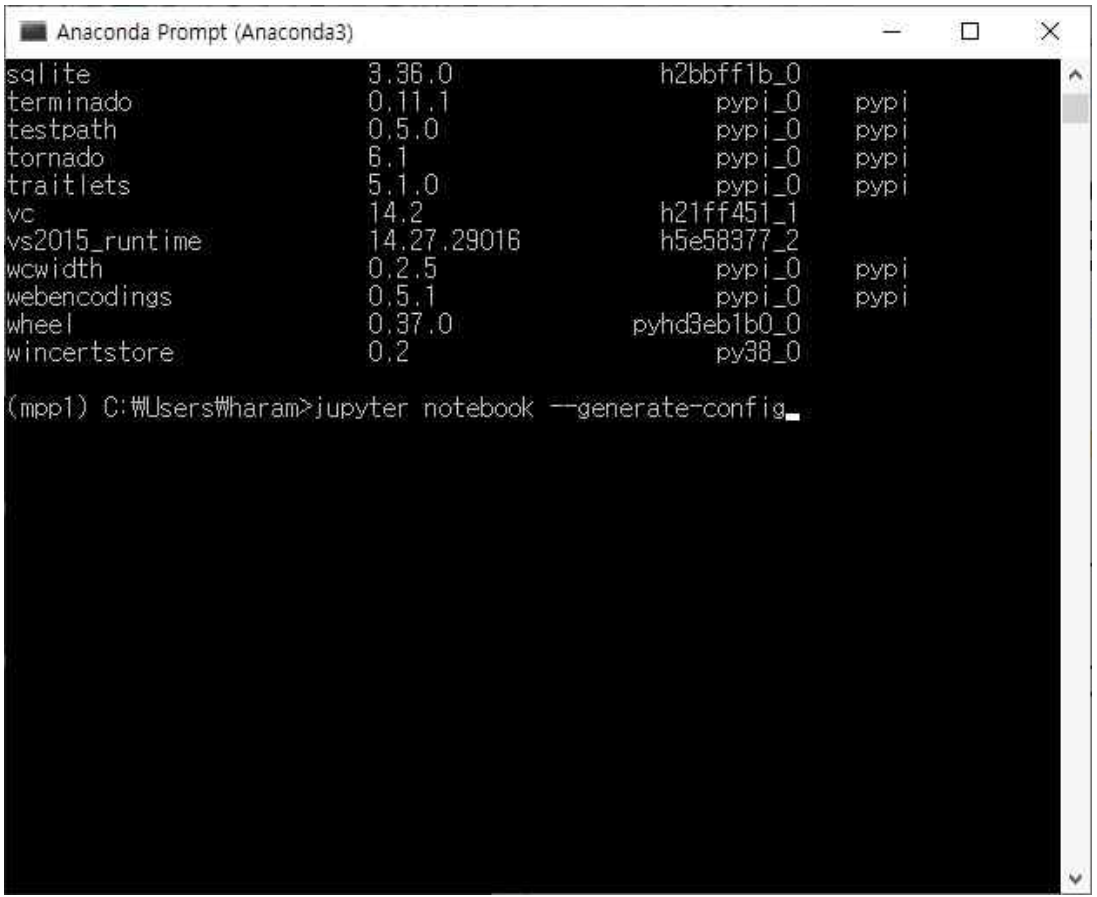
6. Jupyter Notebook 설치



conda list

List 중에 jupyter 설치 확인

7. Jupyter Notebook 환경설정



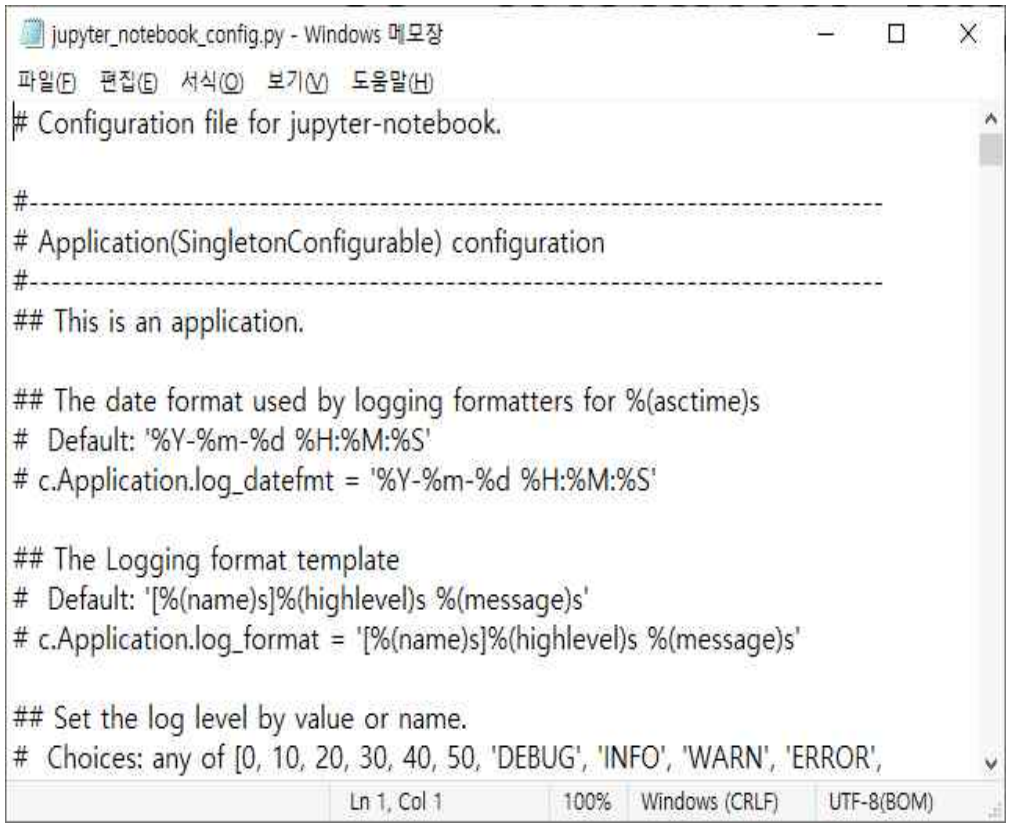
주피터 노트북을 실행하면
시작 폴더가 c:/로되어 있어 불편

1. 프로젝트 폴더를 따로 만들고
2. Jupyter Notebook 시작 폴더를 프로젝트 폴더로 변경

Jupyter notebook -–generate-config
로 설정파일 생성

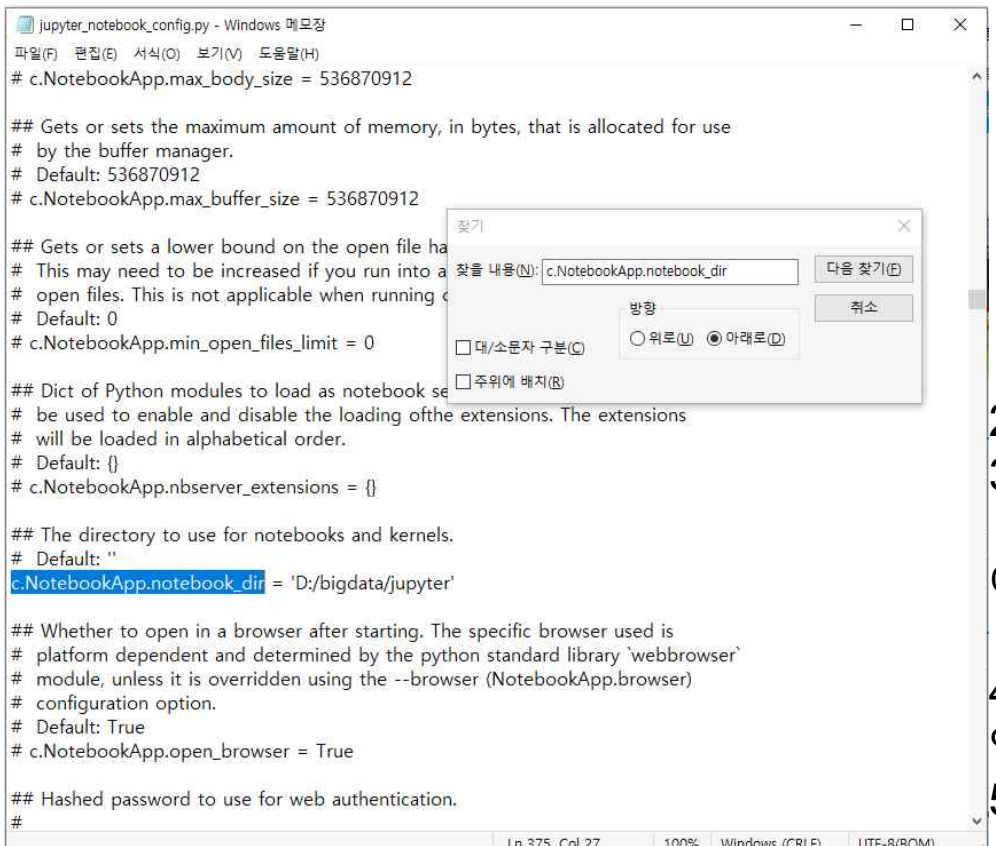
Writing default config to: 뒤에 나오는 경로 복사 후 윈도우 탐색기 경로창에 붙여넣기

7. Jupyter Notebook 환경설정



메모장 혹은 다른 텍스트 에디터를 이용해서
jupyter_notebook_config.py 열기

7. Jupyter Notebook 환경설정



1. F3키를 누른 뒤 찾을 내용에

notebook_dir

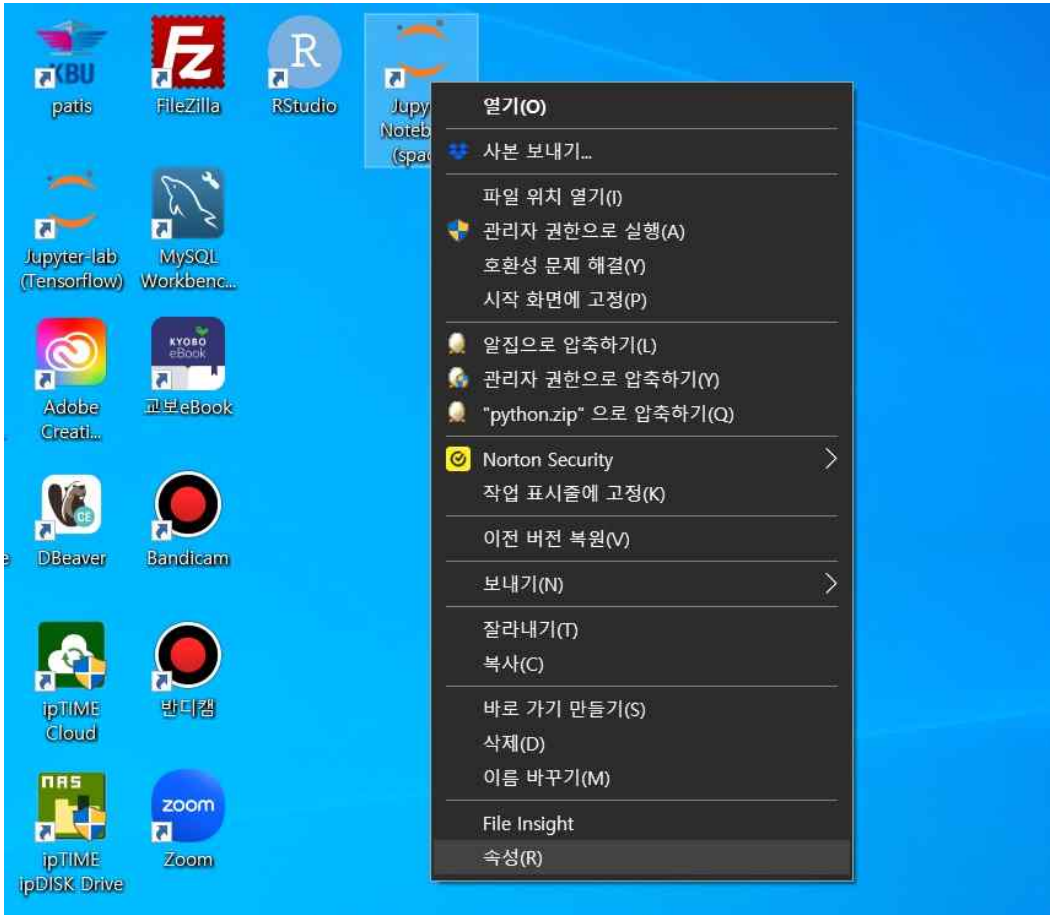
입력 후 **다음 찾기** 클릭

2.앞에 **# 삭제** 후 **공백 없이 왼쪽 끝으로** 붙임
3. dir = '' **따옴표 안쪽**에 프로젝트 경로 지정

예) 'C:/fintech_service'

4. #삭제 후 공백 없애기, ' '안에 경로 설정 확인
5. 파일 → 저장

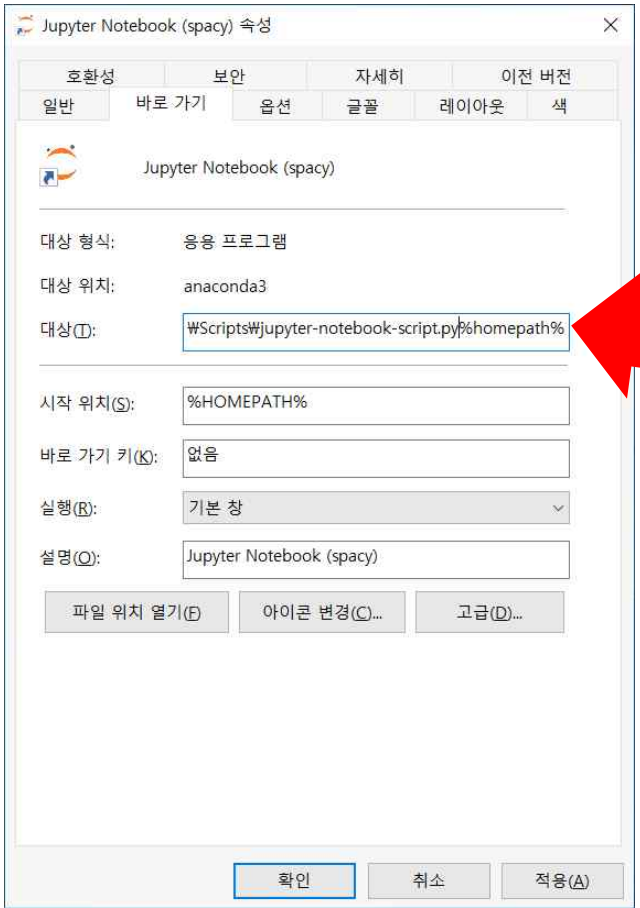
7. Jupyter Notebook 환경설정



- Jupyter Notebook 아이콘에 마우스를 올려놓고 우측 버튼 클릭.

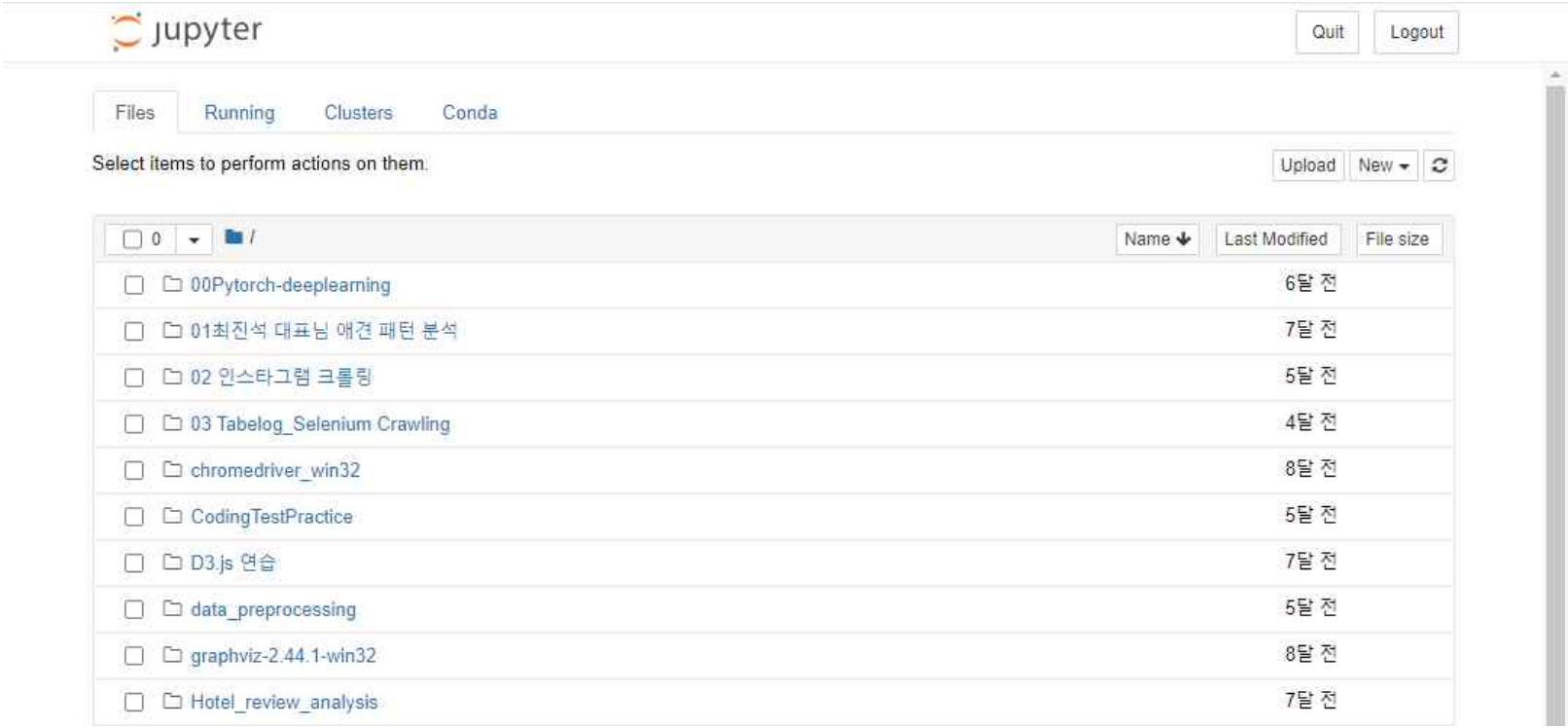
- 가장 아래의 속성 클릭
- 가장 아래의 속성 클릭

7. Jupyter Notebook 환경설정



- 대상(T): 내용중 script.py 뒤의 %로 시작하는 부분 삭제

7. Jupyter Notebook 환경설정



Jupyter notebook을 실행했을 때 설정한 폴더가 루트로 나오면 끝