

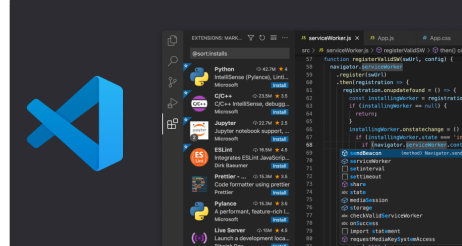
# 1.2 Docker Jupyter 환경 구축

- docker-compose를 통한 실습 이미지 만들기
- visual source code 실행 - 없는 경우 설치 진행

Download Visual Studio Code - Mac, Linux, Windows

Visual Studio Code is free and available on your favorite platform - Linux, macOS, and Windows. Download Visual Studio Code to experience a redefined code editor, optimized for

<https://code.visualstudio.com/Download>



## 1.2.1 실습을 위한 폴더 생성

- 아래의 명령어 실행 : (win+R) 복사하기>붙여넣기

```
mkdir class-source\source ; class-source\share ; class-source\docker
```

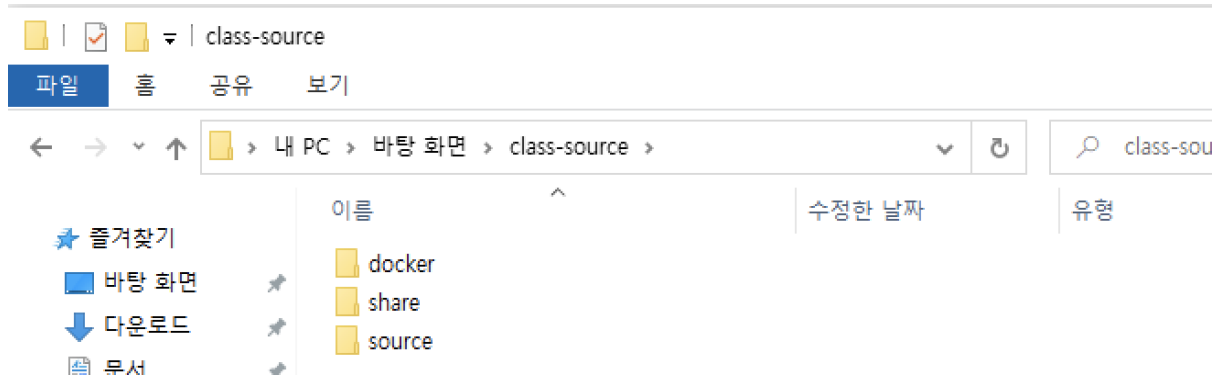
cd : change directory

```
명령 프롬프트
C:\Users\User\Desktop>mkdir class-source\source ; class-source\share ; class-source\docker
C:\Users\User\Desktop>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: D030-6E96

C:\Users\User\Desktop 디렉터리
2022-05-29 오전 11:34 <DIR> .
2022-05-29 오전 11:34 <DIR> ..
2022-05-29 오전 11:34 <DIR> class-source
2022-05-29 오전 10:56      2,140 Docker Desktop.lnk
2022-05-29 오전 11:29      1,401 Visual Studio Code.lnk
                2개 파일      3,541 바이트
                3개 디렉터리 19,189,469,184 바이트 남음

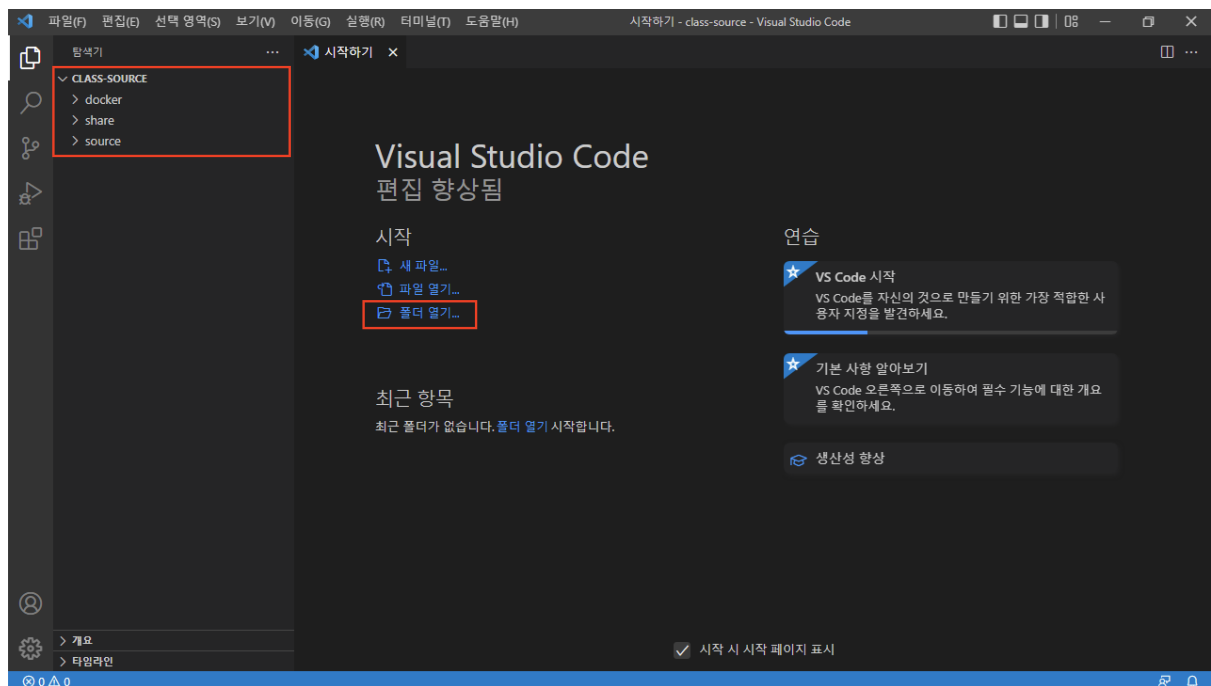
C:\Users\User\Desktop>dir class-source
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: D030-6E96

C:\Users\User\Desktop\class-source 디렉터리
2022-05-29 오전 11:34 <DIR> .
2022-05-29 오전 11:34 <DIR> ..
2022-05-29 오전 11:34 <DIR> docker
2022-05-29 오전 11:34 <DIR> share
2022-05-29 오전 11:34 <DIR> source
                0개 파일      0 바이트
                5개 디렉터리 19,188,547,584 바이트 남음
```



## 1.2.2 실습 코드 작성을 위해 폴더 열기

- visual source code 실행
- 생성된 폴더 열기



## 1.2.3 docker 폴더에 새 파일 생성

- file name : class-docker-jupyter.yml

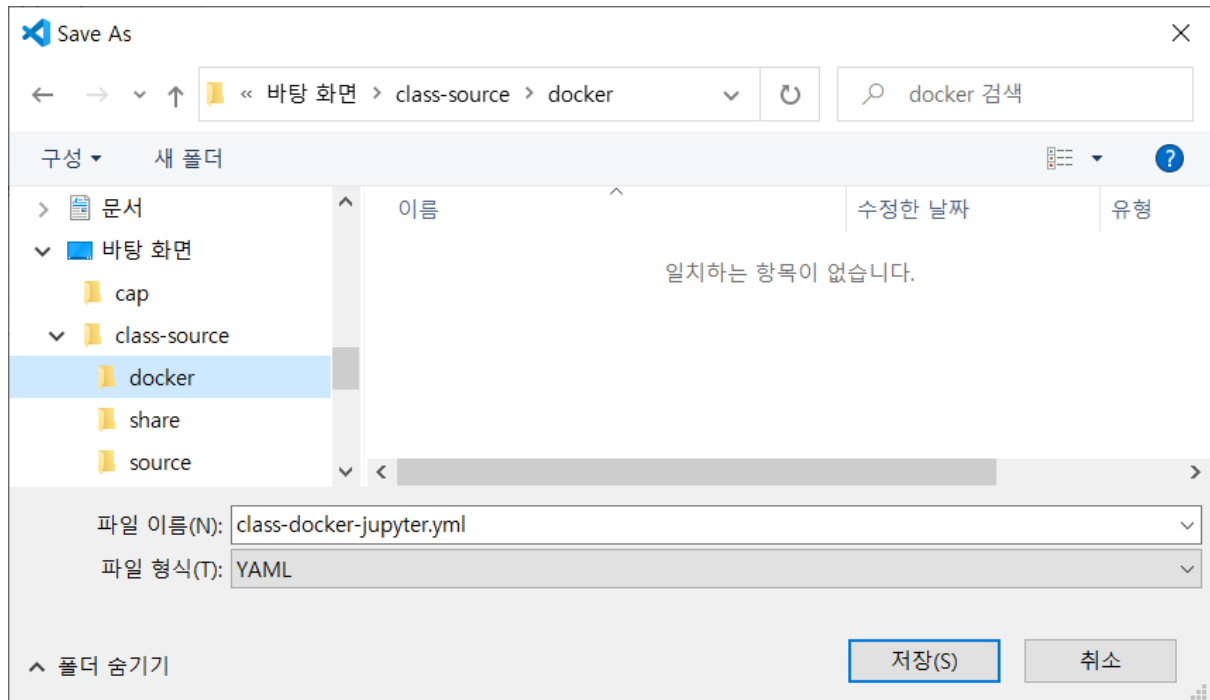
```
version: "3"
services:
  python_class:
    image: jupyter/base-notebook:python-3.9.12
    build :
```

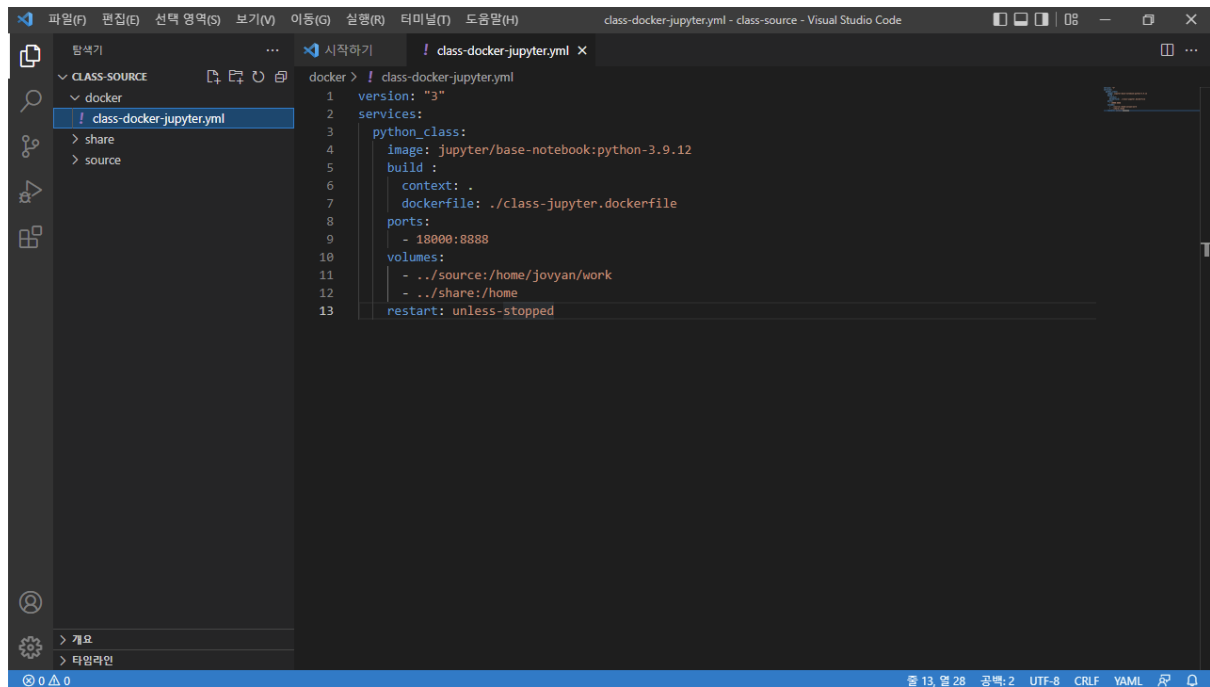
```

context: .
dockerfile: ./class-jupyter.dockerfile
ports:
  - 18000:8888
volumes:
  - ../source:/home/jovyan/work
restart: unless-stopped

```

저장하기 File>Save As> `class-docker-jupyter.yml`





## 1.2.4. Jupyter Image 생성시 기본 Python Library 설치

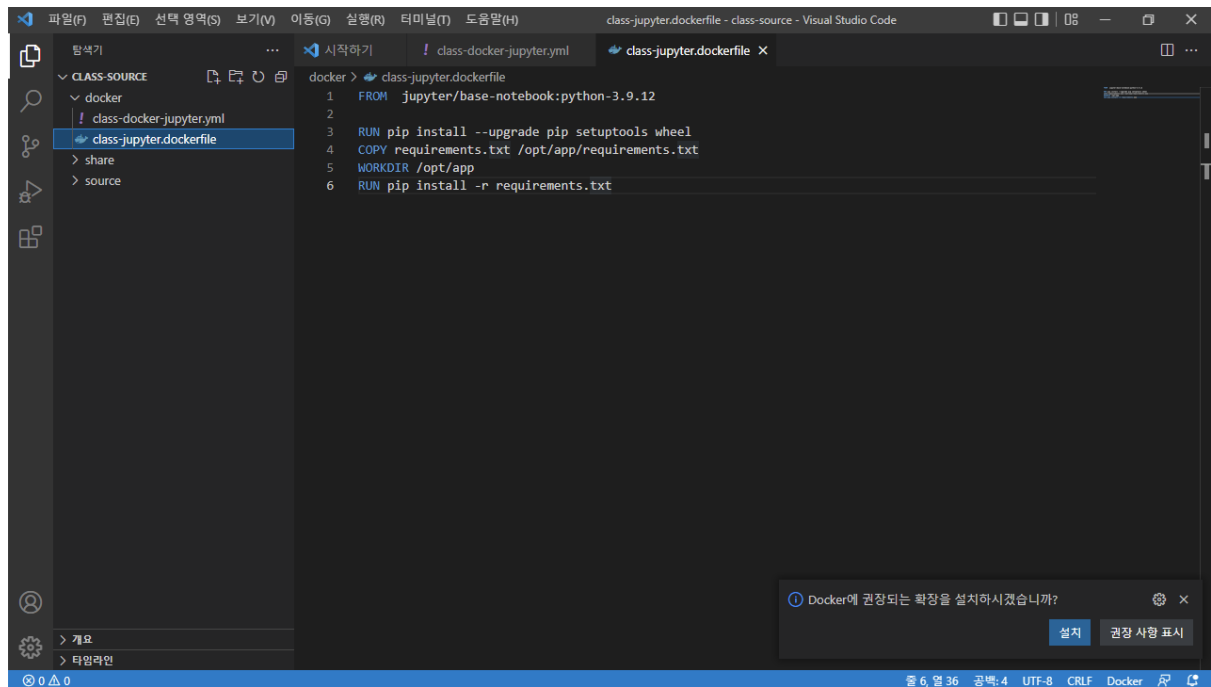
- file name : class-jupyter.dockerfile

File>New Text File

```
FROM jupyter/base-notebook:python-3.9.12

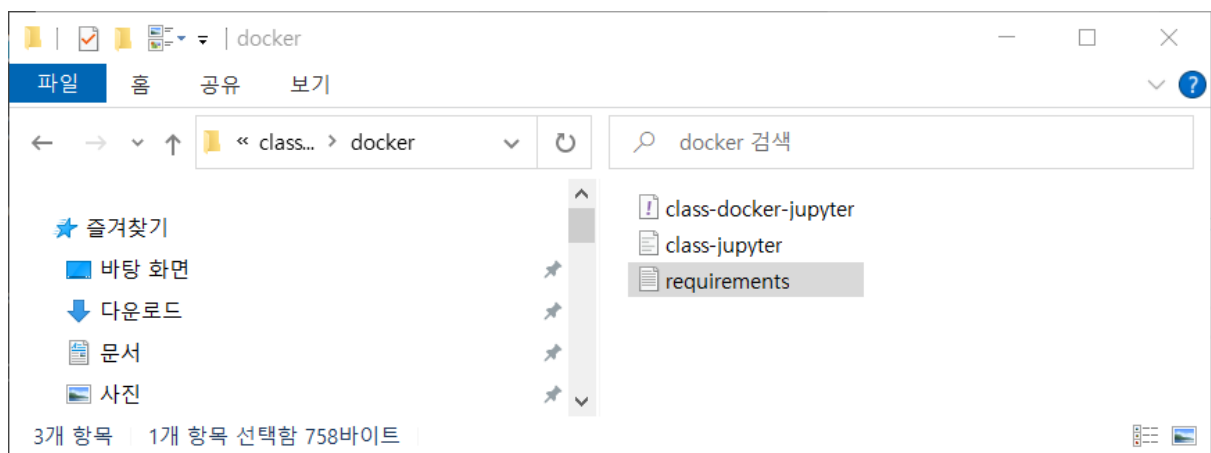
RUN pip install --upgrade pip setuptools wheel
COPY requirements.txt /opt/app/requirements.txt
WORKDIR /opt/app
RUN pip install -r requirements.txt
WORKDIR /home/jovyan/work
```

저장하기 File>Save As> class-jupyter.dockerfile



## 1.2.5. python library 목록 다운로드

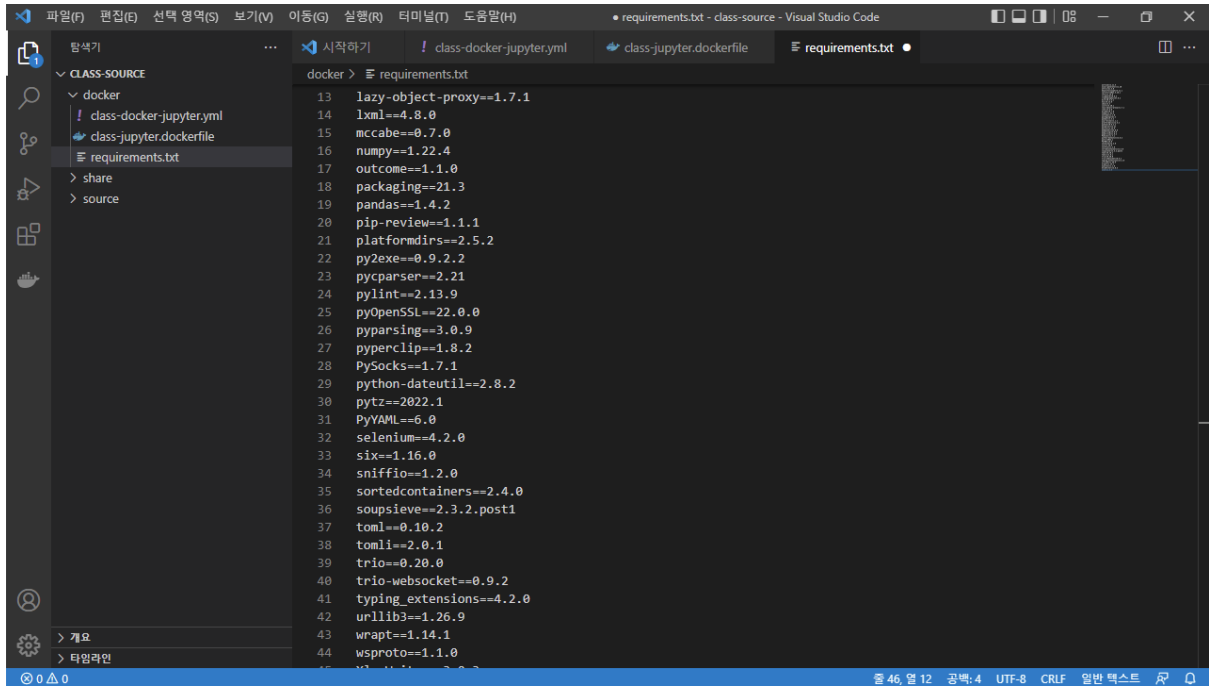
- requirements.txt 파일을 docker 폴더 밑에 다운로드



- 아래의 파일은 전체 실습과정에 필요한 라이브러리 목록으로 변경 필요

`pip freeze > requirements.txt` 추후에 사용예정 인스톨없이 사용가능함.

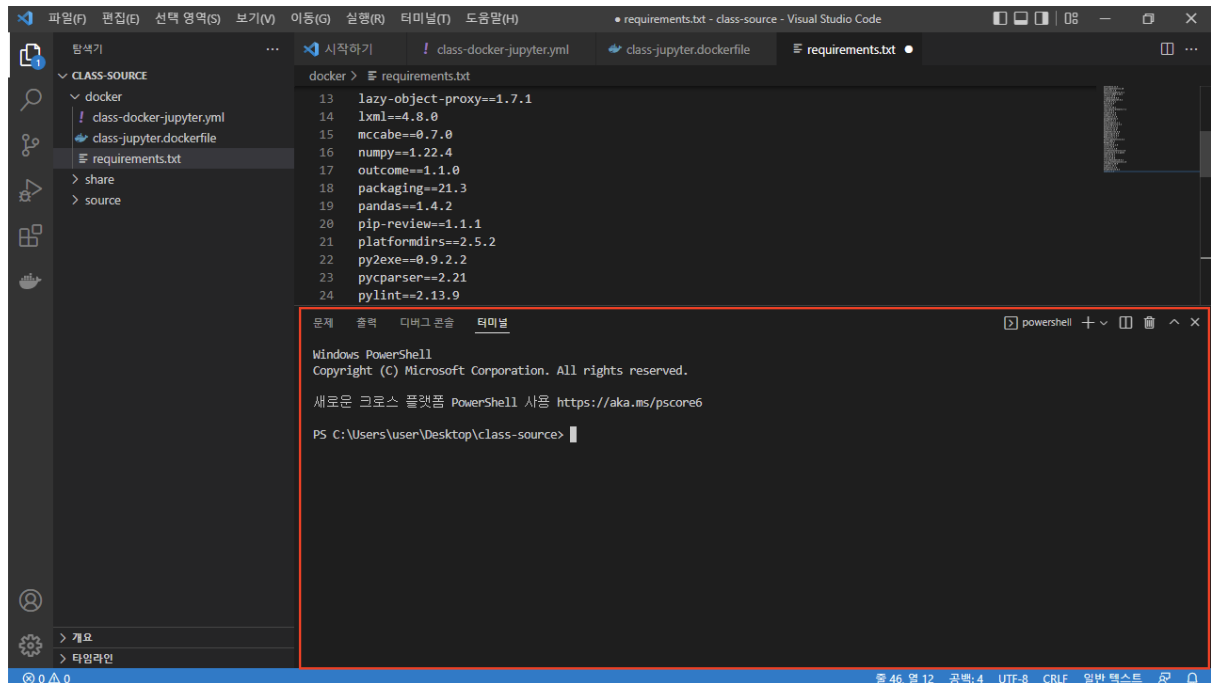
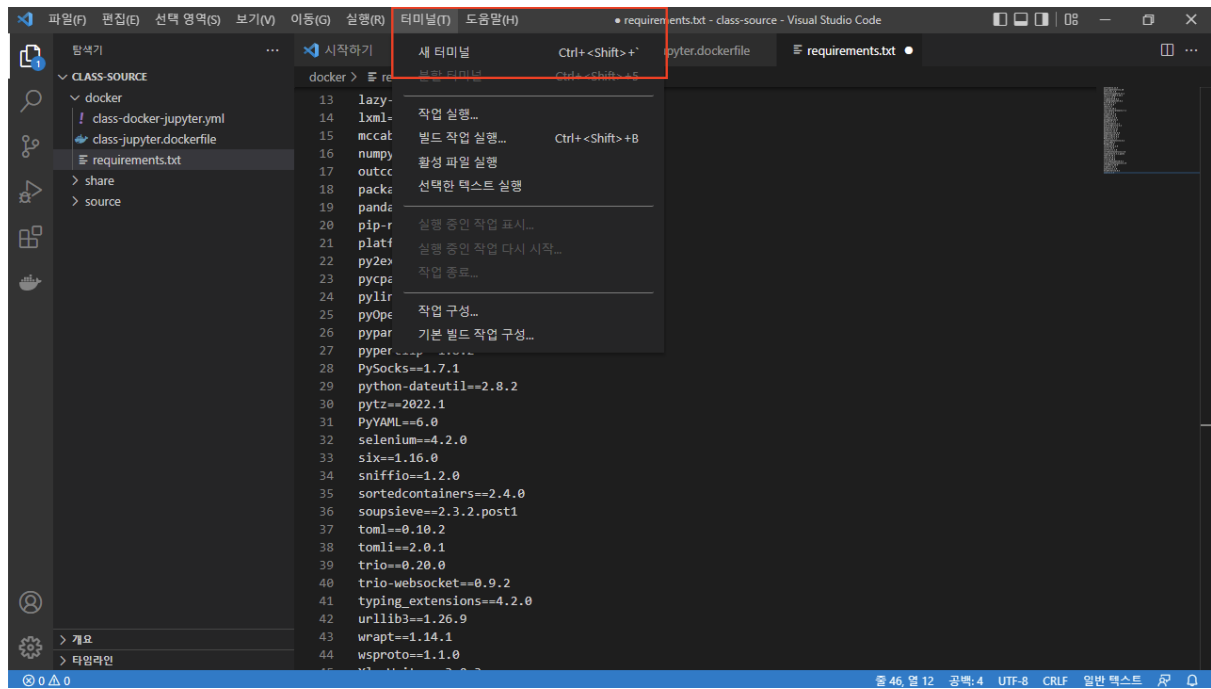
<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/6d51fb0c-dfe3-4fff-976c-68281bcd2881/requirements.txt>



```
13 lazy-object-proxy==1.7.1
14 lxml==4.8.0
15 mccabe==0.7.0
16 numpy==1.22.4
17 outcome==1.1.0
18 packaging==21.3
19 pandas==1.4.2
20 pip-review==1.1.1
21 platformdirs==2.5.2
22 py2exe==0.9.2.2
23 pycparser==2.21
24 pylint==2.13.9
25 pyOpenSSL==22.0.0
26 pyparsing==3.0.9
27 pyperclip==1.8.2
28 PySocks==1.7.1
29 python-dateutil==2.8.2
30 pytz==2022.1
31 PyYAML==6.0
32 selenium==4.2.0
33 six==1.16.0
34 sniffio==1.2.0
35 sortedcontainers==2.4.0
36 soupsieve==2.3.2.post1
37 toml==0.10.2
38 toml==2.0.1
39 trio==0.20.0
40 trio-websocket==0.9.2
41 typing_extensions==4.2.0
42 urllib3==1.26.9
43 wrapt==1.14.1
44 wsproto==1.1.0
```

## 1.2.6. Docker CLI 실행 위한 터미널 열기

- visual source code 에서 터미널 오픈

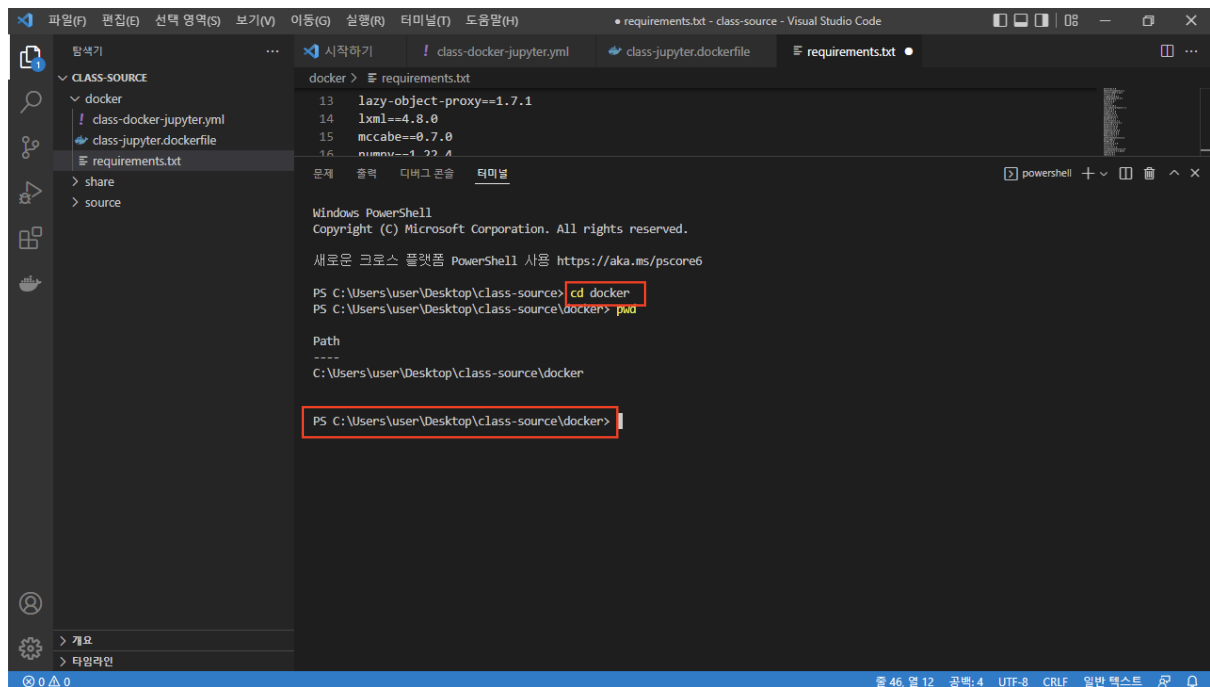


## 1.2.6. Jupyter Image Building

- docker 폴더로 이동

cd docker

pwd



```
requirements.txt
13 lazy-object-proxy==1.7.1
14 lxml==4.8.0
15 mccabe==0.7.0
16 numpy==1.22.4

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/pscore6

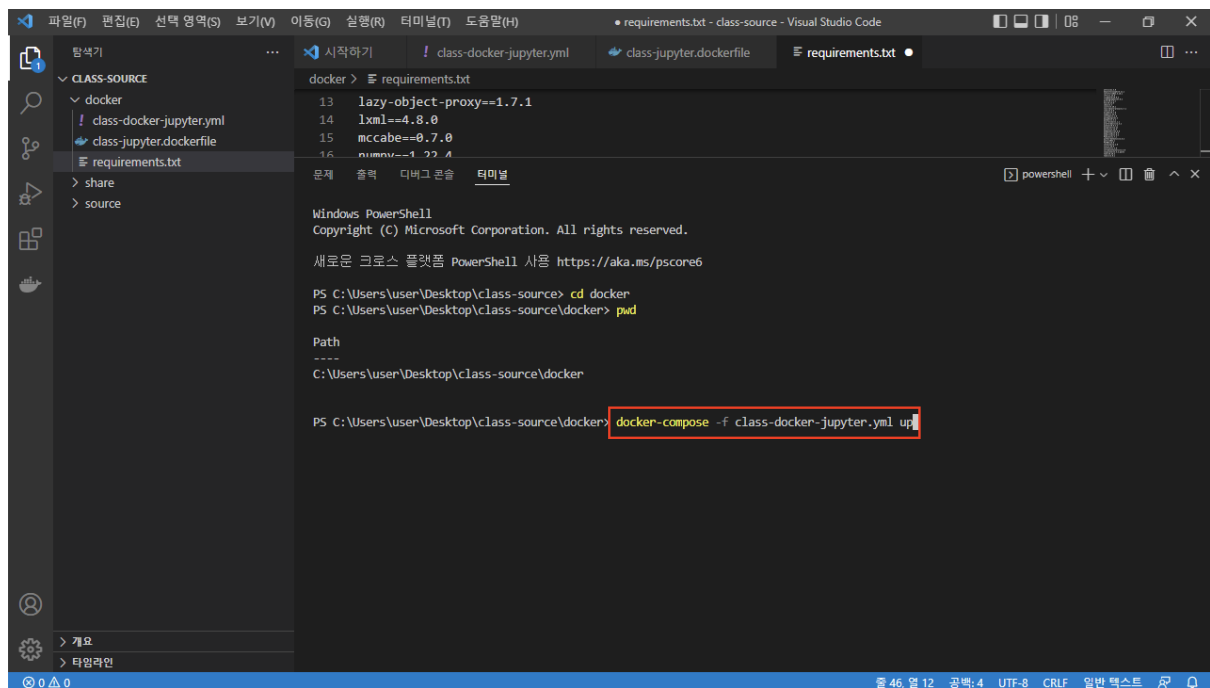
PS C:\Users\user\Desktop\class-source> cd docker
PS C:\Users\user\Desktop\class-source\docker> pwd

Path
----
C:\Users\user\Desktop\class-source\docker

PS C:\Users\user\Desktop\class-source\docker>
```

- 아래의 명령어 입력

```
docker-compose -f class-docker-jupyter.yml up
```



```
requirements.txt
13 lazy-object-proxy==1.7.1
14 lxml==4.8.0
15 mccabe==0.7.0
16 numpy==1.22.4

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/pscore6

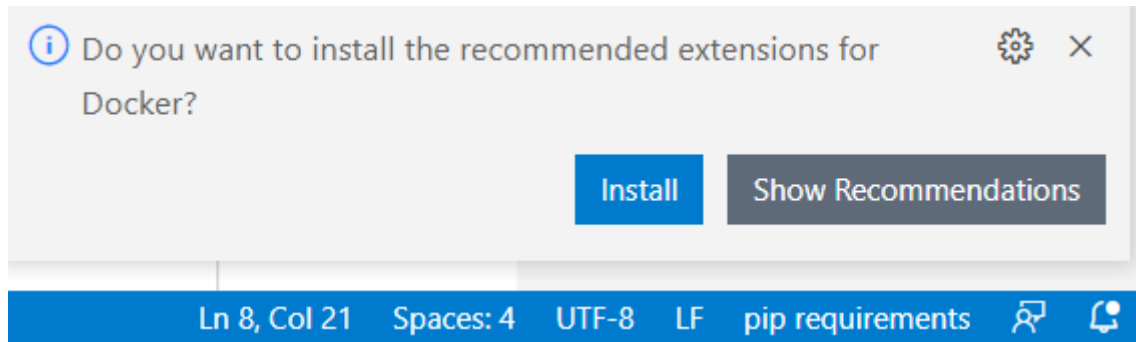
PS C:\Users\user\Desktop\class-source> cd docker
PS C:\Users\user\Desktop\class-source\docker> pwd

Path
----
C:\Users\user\Desktop\class-source\docker

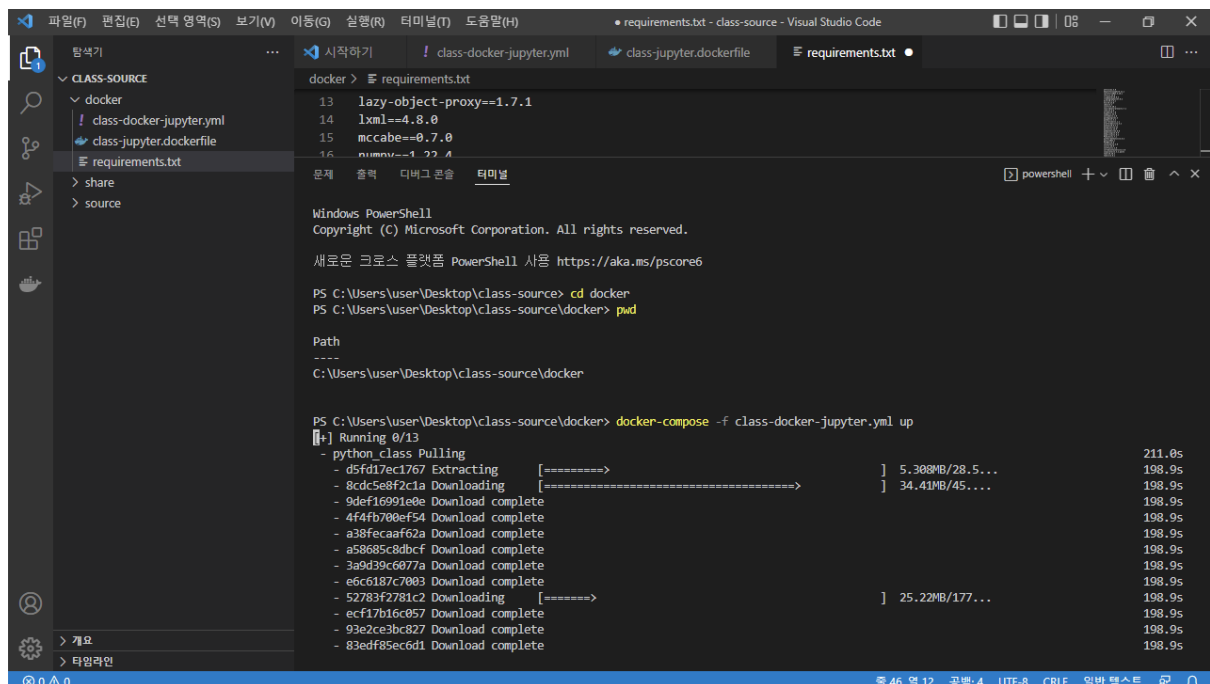
PS C:\Users\user\Desktop\class-source\docker> docker-compose -f class-docker-jupyter.yml up
```

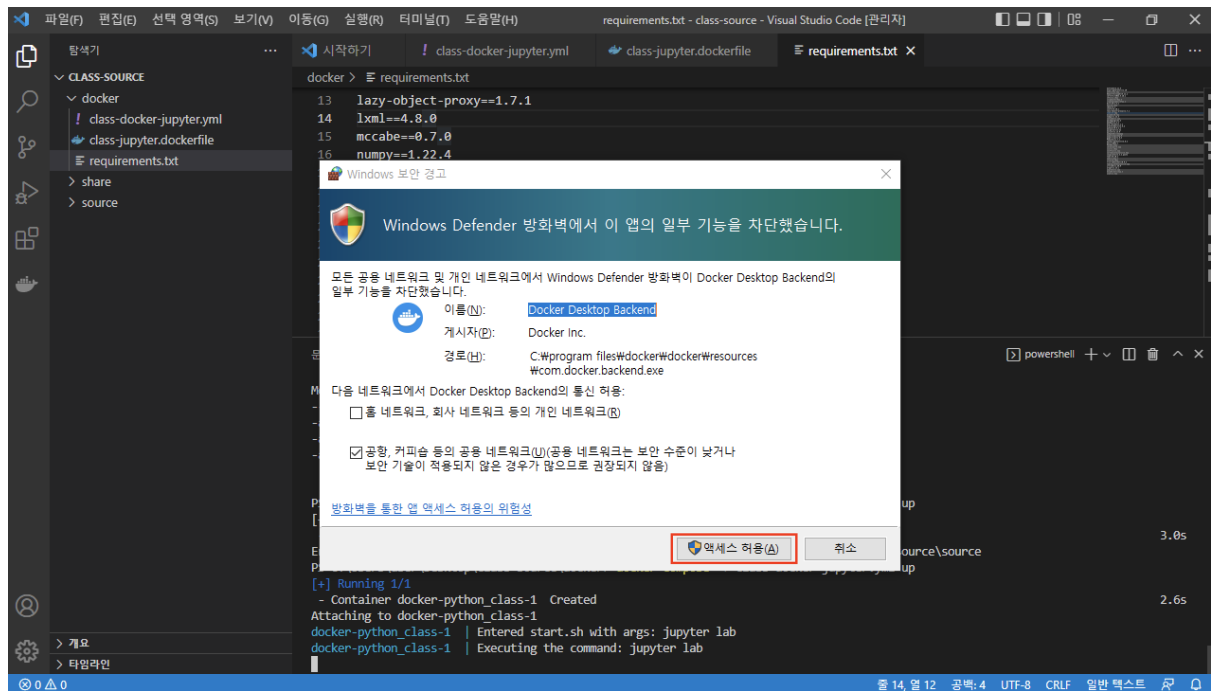


아래와 같은 설명이 나오면 Install 하시오.

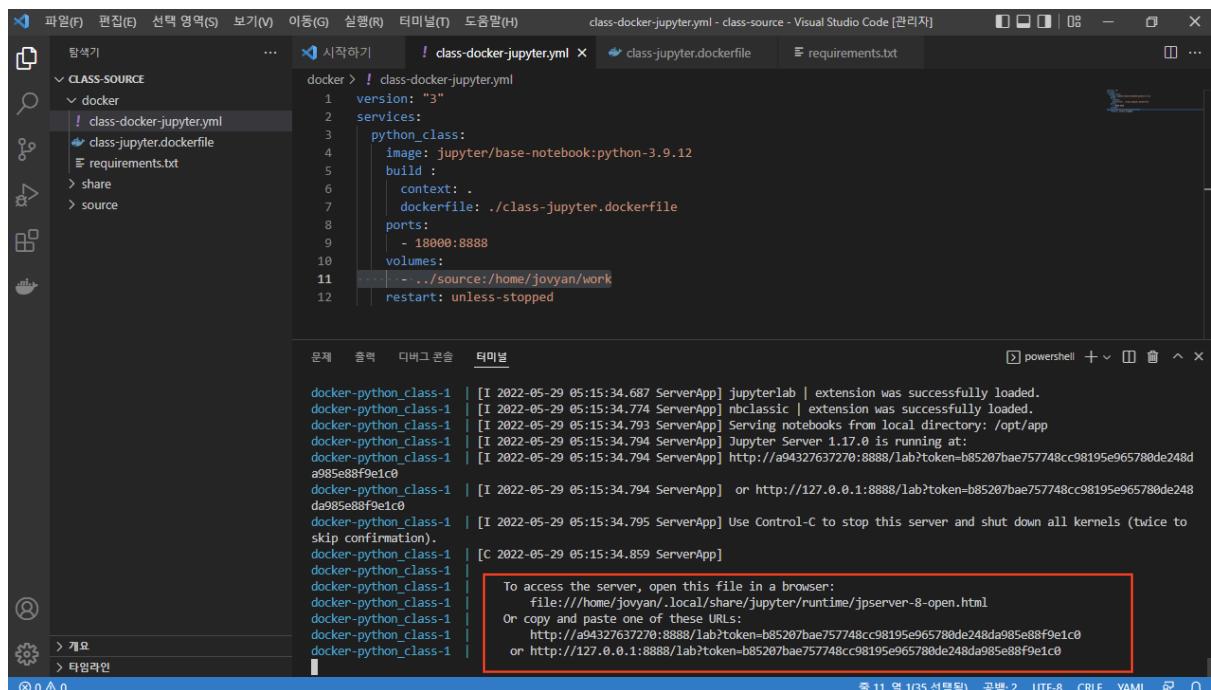


- docker 이미지 다운로드 및 파이선 라이브러리 자동 설치 (5 ~ 10분 소요 - 네트워크 속도에 따라 다름)





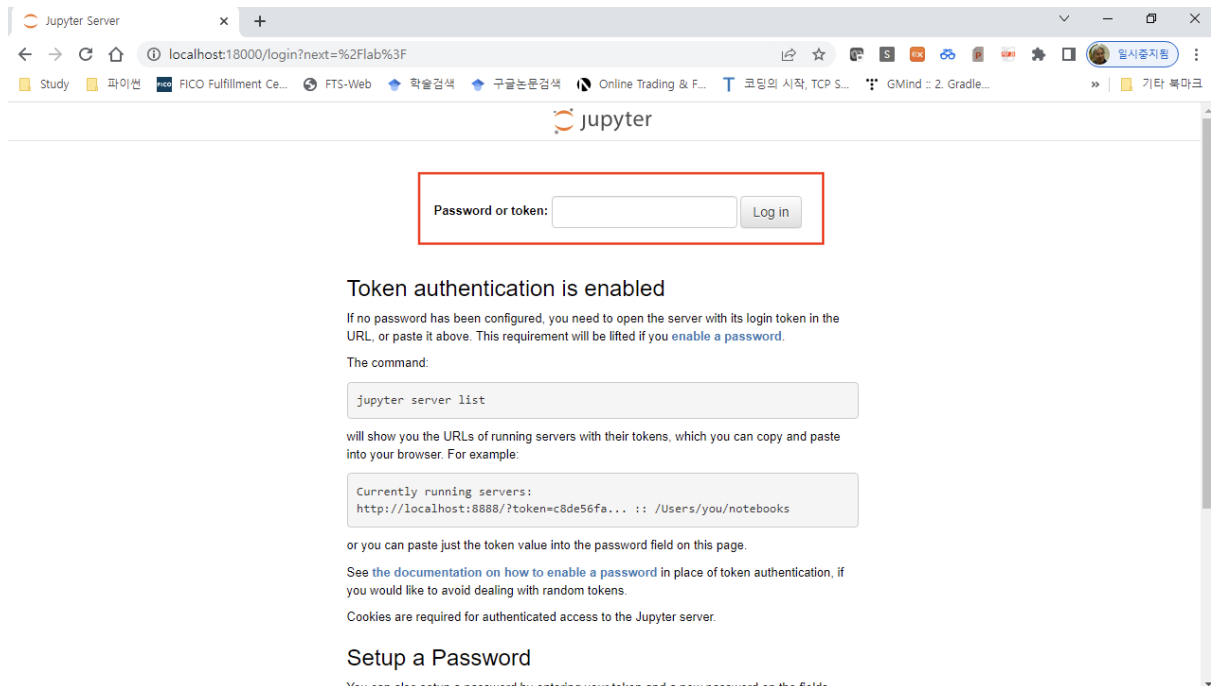
- 서비스 실행 완료시 아래와 같이 접속 정보가 표시됨



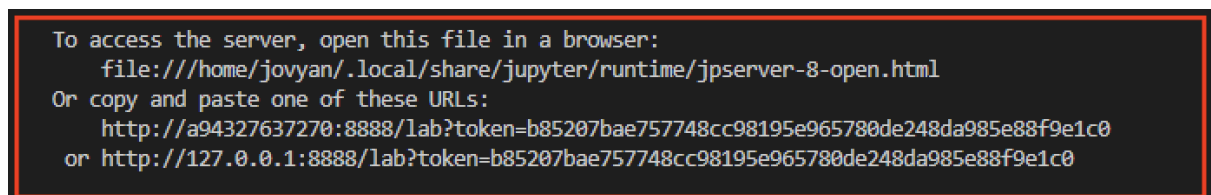
Tip: Ctrl + C

## 1.2.7. Jupyter Lab 접속

- 설치된 Jupyter Lab의 포트는 18000으로 아래와 같이 접속한다.
- <http://127.0.0.1:18000/>



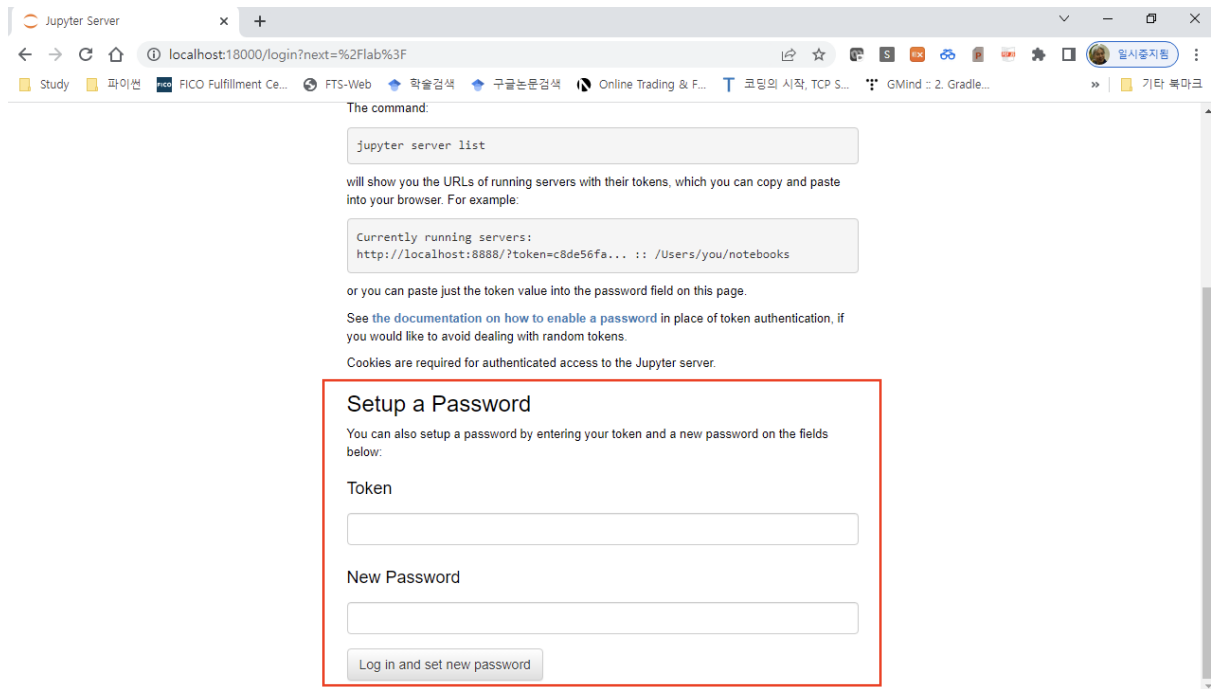
- 아래의 정보에서 token 값을 복사하여 붙여넣기하여 접속하세요.



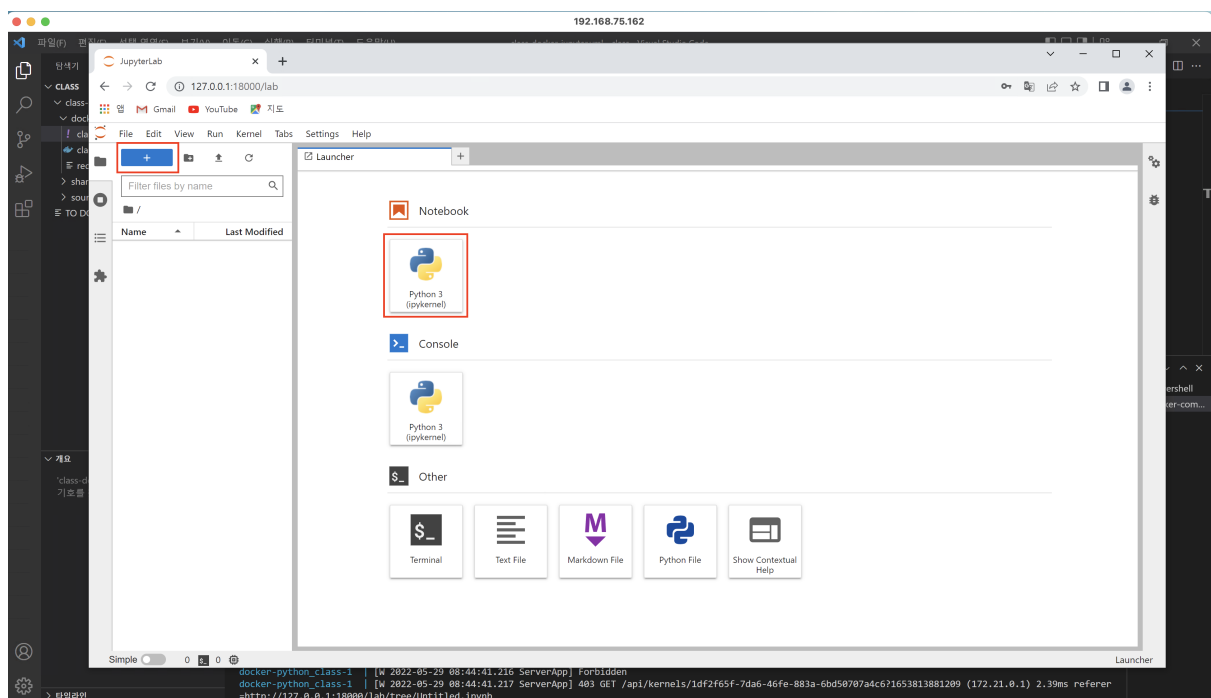
- token값을 붙여 넣고 새로운 패스워드 설정후 접속

```
docker-python_class-1 | To access the server, open this file in a browser:
docker-python_class-1 | file:///home/jovyan/.local/share/jupyter/runtime/jpserver-7-open.html
docker-python_class-1 | Or copy and paste one of these URLs:
docker-python_class-1 | http://c1175c0c964f:8888/lab?token=d7788a4aac5317451d82d6aae908b6d1ad5ec41cf57291e1
docker-python_class-1 | or http://127.0.0.1:8888/lab?token=d7788a4aac5317451d82d6aae908b6d1ad5ec41cf57291e1
```

아래는 생략가능합니다.

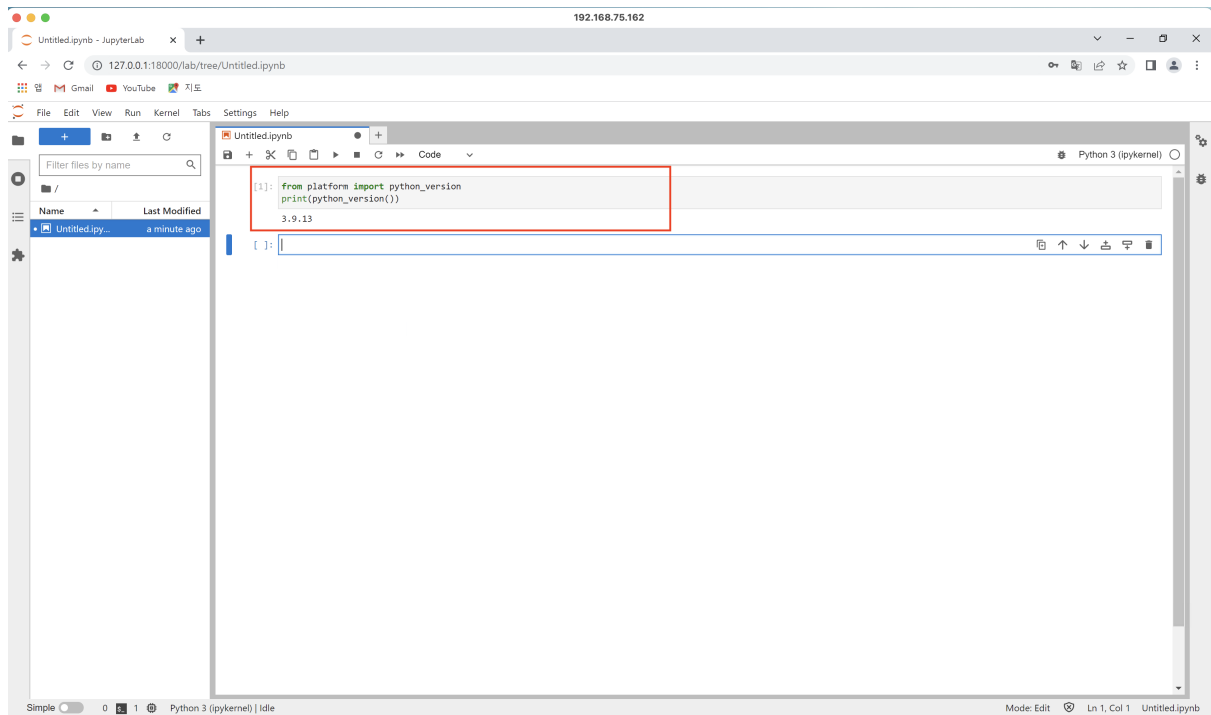


## 1.2.8. Python 버전 확인 하기



```
from platform import python_version
print(python_version())
```

명령어 입력후 실행버튼(삼각형)을 클릭합니다



```
pip list
```

## 1.2.9. Docker 실습

```
pip install slack
```

## 1.2.10. Docker 명령어

- class-source\docker 폴더 이동
- Docker 서비스 중지

```
docker-compose -f class-docker-jupyter.yml stop
```

- Docker 서비스 시작

```
docker-compose -f class-docker-jupyter.yml start
```

- Docker log 보기

```
docker-compose -f class-docker-jupyter.yml logs
```

- Docker 이미지 재 생성, requirements.txt 파일 변경이 필요할 때

```
docker images  
  
docker rmi <CONTAINER ID>  
  
docker-compose -f class-docker-jupyter.yml up
```

- Docker 이미지 재 생성, requirements.txt 파일 변경이 필요할 때

```
docker exec -it docker-python_class-1 /bin/bash  
  
or  
  
docker exec -it <CONTAINER ID> /bin/bash
```

---

## 오류 정보



RuntimeError: Permissions assignment failed for secure file:  
'/home/jovyan/.local/share/jupyter/runtime/jpserver-8.json'. Got '0o655'  
instead of '0o0600'.

volumes 에서 ../share:/home 삭제함

## jupyter password 입력 해제 할 경우



```
jupyter notebook --ip='*' --NotebookApp.token="" --  
NotebookApp.password=""
```