

Chapter 01-2

Exercise environment

Sejong Oh

Exercise environment

- PC
 - Pyhon
 - VS code (IDE for python)
- Cloud
 - Google colab

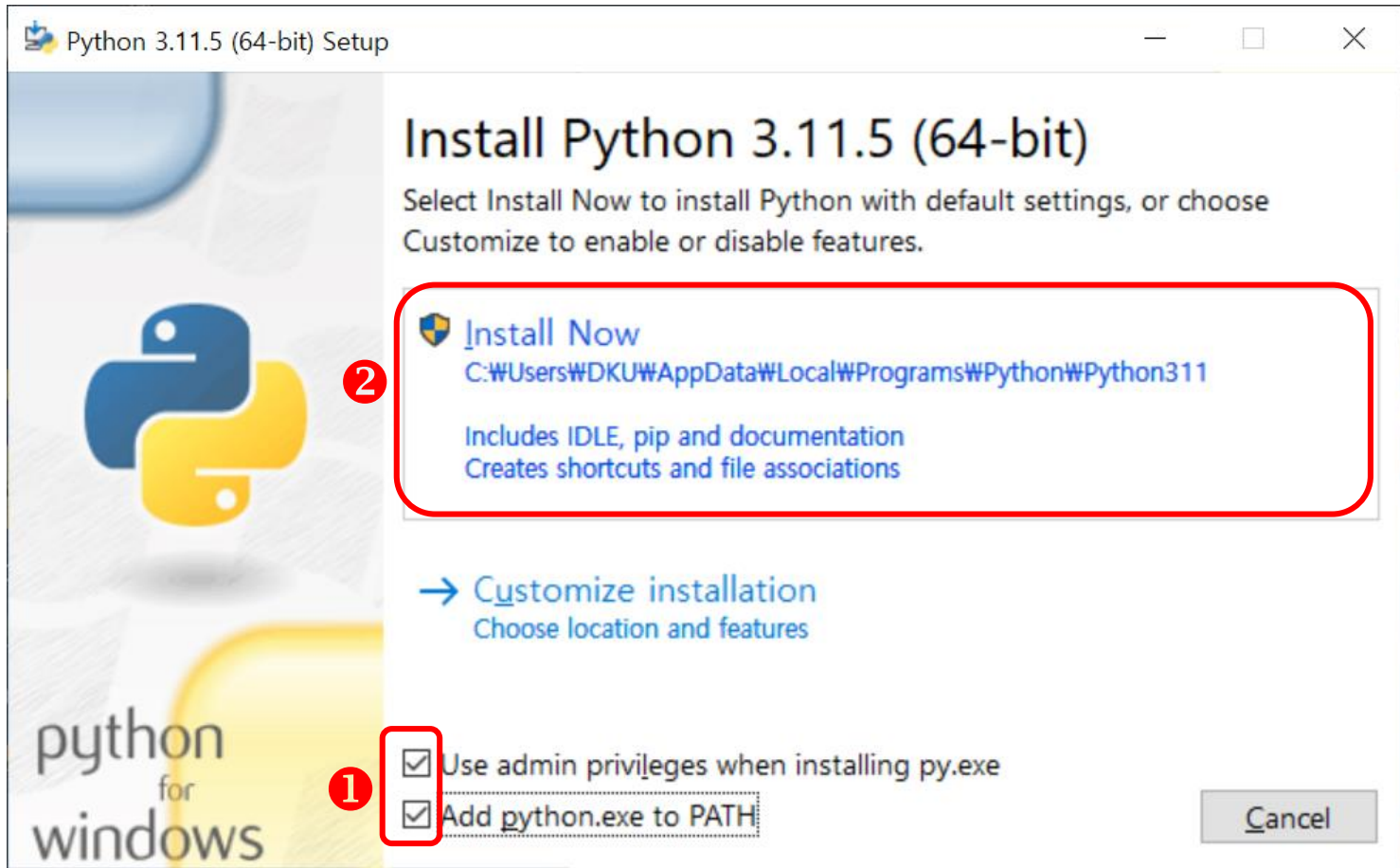
1. Install Python

- 1. Download site
- <https://www.python.org/downloads/>



1. Install Python

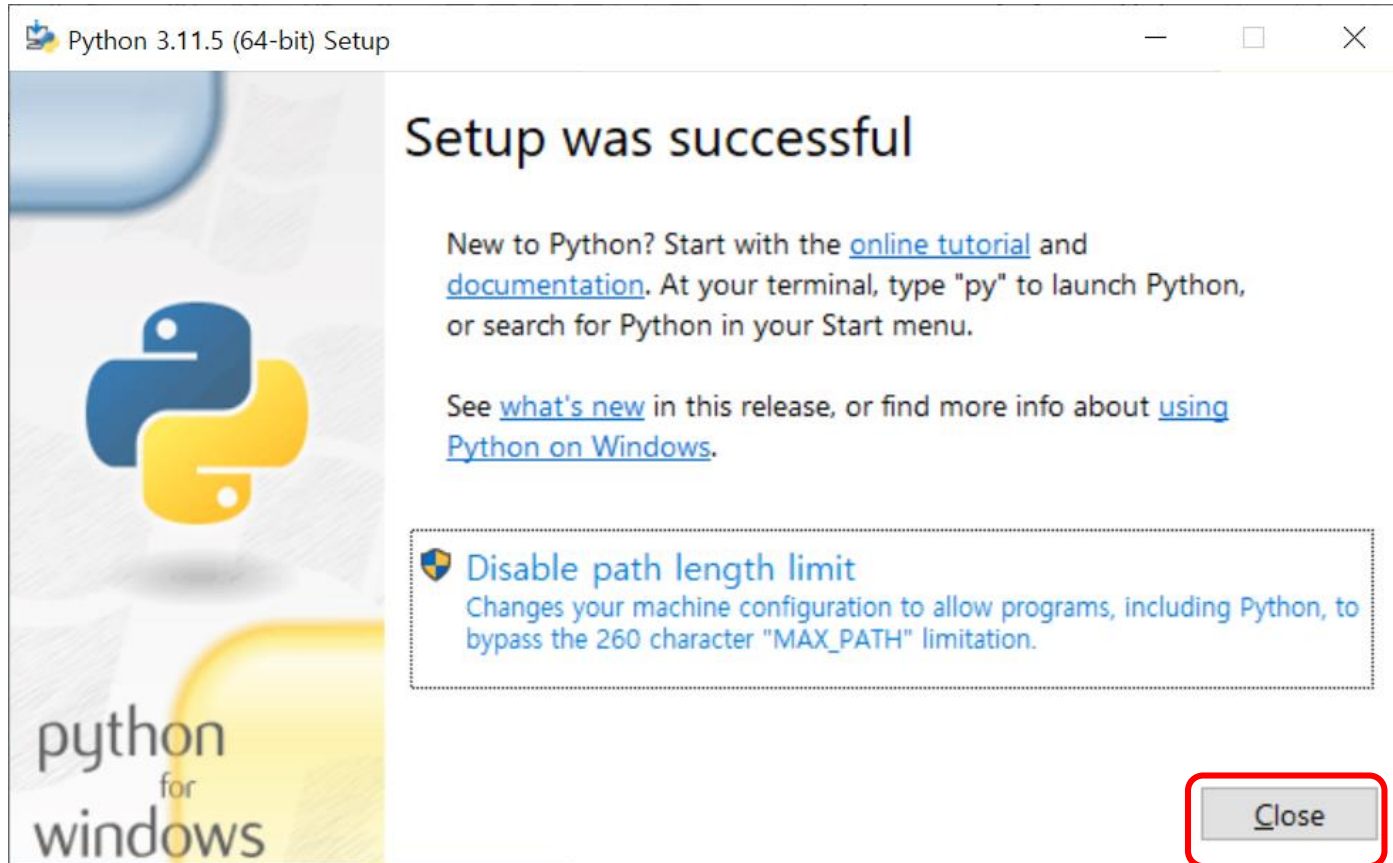
- 2. 다운받은 설치 파일을 실행



중요!!

1. Install Python

- 3. 설치종료






2. VS code : python IDE

- Install
- <https://code.visualstudio.com/download>

Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.



↓ Windows

Windows 8, 10, 11

↓ .deb

Debian, Ubuntu

↓ .rpm

Red Hat, Fedora, SUSE

↓ Mac

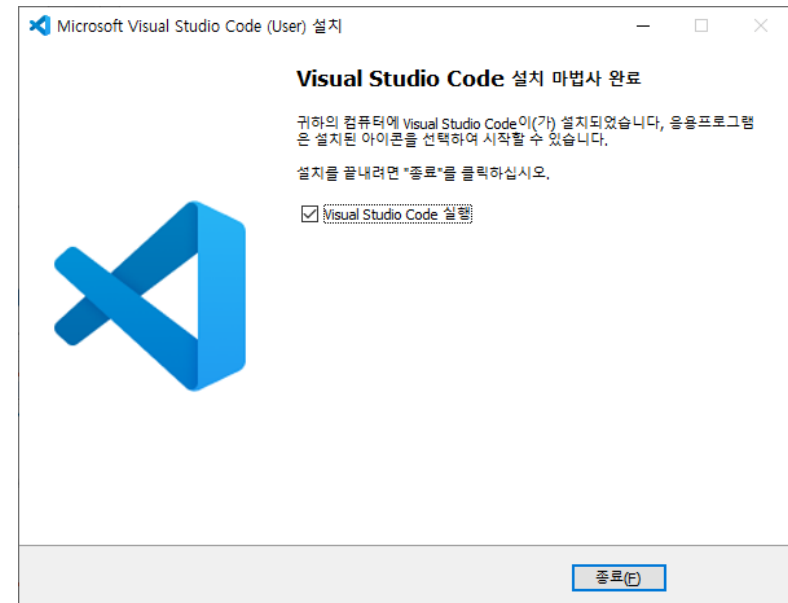
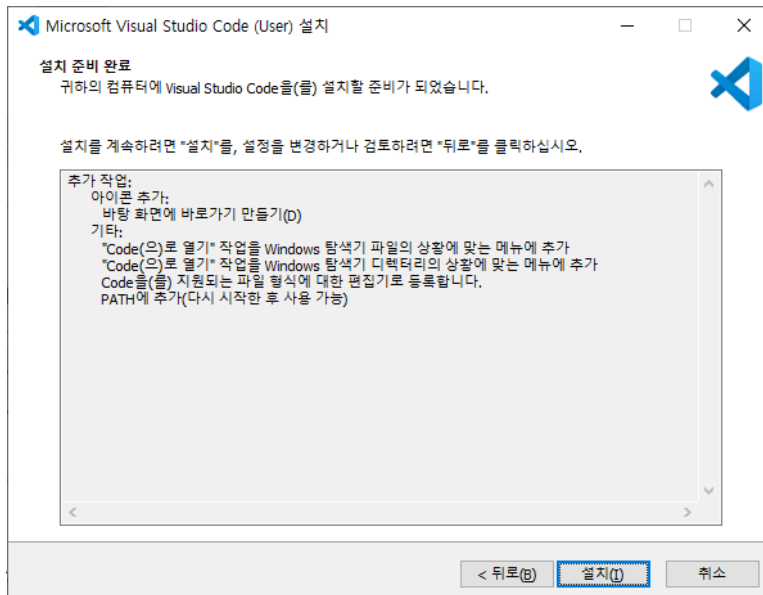
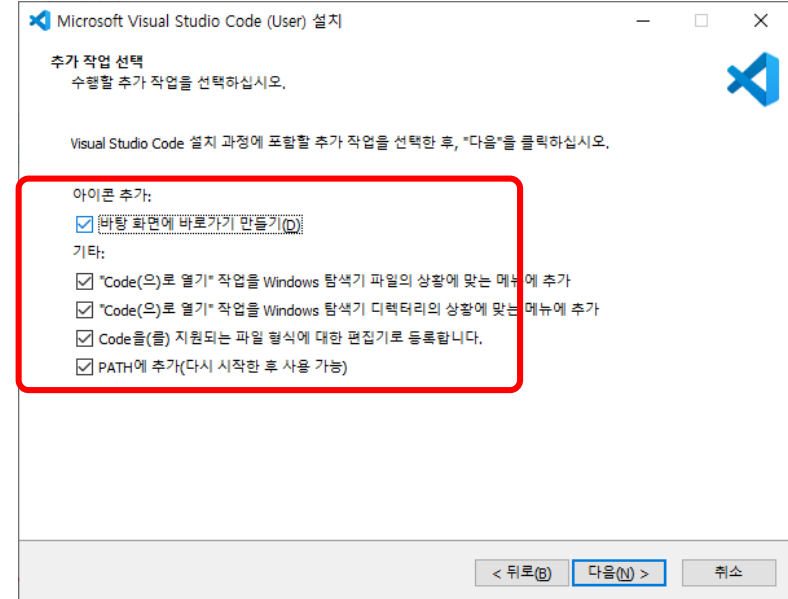
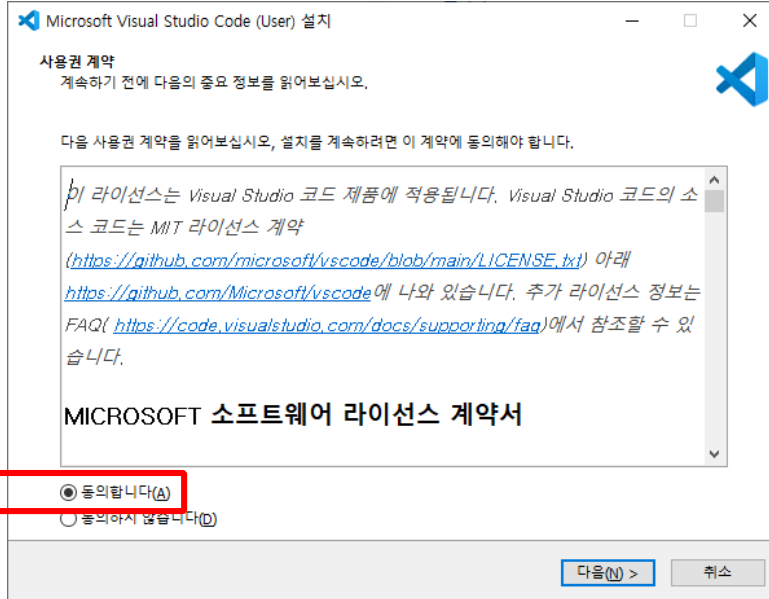
macOS 10.11+

User Installer	x64	x86	Arm64
System Installer	x64	x86	Arm64
.zip	x64	x86	Arm64
CLI	x64	x86	Arm64

.deb	x64	Arm32	Arm64
.rpm	x64	Arm32	Arm64
.tar.gz	x64	Arm32	Arm64
Snap	Snap Store		

.zip	Intel chip	Apple silicon	Universal
CLI	Intel chip	Apple silicon	

2. VS code : python IDE

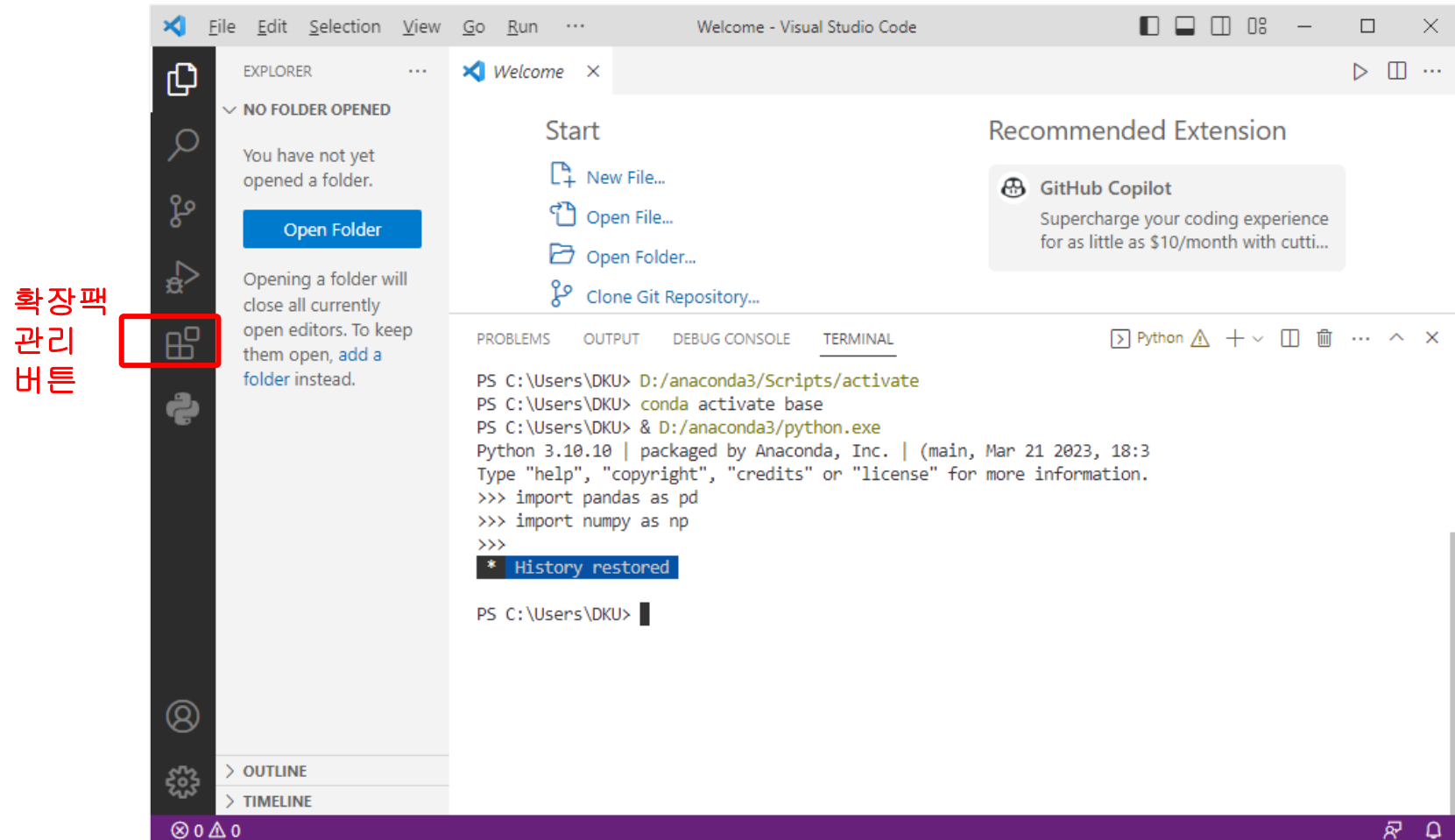


2. VS code : python IDE

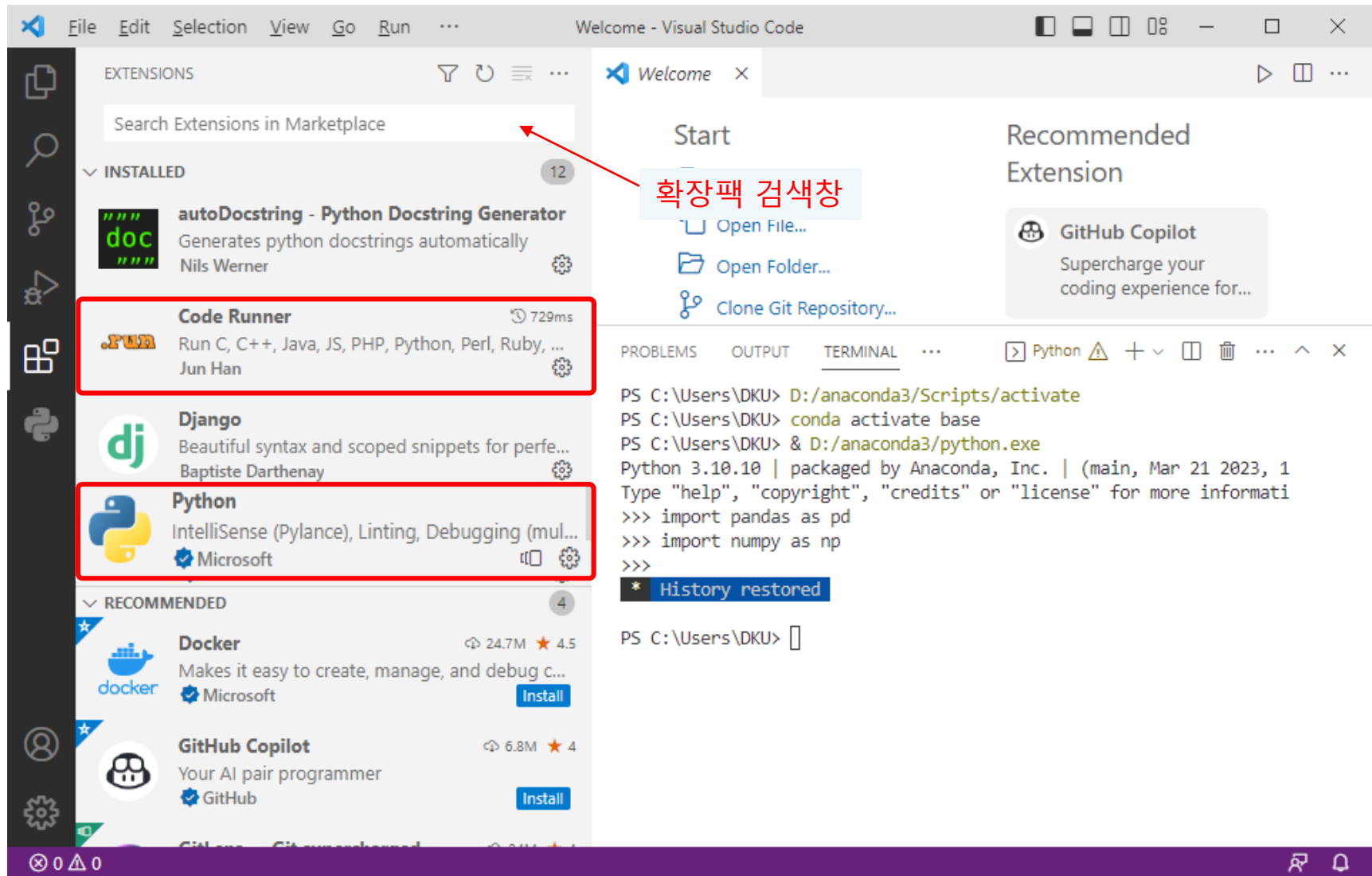
- Python, VS code 설치 후 시스템을 재시작 한다

2. VS code : python IDE

- Python 을 이용하기 위한 확장 팩 설치



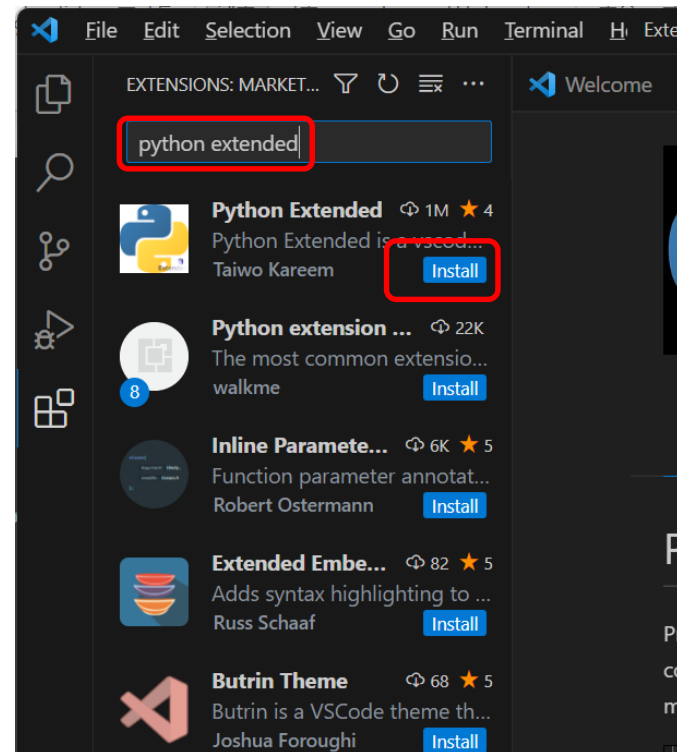
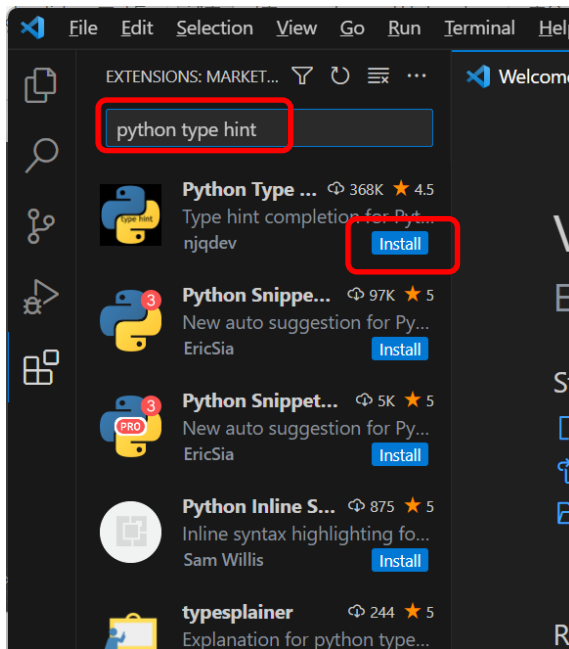
2. VS code : python IDE



- Installed 목록에 code runner, python 이 있는지 확인
- 없으면 검색창에서 검색하여 설치 후 VS code 재시작

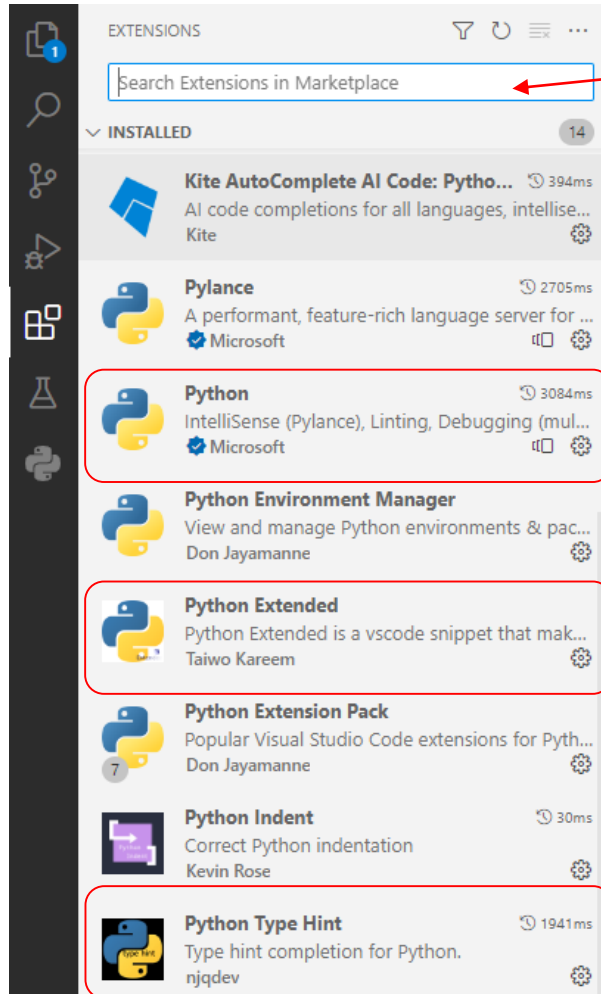
2. VS code : python IDE

- 파이썬을 편리하게 사용하기 위한 추가 확장팩 설치
 - python type hint
 - python extended



2. VS code : python IDE

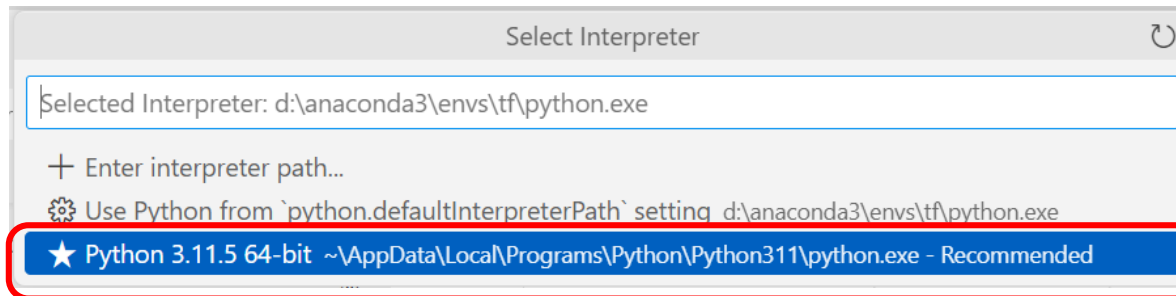
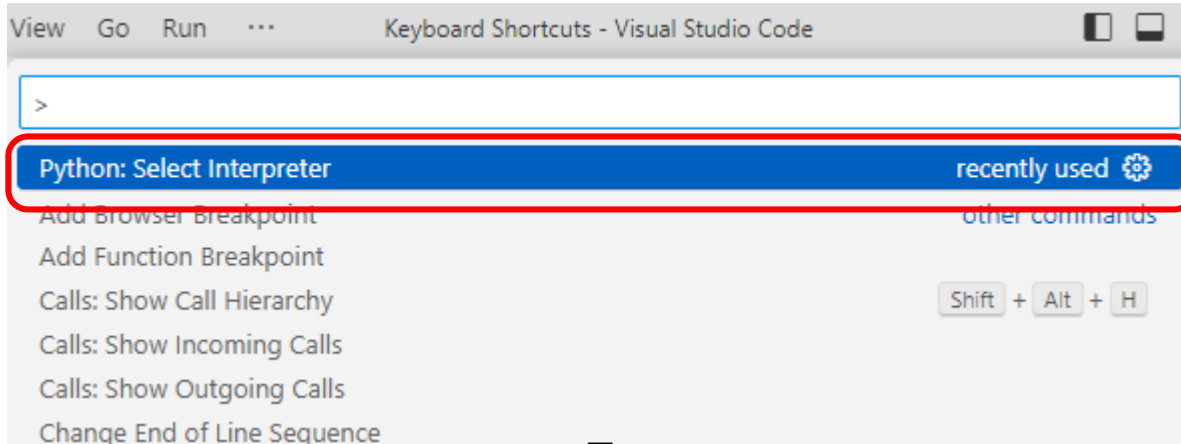
- 설치된 확장팩 확인
 - 검색어 부분을 지우면 다음과 같이 추가 설치 요소들을 확인 할수 있음



여기를 공백으로 만든다.

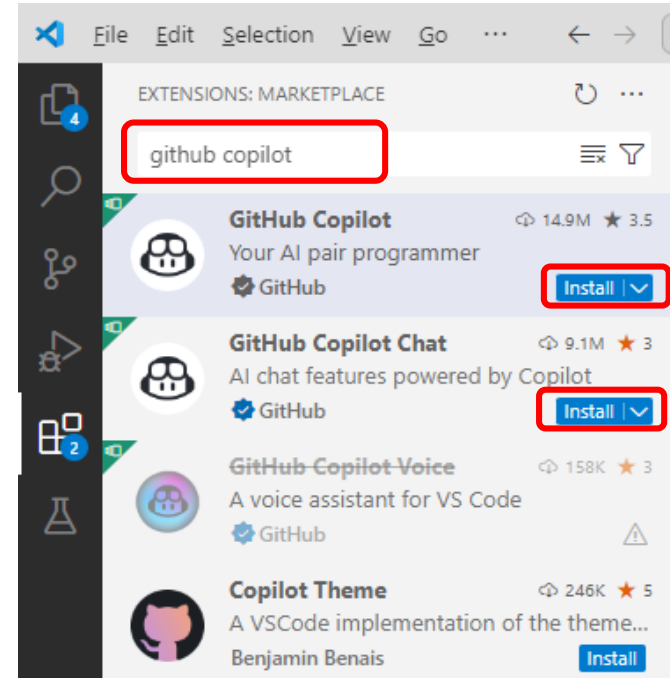
2. VS code : python IDE

- VS code를 파이썬 편집기로 설정하기
- <Ctrl> + <Shift> + <p> 후 Python: Select interpreter 클릭



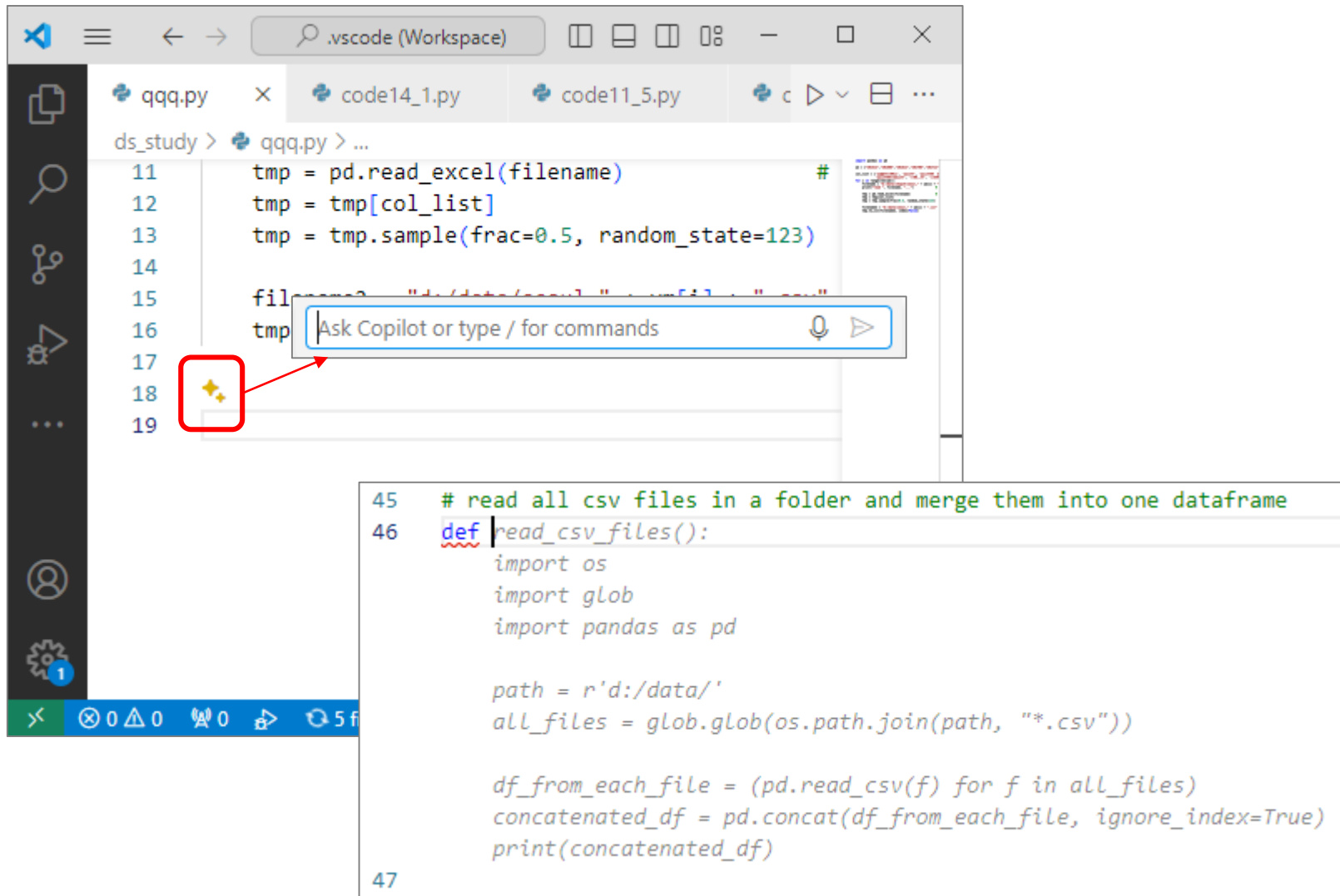
2. VS code : python IDE

- Github Copilot 확장팩
 - 깃허브(github)에서는 파이썬 코딩을 도와주는 인공지능 비서인 코파일럿(copilot)을 출시
 - 파이썬 외에도 C++, Python, Go, JS/TS Ruby, Swift, Java, Kotlin, Rust, PHP등 다양한 언어를 지원
 - 코파일럿을 잘 활용하면 개발 시간을 획기적으로 단축
 - 비용을 지불해야 하는 유료 서비스이지만 교육기관 종사자(학생, 교강사)를 위해 무료 서비스를 지원



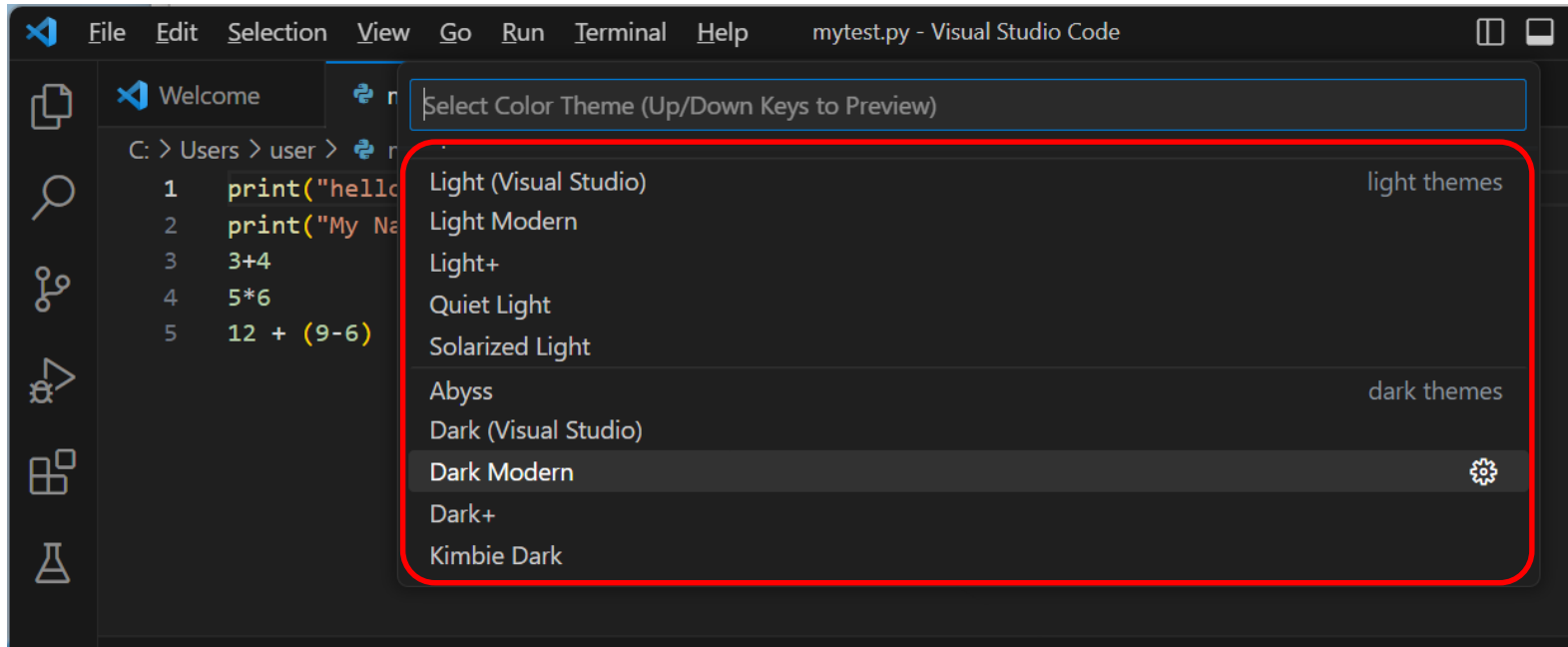
2. VS code : python IDE

- Github Copilot 확장팩



2. VS code : python IDE

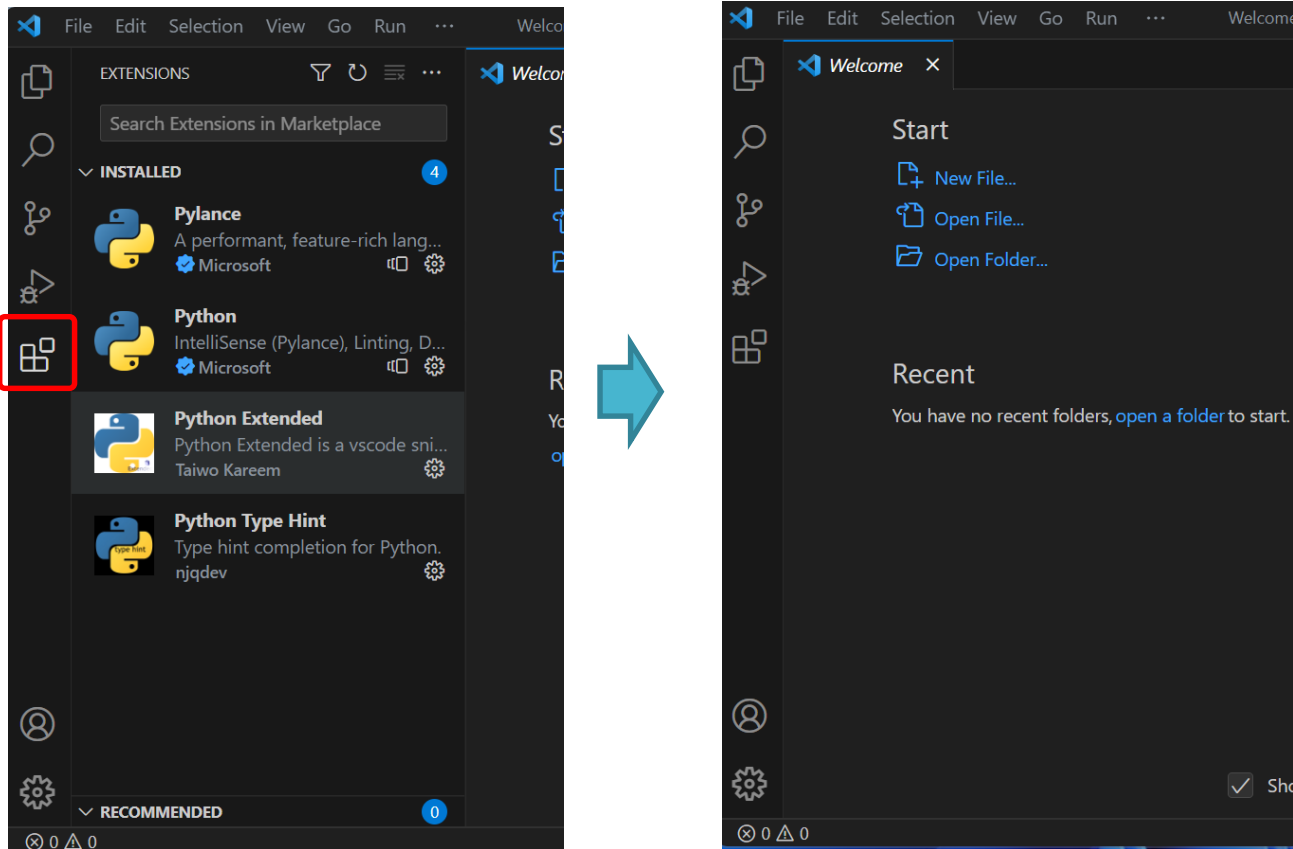
- <Note> 배경 테마 바꾸기
 - [File]→[Preferences] →[Theme] →[Color Themes]



기본 테마는 Dark Modern
흰색 바탕 테마는 Light (Visual Studio) 추천

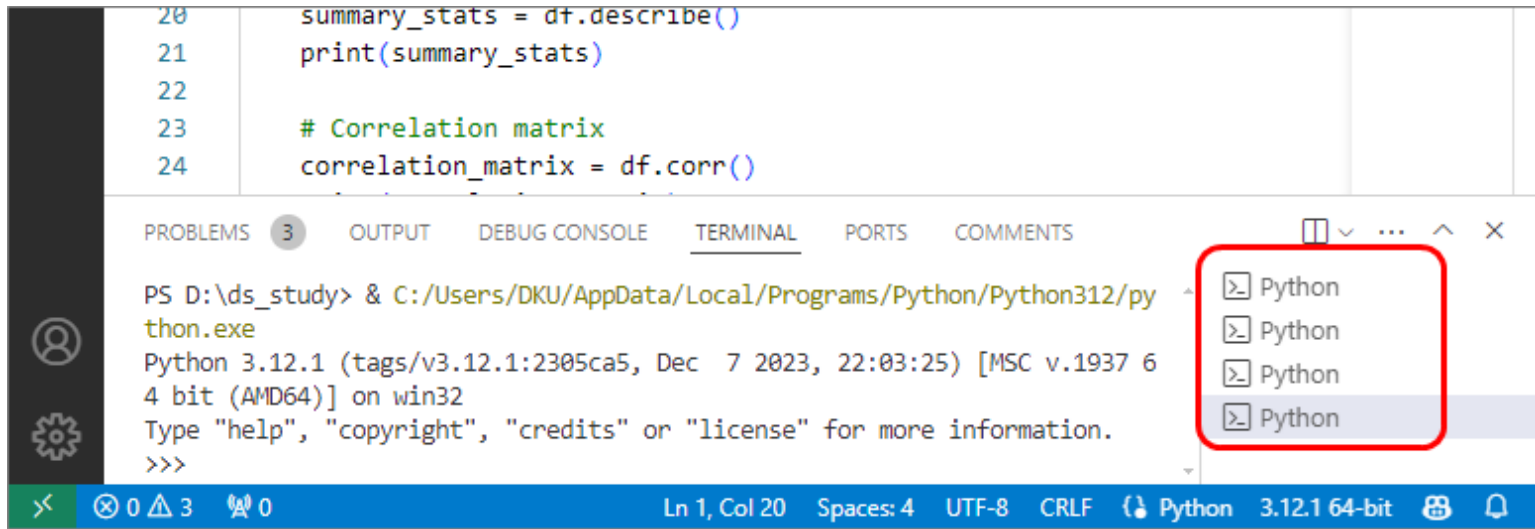
2. VS code : python IDE

- 모든 확장 팩 설치 및 환경 설정을 마쳤으므로 아래 버튼을 클릭하여 확장 팩 설치창 부분을 닫는다



2. VS code : python IDE

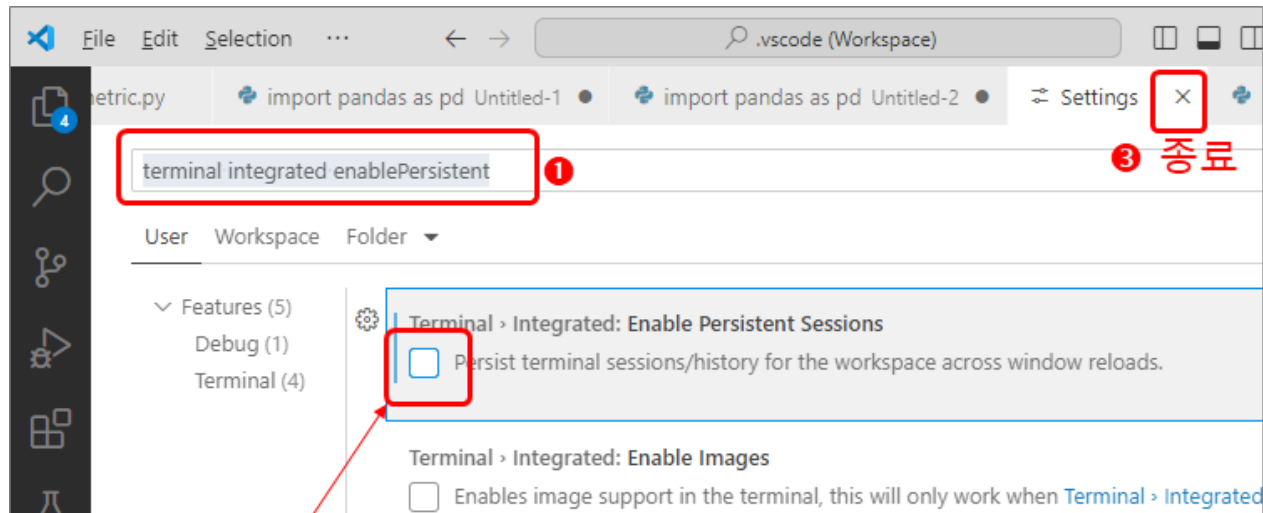
- VS Code의 이전 세션 연결 끊기
 - VS Code 는 작업을 종료하여도 이전 작업에서 생성한 세션정보를 계속 유지
 - 그런데 VS Code 를 재시작하면 이전 세션을 이용하지 않고 새로운 세션을 생성
 - 따라서 VS Code 의 종료와 재시작을 반복하다 보면 아래와 같이 터미널 세션이 여러 개 생겨난다.
 - 이를 방지하기 위해 설정 변경 필요



2. VS code : python IDE

- VS Code의 이전 세션 연결 끊기

- ① 메인 메뉴에서 [File] → [Preferences] → [Settings]를 선택한다.
- ② 검색창에서 'terminal integrated enable persistent'를 입력하면 나오는 Terminal Integrated Enable Persistent Sessions 항목의 선택을 해제한다.
- ③ 설정값 변경 작업을 종료한다.

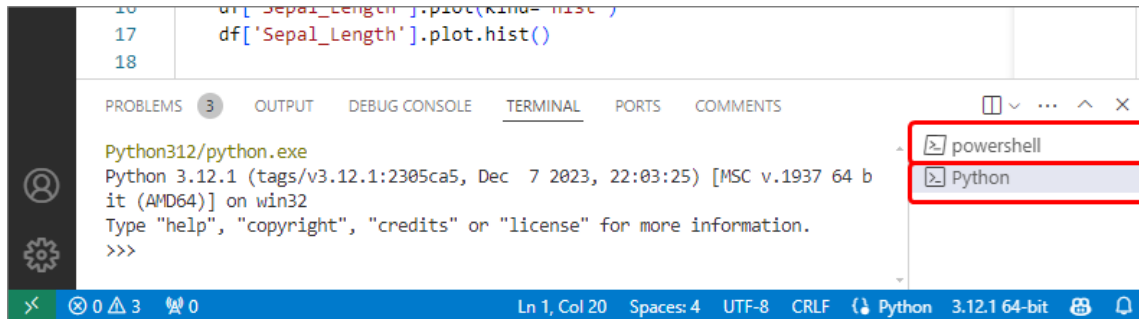


② 선택해제

2. VS code : python IDE

- [note]

- 결과창의 터미널 모드는 윈도우 명령 프롬프트 모드와 ipython 모드의 상호 변환이 가능
- 파이썬 명령문의 아무 곳에서나 <Shift> + <Enter>를 누르면 아래와 같이 명령 프롬프트 모드와 ipython 프롬프트가 표시되는데, 이곳을 클릭하면 원하는 모드로의 변경이 가능

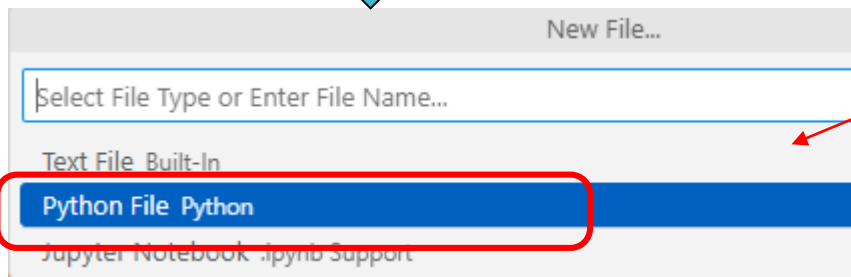
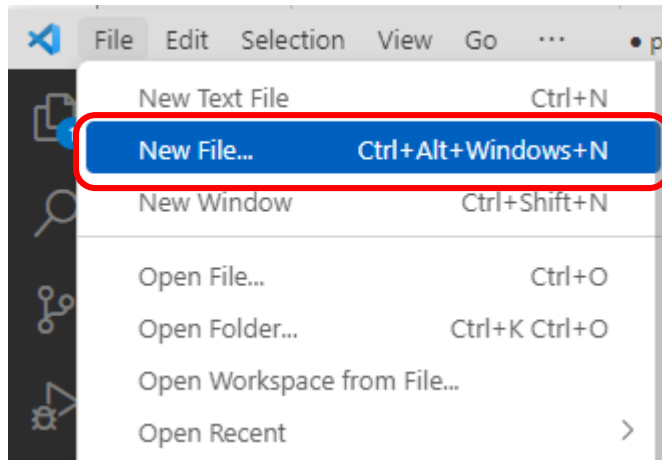


명령 프롬프트

ipython 프롬프트

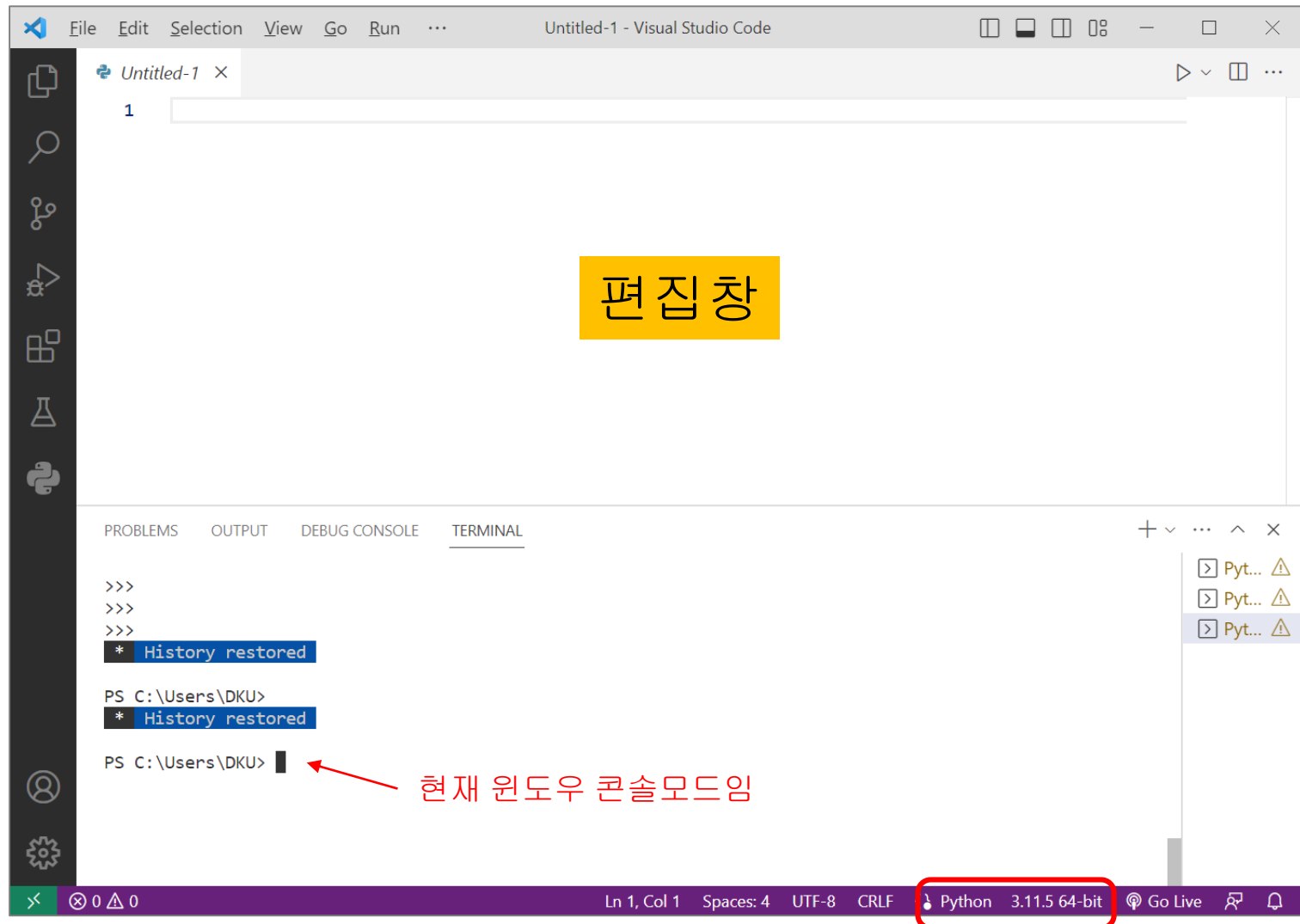
2. VS code : python IDE

- 편집 창 열기



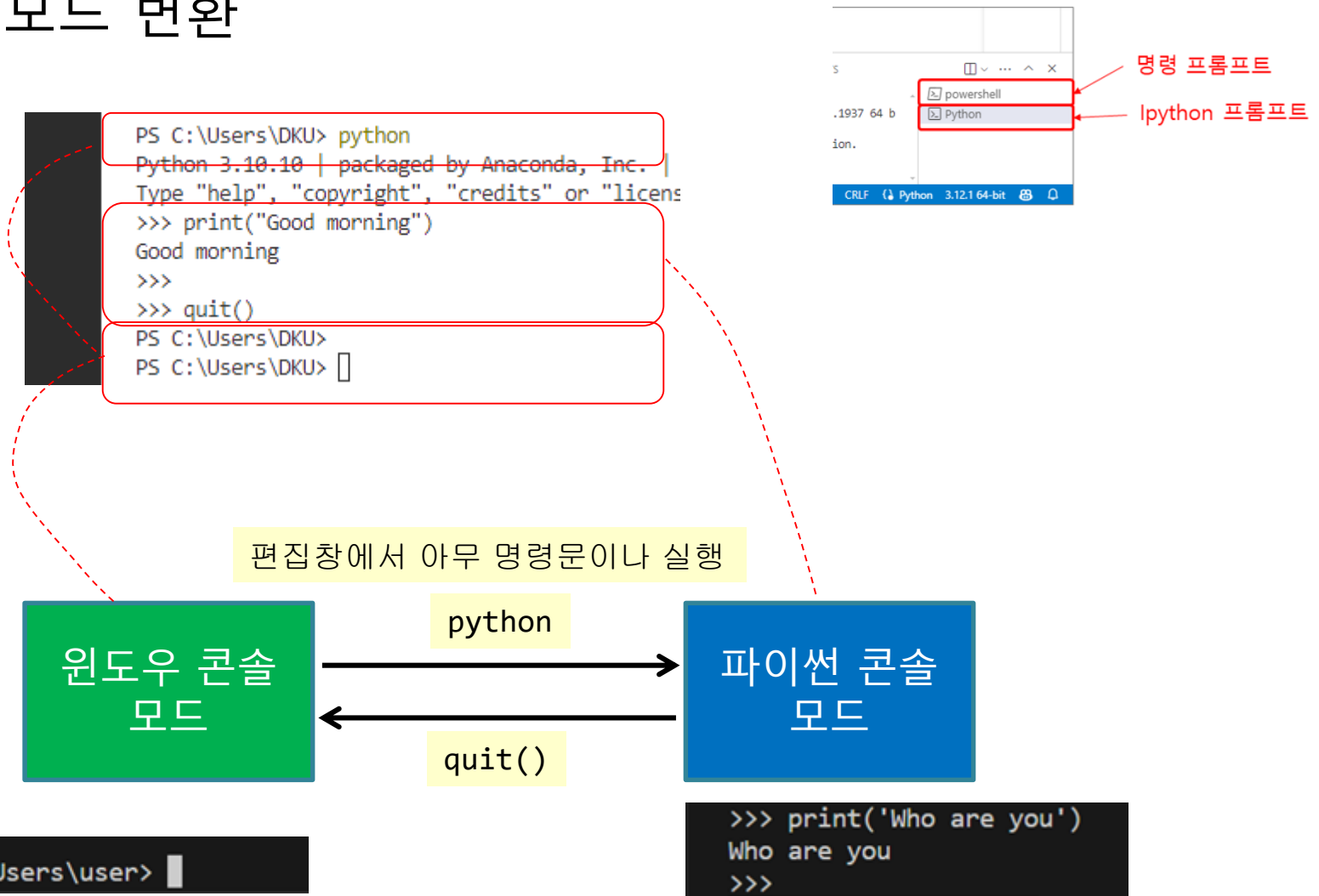
<주의>
Text File 을 선택하면
파이썬 명령문이
실행되지 않음

2. VS code : python IDE



2. VS code : python IDE

- 터미널 모드 변환

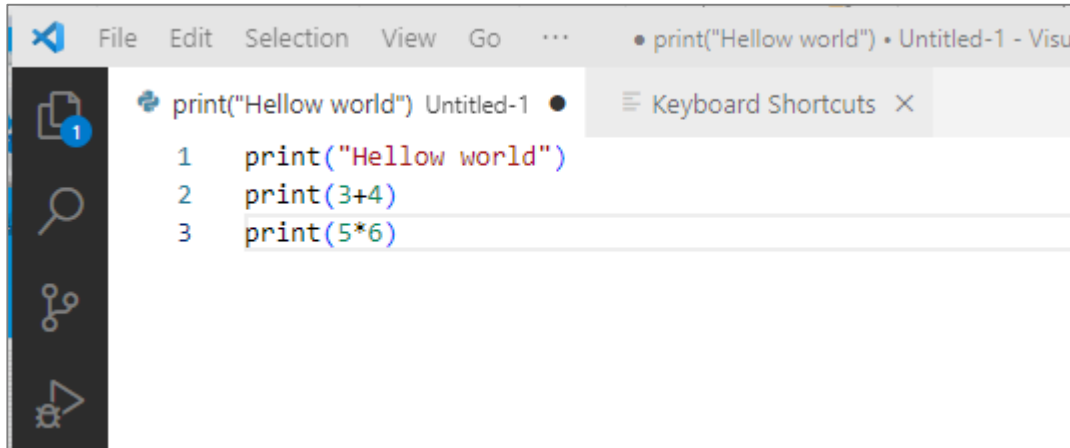


파이썬에 필요한 패키지를 설치할 수 있다. (pip install)

파이썬 명령문을 실행할 수 있다.

2. VS code : python IDE

- 프로그램의 작성

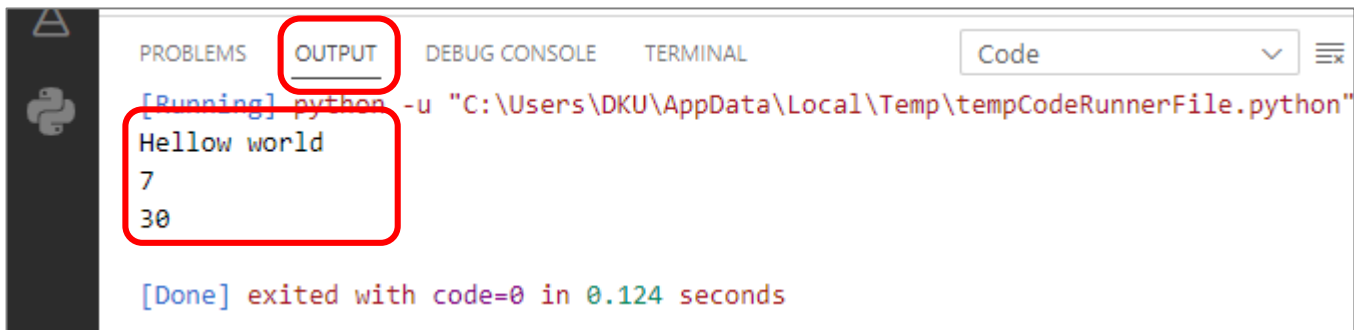
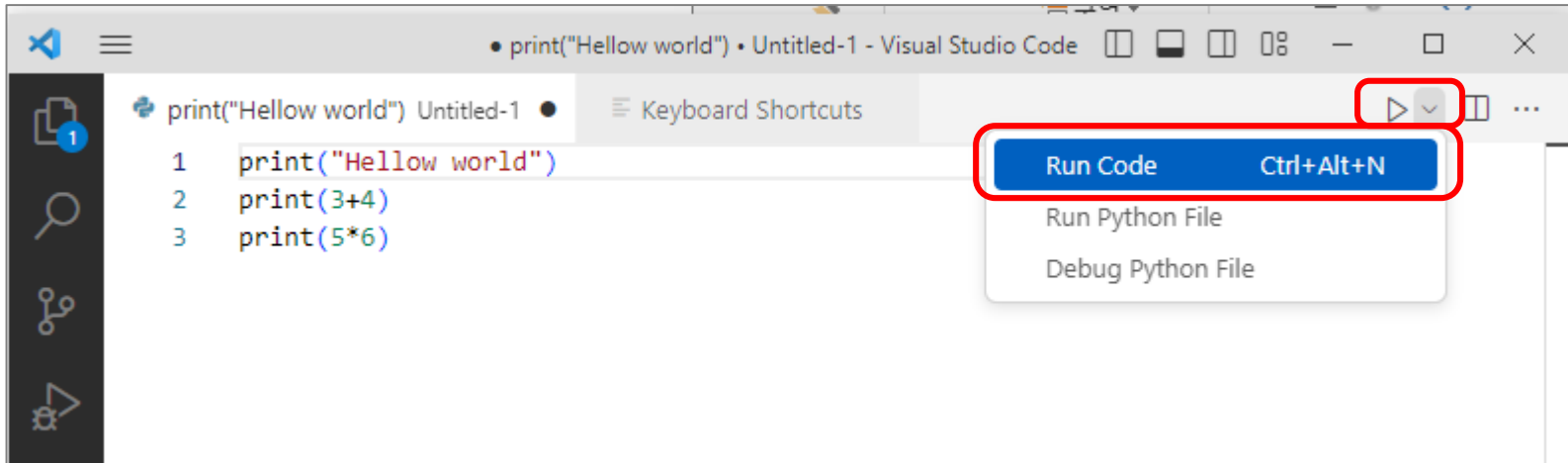


```
print("Hellow world")  
1 print("Hellow world")  
2 print(3+4)  
3 print(5*6)
```


2. VS code : python IDE

- 프로그램의 실행

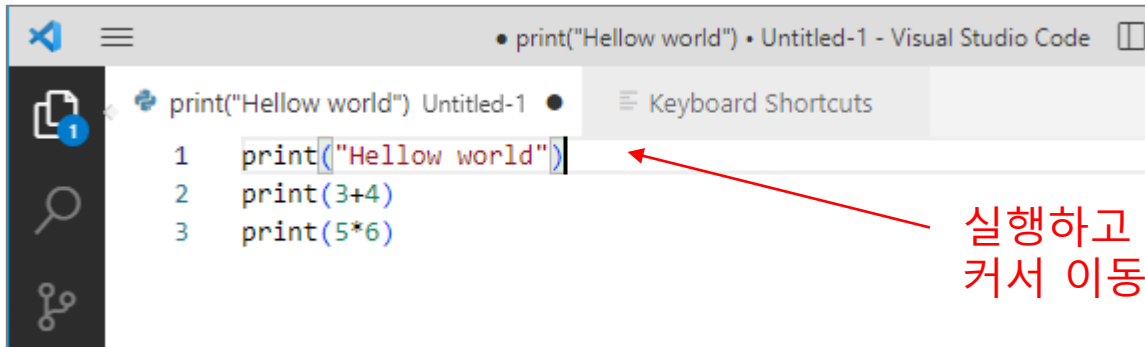
- 현재 편집창에 작성한 **전체** 명령문 실행 (<ctrl> + <Alt> + <N>)



실행 결과가 output 탭에 표시됨

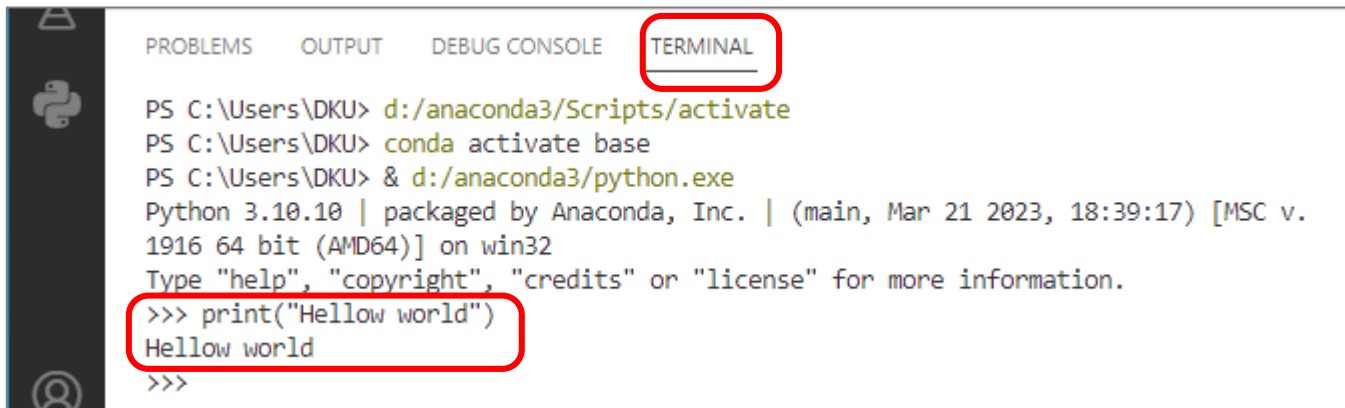
2. VS code : python IDE

- 프로그램의 실행
 - 한번에 **한줄씩** 실행하기(<Shift> + <Enter>)



```
print("Hellow world")
print(3+4)
print(5*6)
```

실행하고 싶은 명령문 라인으로
커서 이동 후 <Shift> + <Enter>

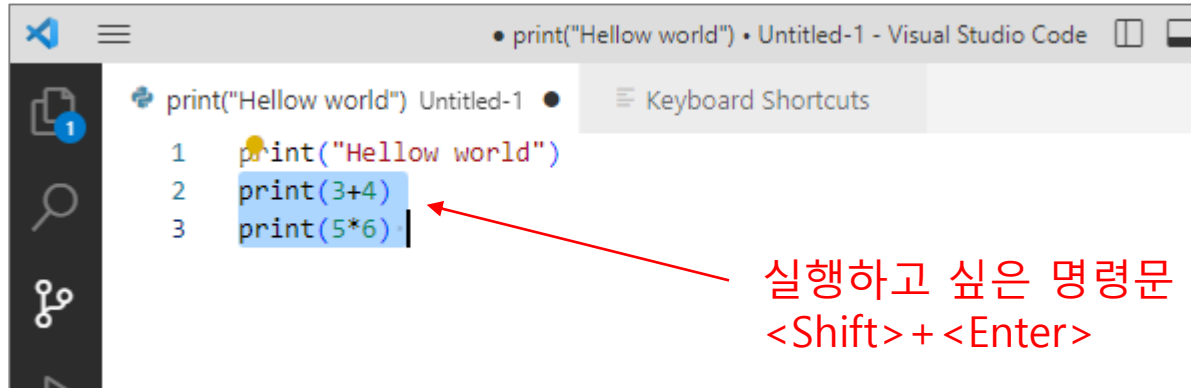


```
PS C:\Users\DKU> d:/anaconda3/Scripts/activate
PS C:\Users\DKU> conda activate base
PS C:\Users\DKU> & d:/anaconda3/python.exe
Python 3.10.10 | packaged by Anaconda, Inc. | (main, Mar 21 2023, 18:39:17) [MSC v.
1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hellow world")
Hellow world
>>>
```

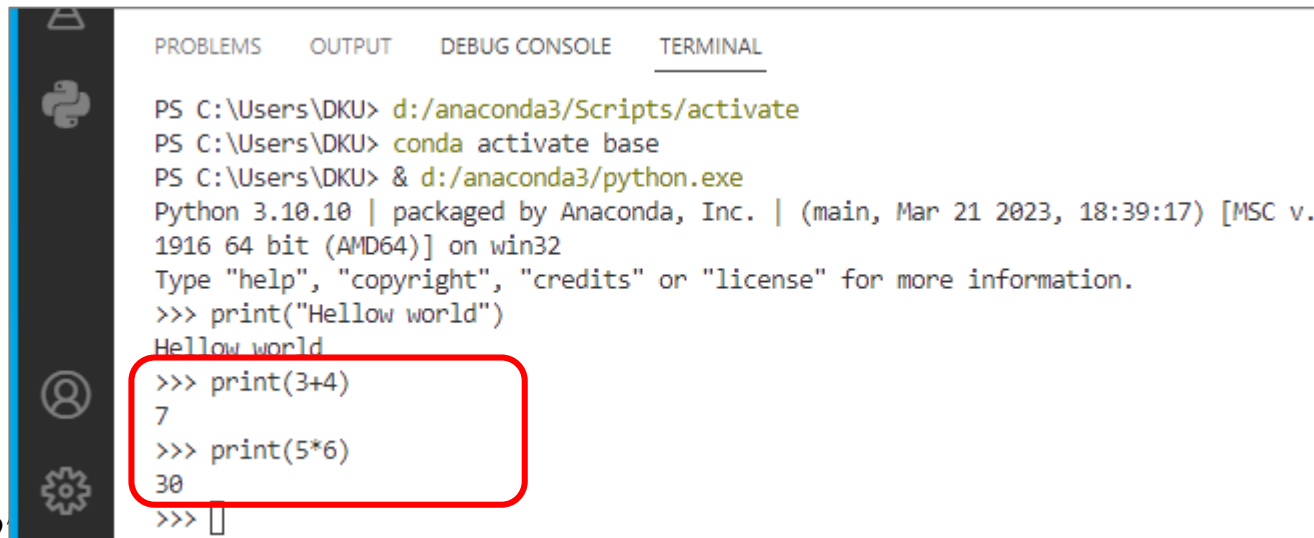
2. VS code : python IDE

- 프로그램의 실행

- 한번에 여러 줄(블록) 실행하기(<Shift> + <Enter>)



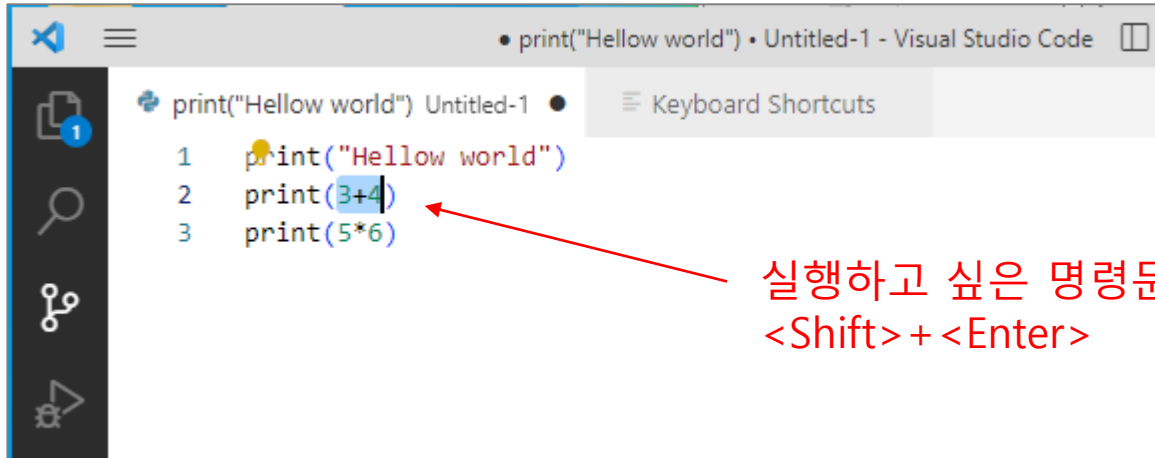
```
print("Hellow world")
print(3+4)
print(5*6)
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\DKU> d:/anaconda3/Scripts/activate
PS C:\Users\DKU> conda activate base
PS C:\Users\DKU> & d:/anaconda3/python.exe
Python 3.10.10 | packaged by Anaconda, Inc. | (main, Mar 21 2023, 18:39:17) [MSC v.
1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hellow world")
Hellow world
>>> print(3+4)
7
>>> print(5*6)
30
>>>
```

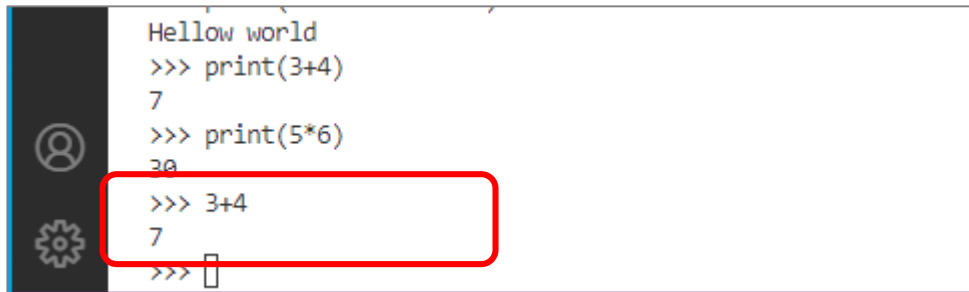
2. VS code : python IDE

- 프로그램의 실행
 - 명령문의 **일부분** 실행(<Shift> + <Enter>)



```
print("Hellow world")
1 print("Hellow world")
2 print(3+4)
3 print(5*6)
```

실행하고 싶은 명령문의 일부분 선택 후
<Shift> + <Enter>

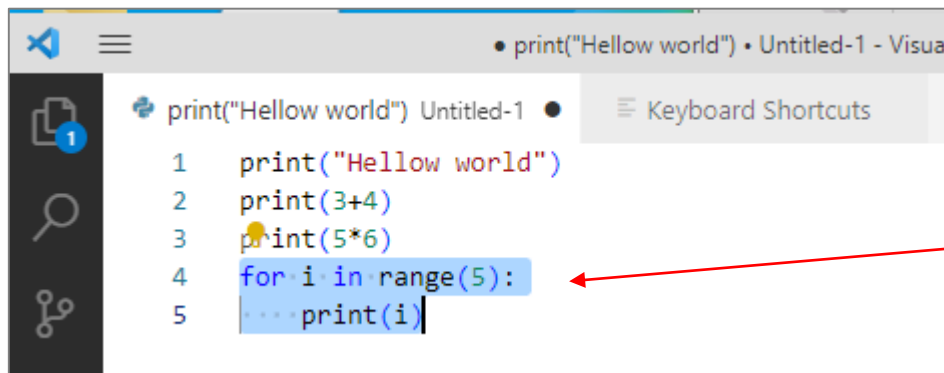


```
Hellow world
>>> print(3+4)
7
>>> print(5*6)
30
>>> 3+4
7
>>> 
```

2. VS code : python IDE

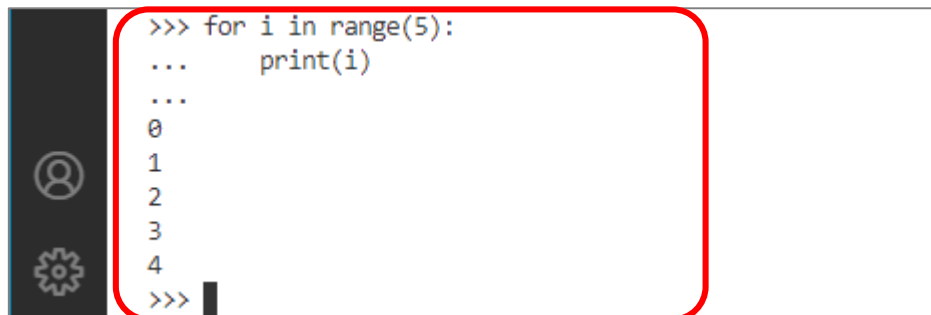
- 프로그램의 실행

- 주의: 하나의 명령문이 여러줄에 걸쳐 작성된 경우 (ex: if문, for문) 전체 명령문을 모두 선택 후 실행해야 에러가 발생하지 않는다.



```
print("Hellow world") Untitled-1 • Keyboard Shortcuts
1 print("Hellow world")
2 print(3+4)
3 print(5*6)
4 for i in range(5):
5     print(i)
```

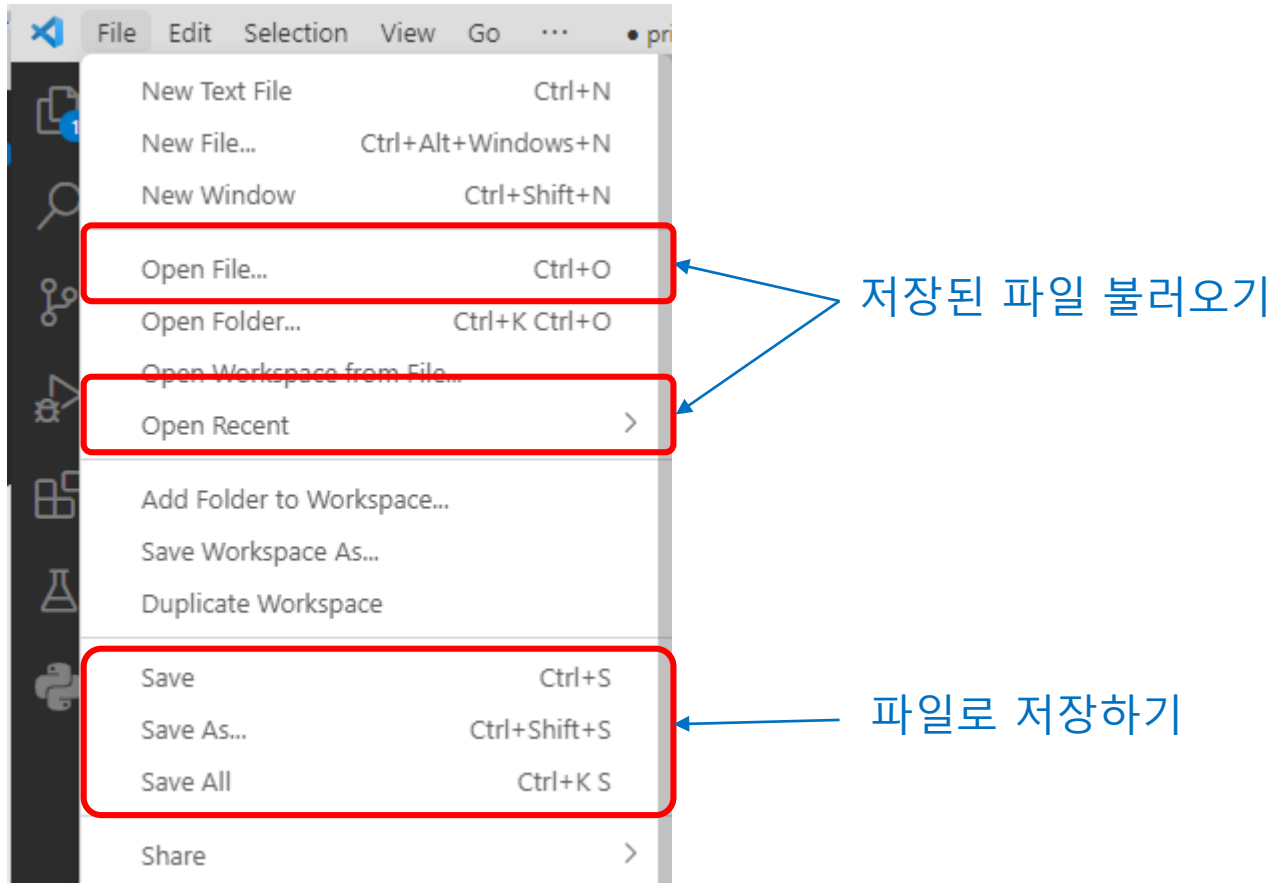
한번에 한줄씩 실행할 수 없음



```
>>> for i in range(5):
...     print(i)
...
0
1
2
3
4
>>>
```

2. VS code : python IDE

- 프로그램의 저장& 불러오기



2. VS code : python IDE

- VS code의 재설치
 - 정상적으로 작동되던 VS code에 문제가 생기는 경우가 종종 있음
 - 단순히 SW 를 uninstall 한 후 다시 install 하는 것으로는 문제가 해결되지 않음
 - 재설치 필요시 다음의 게시글 참조.
 - <https://citylock77.tistory.com/142>
 - 수업 도중 문제가 생기면 일단 colab으로 실습을 하고 수업후 문제를 해결한다.

2. VS code : python IDE

- pip 실행 오류시

보통은 이렇게 하면 정상 실행

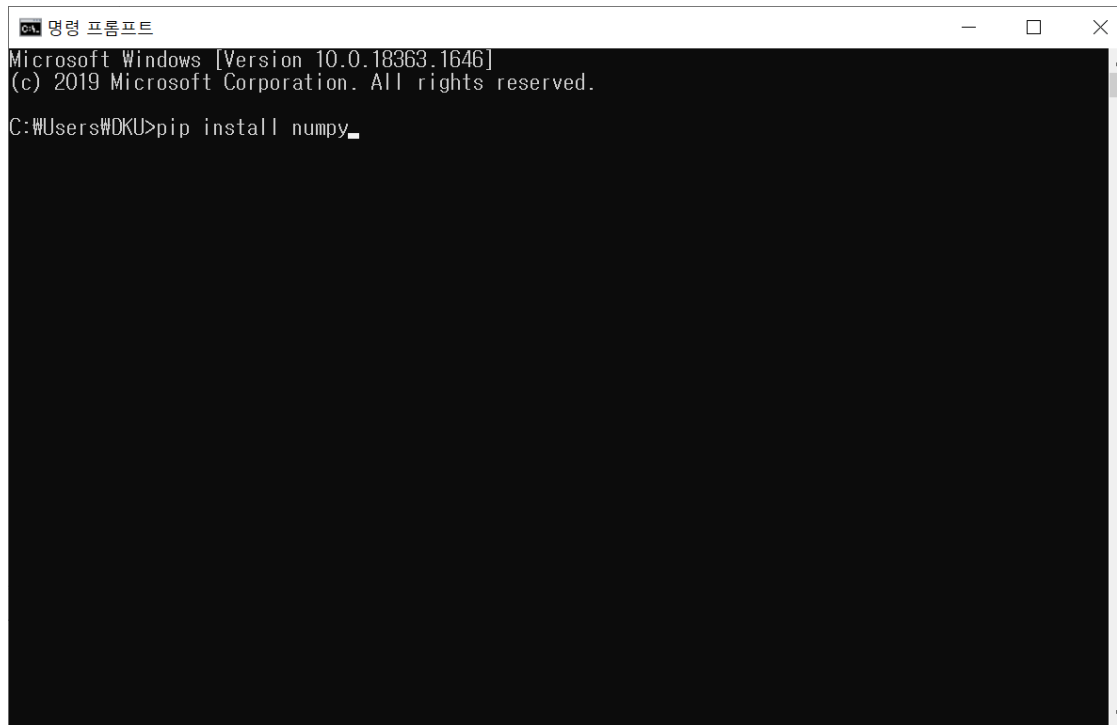
```
PS C:\Users\DKU> pip list  
Fatal error in launcher: Unable to create process using '"c:\users\dku\python37\python.exe"' "c:\users\dku\python37\Scripts\pip.exe" list': ??? ??? ?? ? ????.
```

```
PS C:\Users\DKU> python -m pip list  
Package      Version  
-----  
pip          23.2.1  
setuptools   65.5.0  
PS C:\Users\DKU> _
```

오류 발생시 pip 앞에 python -m 을 붙인다.

2. VS code : python IDE

- VS code 외부에서 package 설치하기
 - Window 명령 프롬프트에서도 설치 가능



```
명령 프롬프트
Microsoft Windows [Version 10.0.18363.1646]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\WDKU>pip install numpy_
```

2. VS code : python IDE

- 사전에 설치할 패키지 리스트
 - numpy
 - pandas
 - matplotlib
 - Seaborn
 - scikit-learn

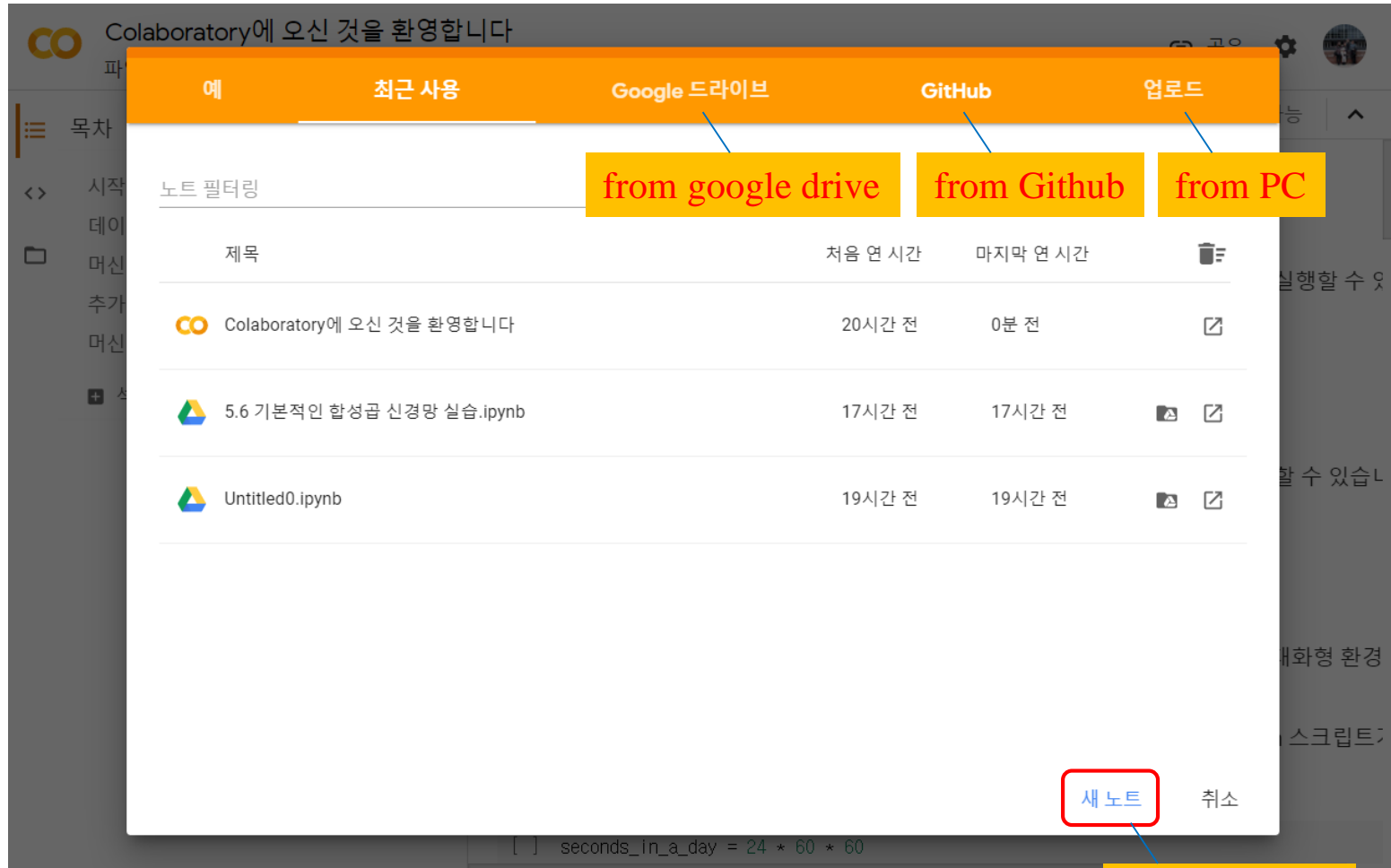
3. Google colab (cloud)

- Google product
 - provide **Jupyter Notebooks** in the browser — albeit with their own unique flavors
 - offer free **GPUs**
 - are designed to foster collaboration for machine learning.
 - are imperfect, but are pretty useful in many situations — particularly when you are starting out in deep learning.
 - don't provide great info on their hardware specs
-
- Kaggle supports similar service

Cloud Service	NVIDIA GPU	CUDA Version	GPU RAM (GB)	CPU Chip	Chip Speed (GHz)	CPU Cores	CPU RAM (Total GB)	L3 Cache (MB)	Disk Space (Total GB)	Max Idle Time (hrs)	Max Session Time (hrs)	Max Commit Time (hrs)
Colab	Tesla K80	10.0	12.0	Intel Xeon CPU	2.2 or 2.3	2	13.3	56	359	1.5	12	n/a
Kaggle	Tesla P100	9.2	17.1	Intel Xeon CPU	2.2 or 2.3	2	16.4	46	220	1	9	6
Command	<code>!nvidia-smi</code>	<code>!cat /usr/local/cuda/version.txt</code>	<code>!nvidia-smi</code>	<code>!cat /proc/cpuinfo</code>	<code>!cat /proc/cpuinfo</code>	<code>multiprocessing.cpu_count()</code>	<code>!cat /proc/meminfo</code>	<code>!cat /proc/cpuinfo</code>	<code>!df -h</code>	<code>docs</code>	<code>docs</code>	<code>docs</code>

3. Google colab

- <https://colab.research.google.com/>



Requires google account !

3. Google colab

The screenshot shows the Google Colab web interface. At the top, the title bar displays the Colab logo, the file name 'Untitled9.ipynb', and a star icon. Below this is a menu bar with options: '파일' (File), '수정' (Edit), '보기' (View), '삽입' (Insert), '런타임' (Runtime), '도구' (Tools), '도움말' (Help), and '모든 변경사항이 저장됨' (All changes are saved). On the right side of the title bar are icons for '댓글' (Comments), '공유' (Share), and '설정' (Settings). Below the title bar, there are tabs for '+ 코드' (Code) and '+ 텍스트' (Text). The main workspace shows a code cell with the text '3+4' and a play button icon. Below the code cell, the output '7' is displayed. On the right side of the workspace, there are status indicators for 'RAM' and '디스크' (Disk) usage, along with a vertical expand/collapse arrow. A toolbar at the bottom right of the workspace contains icons for undo, redo, link, comment, settings, copy, and delete. Three yellow callout boxes with red text provide instructions: 'Set GPU' points to the '런타임' menu; '메뉴 보이기/숨기기' (Show/Hide menu) points to the expand/collapse arrow; and 'Run code (<ctrl>+<Enter>)' points to the play button icon.

Set GPU

메뉴 보이기/숨기기

Run code
(<ctrl>+<Enter>)

3. Google colab

- GPU 사용 설정

The image shows the Google Colab interface. At the top, the 'Run' menu is open, with '런타임' (Runtime) highlighted. Below it, the '런타임 유형 변경' (Change runtime type) option is selected. An arrow points to the '노트 설정' (Notebook Settings) dialog. In this dialog, the '하드웨어 가속기' (Hardware accelerator) dropdown is set to 'GPU', which is highlighted with a red box. The '저장' (Save) button is also highlighted with a red box.

Colaboratory에 오신 것을 ...
파일 수정 보기 삽입 런타임 도구 도움말 변경사항을 저장할 수

목차

- 시작하기
- 데이터 과학
- 머신러닝
- 추가 리소스
- 추천 예시
- 새 섹션
- 섹션

모두 실행 Ctrl+F9
이전 셀 실행 Ctrl+F8
초점이 맞춰진 셀 실행 Ctrl+Enter
선택항목 실행 Ctrl+Shift+Enter
이후 셀 실행 Ctrl+F10
실행 중단 Ctrl+M I
런타임 다시 시작 Ctrl+M .
다시 시작 및 모두 실행
런타임 연결 해제 및 삭제

런타임 유형 변경

노트 설정

하드웨어 가속기

None
GPU
TPU

실행

이 노트를 저장할 때 코드 셀 출력 생략

취소 저장

3. Google colab

- Colab 에서 패키지 설치

```
!pip install pandas
```

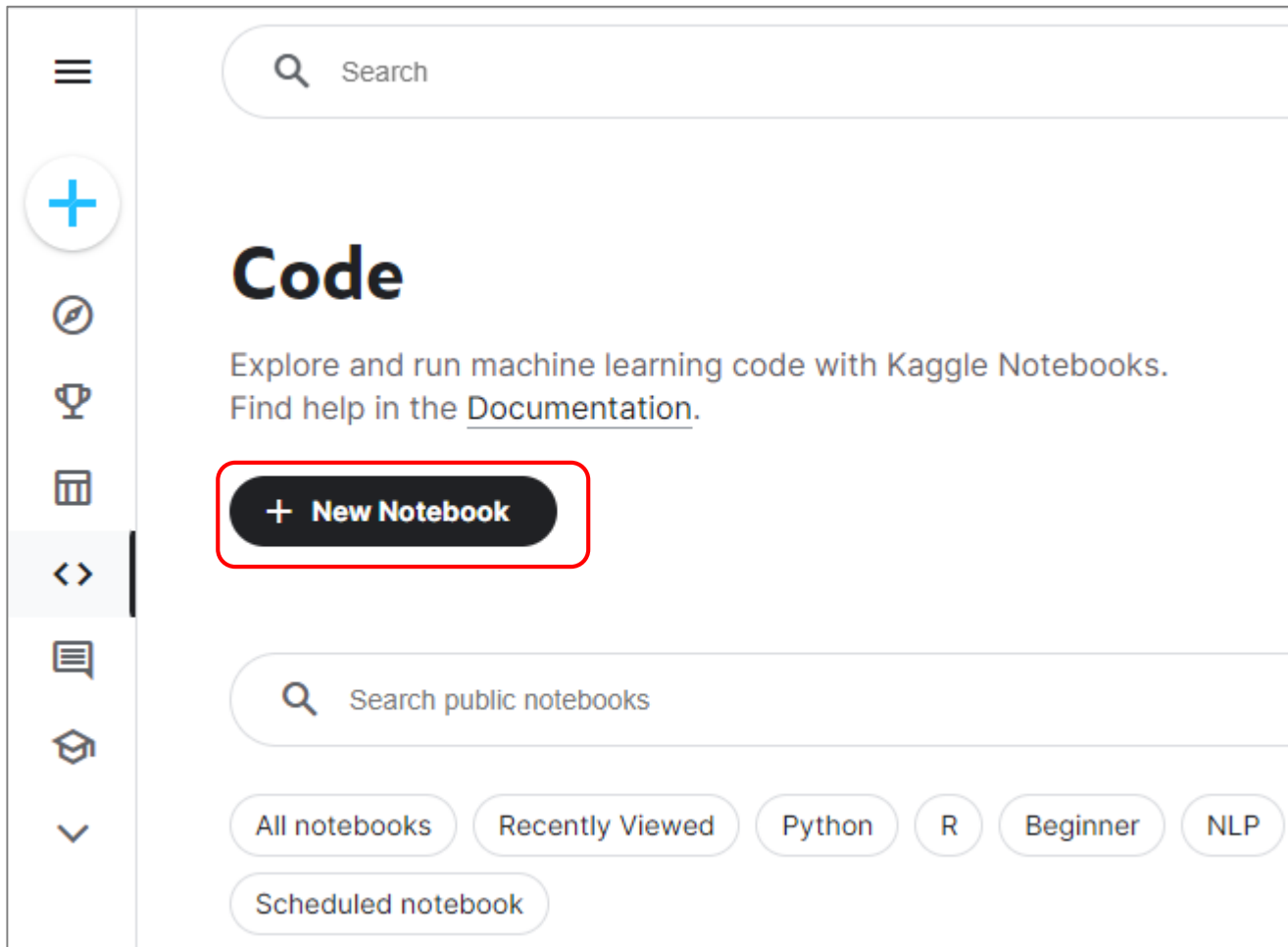
- 자주 사용되는 패키지 들은 대부분 설치되어 있다.

- Colab 에서 데이터 파일 읽기

- (1) 데이터 파일을 Colab 으로 업로드
- (2) 업로드한 파일을 읽는다.

3. Google colab

- Kaggle notebook
- <https://www.kaggle.com/code>



3. Google colab

- Kaggle notebook

Explore machine learning with Kaggle. Logged out sessions end after 15 minutes

Sign in or Register

kernel111beaa572 Draft saved

File Edit View Run Help

+ Run All Draft Session (11h:1m) HDD CPU RAM

```
# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will list the files in the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 5GB to the current directory (/kaggle/working/)
# You can also write temporary files to /kaggle/temp/, but they won't be saved outside of the current session
```

+ Code + Markdown

Data + Add data

Settings

Code Help

Find Code Help

Search for examples of how to do things

Add code

Add text

3. Google colab

- Reference
- Colab vs Kaggle notebook
 - <https://towardsdatascience.com/kaggle-vs-colab-faceoff-which-free-gpu-provider-is-tops-d4f0cd625029>
- Keras in colab
 - <http://corazzon.github.io/keras-gpu>
- Escape colab idle time check
 - <https://teddylee777.github.io/colab/google-colab-%EB%9F%B0%ED%83%80%EC%9E%84-%EC%97%B0%EA%B2%B0%EB%81%8A%EA%B9%80%EB%B0%A9%EC%A7%80>