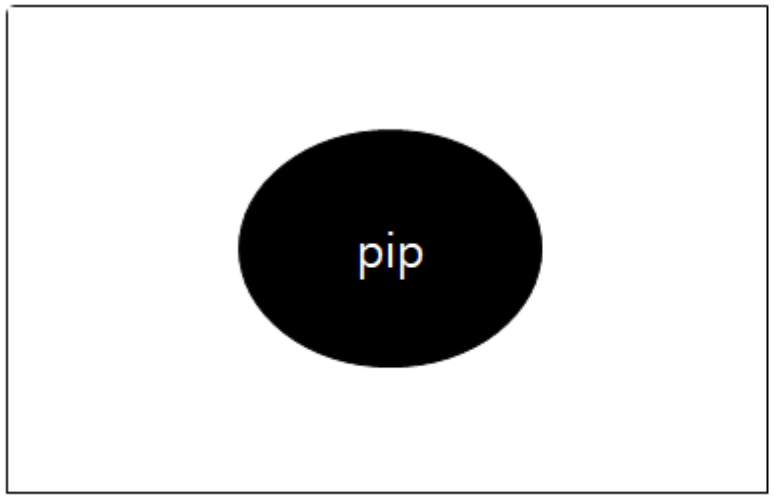
강원대학교
AI 소프트웨어학과

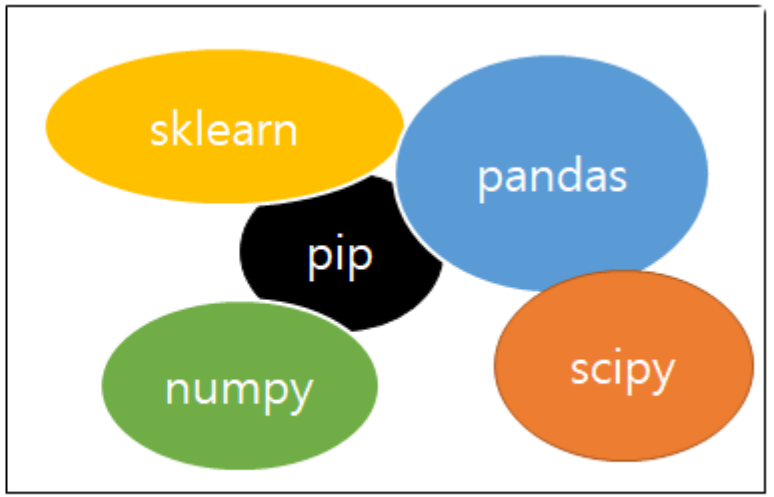
인공지능
- 파이썬 설치 및 기초 -

KNU 강원대학교

- Python : pip라는 툴을 가지고 있고 모두 수동으로 패키지를 설치해 사용해야함
- Anaconda : 아나콘다는 여러가지 수학 및 과학 패키지(Package)들을 기본적으로 포함하고 있는 파이썬 배포판때문에 머신러닝(Machine learning)이나 데이터 분석(Data analysis)을 할 때, 다양한 패키지가 필요하므로 이를 사용

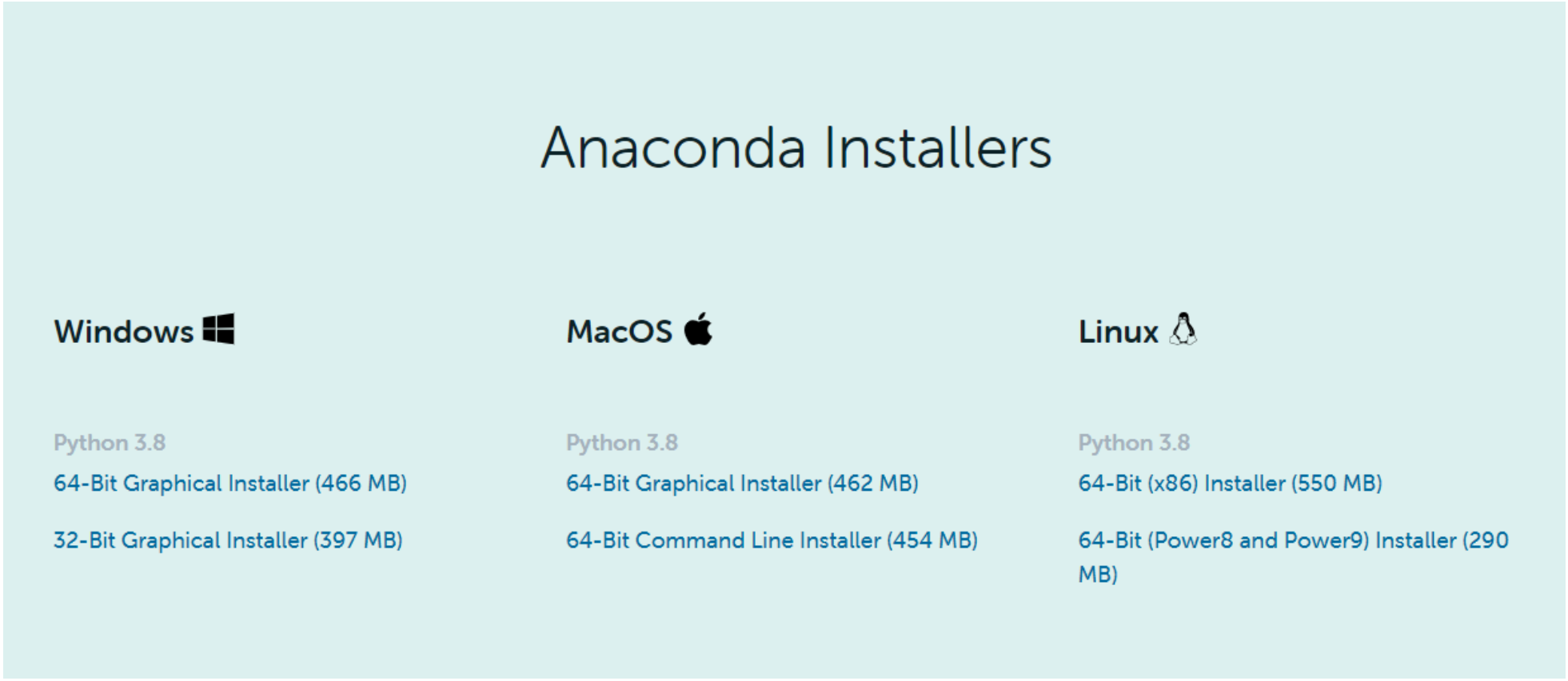


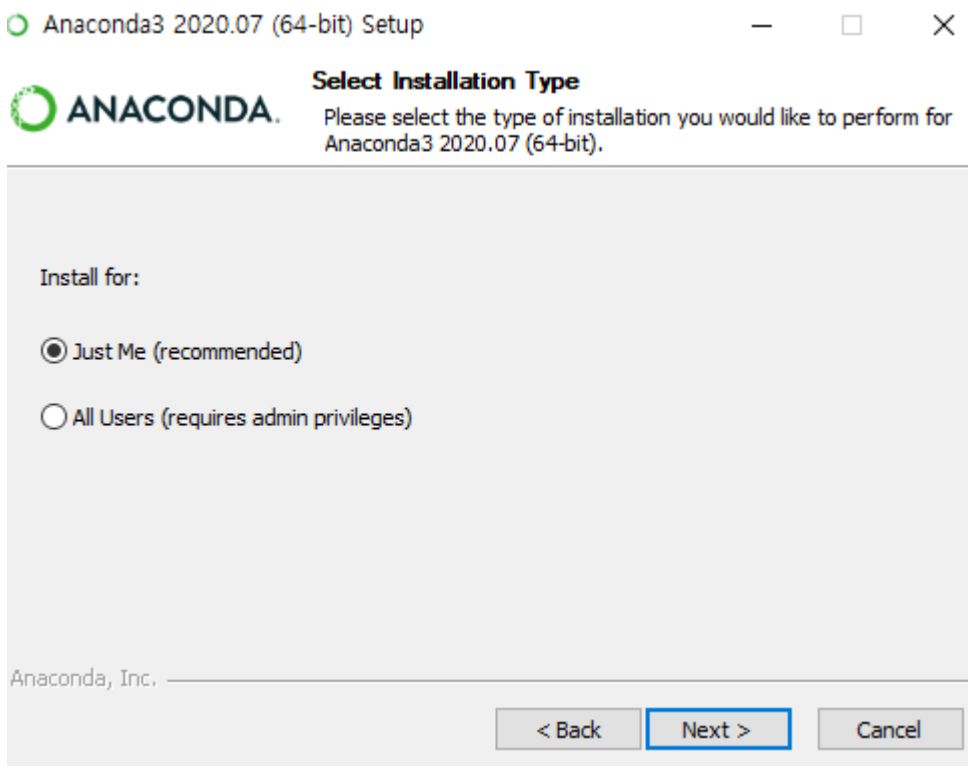
Python

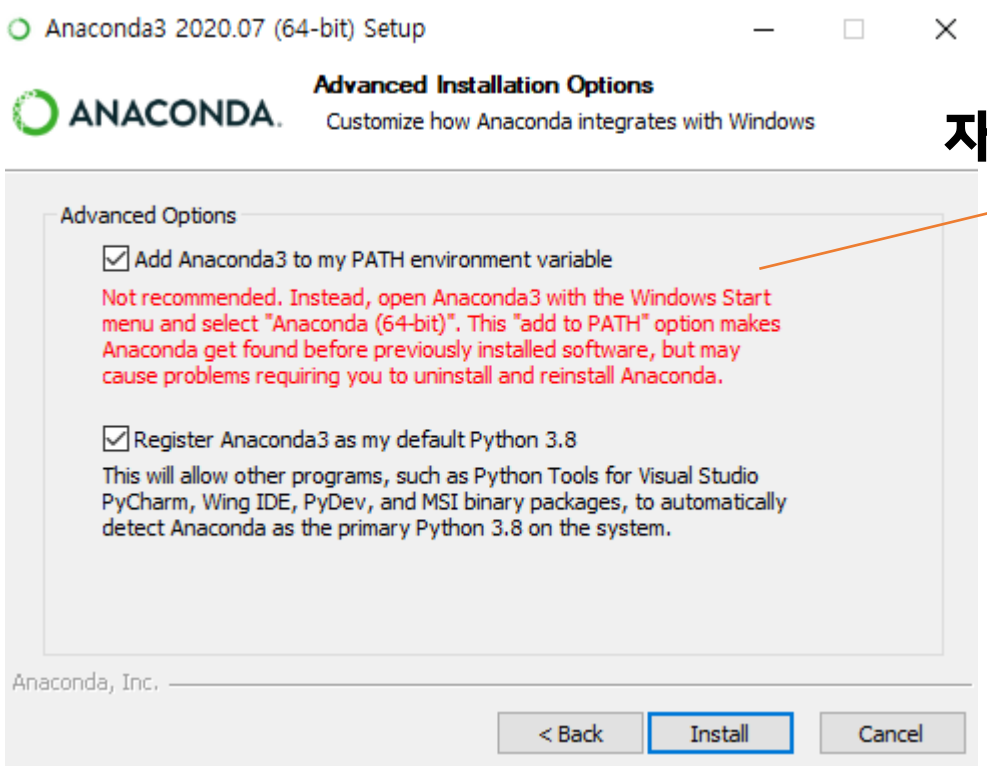


Anaconda

- https://www.anaconda.com/products/individual

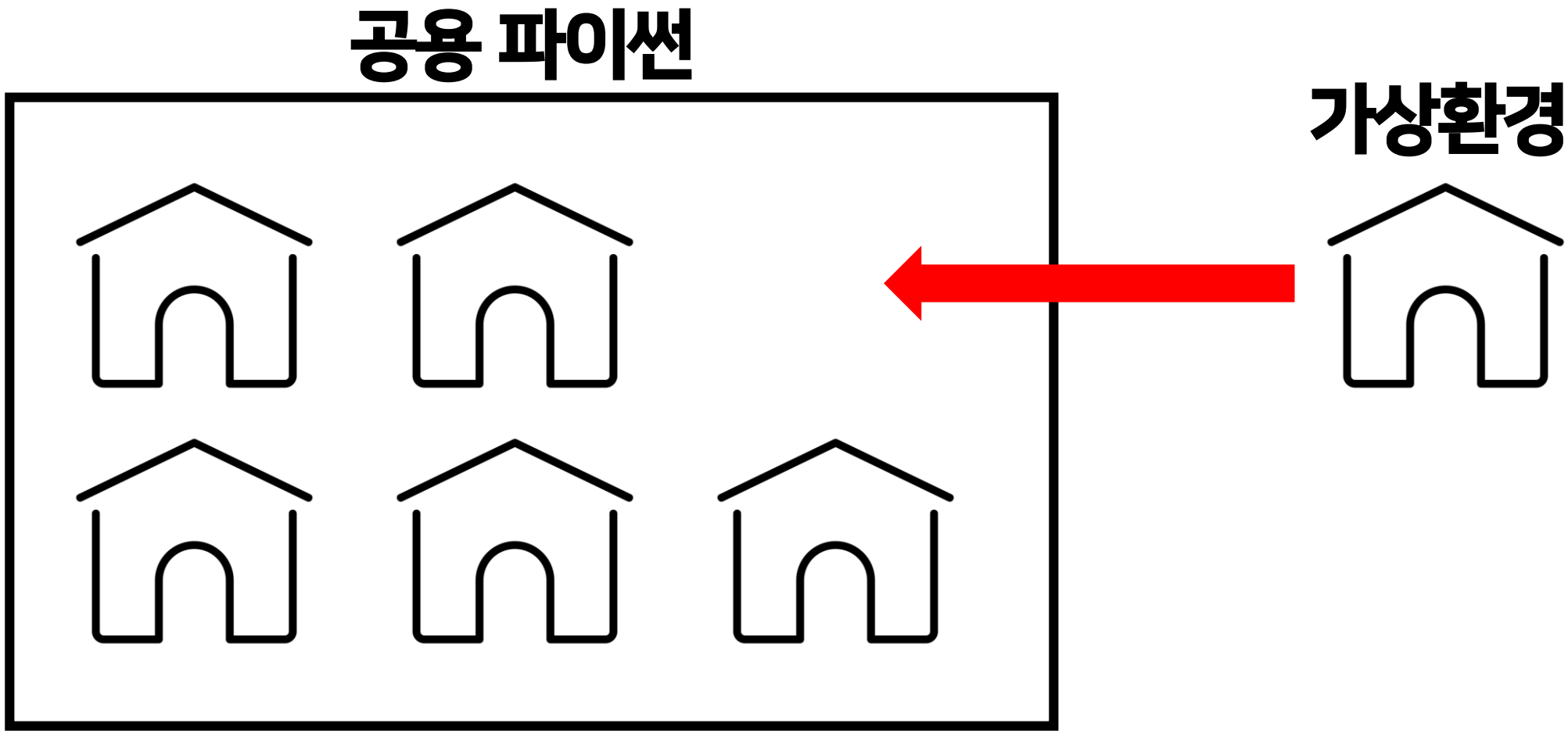



Anaconda3 2020.07 (64-bit) Setup
ANACONDA.
Select Installation Type
Please select the type of installation you would like to perform for Anaconda3 2020.07 (64-bit).
Install for:
Just Me (recommended)
All Users (requires admin privileges)
Anaconda, Inc.
< Back
Next >
Cancel

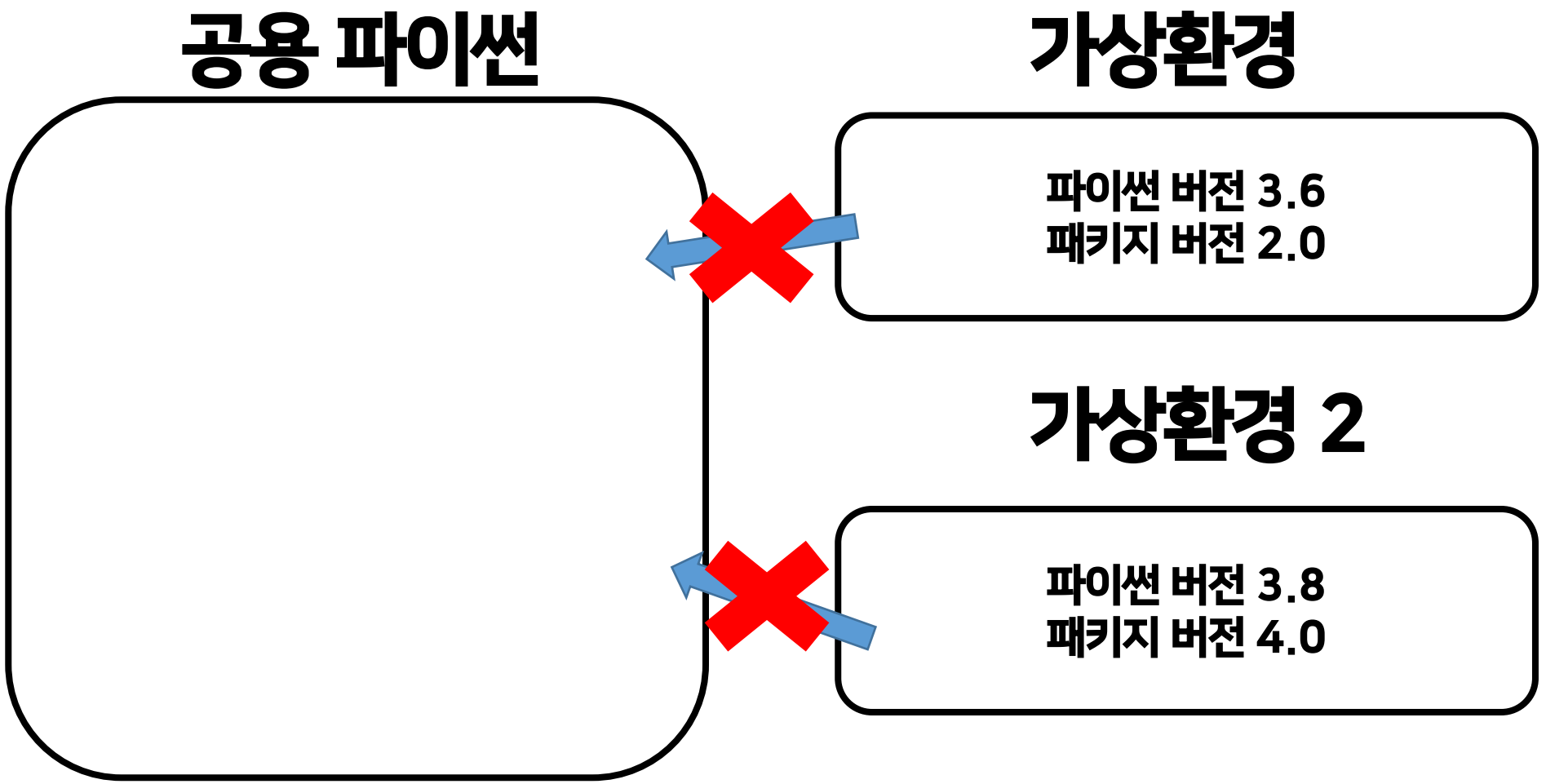

Anaconda3 2020.07 (64-bit) Setup
ANACONDA.
Advanced Installation Options
Customize how Anaconda integrates with Windows
Advanced Options
Add Anaconda3 to my PATH environment variable
Not recommended. Instead, open Anaconda3 with the Windows Start menu and select "Anaconda (64-bit)". This "add to PATH" option makes Anaconda get found before previously installed software, but may cause problems requiring you to uninstall and reinstall Anaconda.
Register Anaconda3 as my default Python 3.8
This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 3.8 on the system.
Anaconda, Inc.
< Back
Install
Cancel

자동 path 설정

- 가상환경 : 프로젝트별로 패키지를 따로 관리하는 공간
- 패키지 별로 호환의 문제가 발생하기 때문에



- 가상환경 : 프로젝트별로 패키지를 따로 관리하는 공간
- 패키지 별로 호환의 문제가 발생하기 때문에



- **가상환경 : 프로젝트별로 패키지를 따로 관리하는 공간**
- **패키지 별로 호환의 문제가 발생하기 때문에**

Overview

neuralnetworksanddeeplearning.com integrated scripts for Python 3.5.2 and Theano with CUDA support

These scrips are updated ones from the **neuralnetworksanddeeplearning.com** gitHub repository in order to work with Python 3.5.2

The testing file (**test.py**) contains all three networks (network.py, network2.py, network3.py) from the book and it is the starting point to run (i.e. *train and evaluate*) them.

Brief Summary

Last updated: 6/22/2019 with TensorFlow v1.13.1

This repository is a tutorial for how to use TensorFlow's
multiple objects on Windows 10, 8, or 7. (It will also wo
originally written using TensorFlow version 1.5, but will

Environment

I have tested on Ubuntu 16.04/18.04. The code may work on other systems.

[Ubuntu-Deep-Learning-Environment-Setup]

- **Ubuntu 16.04 / 18.04**
- **ROS Kinetic / Melodic**
- **GTX 1080Ti / RTX 2080Ti**
- **python 2.7 / 3.6**

- Win키+R :CMD창 열기 or win창에서 cmd검색 후 명령 프롬프트 클릭

- Anaconda 안에 있는 파이썬 버전 확인

→ conda search python

```
C:₩Users₩cg>conda search python
Loading channels: done
# Name                       Version           Build  Channel
python                        2.7.13     h1b6d89f_16  pkgs/main
python                        2.7.13     h9912b81_15  pkgs/main
python                        2.7.13     hb034564_12  pkgs/main
python                        2.7.14     h2765ee6_18  pkgs/main
python                        2.7.14     h3e68818_15  pkgs/main
python                        2.7.14     h4084c39_22  pkgs/main
python                        2.7.14     h4a10d90_30  pkgs/main
python                        2.7.14     h4a10d90_31  pkgs/main
python                        2.7.14     h59f5a59_20  pkgs/main
python                        2.7.14     h819644d_16  pkgs/main
python                        2.7.14     h8c3f1cb_23  pkgs/main
python                        2.7.15      h2880e7c_2  pkgs/main
python                        2.7.15      h2880e7c_3  pkgs/main
python                        2.7.15      h2880e7c_4  pkgs/main
python                        2.7.15     hcb6e200_15  pkgs/main
python                        2.7.15      hcb6e200_5  pkgs/main
python                        2.7.15      hcb6e200_7  pkgs/main
python                        2.7.15      he216670_0  pkgs/main
python                        2.7.16      hcb6e200_0  pkgs/main
```

- Anaconda에서 가상환경 생성
- conda create -n 생성하고 싶은 이름 python=설치 버전

```
C:₩Users₩cg>conda create -n chang python=3.6
WARNING: A conda environment already exists at 'C:₩Users₩cg₩anaconda3₩envs₩chang'
Remove existing environment (y/[n])? y
```

- Anaconda에서 가상환경 확인
- Conda env list

```
C:₩Users₩cg>conda env list
# conda environments:
#
base                  *  C:₩Users₩cg₩anaconda3
chang                    C:₩Users₩cg₩anaconda3₩envs₩chang
```

- Anaconda에서 가상환경 삭제
- conda env remove -n 삭제하고 싶은 이름

```
C:₩Users₩cg>conda env remove -n chang
```

- 현재 설치돼 있는 파이썬의 환경에서 설치된 package 종류 확인
- pip freeze

```
C:₩Users₩cg>pip freeze
alabaster==0.7.12
anaconda-client==1.7.2
anaconda-navigator==1.9.12
anaconda-project==0.8.3
argh==0.26.2
asn1crypto==1.3.0
astroid @ file:///C:/ci/astroid_1592487315634/work
astropy==4.0.1.post1
atomicwrites==1.4.0
attrs==19.3.0
autopep8 @ file:///tmp/build/80754af9/autopep8_1592412889138/work
Babel==2.8.0
backcall==0.2.0
backports.functools-lru-cache==1.6.1
backports.shutil-get-terminal-size==1.0.0
backports.tempfile==1.0
backports.weakref==1.0.post1
```

```
(chang) C:₩Users₩cg>pip freeze
certifi==2020.6.20
wincertstore==0.2
```

- 생성된 가상환경 들어가기→ conda activate 가상환경이름
- 생성된 가상환경 나가기→ deactivate

```
C:₩Users₩cg>conda activate chang

(chang) C:₩Users₩cg>
```

- 주피터 노트북과 커널을 만들수 있는 패키지를 설치

pip install jupyter notebook
pip install ipykernel

python -m ipykernel install --user --name=커널이름 --display-name 원하는이름

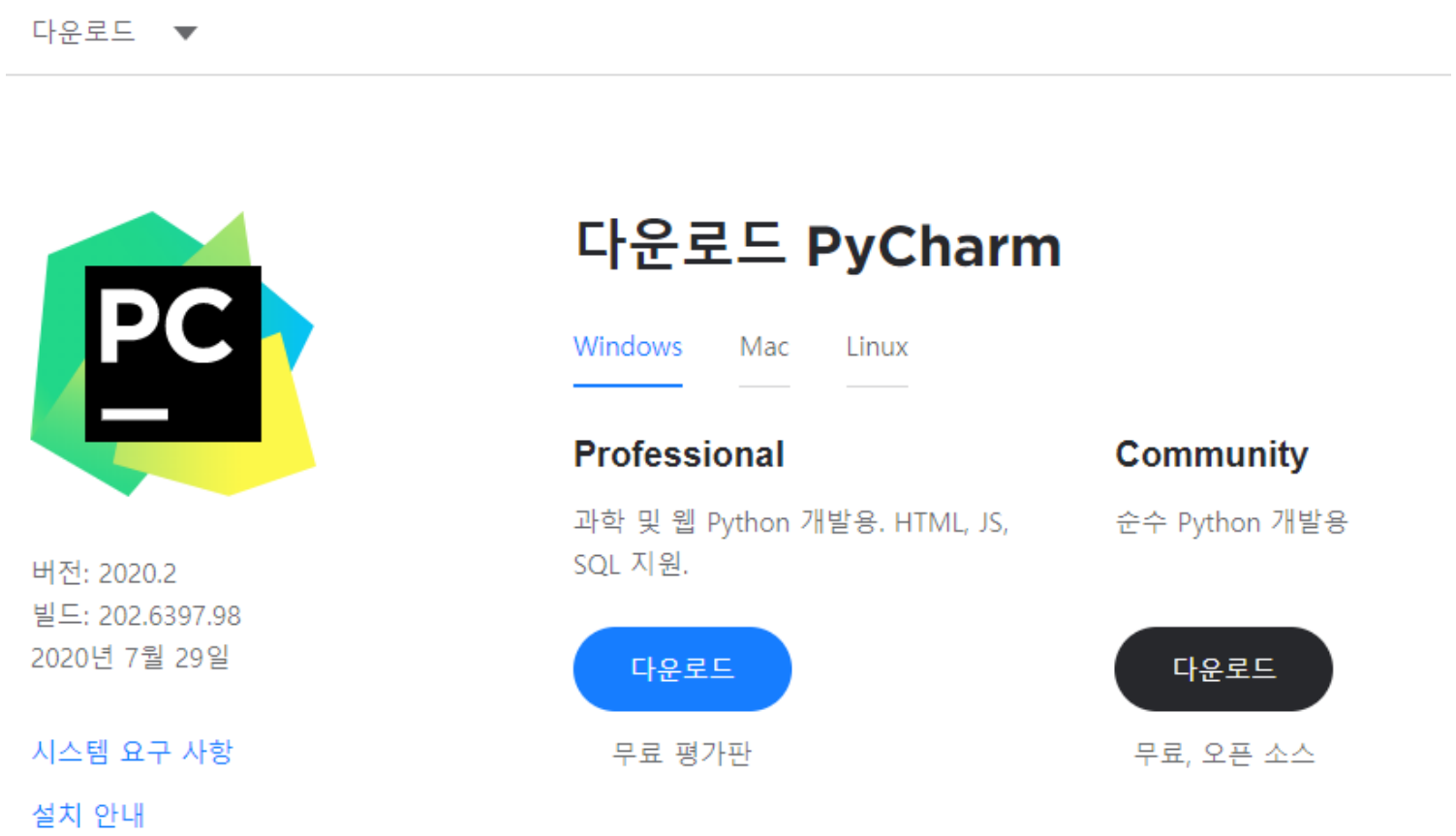
- 주피터에 생성된 커널 목록 확인하기

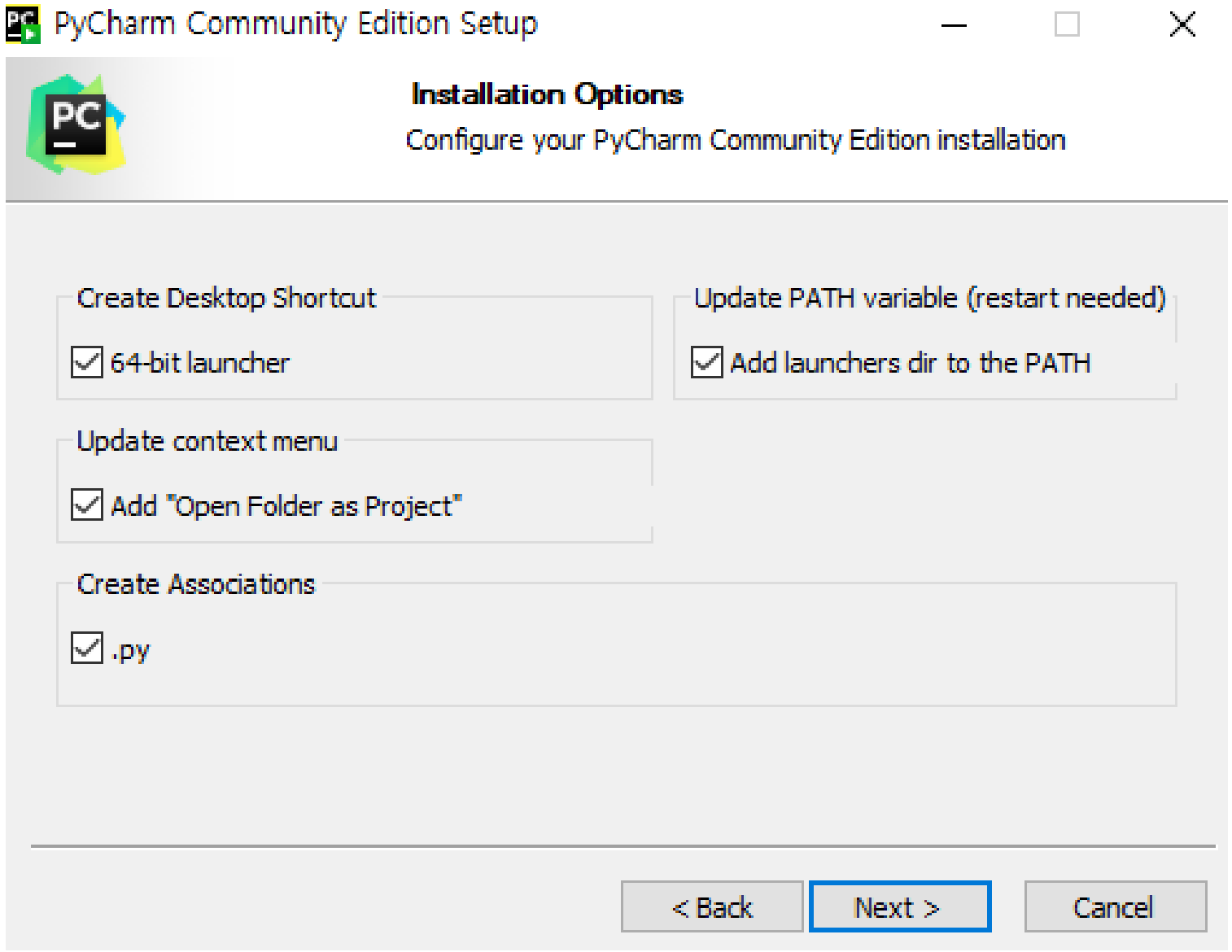
jupyter kernelspec list

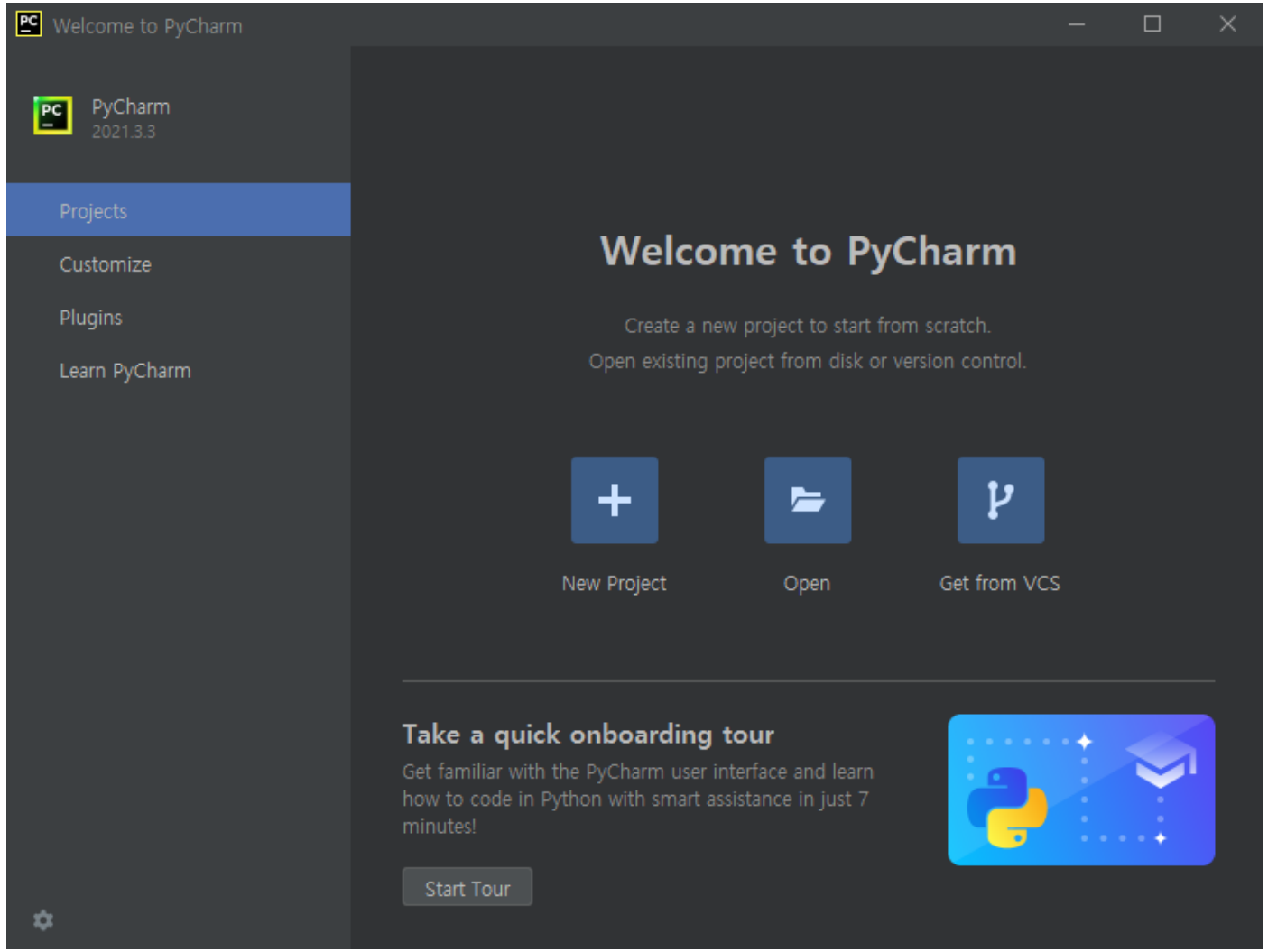
- 주피터에 생성된 커널 삭제하기

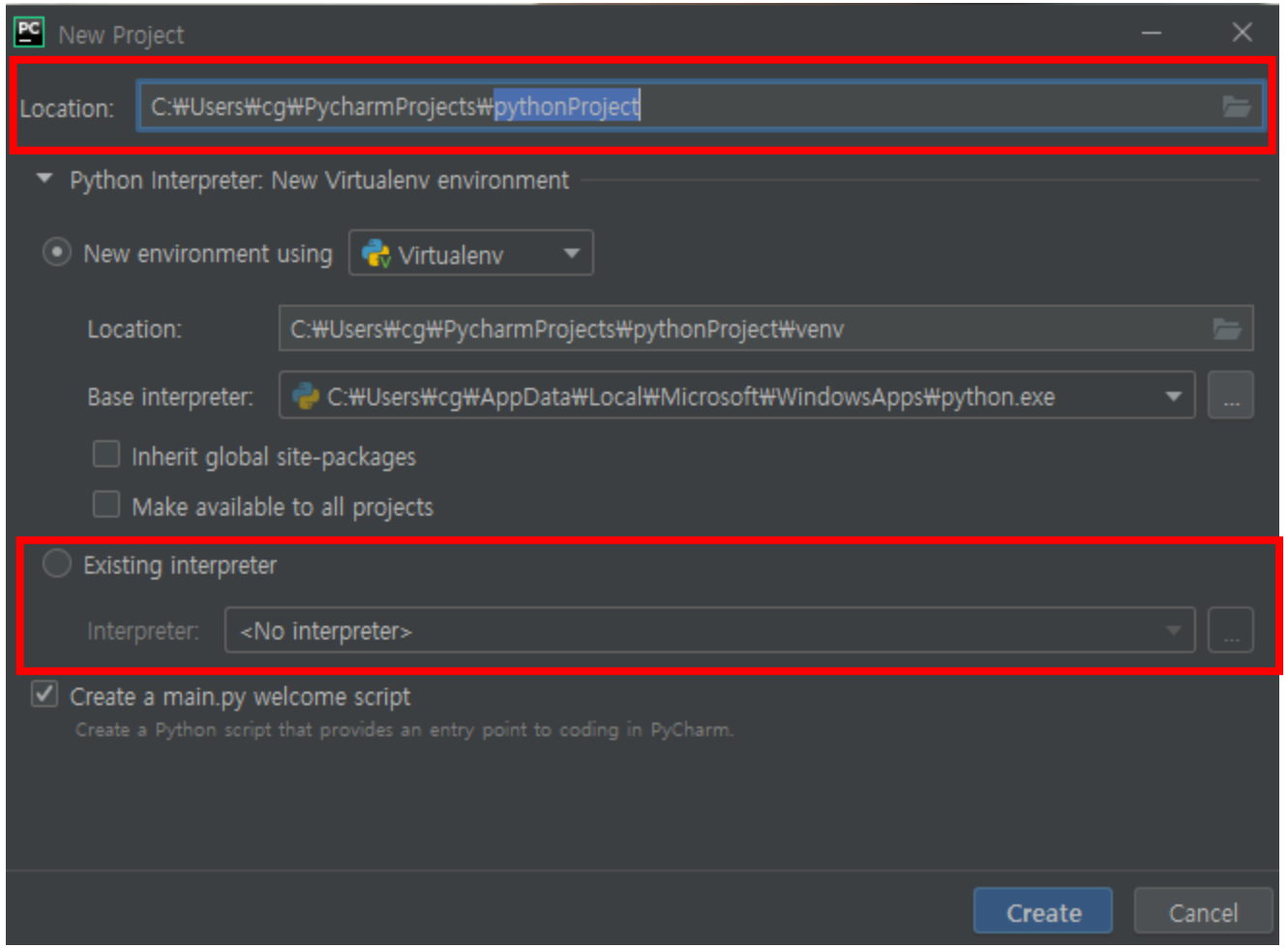
jupyter kernelspec uninstall 커널이름

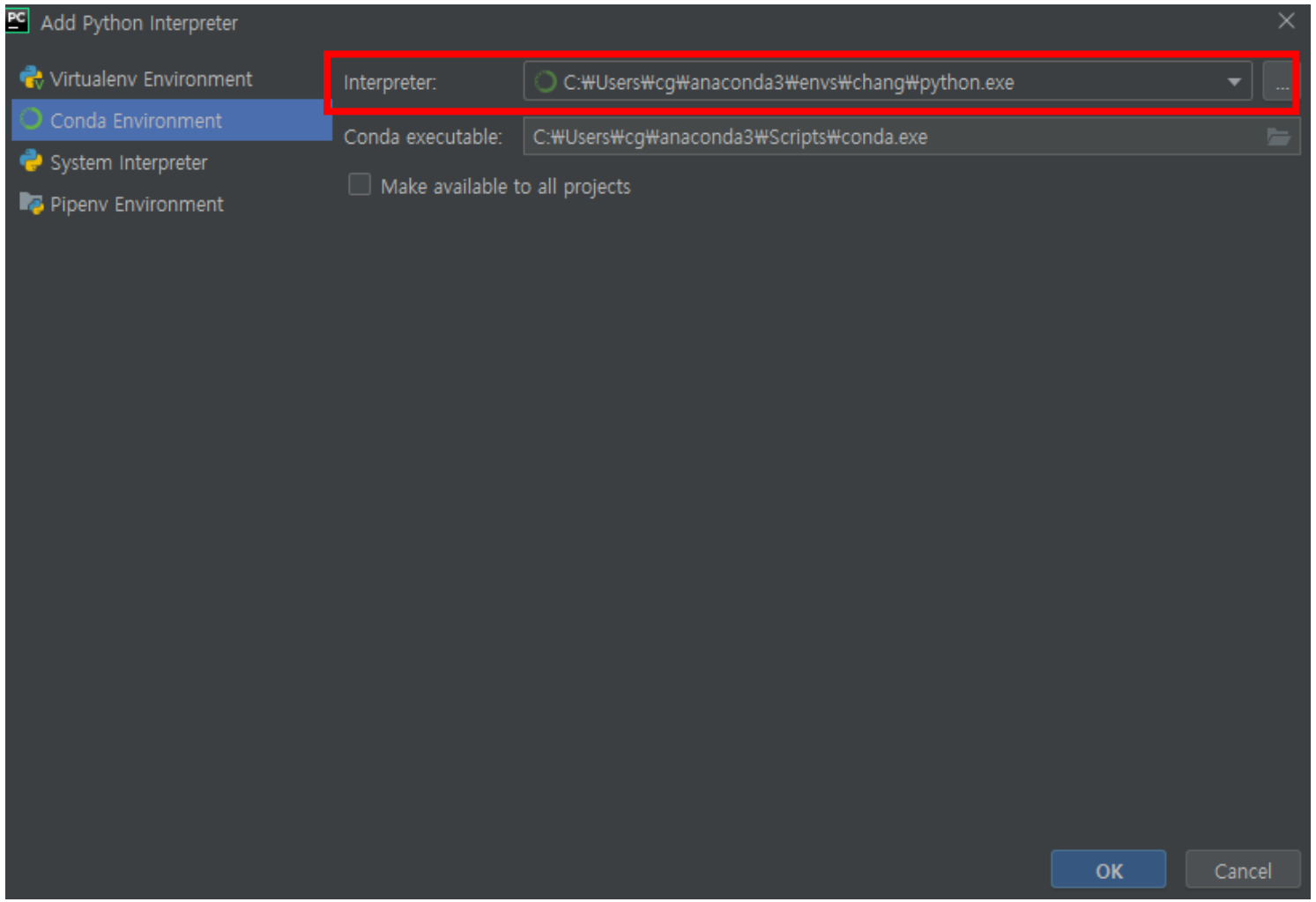
- **https://www.jetbrains.com/ko-kr/pycharm/download/#section=windows**



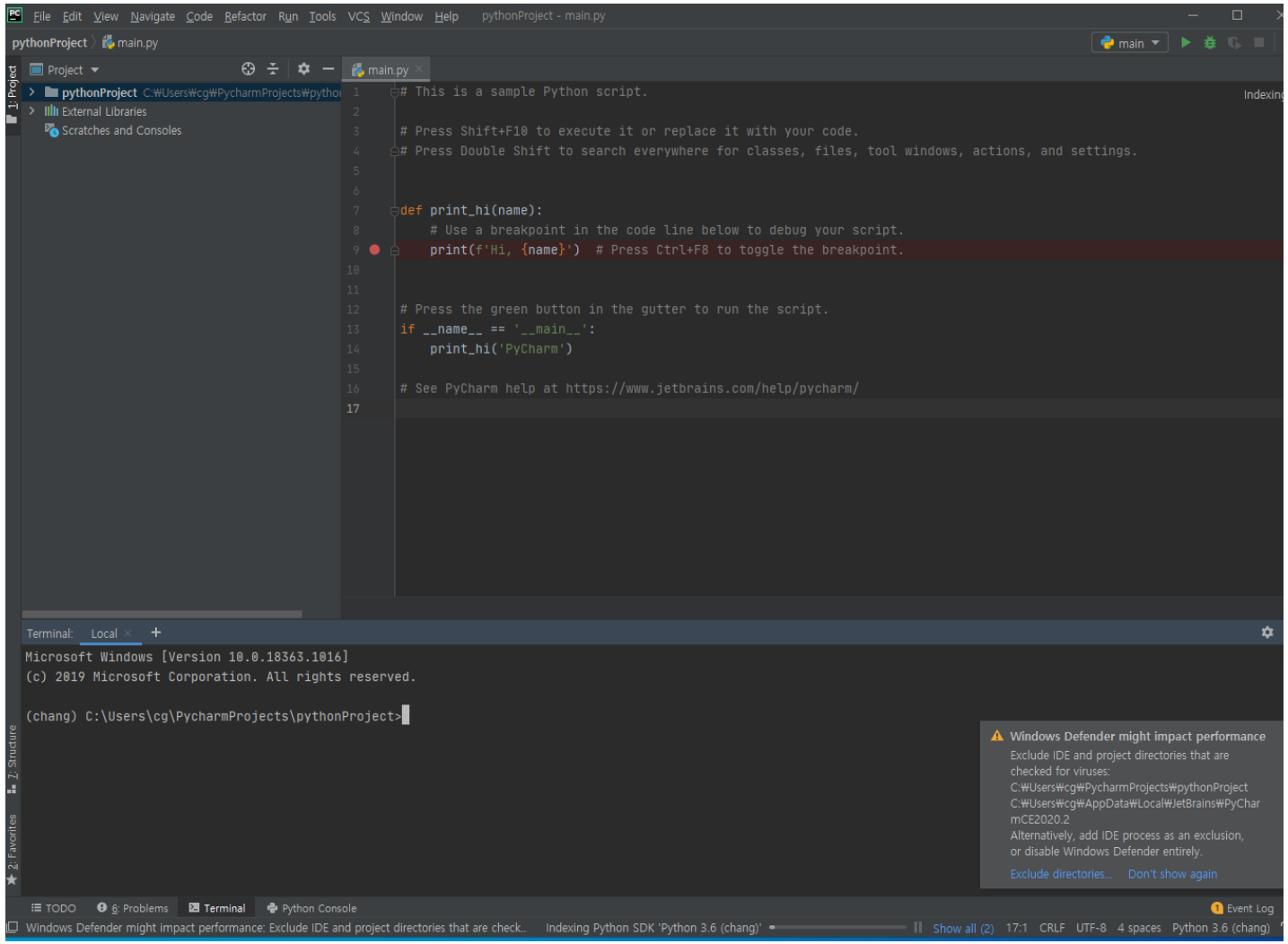

PyCharm Community Edition Setup
Installation Options
Configure your PyCharm Community Edition installation
Create Desktop Shortcut
64-bit launcher
Update PATH variable (restart needed)
Add launchers dir to the PATH
Update context menu
Add "Open Folder as Project"
Create Associations
.py
< Back
Next >
Cancel


Welcome to PyCharm
PyCharm
2021.3.3
Projects
Customize
Plugins
Learn PyCharm
Welcome to PyCharm
Create a new project to start from scratch.
Open existing project from disk or version control.
New Project
Open
Get from VCS
Take a quick onboarding tour
Get familiar with the PyCharm user interface and learn how to code in Python with smart assistance in just 7 minutes!
Start Tour


New Project
Location: C:₩Users₩cg₩PycharmProjects₩pythonProject
Python Interpreter: New Virtualenv environment
New environment using Virtualenv
Location: C:₩Users₩cg₩PycharmProjects₩pythonProject₩venv
Base interpreter: C:₩Users₩cg₩AppData₩Local₩Microsoft₩WindowsApps₩python.exe
Inherit global site-packages
Make available to all projects
Existing interpreter
Interpreter: <No interpreter>
Create a main.py welcome script
Create a Python script that provides an entry point to coding in PyCharm.
Create
Cancel

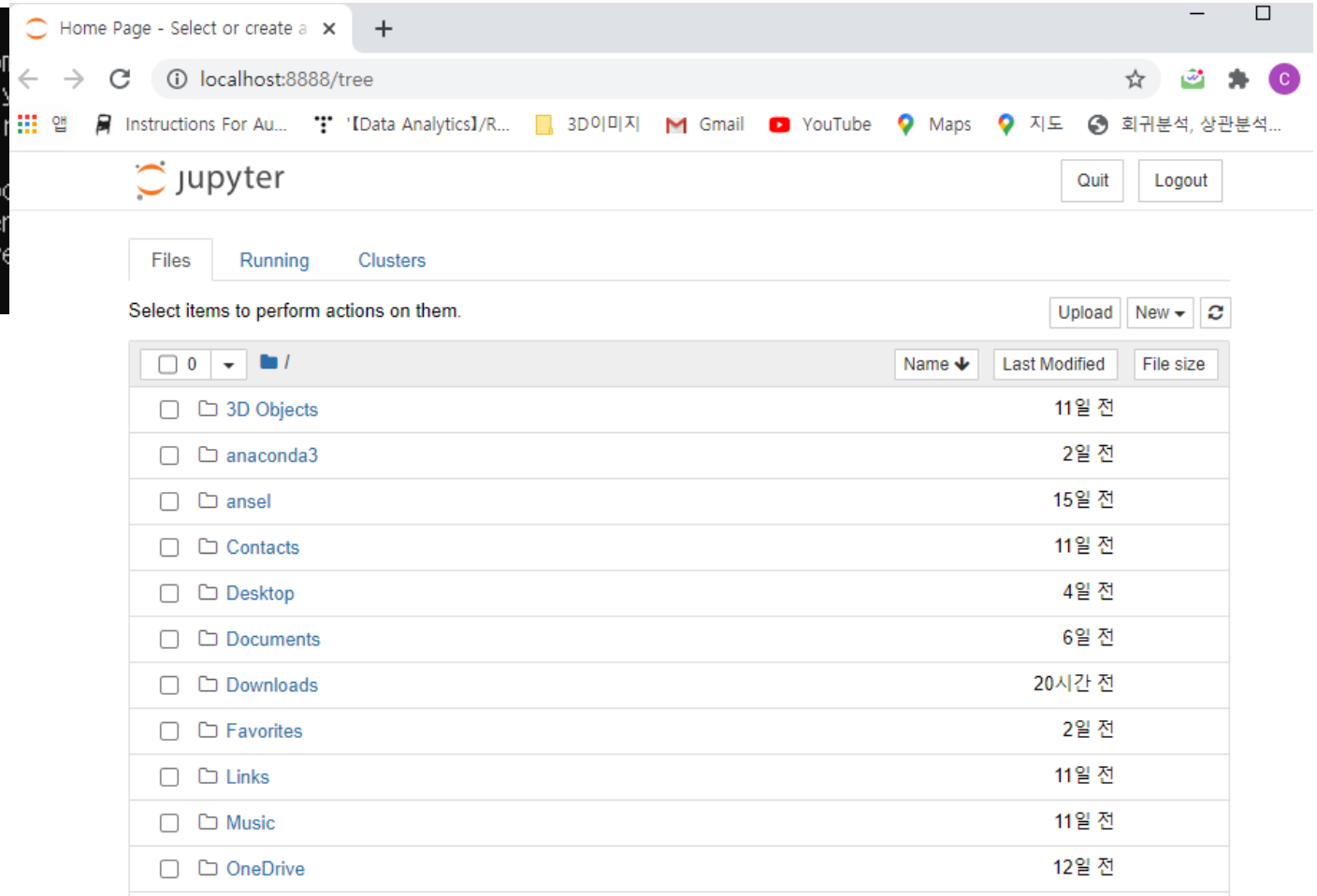

Add Python Interpreter
Virtualenv Environment
Conda Environment
System Interpreter
Pipenv Environment
Interpreter:
C:₩Users₩cg₩anaconda3₩envs₩chang₩python.exe
Conda executable:
C:₩Users₩cg₩anaconda3₩Scripts₩conda.exe
Make available to all projects
OK
Cancel

- Ctrl+/ : 주석처리 및 주석 해제
- Shift+f10 : 전체 실행
- Alt+Shift+e : 한줄 실행



- **Jupyter notebook 실행**

```
C:₩Users₩cg>jupyter notebook
[I 13:28:14.638 NotebookApp] JupyterLab extension loaded fro
[I 13:28:14.639 NotebookApp] JupyterLab application directory
[I 13:28:14.641 NotebookApp] Serving notebooks from local di
[I 13:28:14.641 NotebookApp] The Jupyter Notebook is running
[I 13:28:14.642 NotebookApp] http://localhost:8888/?token=4b
[I 13:28:14.642 NotebookApp]  or http://127.0.0.1:8888/?toke
[I 13:28:14.642 NotebookApp] Use Control-C to stop this serv
[C 13:28:14.689 NotebookApp]
```



Win창에서 anaconda navigator검색 후 실행→Applications on에서 생성한 가상환경 선택→jupyter notebook install → launch