



파이썬을 활용한 데이터 처리 실무자 양성과정

파이썬 학습환경 만들기

강사 조윤실

※ 파이썬을 활용한 데이터 처리 실무자 양성 과정

파이썬 학습환경 만들기

목 차



1.소통 채널 및 코드 저장 환경 만들기

- 1) Slack 워크스페이스 가입하기
- 2) Github Repository 만들기

2.파이썬 소개

- 1) 파이썬 소개

3.파이썬 학습 환경 만들기

- 1) PC에 파이썬 설치하기
- 2) 파이썬 라이브러리 사용 방법
- 3) Visual Studio Code 설치하기
- 4) 구글 코랩(Colab) 사용하기



1. 소통 채널 및 코드 저장 환경 만들기



준비물



환경설정을 위한 준비물

■ 준비물

- PC(노트북)
- 인터넷 웹 브라우저(구글 계정)



구글 크롬

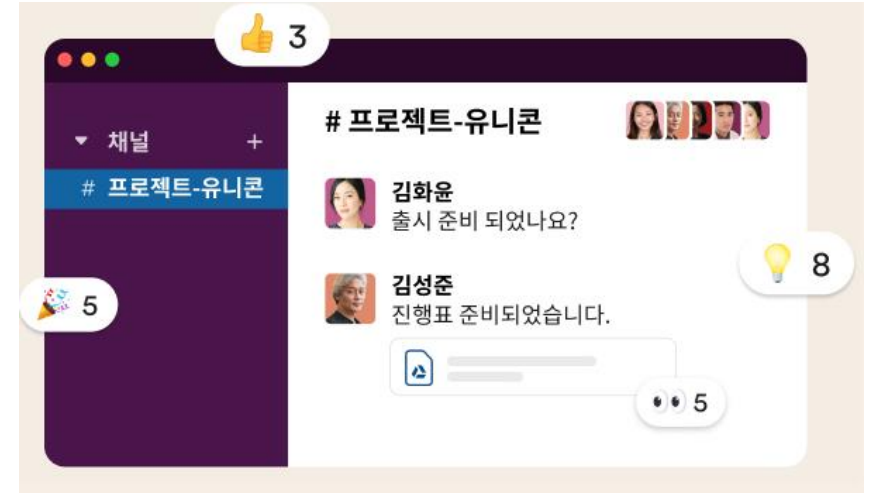
- ① 구글 계정이 없는 사람은
구글 계정을 만들어 주세요.
- ② 크롬 브라우저에
로그인 한 상태로 만들어 주세요.

Slack 워크스페이스 가입하기



Slack 소개

- Slack <https://slack.com/intl/ko-kr/>
 - Global 기업용 메시징 앱,
 - 사내 커뮤니케이션 도구
 - 주로 프로젝트 팀을 위한 메신저로 사용
 - **구글 계정으로 가입하기**
- 우리가 사용할 Slack 프로젝트 링크
<https://gangseo-python.slack.com/>



실습 : Slack 가입하기



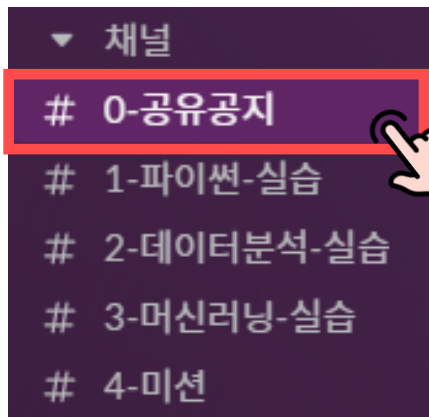
다음 순서대로 Slack에 가입하고 프로젝트에 연결하기

1. 회원 가입하기 (구글계정) : <https://slack.com/intl/ko-kr/>

(화면 아래로 내리면 구글 계정 선택,
가입 후 맨 처음은 Slack에서 보낸 메일에서 클릭해서 들어온다.)

2. 우리 프로젝트 워크스페이스에 가입하기 : <https://gangseo-python.slack.com/>

3. 모든 채널에 가입하기



(모든 채널에 접속 되는지 클릭해보기)

Github Repository 만들기



Github 소개

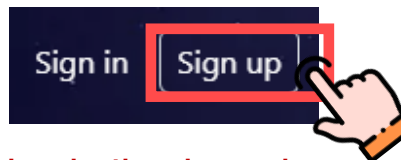
- Github <https://github.com/>
 - Github는 (코드)파일의 버전 관리와 협업을 위한 코드 웹 호스팅 플랫폼.
 - 주로 코드(파일)을 아카이브(저장) 하고 공유하는 용도로 많이 사용
 - 현재 Microsoft사 소유
 - 깃 허브와 OpenAI사가 만든 Copilot 테스트 가능
 - 이메일/비번으로 가입
- 각자 자신의 Github (코드)파일 공간(repository)을 만든 후 수업시간에 만든 파이썬 코드를 업로드할 예정임

실습 : Github에 가입하고 Repository 만들기



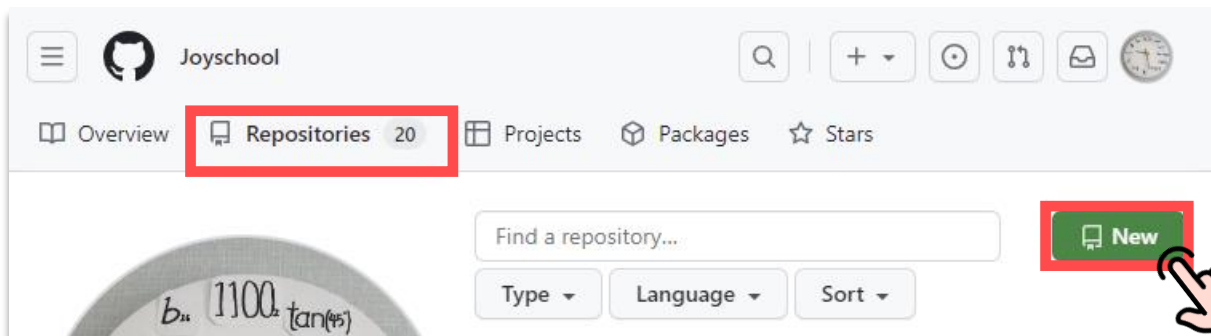
다음 순서대로 Github에 가입하고 자신의 코드파일 저장공간에 연결하기

1. 회원 가입하기 (이메일/비번) : <https://github.com/>



(가입 후 맨 처음은 Github에서 보낸 메일 안내 따르기)

2. Repository 만들기



Repository creation form details:

- Owner *: Joyschool
- Repository name *: **dataAnalysis** (Repository 이름 영어로 입력)
- Description (optional):
- Public (selected): Anyone on the internet can see this repository. You choose who can commit.
- Private: You choose who can see and commit to this repository.
- Initialize this repository with:
 - ☒ Add a README file
- Create repository

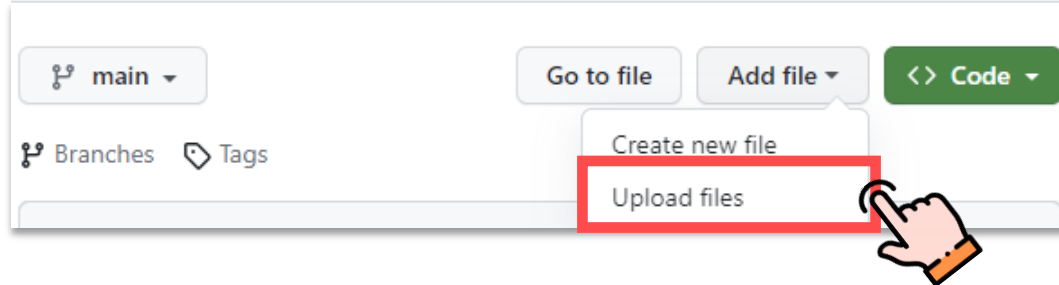
실습 : Repository에 파일 업로드하기



자신의 Repository에 (코드)파일 업로드하기

1. 앞에서 만든 Repository에 연결하기

2. 업로드 메뉴 선택



3. Repository에서 업로드된 파일 확인



2.파이썬 소개

파이썬 소개



파이썬 소개



- 파이썬(Python)은 '귀도 반 로섬'이라는 네덜란드 출신의 소프트웨어 엔지니어가 C언어로 개발해 1991년에 공식으로 발표한 고급 프로그래밍 언어
- 파이썬이라는 이름은 귀도가 좋아하는 코미디 프로그램(Monty Python's Flying Circus)에서 따 왔음
- 2024.03월 현재 최신 (Python 3.12.x)

<https://gvanrossum.github.io/>



프로그래밍 언어의 분류

• 실행 주체에 따른 분류

- 고급언어(High-level Language) : 사람이 이해하기 쉽게 작성된 언어. Python, C, C++, Java 등
- 저급언어(Low-level Language) : 컴퓨터 이해하기 쉽게 작성된 언어. 기계어, Assembly 등

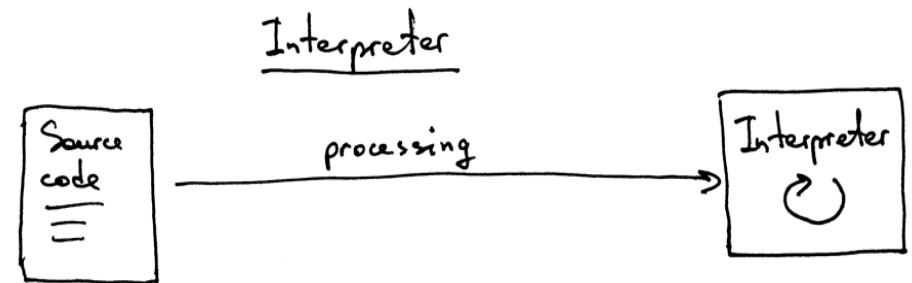
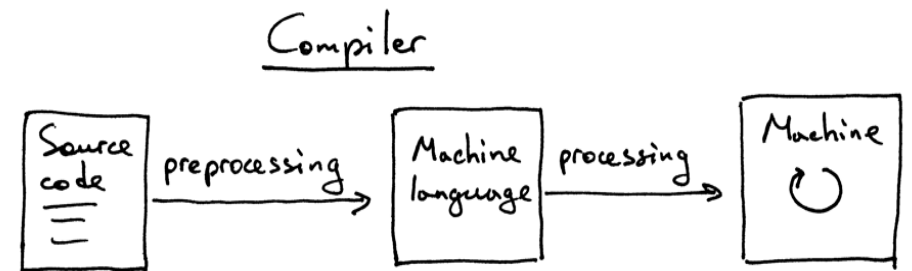
• 실행에 따른 분류

- 컴파일 언어(Compile Language)

원시코드를 실행 가능한 형태의 기계어로 미리 번역하고 빌드하여 구동 시킴. C, C++, Java 등

- 인터프리터 언어(Interpreter Language)

구동 단계에 번역기(Interpreter)에 의해 소스코드를 바로 번역해 실행코드로 변환. Python, Javascript 등





컴파일러 vs. 인터프리터

- 컴파일러(Compiler)의 특징
 - 전체 파일을 스캔하여 한꺼번에 번역한다.
 - 초기 스캔시간이 오래 걸리지만, 한번 실행 파일이 만들어지고 나면 빠르다.
 - 기계어 번역과정에서 더 많은 메모리를 사용한다.
 - 전체 코드를 스캔하는 과정에서 모든 오류를 한꺼번에 출력해주기 때문에 실행 전에 오류를 알 수 있다.
 - 대표적인 언어로 C, C++, JAVA 등이 있다.



컴파일러 vs. 인터프리터

- 인터프리터(Interpreter)의 특징
 - 프로그램 실행 시 한 번에 한 문장 씩 번역한다.
 - 한번에 한 문장 씩 번역 후 실행 시키기 때문에 실행 시간이 느리다.
 - 컴파일러와 같은 오브젝트 코드 생성과정이 없기 때문에 메모리 효율이 좋다.
 - 프로그램을 실행시키고 나서 오류를 발견하면 바로 실행을 중지 시킨다. 실행 후에 오류를 알 수 있다.
 - 대표적인 언어로 Python, Ruby, Javascript 등이 있다.



컴파일러 vs. 인터프리터

- 파이썬과 C 언어의 비교 : <https://pythontutor.com/>

```
Python 3.11
1 list1 = ['This is', 'the only tool']
2 list2 = ['that', 'lets', 'you', 'visually']
3 print(' '.join(list1 + list2))
4 myTuple = ('debug code', 'step-by-step!')
5 print(' '.join(myTuple))
6 fruitSet = {'apple',
7             'banana',
8             'cherry',
9             'durian'}
```

[Edit Code & Get AI Help](#)

→ line that just executed
→ next line to execute

<< First < Prev Next > Last >>

Step 1 of 6

Print output (drag lower right corner to resize)

Frames Objects

```
C (C17 + GNU extensions)
1 #include <stdio.h>
2
3 int main() {
4     int x[] = {10, 20, 30};
5     int* p = &x[1]; // pointer into middle
6     char* fruit[3] = {"apples",
7                      "bananas",
8                      "cherries"};
9
10    printf("I have %d %s\n", *p, fruit[1]);
11    return 0;
12 }
```

[Edit Code & Get AI Help](#)

→ line that just executed
→ next line to execute

<< First < Prev Next > Last >>

Step 1 of 6

Print output (drag lower right corner to resize)

Stack

main

x	array		
	0	1	2
	int	int	int
	?	?	?

p

pointer to int

?

fruit

array	0	1	2
	pointer to char	pointer to char	pointer to char
	?	?	?

Note: ? refers to an uninitialized value

C/C++ details: none [default view]



파이썬 특징

- 파이썬의 특징

배우기 쉽고 결과를 바로 확인할 수 있어 초보자에게 적합한 코딩 언어

- 텍스트 언어이다.

- 다른 언어에 비해 간결하고, 코드의 가독성이 높다.

영어와 유사한 문법으로 프로그램을 작성할 수 있다.

- 인터프리터 언어로 한 번에 한 줄의 코드가 실행되어 실행 상태를 확인하기 쉽다.

불편한 컴파일 과정이 없고, '대화하듯'이 프로그램을 작성하고 실행한다.

- 플랫폼에 독립적이다.

Windows, Mac, Unix/Linux... 등 동작가능. 단, 모바일 환경(안드로이드/iOS)에서는 동작 안함)

라즈베리파이 기반에서 잘 동작하므로 사물인터넷(IoT) 개발 및 운영에 적극 활용



파이썬 특징

- **다양한 라이브러리가 있어 필요한 기능을 찾아 쓰기 쉽다.**

파이썬에서 제공하는 라이브러리뿐 아니라, 외부에서 제공하는 다양한 서드 파티(Third Party) 라이브러리까지 사용 가능

- **다른 언어와 연결하기 쉽다.**

다른 언어로 쓰인 모듈들을 연결하는 풀언어(glue language)로써 자주 이용됨.
많은 사용 프로그램에서 스크립트 언어로 채용되고 있음.

- **객체 지향(object-oriented) 언어이다.**

소프트웨어 개발과 보수를 간편하게 하며, 보다 직관적인 코드 분석이 가능하다.
(C언어의 절차지향적 프로그래밍과 비교)

- **강력한 웹 프레임워크를 사용할 수 있다.**

파이썬의 웹 프레임워크(예: Django, Flask 등)를 사용해 강력하고 빠른 웹 환경을 구축 가능



파이썬이 활용되는 분야

- 시스템 유틸리티 제작
- GUI 프로그래밍
- C/C++와의 결합
- 웹 프로그래밍
- 수치 연산 프로그래밍
- 데이터베이스 프로그래밍
- 데이터 분석, 사물 인터넷
- 머신러닝, 딥러닝 등 인공지능 분야
등등 매우 다양하고 폭 넓게 사용됨



데이터 분석

다양한 파이썬 라이브러리를 활용하면 데이터를 쉽게 분석하고 정리할 수 있습니다.



딥러닝

파이썬 라이브러리 TensorFlow를 활용하세요. 여러분도 '알파고'에 활용된 딥러닝을 배울 수 있어요



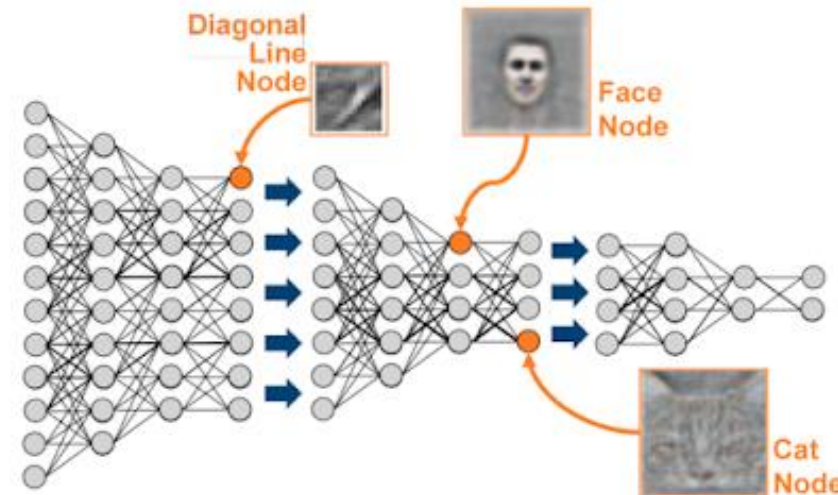
업무 자동화

웹크롤링, 엑셀, 이메일 알림 자동화 등 반복적인 업무를 자동화 시키세요. 일은 컴퓨터가 하고 나는 더 생산적인일에 집중하는 거죠.



웹 개발

파이썬으로 웹 개발도 할 수 있어요. 파이썬의 웹 프레임워크인 Django로 나만의 웹서비스를 만들어보세요.



대학원

교수님이 엄청난 양의 데이터를 놓고 가셨나요? worry마세요. 파이썬과 SciPy를 활용하면 여러 문제들을 컴퓨터로 쉽게 풀 수 있습니다.



3. 파이썬 학습 환경 만들기





파이썬 학습환경

- 파이썬은 다양한 도구(툴)를 사용하여 프로그래밍이 가능하다.

① IDLE(파이썬 설치하면 자동 설치됨)

```
Python 3.8.5 Shell
Python 3.8.5 (tags/020, 15:57:54) [MS
on win32
Type "help", "copy
ense()" for more i
>>>

File Editor
class VideoCamera(object):
    def __init__(self):
        #비디오(웹캠) 사용
        self.video = cv2.VideoCapture(0)

    def __del__(self):
        #비디오 중지
        self.video.release()
```

② 주피터 노트북(라이브러리 설치)

```
jupyter operation_test
Trusted Python 3

File Edit View Insert Cell Kernel Widgets Help

In [10]: #곱셈(*)
3 * 4
Out[10]: 12

In [11]: #나눗셈(/)
9/3.0
```

③ VS Code(에디터)

```
1. numpy 배열

numpy 사용

In [1]: import numpy as np
a = [0,1,2,3,4,5]
print(type(a), a)
na = np.array(a)
print(type(na), na)

Out[1]: <class 'list'> [0, 1, 2, 3, 4, 5]
<class 'numpy.ndarray'> [0 1 2 3 4 5]

1차원 배열 만들기
```

④ Colab(구글 코랩 사이트)

```
Linear_regression_200413.ipynb
파일 수정 보기 삽입 런타임 도구 도움말

더블클릭 또는 Enter 키를 눌러 수정

[ ] tensorflow_version 1.x
TensorFlow 1.x selected.

[ ] import tensorflow as tf

[ ] a = tf.constant(3.0)
b = tf.constant(4.0)
c = tf.constant(5.0)

d = a + b + c
print(d)
# print d node

Tensor("add:0", shape=(), dtype=float32)
```

PC에 파이썬 설치 없이 파이썬 사용 가능

PC에 파이썬 설치하기

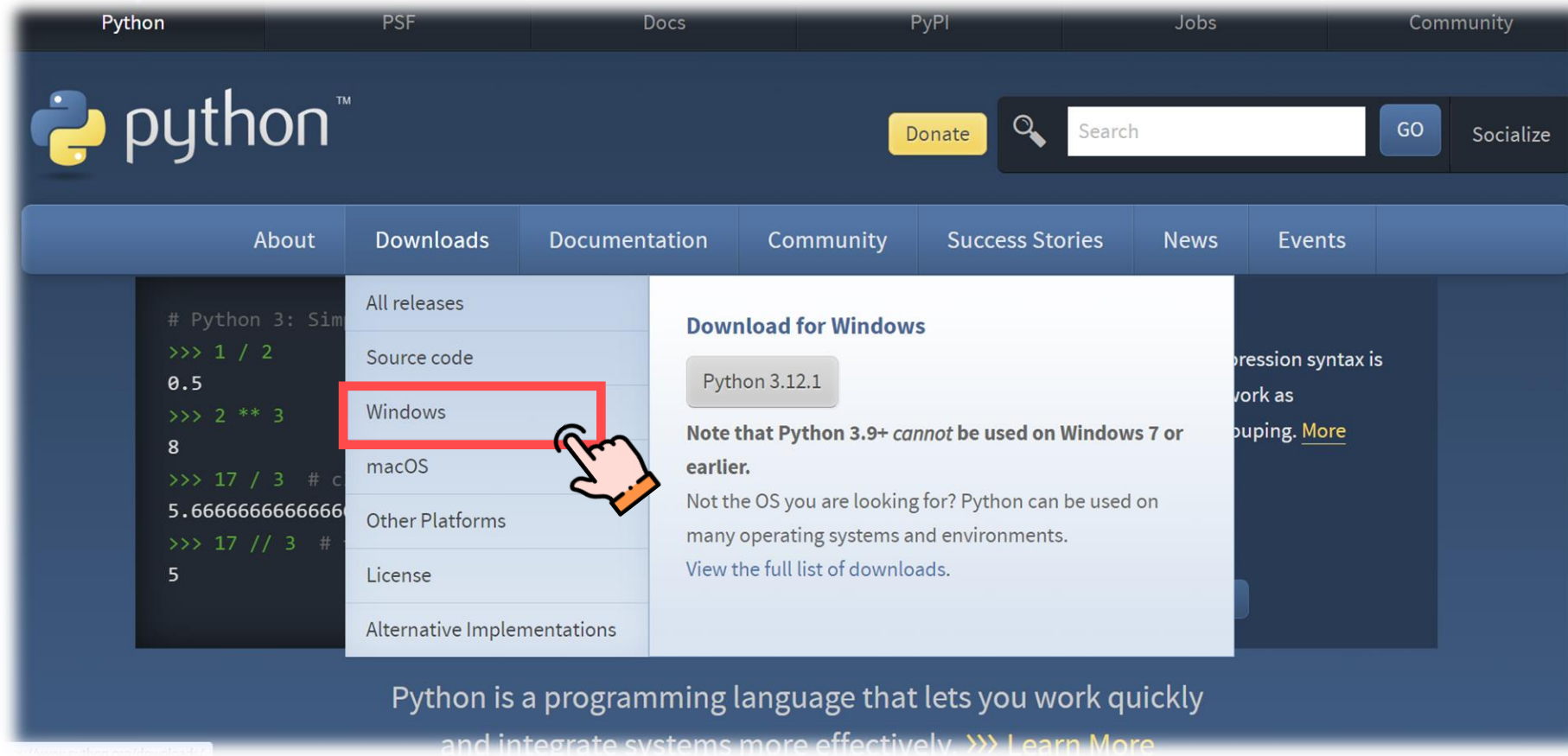
실습 : 파이썬 설치하기



www.python.org

PC 웹 브라우저(chrome, Edge 등)에서 파이썬 공식 사이트를 입력해 주세요.

※ [주의]
이 교육과정에서는
사용자가 Windows OS
PC를 사용한다는
가정에 설치방법이
기술되었습니다.



파이썬 설치하기



www.python.org

Windows용 파이썬을 선택하고 **64-bit용 3.12.x 버전**을 다운로드 해주세요.

Stable Releases

- Python 3.12.1 - Dec. 8, 2023

Note that Python 3.12.1 *cannot* be used on Windows 7 or earlier.

- Download [Windows embeddable package \(32-bit\)](#)
- Download [Windows embeddable package \(64-bit\)](#)
- Download [Windows embeddable package \(ARM64\)](#)
- Download [Windows installer \(32 -bit\)](#)
- Download [Windows installer \(64-bit\)](#)
- Download [Windows installer \(ARM64\)](#)

※ [참고]

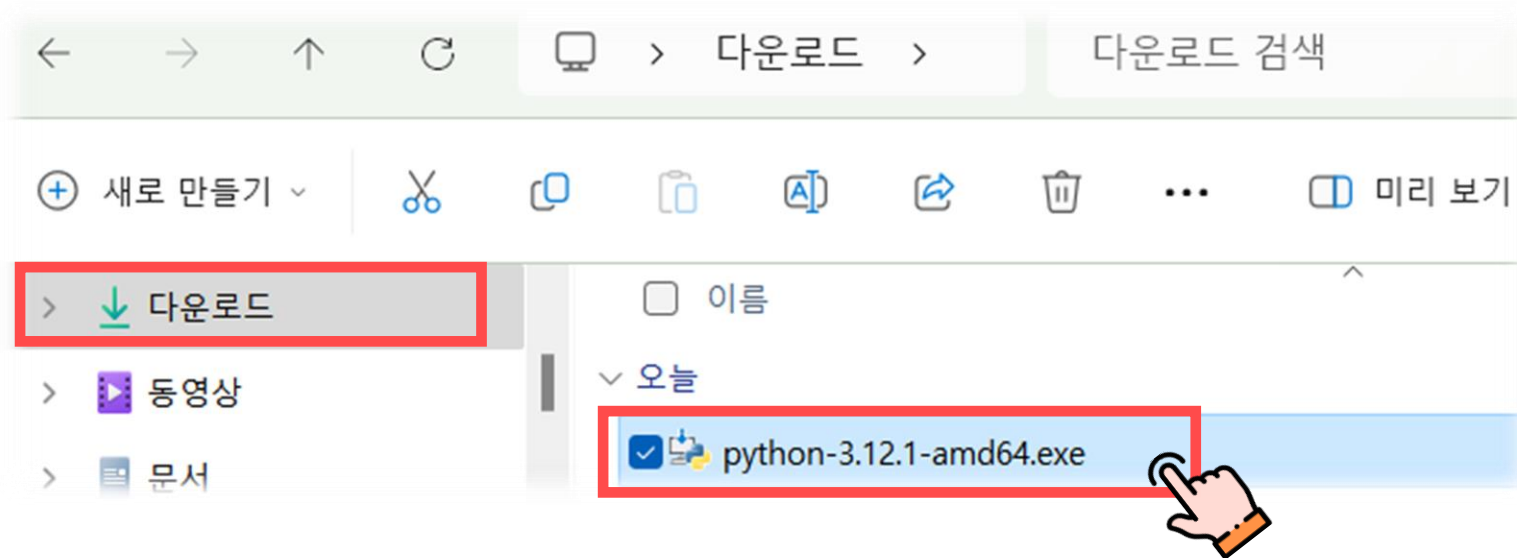
PC에 이미 파이썬이 설치되었다면
3.10.x 이상인 경우
다시 설치 안 하고
설치된 버전을 사용하셔도 됩니다.

파이썬 설치하기

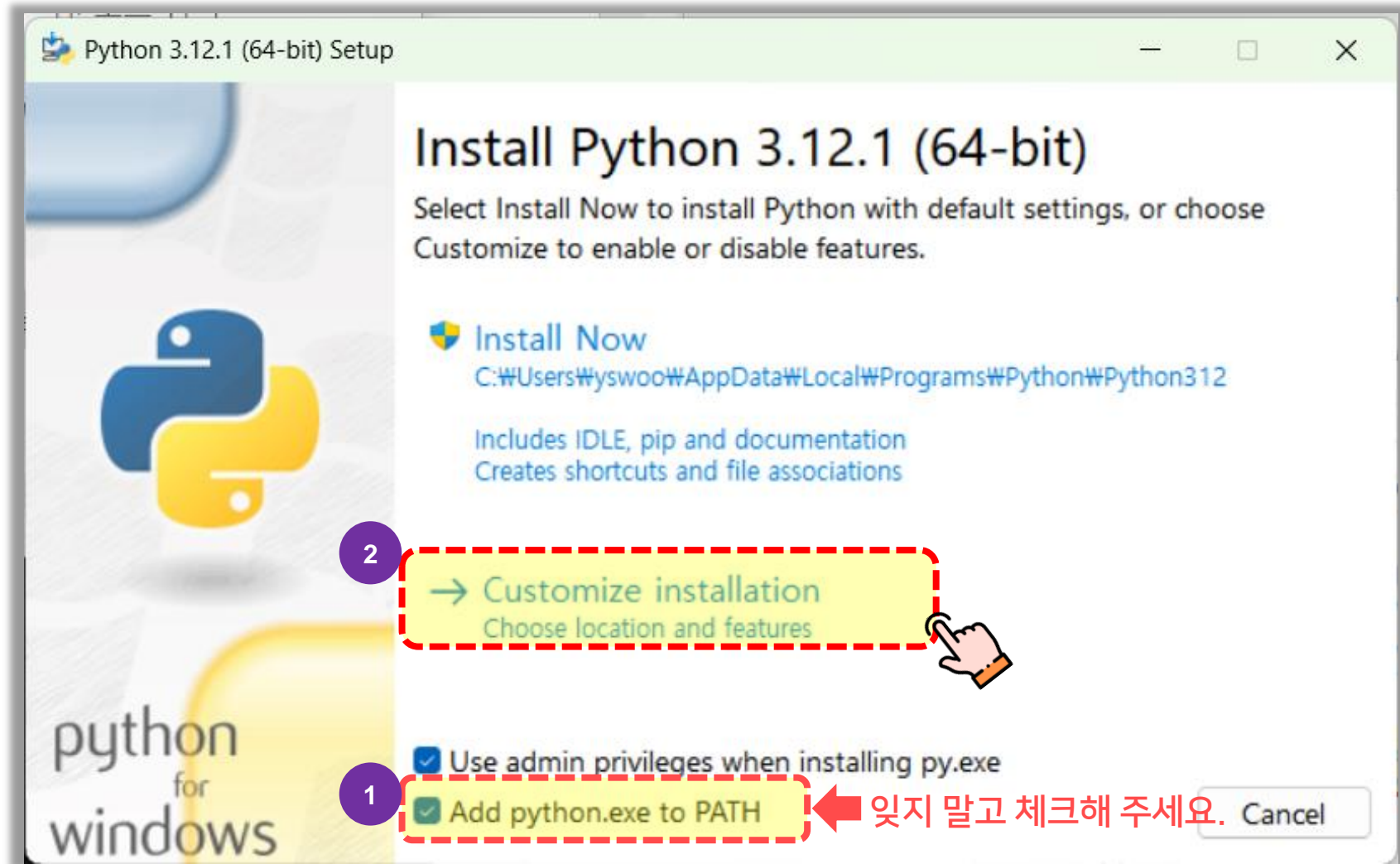


다운로드된 exe파일 실행하기

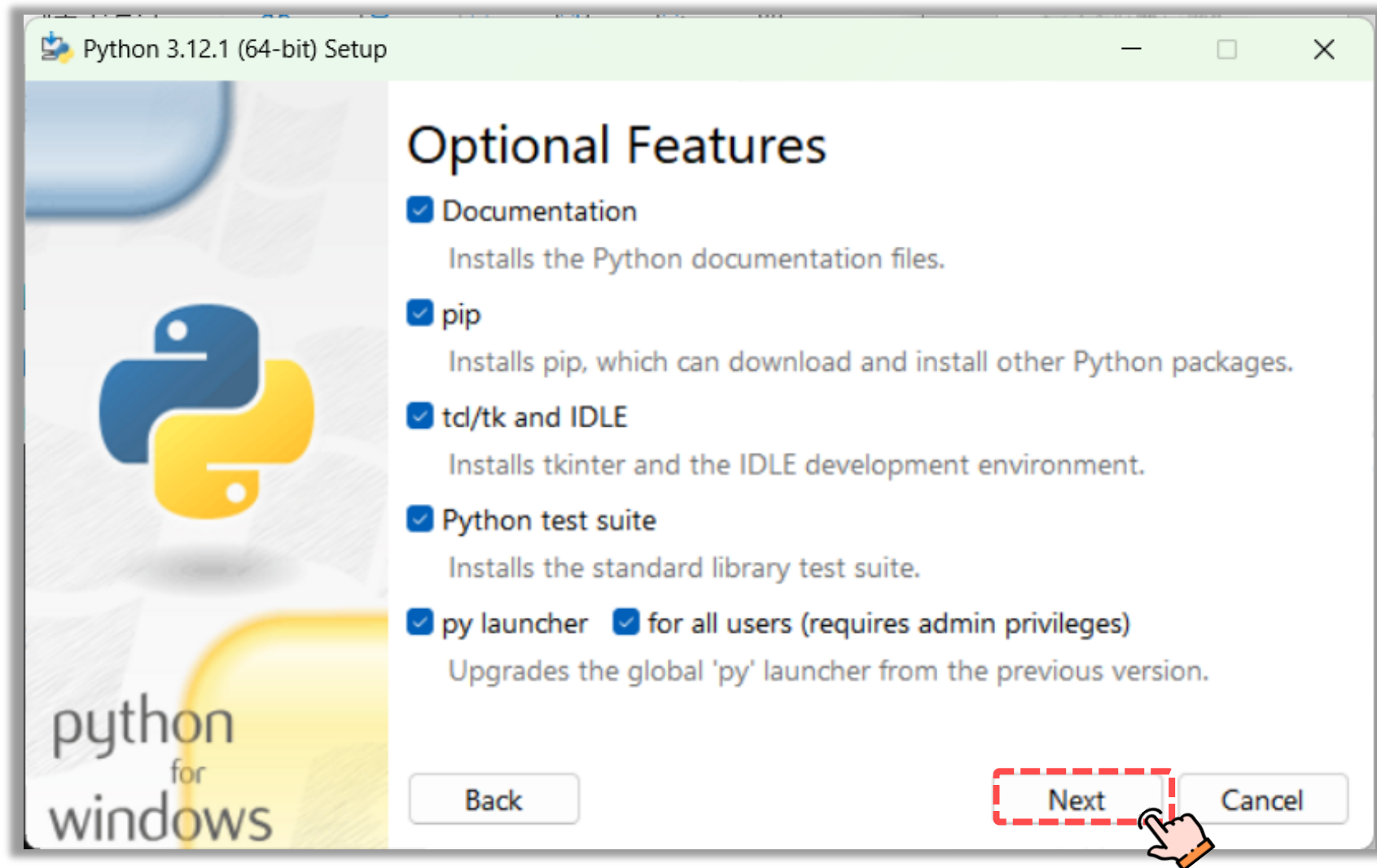
다운로드된 파이썬 설치 파일을 실행해서 파이썬 인터프리터를 설치해 주세요.



파이썬 설치하기

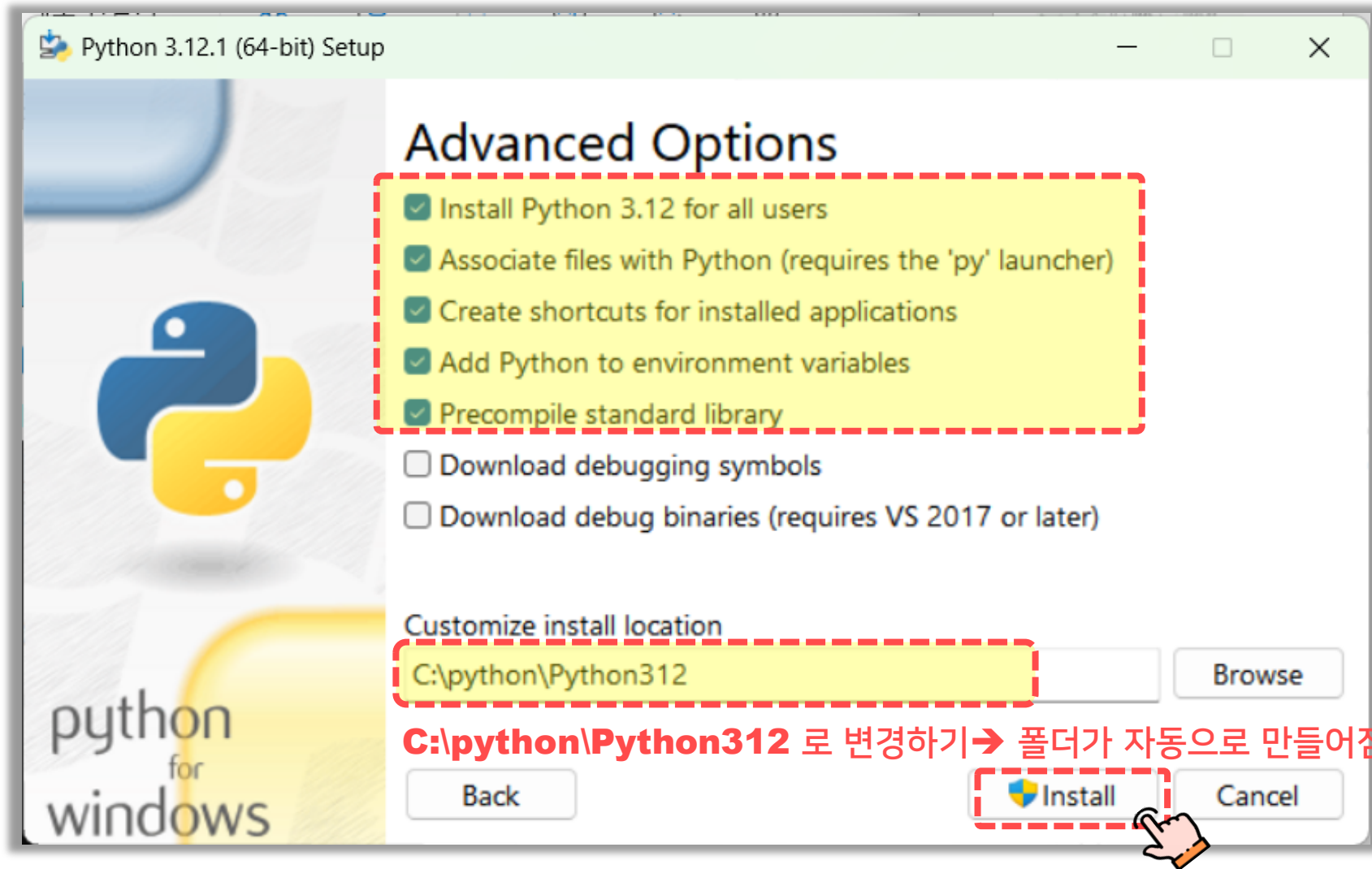


파이썬 설치하기



※본 강의 자료는 본인 학습용으로만 사용 가능하며 무단 복제/배포를 금지합니다.

파이썬 설치하기



파이썬 설치하기



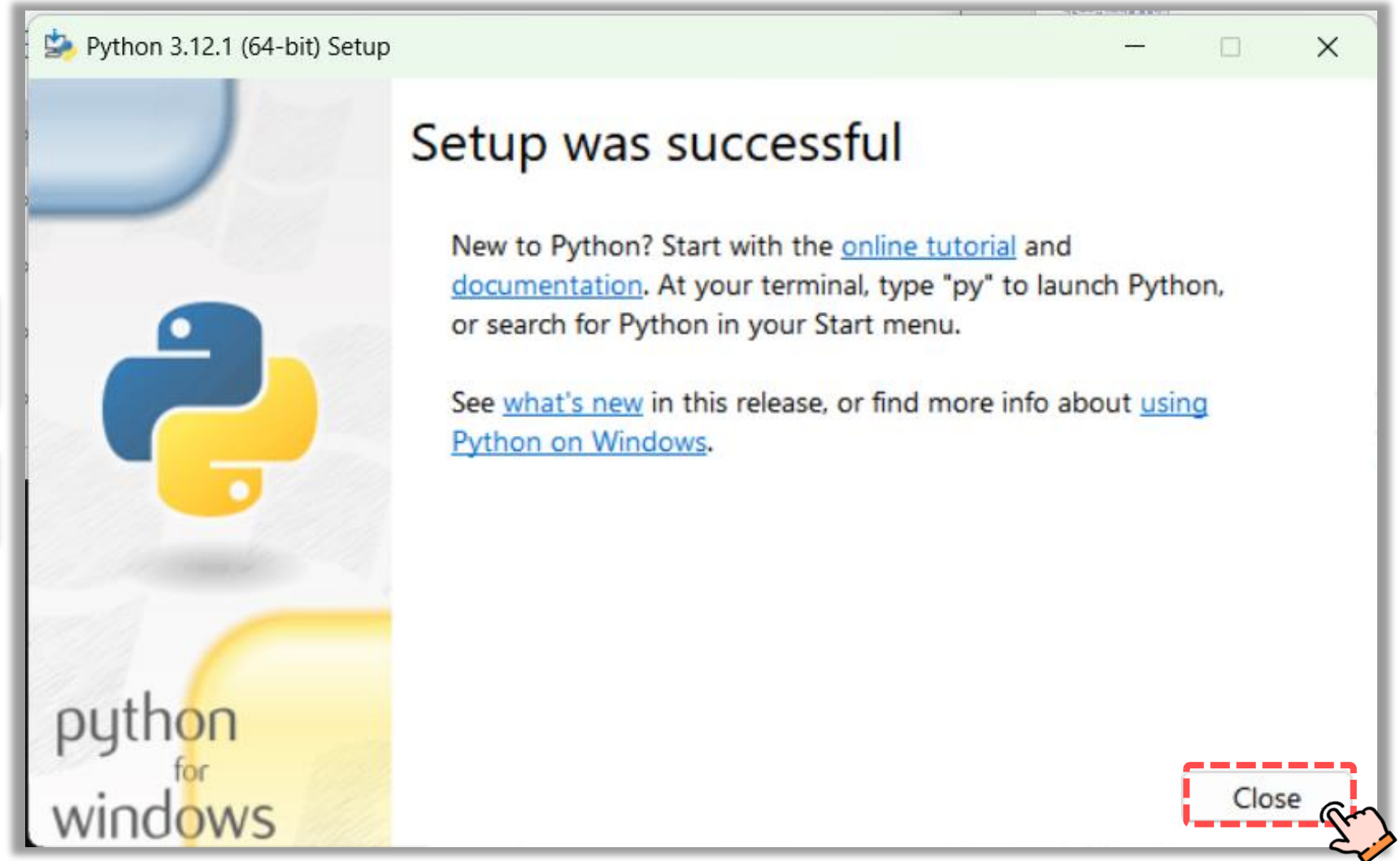
만약 이 메시지가 나오면
예를 선택한다

사용자 계정 컨트롤

이 앱이 디바이스를 변경할 수 있도록 허용하
시겠어요?

예


아니요








※본 강의 자료는 본인 학습용으로만 사용 가능하며 무단 복제/배포를 금지합니다.

파이썬 설치하기



PC화면 왼쪽 하단의 [윈도우창 ] 을 클릭하여 python 3.12이 아래와 같이 설치되었는지 확인!

 Python 3.12 신규	
 IDLE (Python 3.12 64-bit) 신규	→ 파이썬 기본 개발 툴(통합개발환경)
 Python 3.12 (64-bit) 신규	→ 파이썬 셸 프로그램
 Python 3.12 Manuals (64-bit)	→ 파이썬 매뉴얼(PC에 설치된)
 Python 3.12 Module Docs (64-bit) 신규	→ 파이썬 주요 모듈 매뉴얼 (PC에 설치된, 웹 브라우저에서 볼 수 있는 형태)

실습 : IDLE 사용하기



파이썬 코드 실행하기

- IDLE에서 파이썬 코드 실행하기

```
import sys  
  
sys.executable
```

- 파이썬 라이브러리 위치 확인

```
print(sys.path)
```

아래 내용과 완전히 동일하지 않더라도
C:\wwpython\wwpython312~ 로
시작되는 내용이 있으면 됩니다.



파이썬 설치 위치 & 학습 코드 위치 확인하기

교육 중에 사용할 파이썬 설치 위치와 학습 코드 폴더 위치 확인하기

파이썬 설치 위치 **C:\python\Python312**

학습 코드 위치 **C:\python\dataAnalysis**

C:\python\dataAnalysis\data

C:\python\dataAnalysis\image

해당 폴더가 없다면
파일 탐색기 창을 열어
(윈도우즈키+E)
폴더를 만들어 주세요.

파이썬 라이브러리 사용 방법



파이썬 라이브러리 확인

■ 파이썬 설치 경로

```
C:\python\bioinformatics>python
Python 3.10.9 (tags/v3.10.9:1dd9be6, Dec 6 2022, 20:01:21)
[MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more i
nformation.
>>> import sys
>>> sys.executable
'C:\\python\\Python310\\python.exe'
>>> |
```

python

※ 파이썬 셸 들어가기

import sys

sys.executable

■ 라이브러리 설치 위치 확인

```
>>> sys.path
['', 'C:\\python\\Python310\\python310.zip', 'C:\\python\\P
ython310\\DLLs', 'C:\\python\\Python310\\lib', 'C:\\python\\
Python310', 'C:\\python\\Python310\\lib\\site-packages', '
C:\\python\\Python310\\lib\\site-packages\\win32', 'C:\\pyt
hon\\Python310\\lib\\site-packages\\win32\\lib', 'C:\\pytho
n\\Python310\\lib\\site-packages\\Pythonwin']
>>> exit()
```

sys.path

※ 각자 설치 위치에 따라 결과 값이 다를 수 있다

exit()

※ 파이썬 셸 나가기



파이썬 라이브러리 확인

■ 설치된 라이브러리 **전체 확인**

```
명령 프롬프트
C:\python\bioinformatics>pip list
Package            Version
-----
absl-py            1.4.0
abstract_singleton 1.0.1
accelerate         0.16.0
aiohttp            3.8.4
aiosignal          1.3.1
```

`pip list`

또는

`python -m site`

■ 특정 라이브러리 설치 확인

```
C:\python\bioinformatics>pip show numpy
Name: numpy
Version: 1.24.2
Summary: Fundamental package for array computing in Python
Home-page: https://www.numpy.org
Author: Travis E. Oliphant et al.
Author-email:
License: BSD-3-Clause
Location: c:\python\python310\lib\site-packages
```

`pip show [라이브러리명]`



파이썬 라이브러리 설치 & 삭제

■ 특정 라이브러리 설치

```
C:\python\bioinformatics>pip install numpy
Collecting numpy
  Obtaining dependency information for numpy from https://files.pythonhosted.org/packages/b7/db/4d37359e2c9cf8bf071c08b8a6f7374648a5ab2e76e2e22e3b808f81d507/numpy-1.25.2-cp310-cp310-win_amd64.whl.metadata
  Downloading numpy-1.25.2-cp310-cp310-win_amd64.whl.metadata (5.7 kB)
  Downloading numpy-1.25.2-cp310-cp310-win_amd64.whl (15.6 MB)
  ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 15.6/15.6 MB 8.7 MB/s eta 0:00:00
Installing collected packages: numpy
Successfully installed numpy-1.25.2
```

pip install [라이브러리명]

■ 설치된 라이브러리 삭제

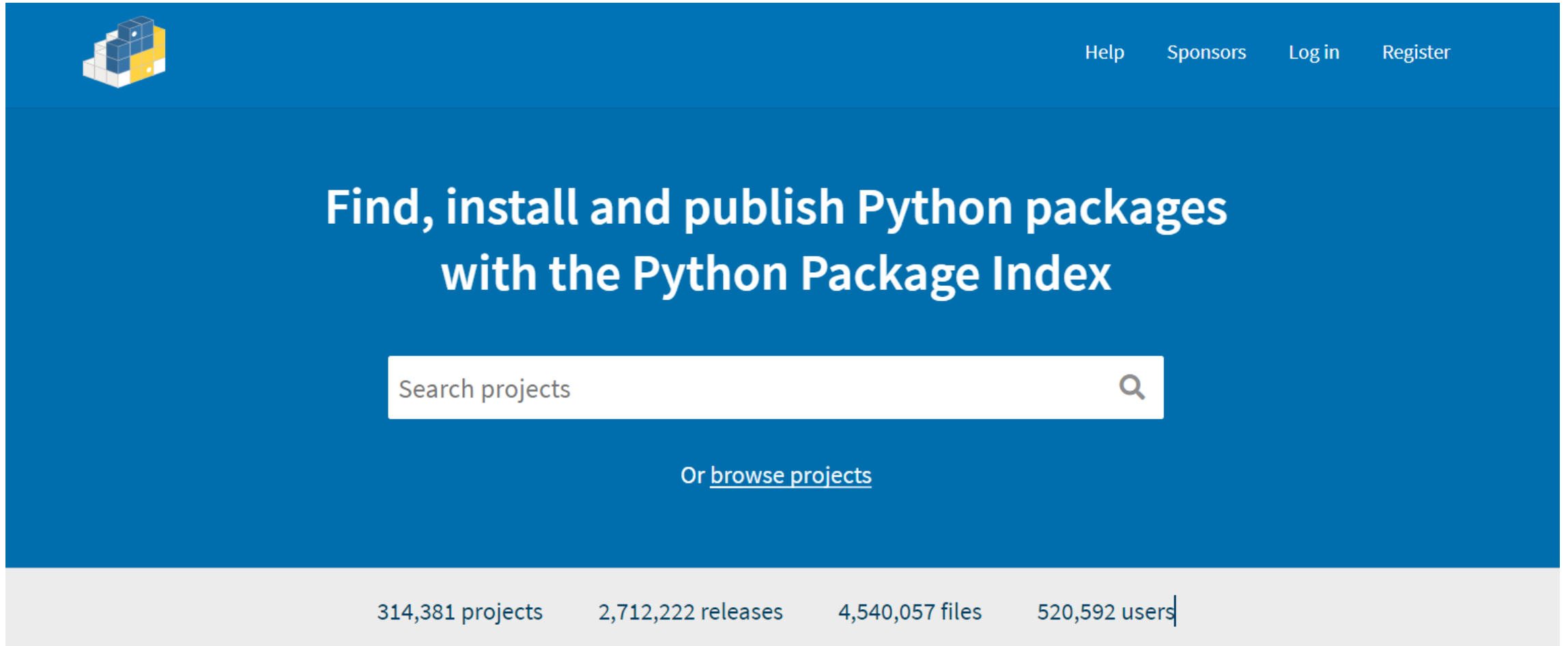
```
C:\python\bioinformatics>pip uninstall numpy
Found existing installation: numpy 1.24.2
Uninstalling numpy-1.24.2:
  Would remove:
    c:\python\python310\lib\site-packages\numpy-1.24.2.dist-info\*
    c:\python\python310\lib\site-packages\numpy\*
    c:\python\python310\scripts\f2py.exe
Proceed (Y/n)? Y
Successfully uninstalled numpy-1.24.2
```

pip uninstall [라이브러리명]



최신 파이썬 라이브러리 버전 확인

- <https://pypi.org/>



Find, install and publish Python packages
with the Python Package Index

Search projects

Or [browse projects](#)

314,381 projects 2,712,222 releases 4,540,057 files 520,592 users

MS VSC(Visual Studio Code) 설치하기

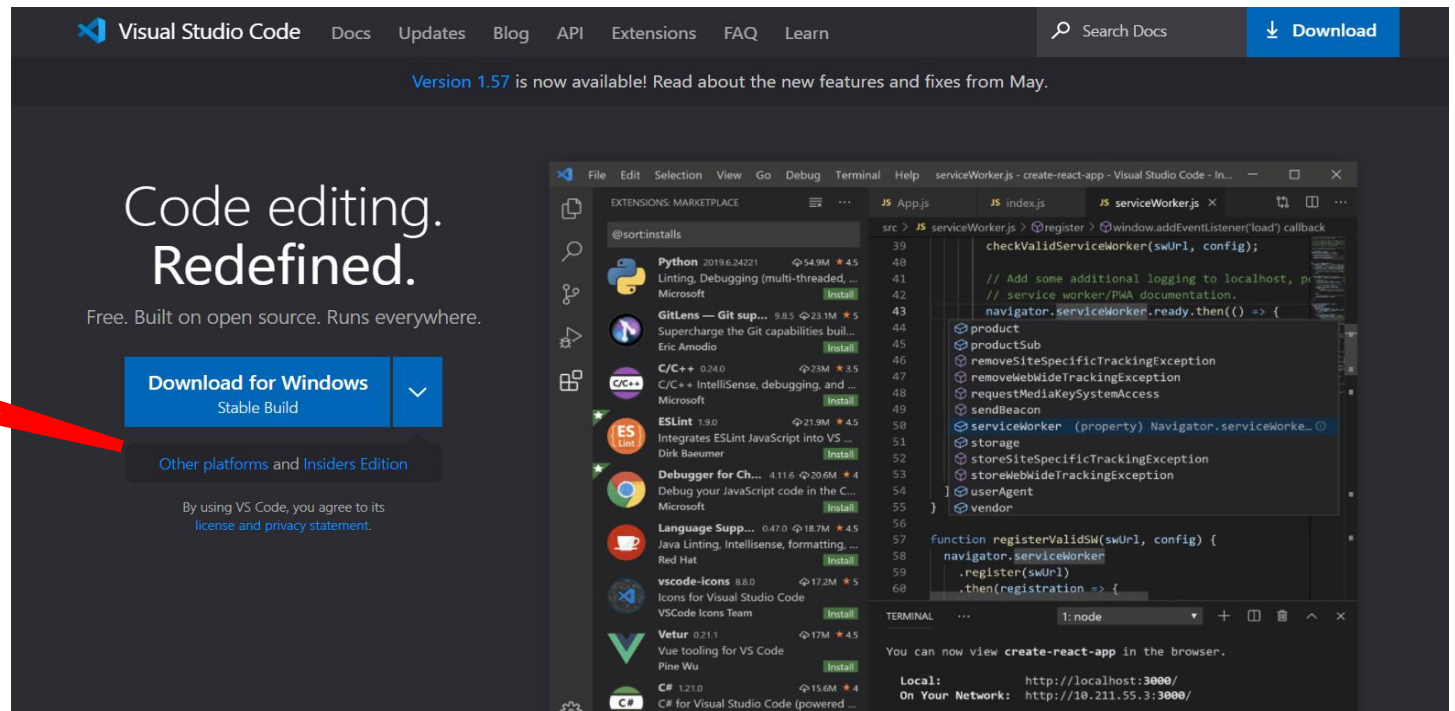
VSC 설치하기



Visual Studio Code

- Microsoft에서 개발한 범용 소스 코드 편집기(Windows, Linus, MacOS)
- 설치 프로그램 다운로드: <https://code.visualstudio.com/>
- 다운로드한 exe파일 실행시켜 설치하기 노트 열기


OS에 맞게 다운로드 버튼이
달라짐



VSC 환경 설정

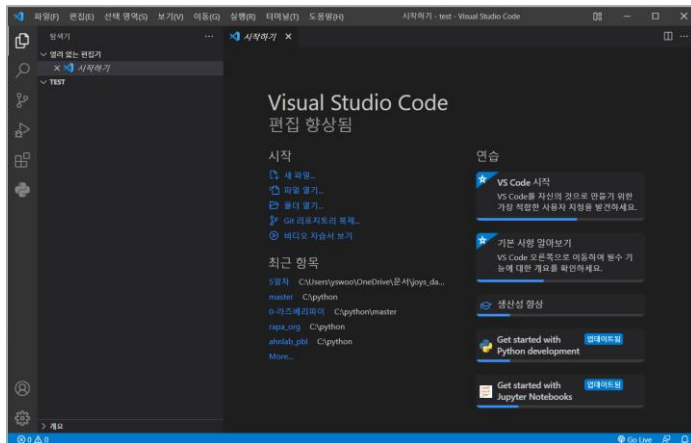


한글팩 설치(한글 메뉴로 보이지 않는 경우 설치하기)

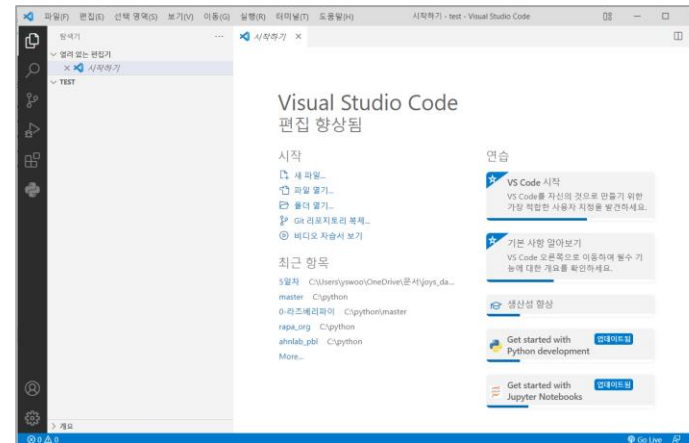
- ① 확장 버튼  클릭: 키보드로 Korean 입력해서 Korean Language Pack~ 선택
- ② 팝업창에서 [restart] 버튼 선택해서 VSC 재시작

화면 색상 바꾸기

- ① 파일 > 기본설정 > 색 테마 : 원하는 색상 테마 선택 :
선택 예: 어둡게+ (기본 어둡게)



- 선택 예: 밝게+ (기본 밝게)

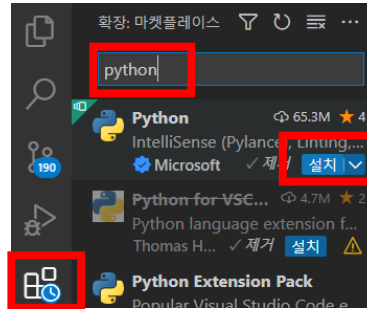


VSC 환경 설정



확장 프로그램 설치 : Python, Jupyter 용 확장 프로그램 설치

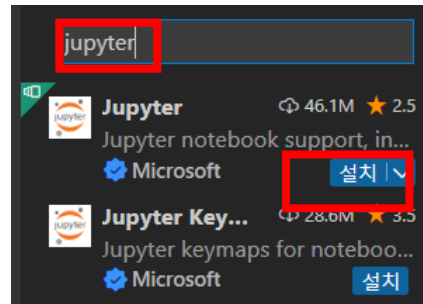
① 확장 버튼  클릭: Python Install



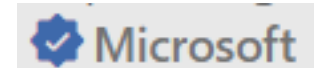
(Microsoft 확장 팩 선택)



② 확장 버튼  클릭: Jupyter Install



(Microsoft 확장 팩 선택)



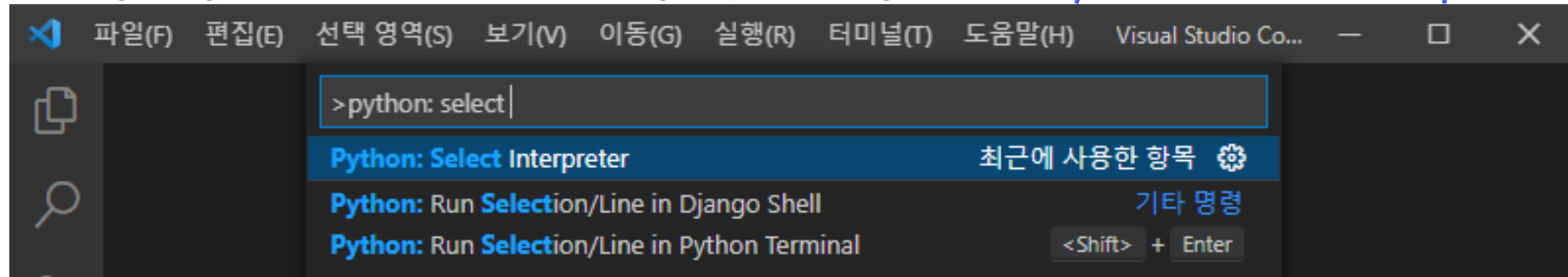
VSC 환경 설정



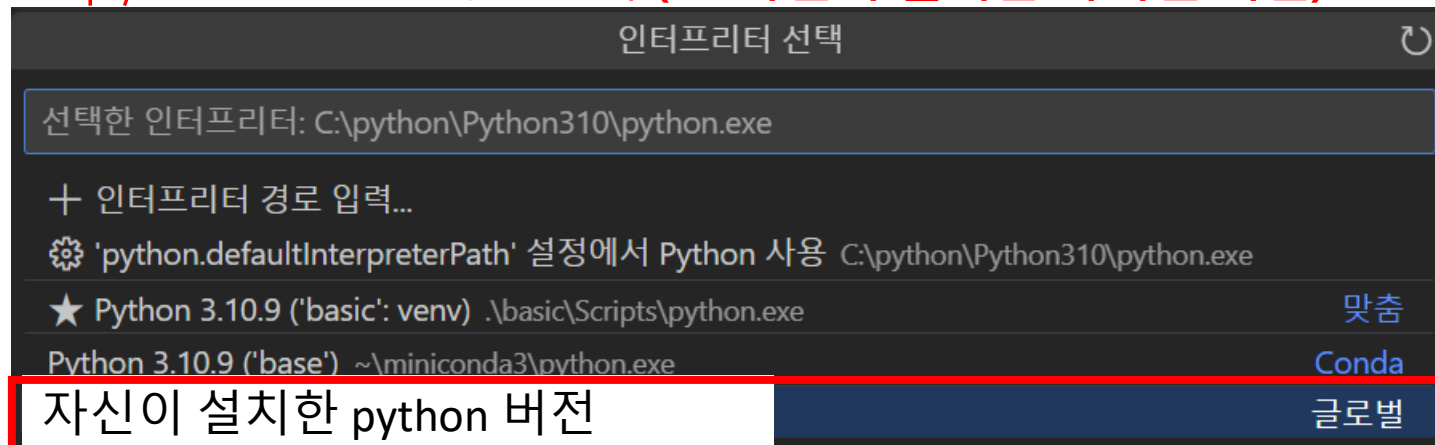
파이썬 인터프리터 선택 : python 3.12.x 선택

- ① 파이썬 인터프리터 선택하기: (ctrl + shift + p)

view(보기) > Command Palette(명령 팔레트) 클릭 > Python:Select Interpreter 선택



- ② → python3.12.x 버전 선택 (← 자신이 설치한 파이썬 버전)



구글 코랩(Colab) 사용하기

구글 코랩 사용하기



구글 코랩 사이트 접속하기: <https://colab.research.google.com>

- 클라우드 기반 파이썬 실행 환경(파이썬+주피터 노트북)

구글 코랩 사용하기



새 노트 작성하기: <https://colab.research.google.com>

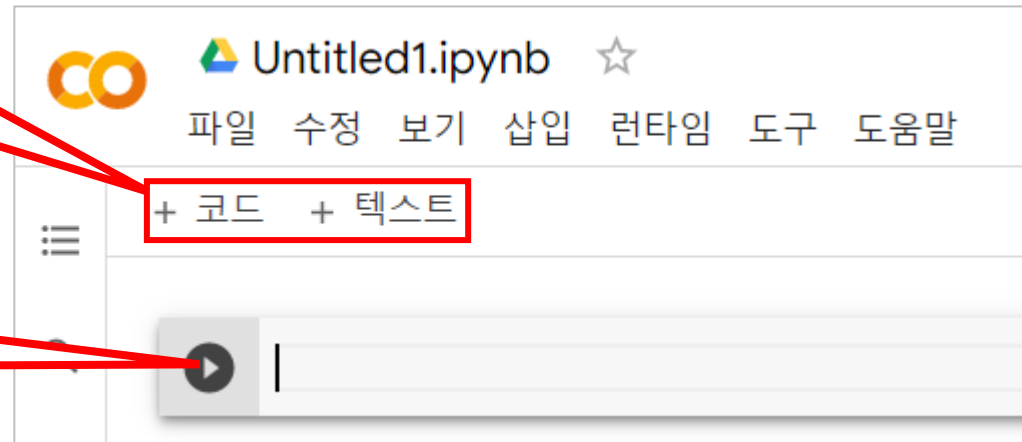
- 메뉴: 파일 > 새 노트

파이썬 코딩을 위해
새 노트를 선택한다.

셀 추가 유형

- ① +코드 : 파이썬 코드
- ② +텍스트 : 일반 텍스트(설명)

파이썬 코드 작성 후
버튼 클릭해서 코드 실행하기



구글 코랩 사용하기

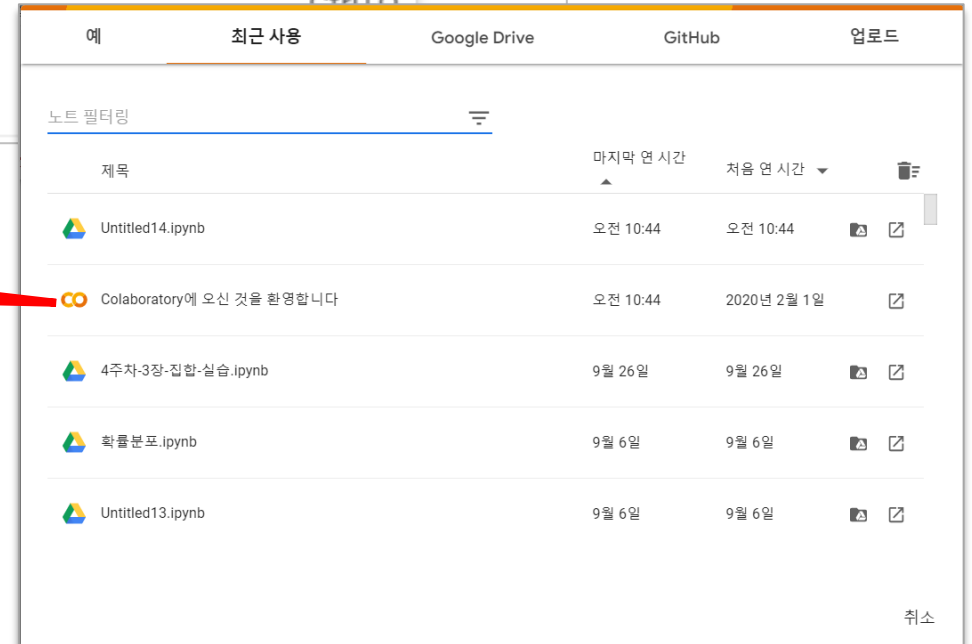


노트 열기

- 메뉴: 파일 > 노트 열기

기존에 작성된 파이썬 코드를 불러오기 위한 메뉴

기존에 작성된 파이썬 코드 (*.ipynb) 선택해서 불러오기



구글 코랩 사용하기

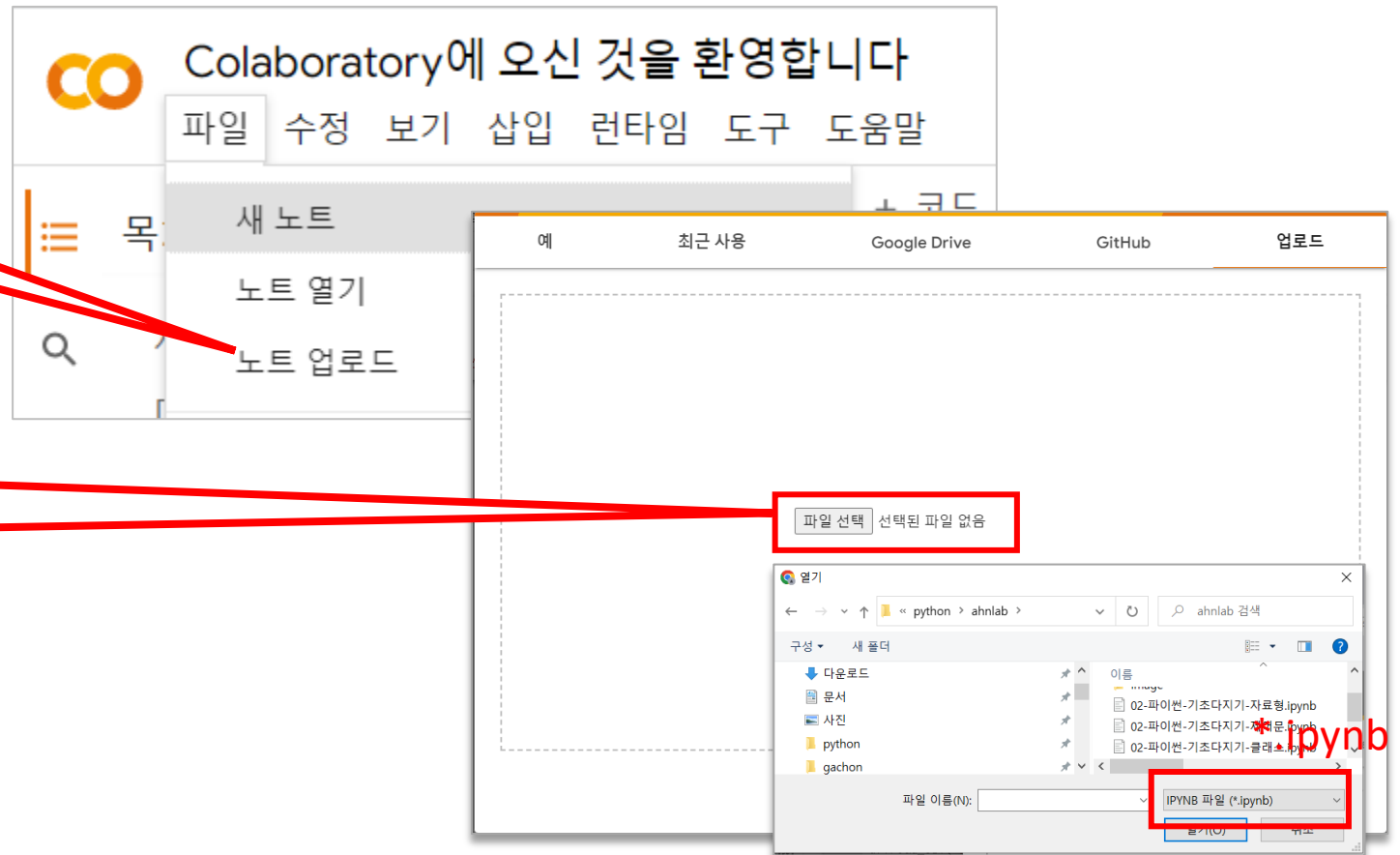


노트 업로드

- 메뉴: 파일 > 노트 업로드

PC에서 작성된 파이썬 코드를 불러오기 위한 메뉴.

[파일선택] 버튼을 눌러서 PC에 있는 파이썬 코드(*.ipynb) 가져오기



실습 : 구글 코랩 사용하기



파이썬 코드 실행하기

- (Cell단위) 파이썬 코드 실행

```
import sys  
  
sys.executable
```

- 파이썬 라이브러리 위치 확인

```
print(sys.path)
```



파이썬 코드 파일

- 파이썬 코드 확장자

일반적인 개발 툴

test.py

주피터 노트북
or 코랩

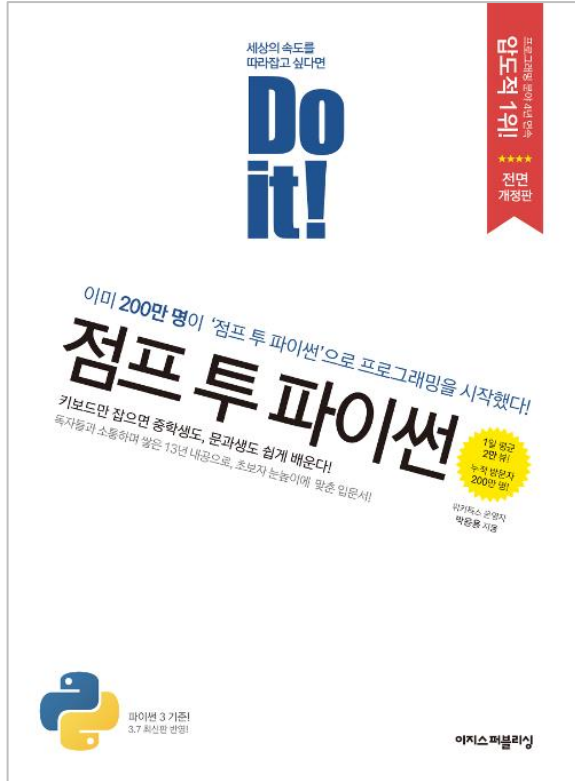
test.ipynb

※jupyter notebook & Colab에서 만든 파이썬 코드 확장자

참고 도서



[참고도서] : 파이썬 기초 문법 이해



<https://wikidocs.net/5>



<https://wikidocs.net/book/922>



<https://wikidocs.net/84366>



초보자용: 본인이 학습하기 제일 좋은 책(사이트)로 학습한다.



[참고도서] : 파이썬 기초 문법 이해



<http://www.yes24.com/Product/Goods/30231768>

중급자를 위한 파이썬

다음 시간에 만나요!