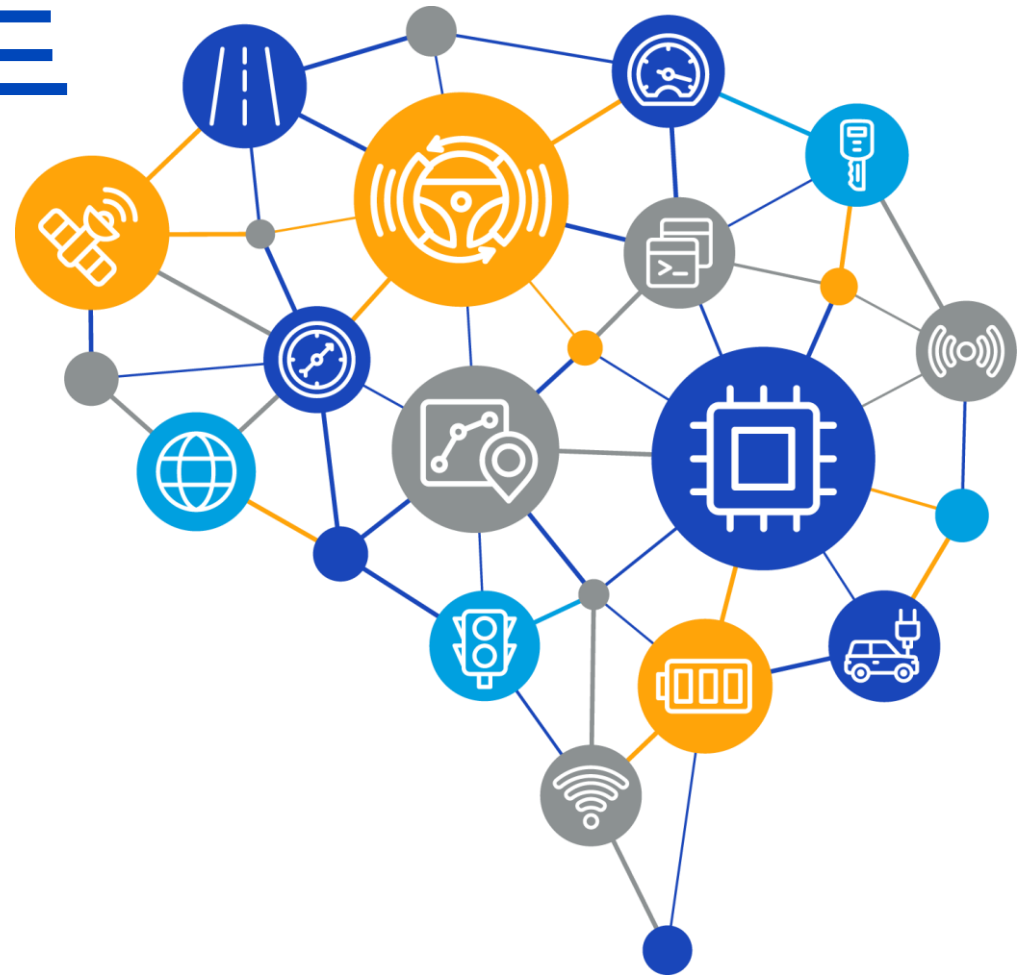


비주얼 스튜디오 코드 파이썬 환경과 주피터 노트북 실행



강환수 교수

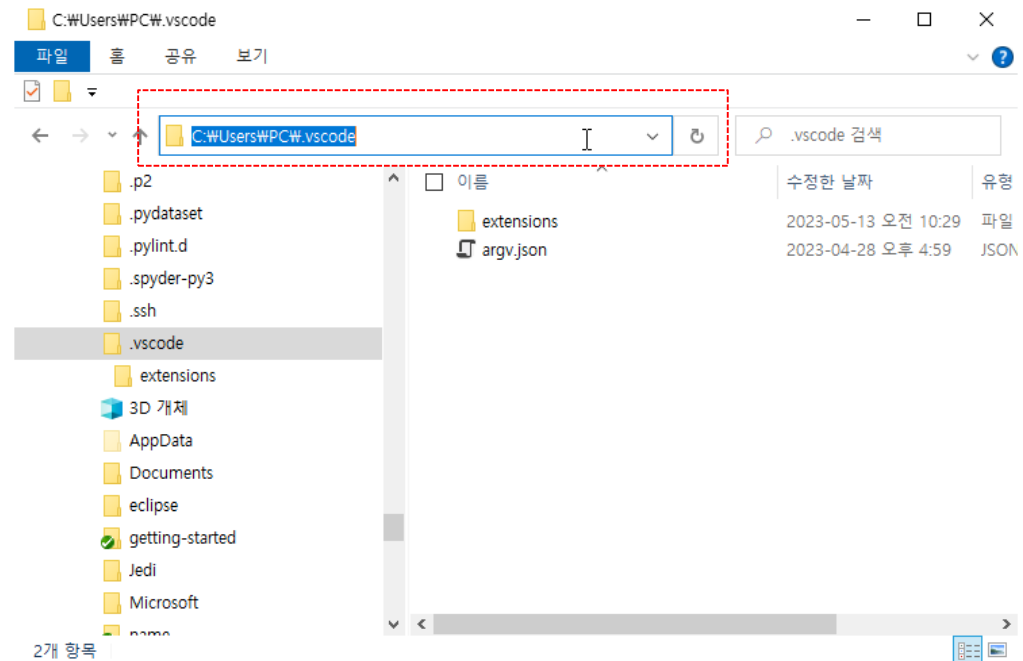
AI Experts
Who Lead
The Future

01

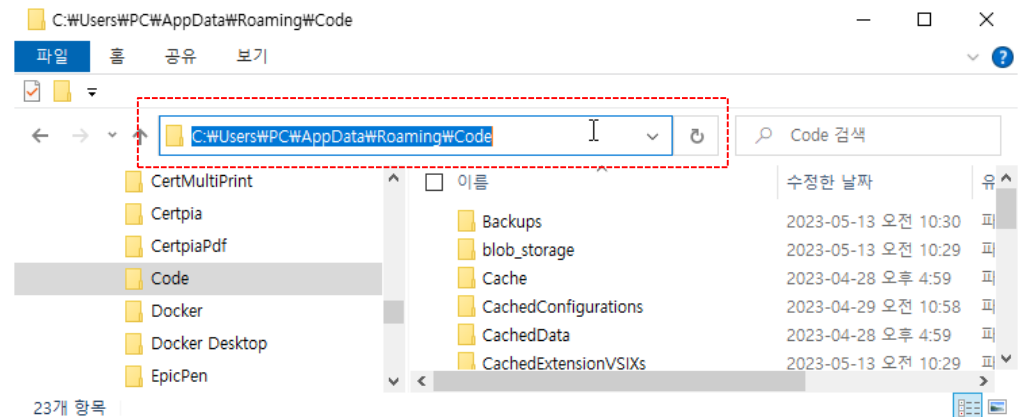
Vs code 파이썬 코드 실행 기초

- Vs code
 - 인터프리터가 반드시 필요
- 즐겨 활용하는 인터프리터
 - ① 표준 Python 인터프리터 설치
 - python.org 에서 Python을 설치
 - ② 또는 아나콘다 등의 배포판 설치

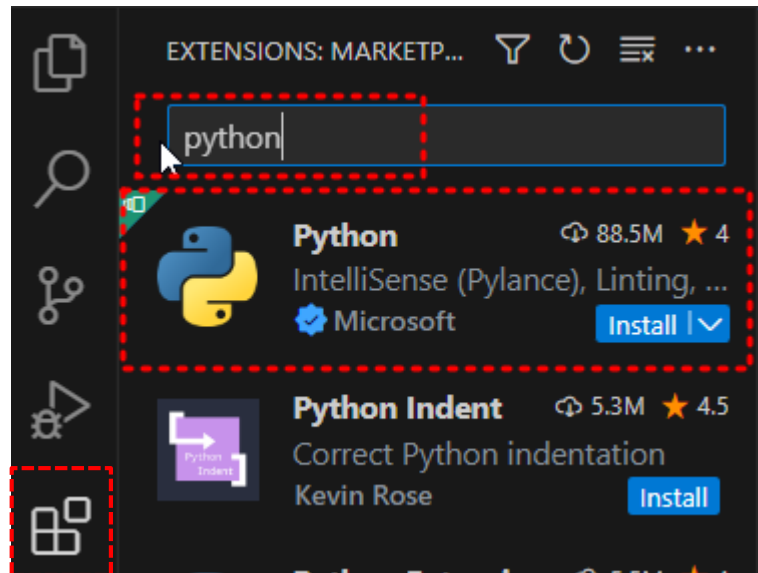
- 확장 등 설정 폴더 삭제
 - C:\Users\W[사용자계정]\W.vscode

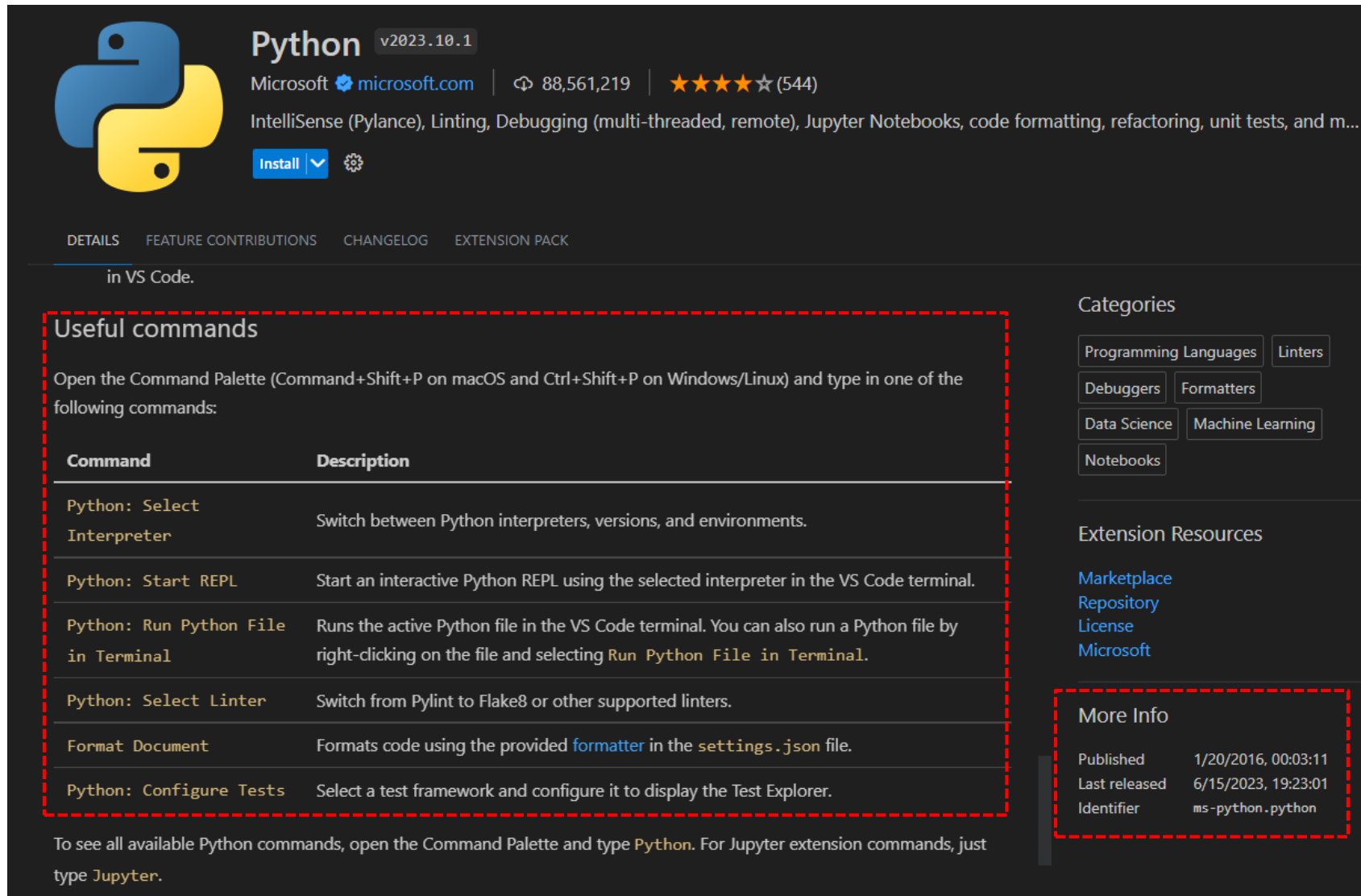


- C:\Users\W[사용자계정]\W\AppData\Roaming\WCode 삭제



- Python 확장 설치





Python v2023.10.1

Microsoft microsoft.com | 88,561,219 | ★★★★★ (544)

IntelliSense (Pylance), Linting, Debugging (multi-threaded, remote), Jupyter Notebooks, code formatting, refactoring, unit tests, and m...

[Install](#) [Settings](#)

[DETAILS](#) [FEATURE CONTRIBUTIONS](#) [CHANGELOG](#) [EXTENSION PACK](#)

in VS Code.

Useful commands

Open the Command Palette (Command+Shift+P on macOS and Ctrl+Shift+P on Windows/Linux) and type in one of the following commands:

Command	Description
Python: Select Interpreter	Switch between Python interpreters, versions, and environments.
Python: Start REPL	Start an interactive Python REPL using the selected interpreter in the VS Code terminal.
Python: Run Python File in Terminal	Runs the active Python file in the VS Code terminal. You can also run a Python file by right-clicking on the file and selecting Run Python File in Terminal .
Python: Select Linter	Switch from Pylint to Flake8 or other supported linters.
Format Document	Formats code using the provided formatter in the <code>settings.json</code> file.
Python: Configure Tests	Select a test framework and configure it to display the Test Explorer.

To see all available Python commands, open the Command Palette and type **Python**. For Jupyter extension commands, just type **Jupyter**.

Categories

- Programming Languages
- Linters
- Debuggers
- Formatters
- Data Science
- Machine Learning
- Notebooks

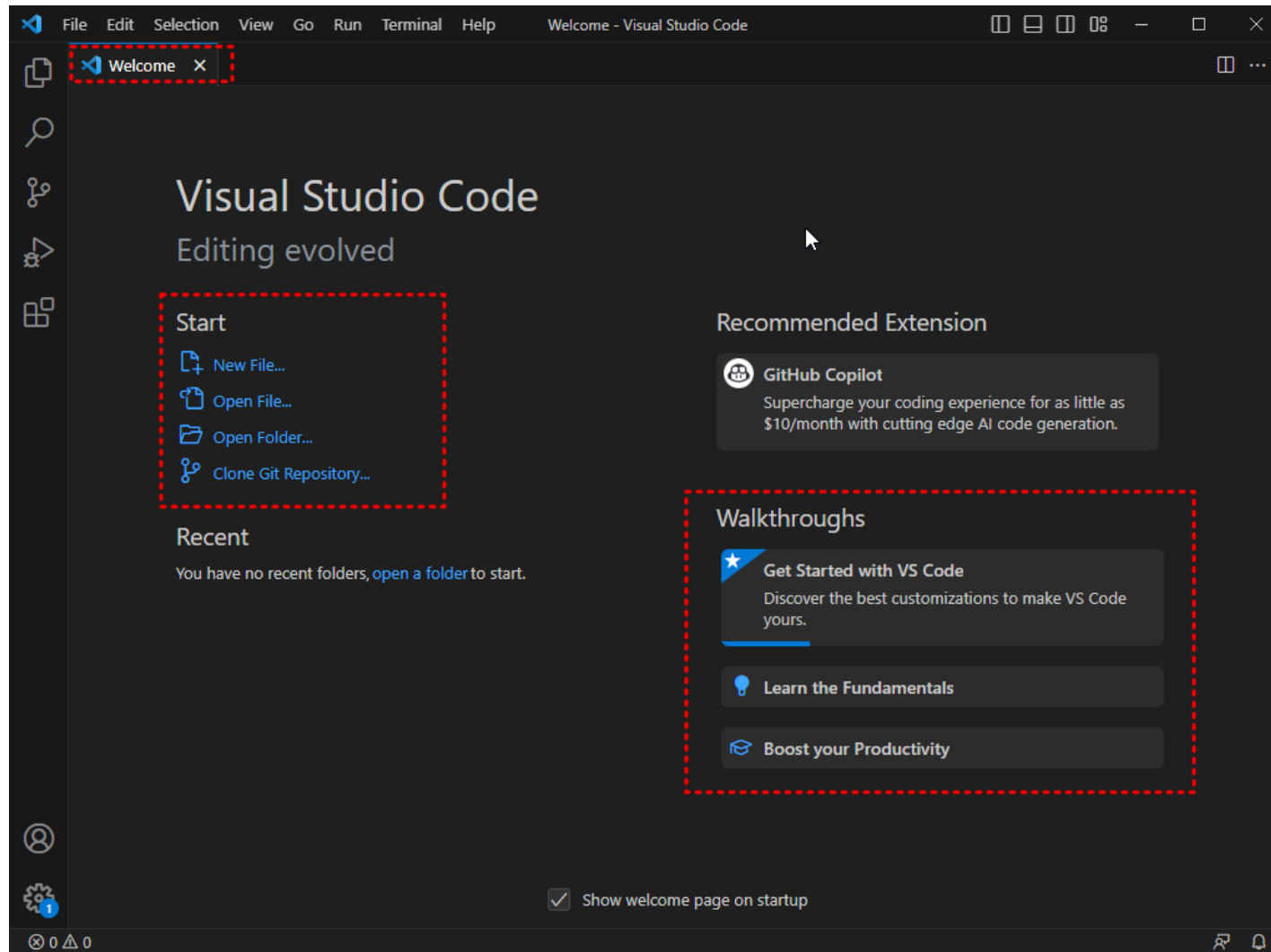
Extension Resources

- [Marketplace](#)
- [Repository](#)
- [License](#)
- [Microsoft](#)

More Info

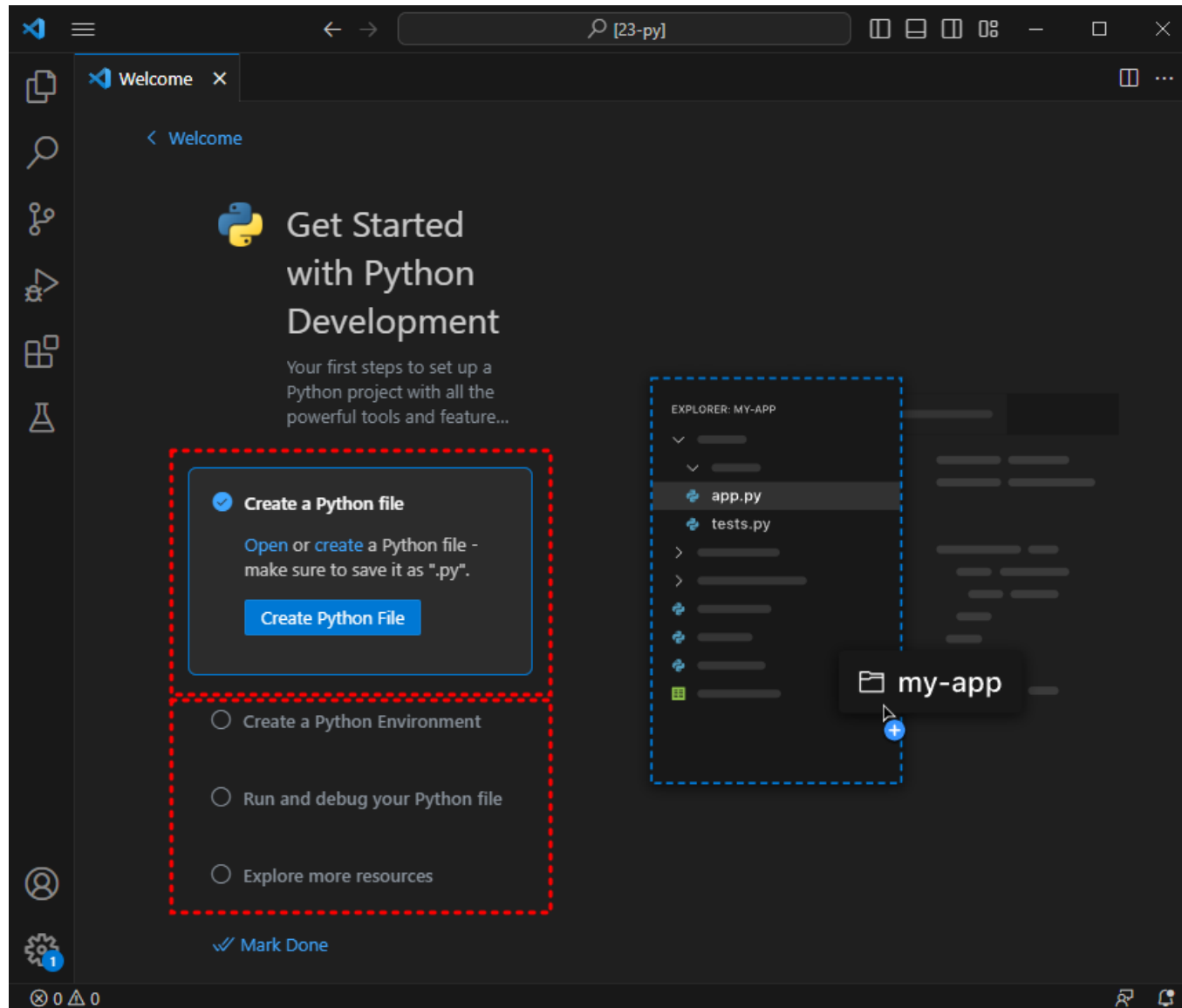
Published	1/20/2016, 00:03:11
Last released	6/15/2023, 19:23:01
Identifier	ms-python.python

- 다를 수 있음

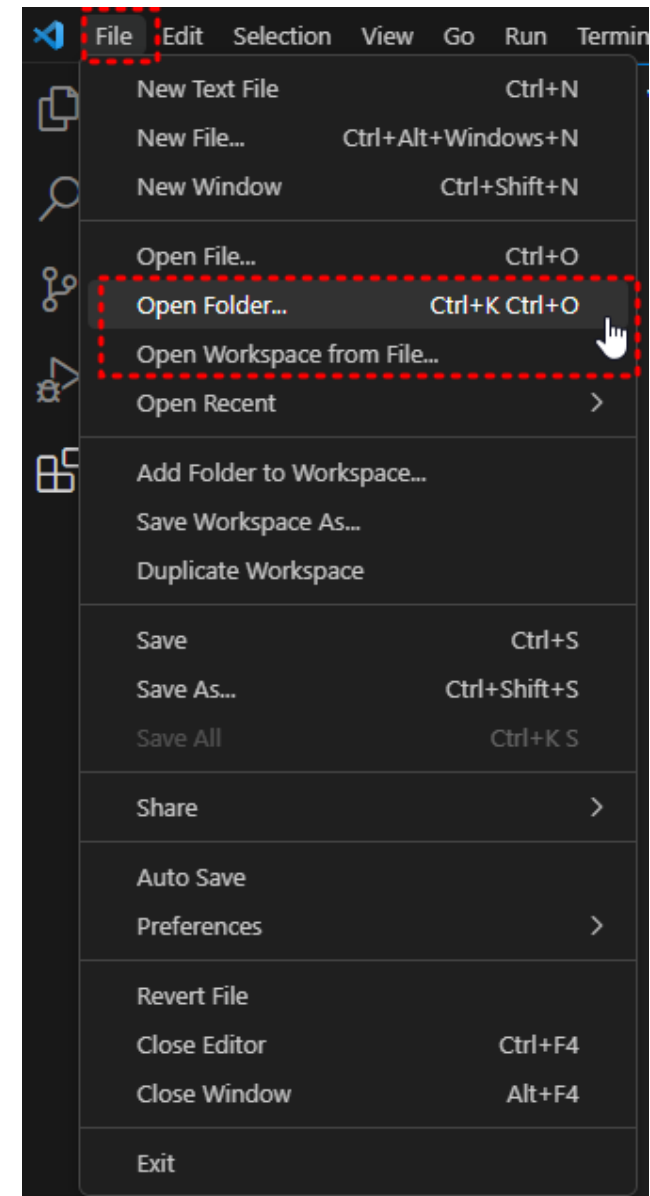
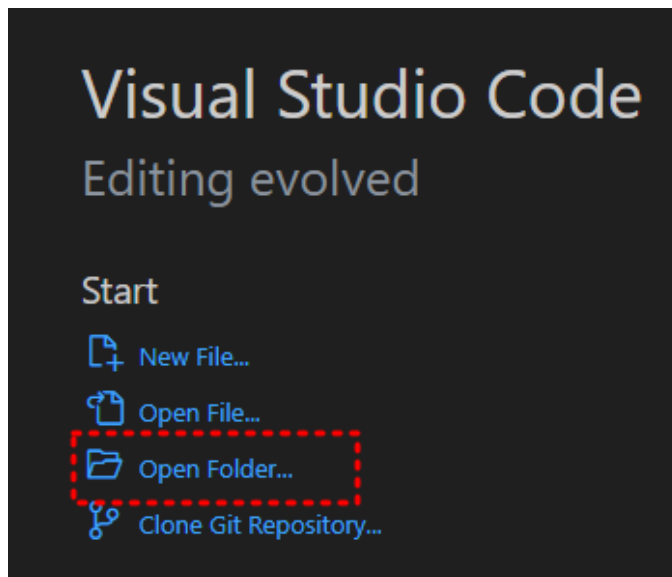


Get Strted with vs Code

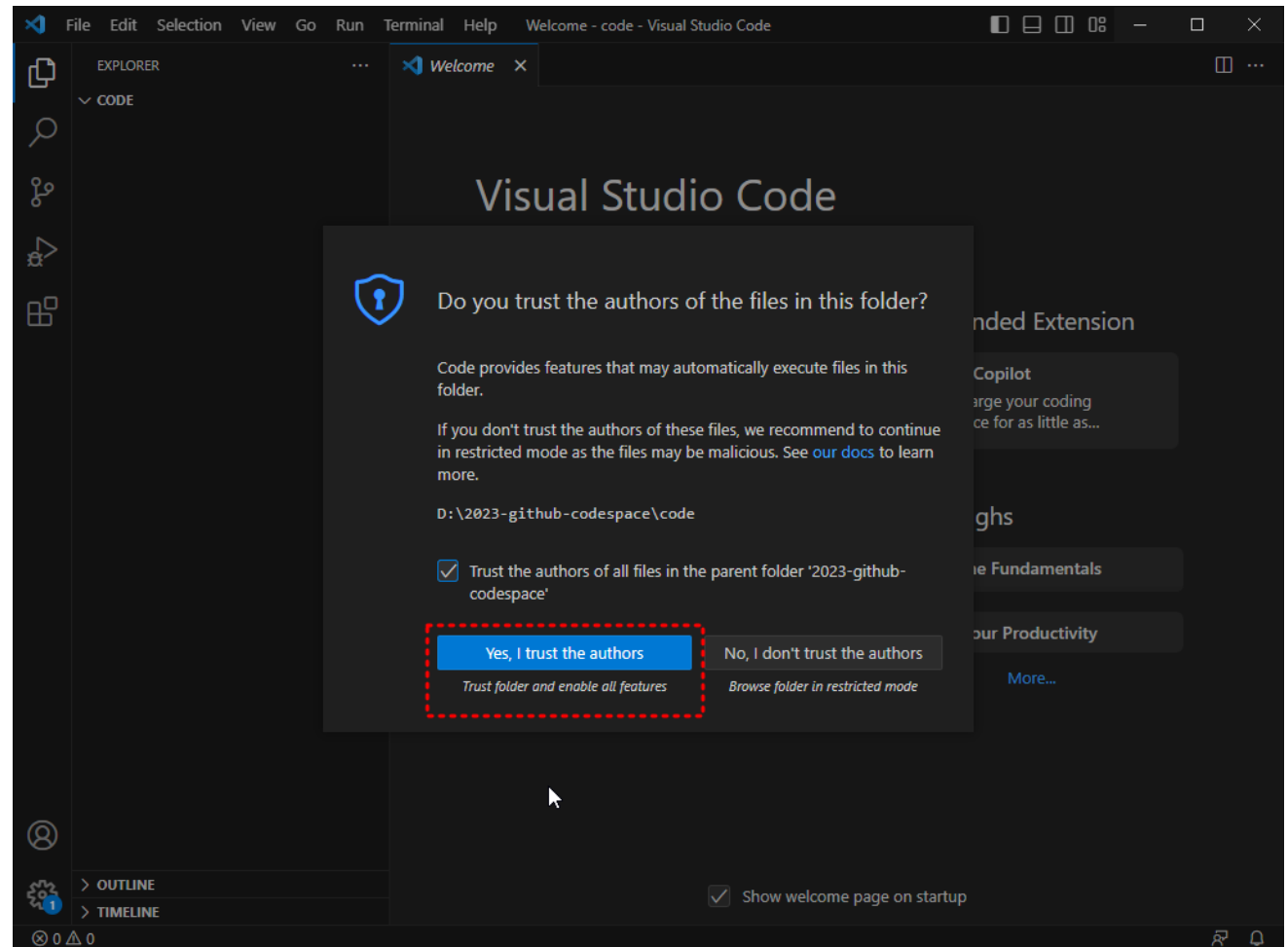
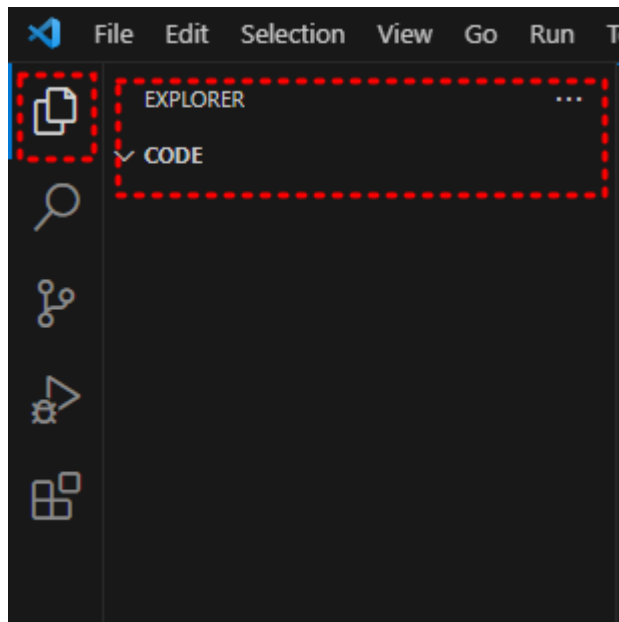
오픈소스 소프트웨어를 위한 깃과 깃허브 Python language



- 폴더 지정
 - 우리 작업 폴더
 - D:\2023-github-codespace\code
 - Ctrl + K + Ctrl + O



- 열린 폴더



- **chatGPT** 파이썬을 처음 배우는 학생에게 도움이 되는
코드 10줄 정도를 코딩해 줘, 설명도 해줘

1. Hello, World! 출력하기

```
print("Hello, World!")
```

2. 변수 사용하기

```
name = input("이름을 입력하세요: ")
```

```
print("안녕하세요, " + name + "님!")
```

3. 숫자 계산하기

```
num1 = 10
```

```
num2 = 5
```

```
result = num1 + num2
```

```
print("결과:", result)
```

4. 조건문 사용하기

```
age = int(input("나이를 입력하세요: "))
```

```
if age >= 18:
```

```
    print("성인입니다.")
```

```
else:
```

```
    print("미성년자입니다.")
```

5. 반복문 사용하기

```
for i in range(5):
```

```
    print("반복문 예제:", i)
```

6. 리스트 활용하기

```
fruits = ["사과", "바나나", "오렌지"]
```

```
print("과일 목록:", fruits)
```

7. 함수 정의하기

```
def square(x):
```

```
    return x * x
```

```
result = square(5)
```

```
print("결과:", result)
```

8. 파일 입출력

```
file_name = "example.txt"
```

```
with open(file_name, "w") as f:
```

```
    f.write("파일 입출력 예시")
```

9. 모듈 사용하기

```
import random
```

```
random_number = random.randint(1, 100)
```

```
print("랜덤 숫자:", random_number)
```

10. 예외 처리하기

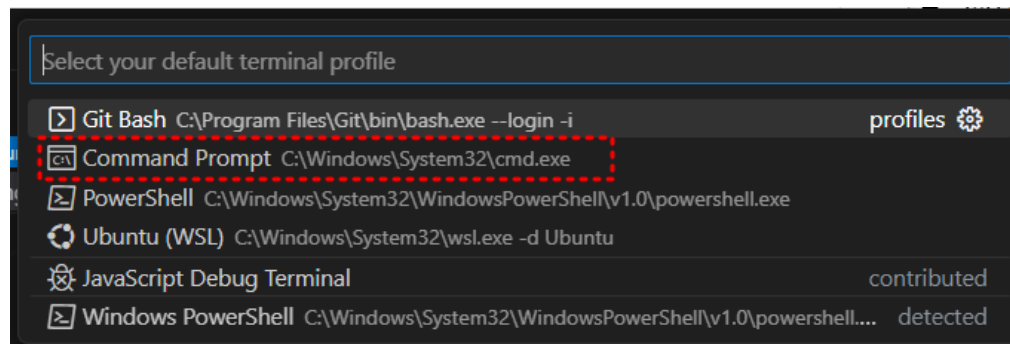
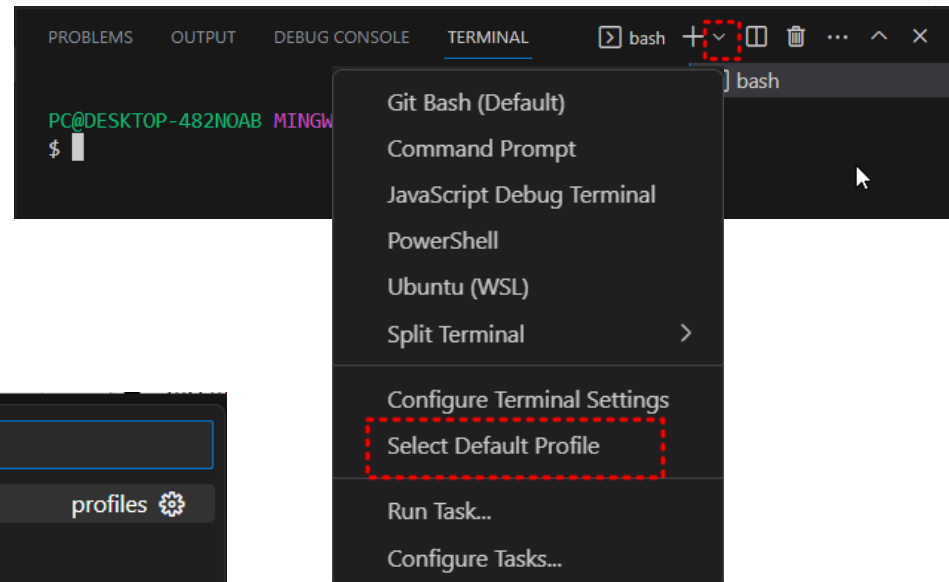
```
try:
```

```
    x = 10 / 0
```

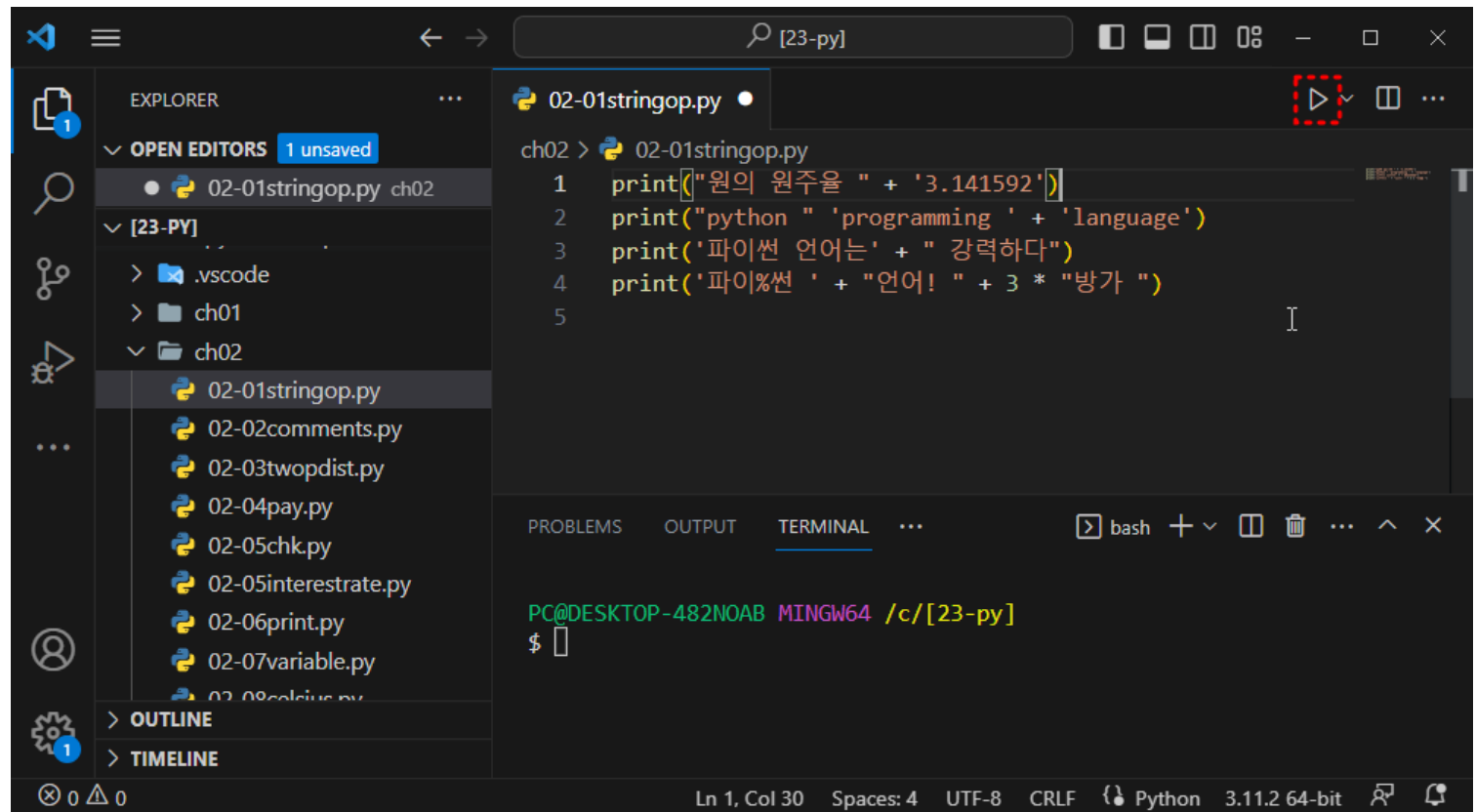
```
except ZeroDivisionError:
```

```
    print("0으로 나눌 수 없습니다.")
```

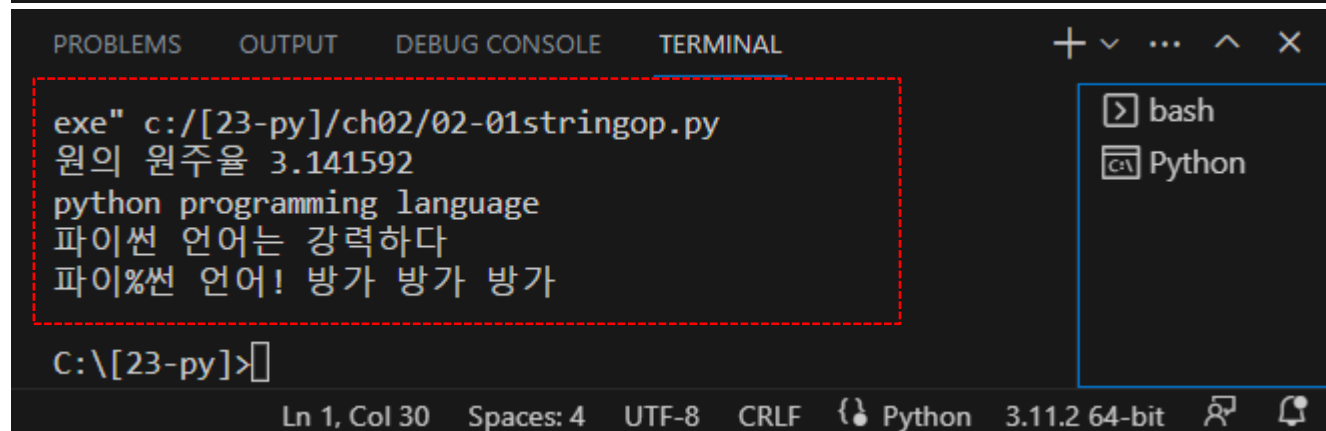
- 파일 생성
 - hello.py
- 터미널 실행
 - Ctrl + `(backquote)[왼쪽 Tab 키 위]
- 터미널 프로파일 설정
 - 터미널 아이콘 메뉴 아래 화살표 클릭
 - Select Default Profile
 - 원하는 셸 선택
 - Command Prompt



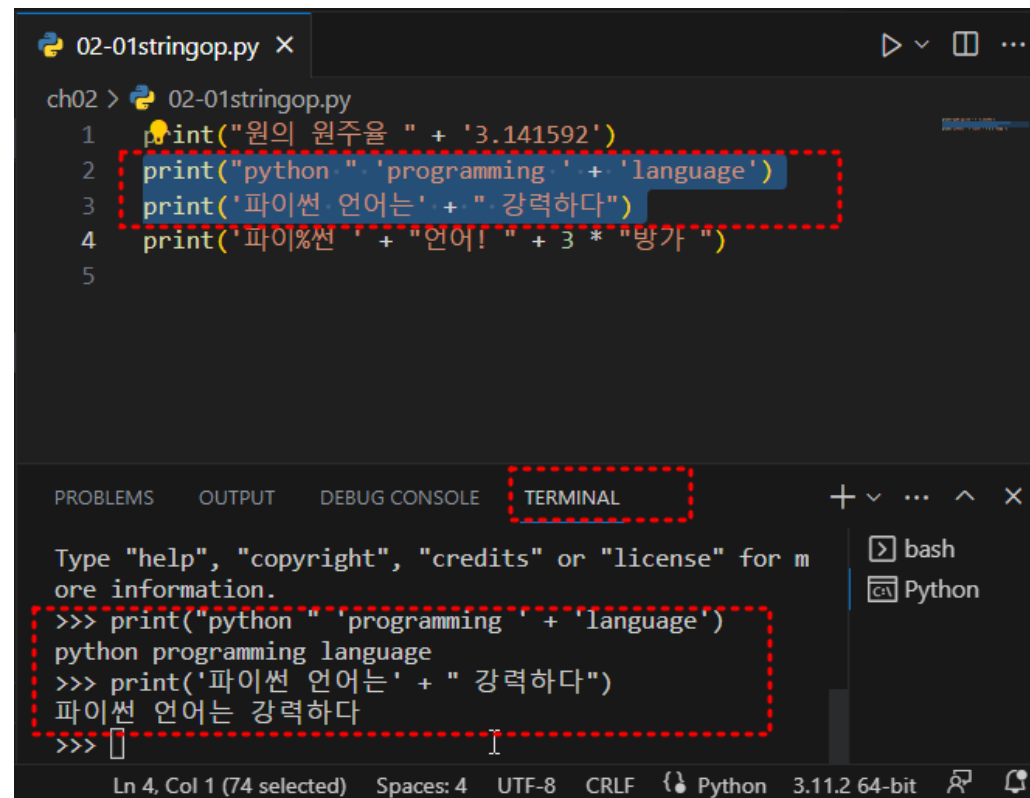
- 실행 아이콘 클릭



- 터미널에서 실행



- 편집기 창 클릭 > 마우스 오른쪽 버튼으로 클릭
 - Run Python | Run Python File in Terminal
- 하나 이상의 라인을 선택
 - 마우스 오른쪽 메뉴
 - Run Python | Run Selection/Line in Python Terminal
 - Shift + Enter



The screenshot shows an IDE window with a file named '02-01stringop.py'. The code in the editor is as follows:

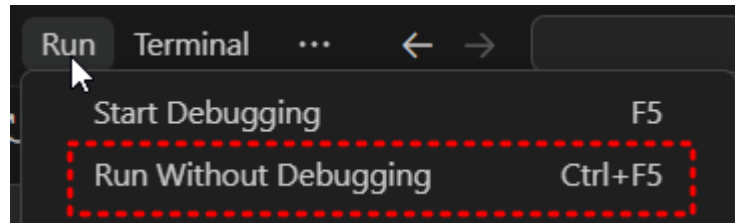
```
1 print("원의 원주율 " + '3.141592')
2 print("python " + 'programming ' + 'language')
3 print('파이썬 언어는 ' + " 강력하다")
4 print('파이썬 ' + "언어! " + 3 * "방가 ")
5
```

Lines 2 and 3 are selected. The 'TERMINAL' tab is active, showing the output of the selected lines:

```
Type "help", "copyright", "credits" or "license" for more information.
>>> print("python " + 'programming ' + 'language')
python programming language
>>> print('파이썬 언어는 ' + " 강력하다")
파이썬 언어는 강력하다
>>>
```

The status bar at the bottom indicates 'Ln 4, Col 1 (74 selected) Spaces: 4 UTF-8 CRLF Python 3.11.2 64-bit'.

- **Ctrl + F5**
 - Run | Run Without Debugging

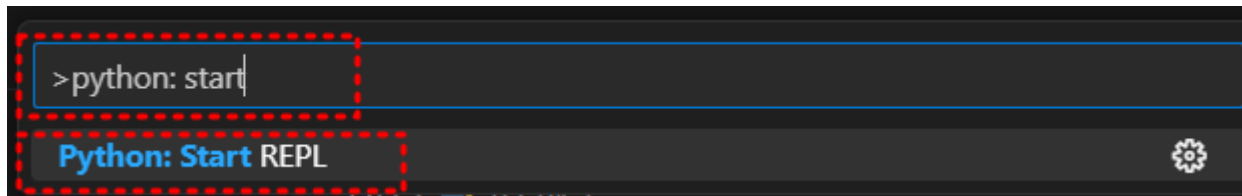


Python 인터프리터에 대한 REPL 터미널 열기

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

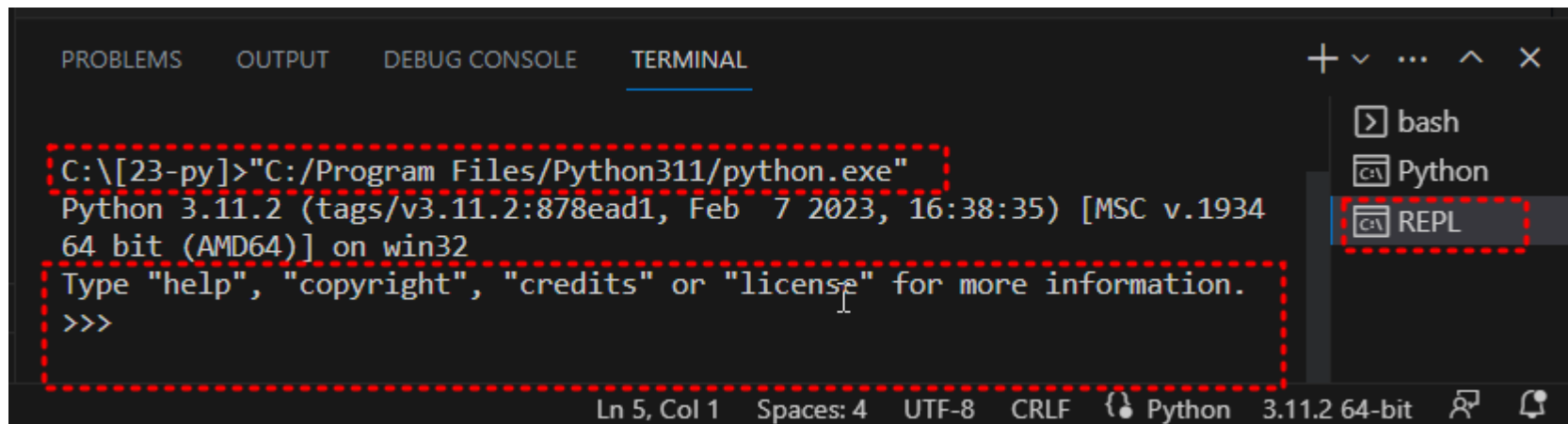
- **Command Palette**

- Ctrl+Shift+P



- **다음 입력**

- Python: Start REPL
- REPL에서
 - 코드 행을 한 번에 하나씩 입력하고 실행

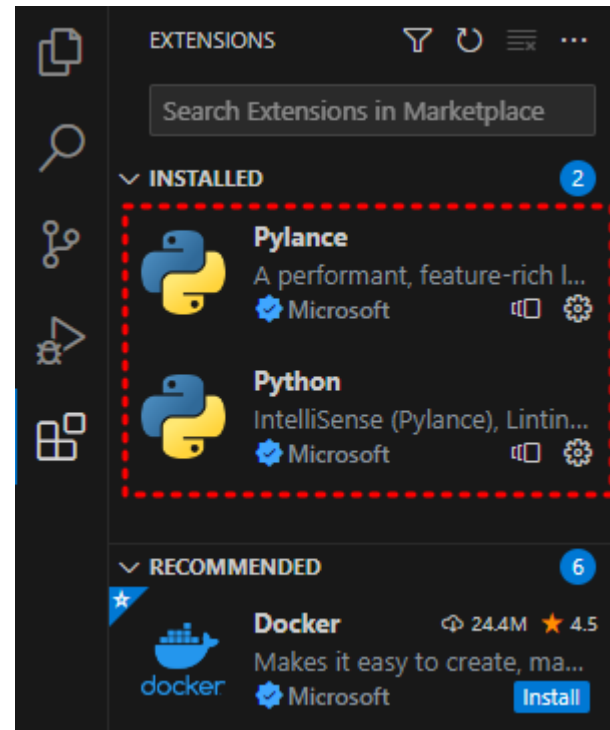


AI Experts
Who Lead
The Future

02

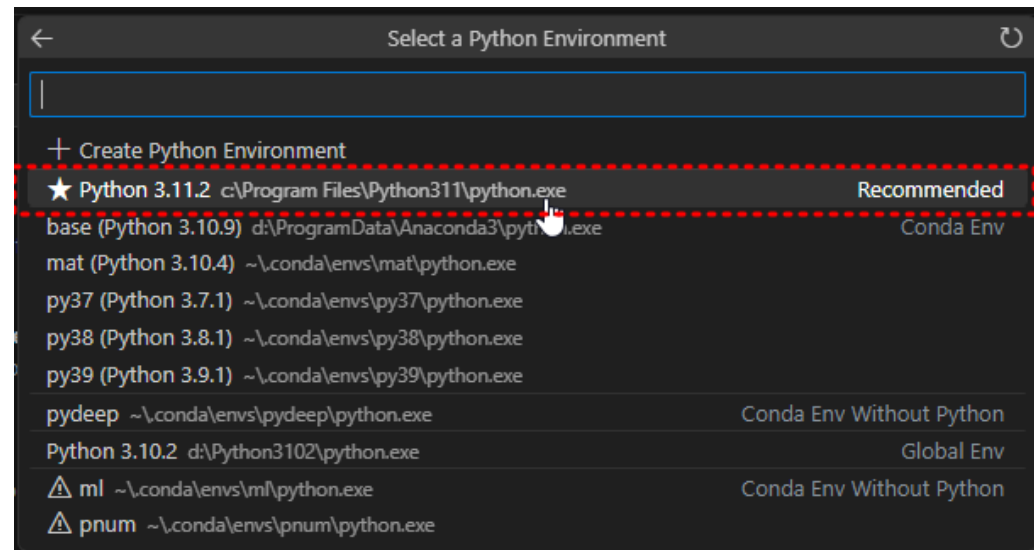
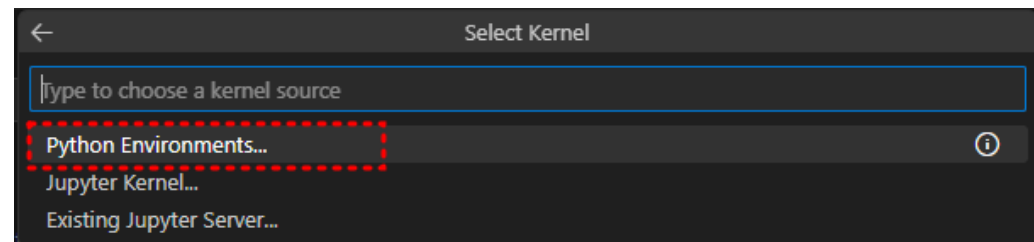
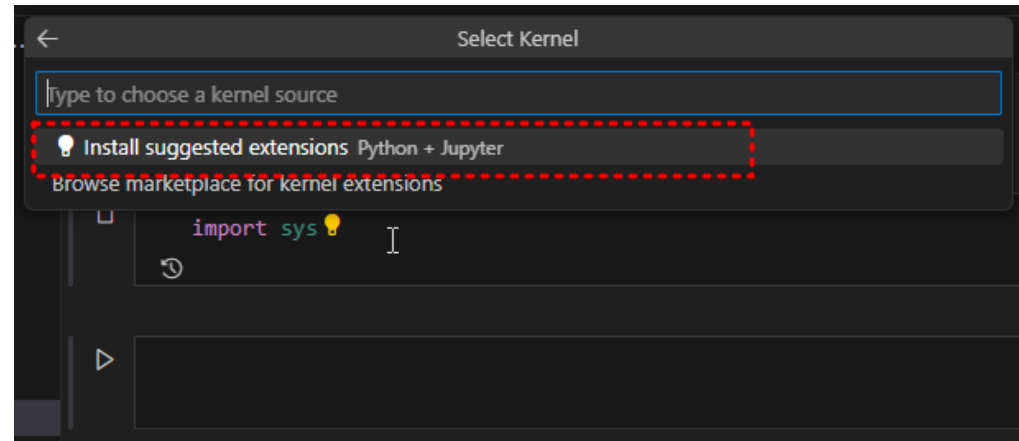
주피터 노트북에서 파이썬 실행

- 기본

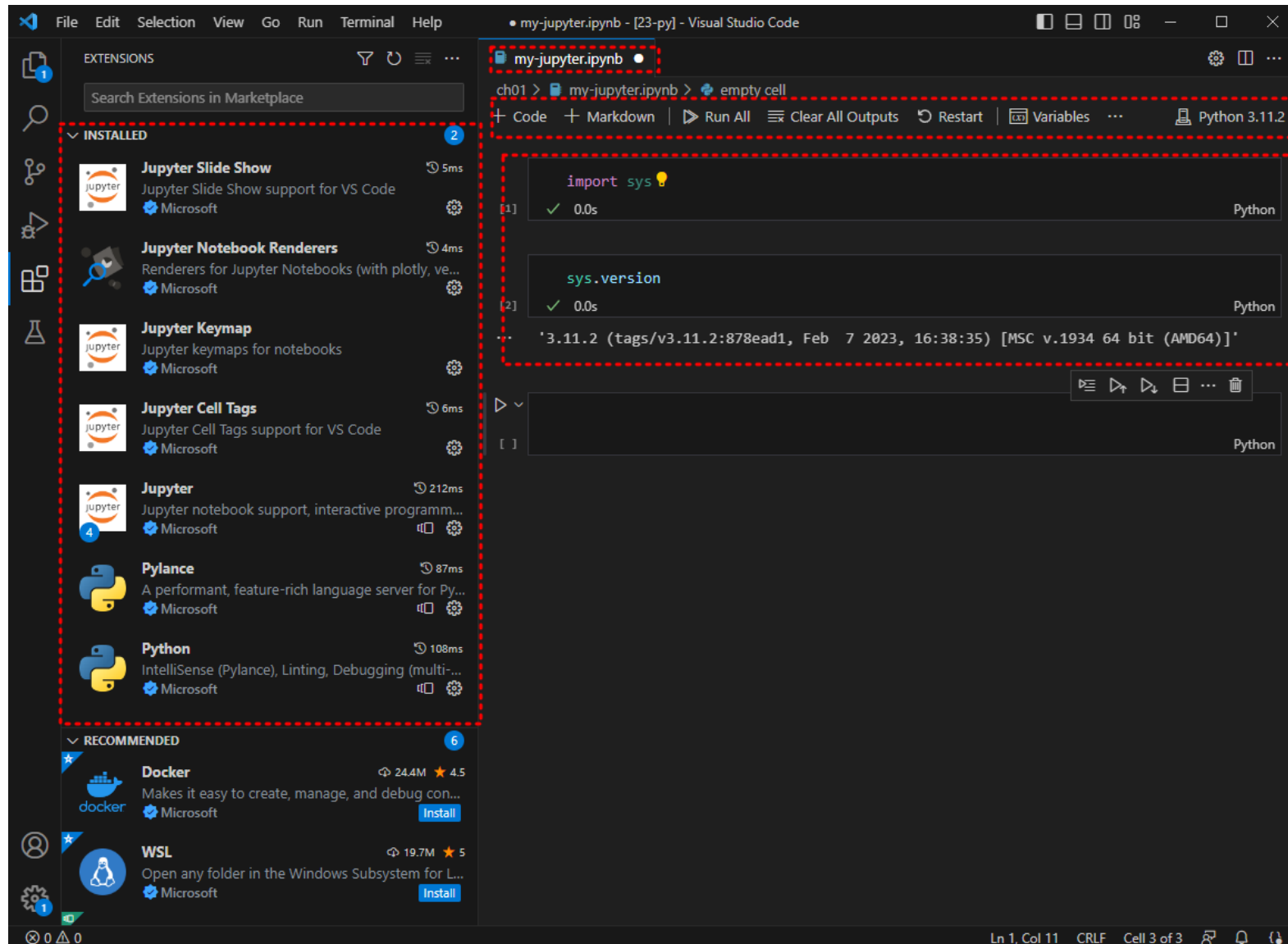


- 대화형이 강화된 인터프리터
 - 주피터(이전의 Ipython Notebook)
 - 마크다운 텍스트와 실행 가능한 파이썬 소스 코드를 노트북이라는 하나의 캔버스에 쉽게 결합할 수 있는 오픈 소스 프로젝트
 - 코드 셀로 작업

- 파일 my-jupyter.ipynb
- 첫 셀 코딩
 - import sys
 - Shift + Enter
- 선택 install suggested extensions
 - 주피터 관련 여러 개의 확장 설치
- 다시 실행
 - 선택 Python Environments...
 - 선택 Python 3.11.2



- 주피터 관련 여러 개의 확장 설치



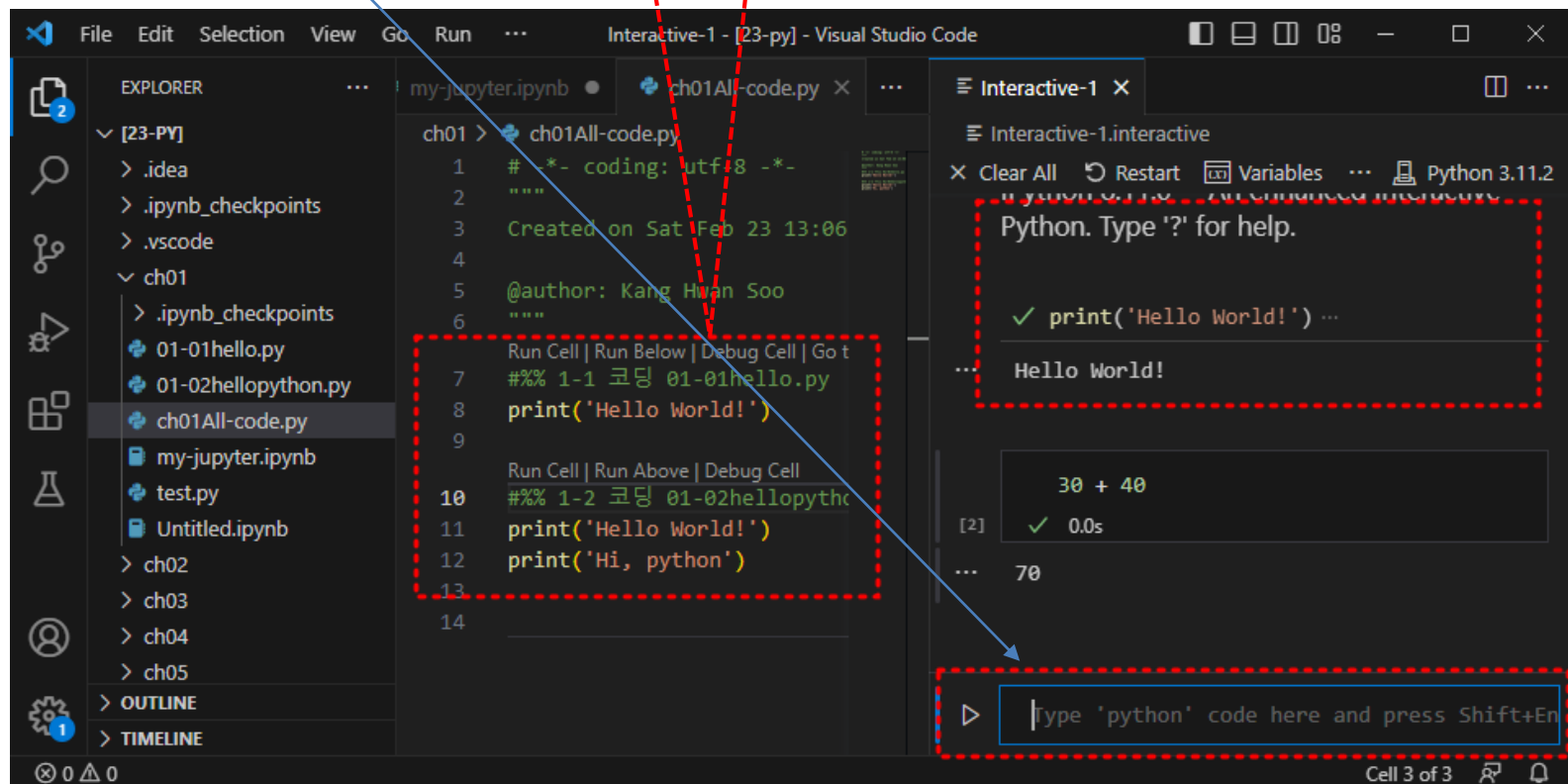
- 파일 *.py
 - # %%

일반 파이썬 파일에서 주피터 코드 셀 실행

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- # %%
 - 코드 셀 표시
- 실행
 - Shift + Enter
- 실행된 대화형 창의 맨 하부 셀
 - 코딩 후 실행
 - Ctrl + Enter 또는 Shift + Enter

코드 셀을 인식해야
자동으로 메뉴가 보임



명령	바로 가기 키
파이썬: 다음 셀로 이동	Ctrl+Alt+]
파이썬: 이전 셀로 이동	Ctrl+Alt+[
Python: 위의 셀을 기준으로 선택 영역 확장	Ctrl+Shift+Alt+[
Python: 아래 셀별 선택 영역 확장	Ctrl+Shift+Alt+]
Python: 선택한 셀을 위로 이동	Ctrl+; U
Python: 선택한 셀 아래로 이동	Ctrl+; D
Python: 위에 셀 삽입	Ctrl+; A
Python: 아래에 셀 삽입	Ctrl+; B
Python: 위치 아래에 셀 삽입	Ctrl+; S
Python: 선택한 셀 삭제	Ctrl+; X
파이썬: 셀을 코드로 변경	Ctrl+; C
파이썬: 셀을 마크다운으로 변경	Ctrl+; M