

# 컴퓨터정보과 파이선 프로그래밍

---

02주차 기본 자료형

# [개발환경 구축] 1주차 관련 내용

---

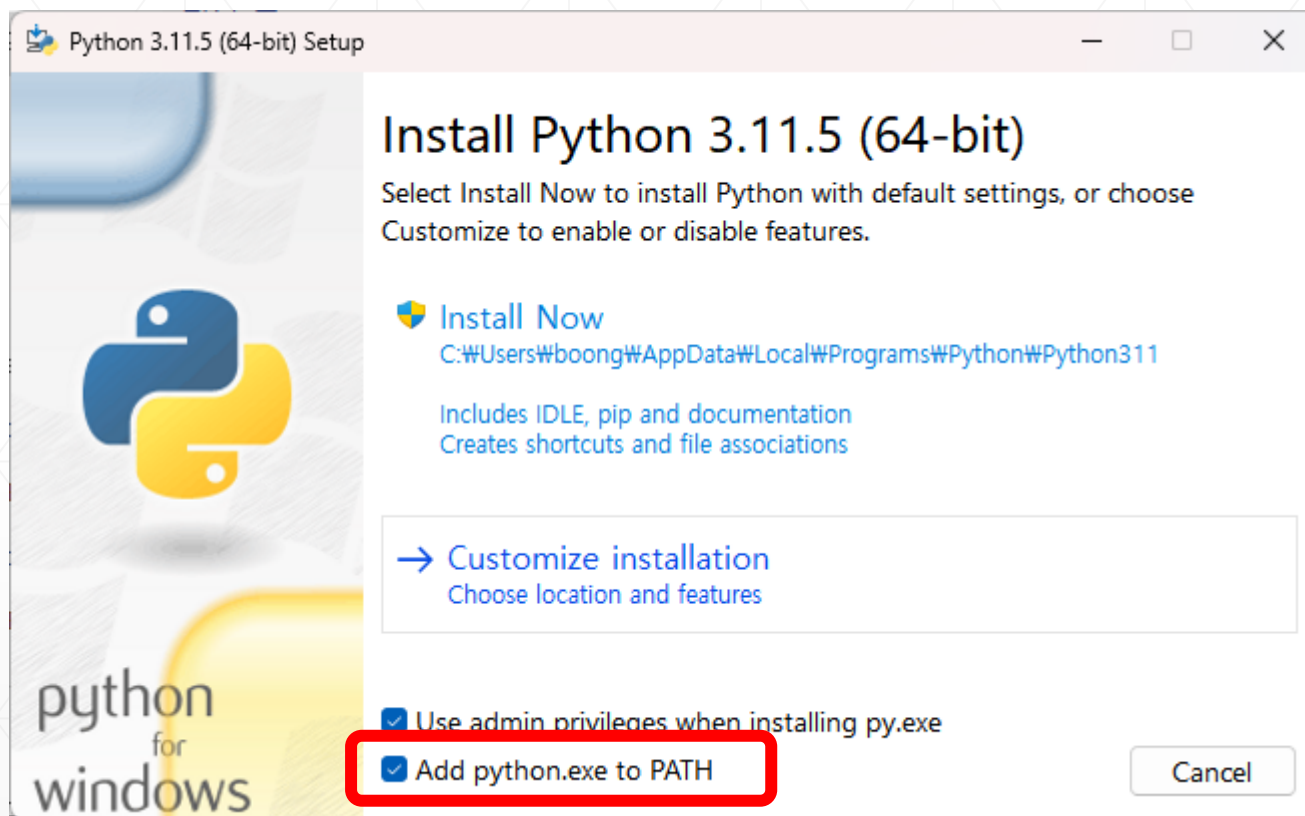
python + VS Code

# 파이썬 설치

---

1주차 참고

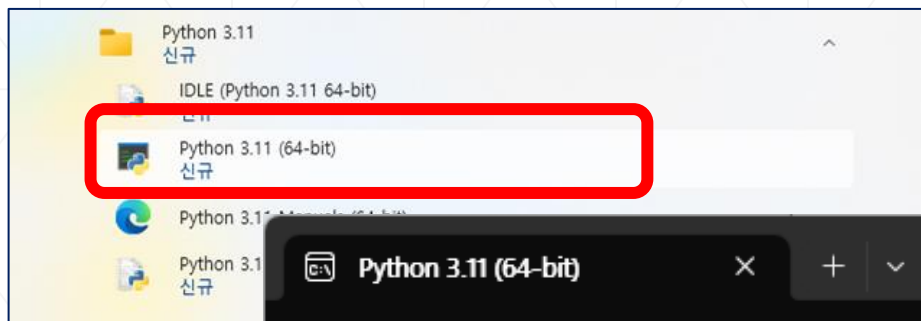
# 설치



# 설치 내용

- Python Interpreter
- Python Shell
- Python IDE
  - IDLE

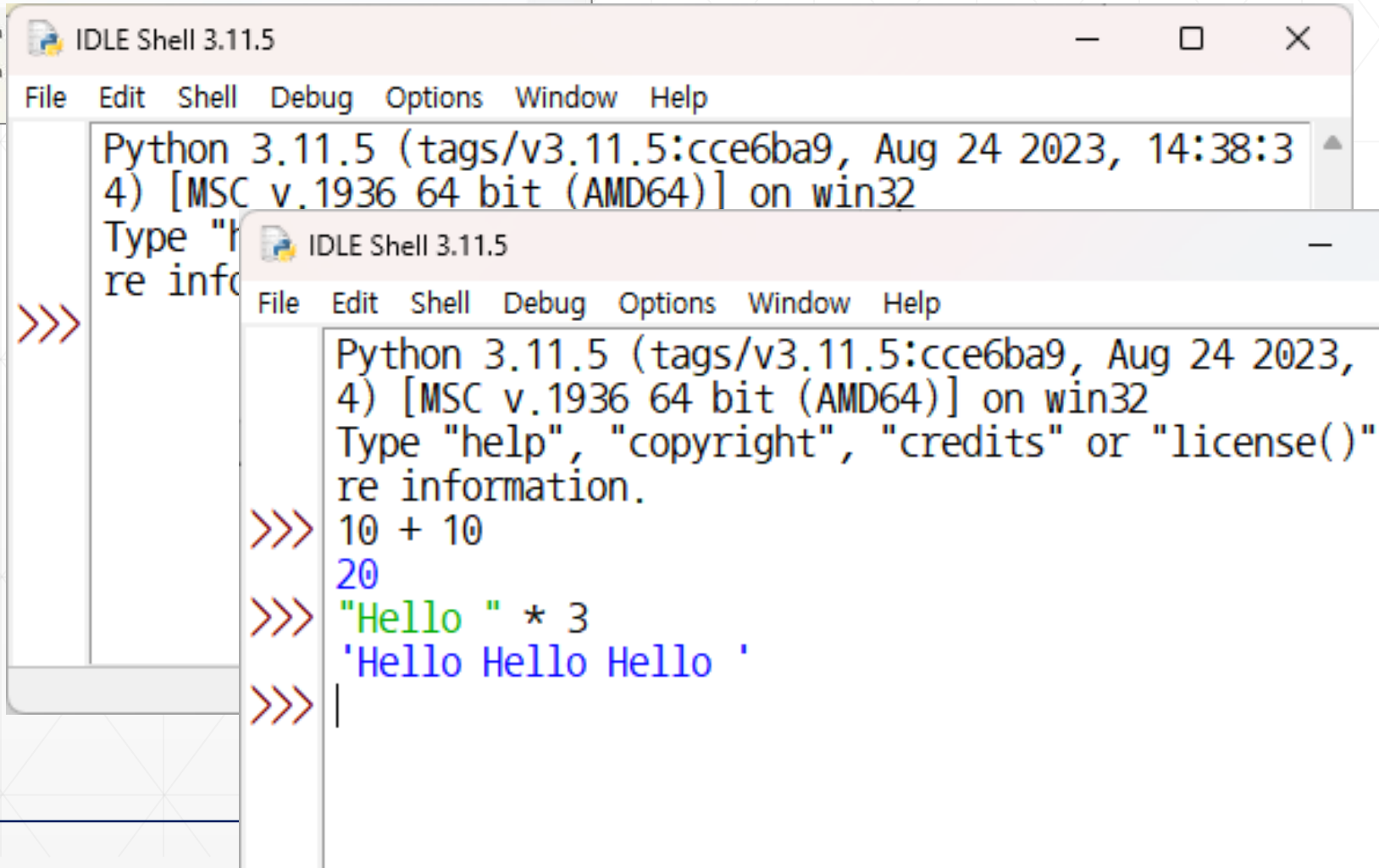
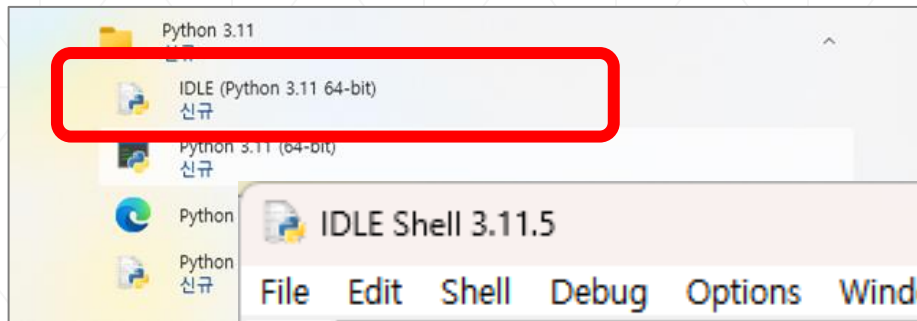
# Python Shell



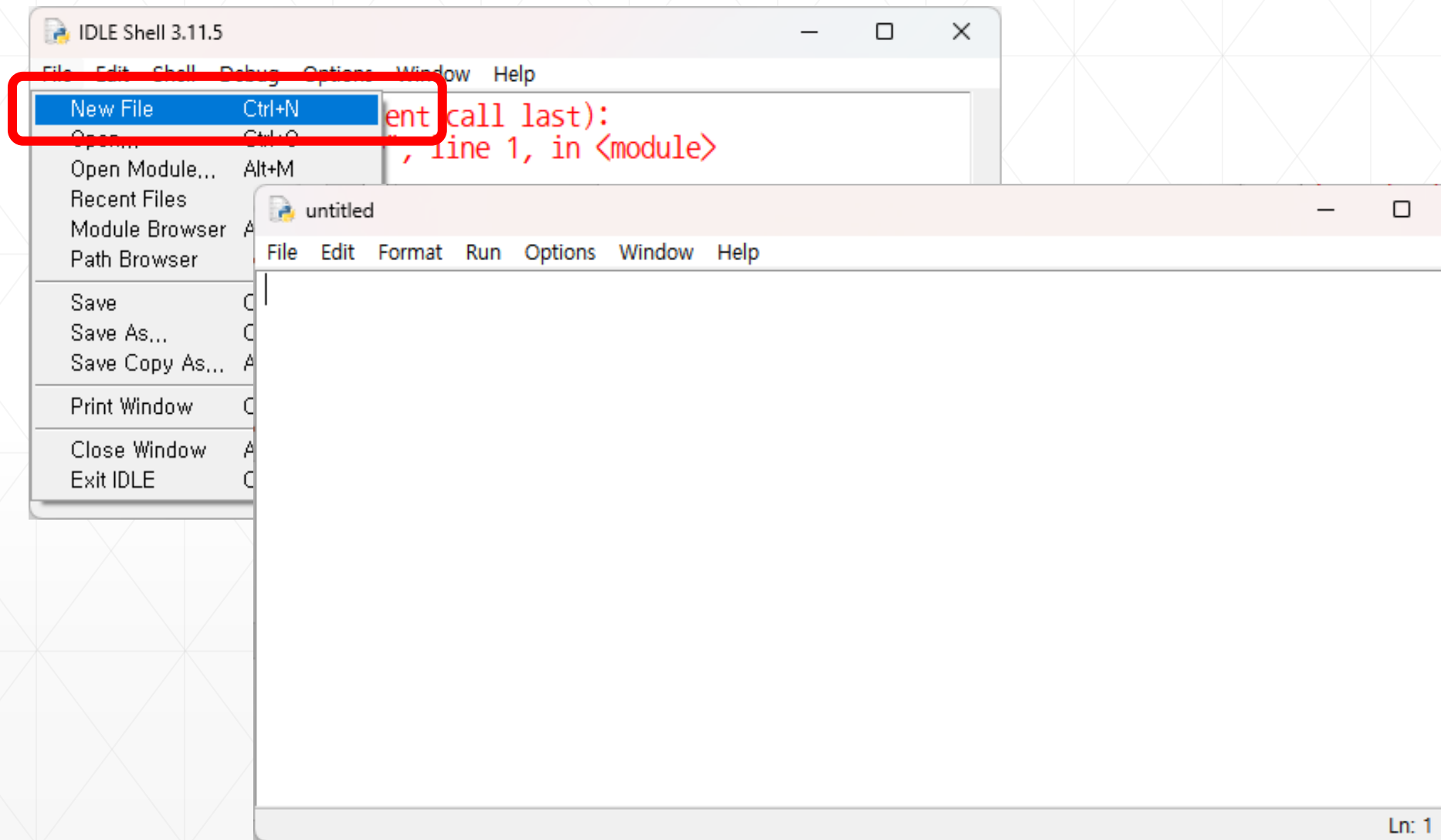
```
Python 3.11 (64-bit)
Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> |
```

```
Python 3.11 (64-bit)
Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> 10 + 10
20
>>> "Hello " * 3
'Hello Hello Hello '
>>> |
```

# IDLE - Shell + Edit



# IDLE로 파일 만들어 실행하기





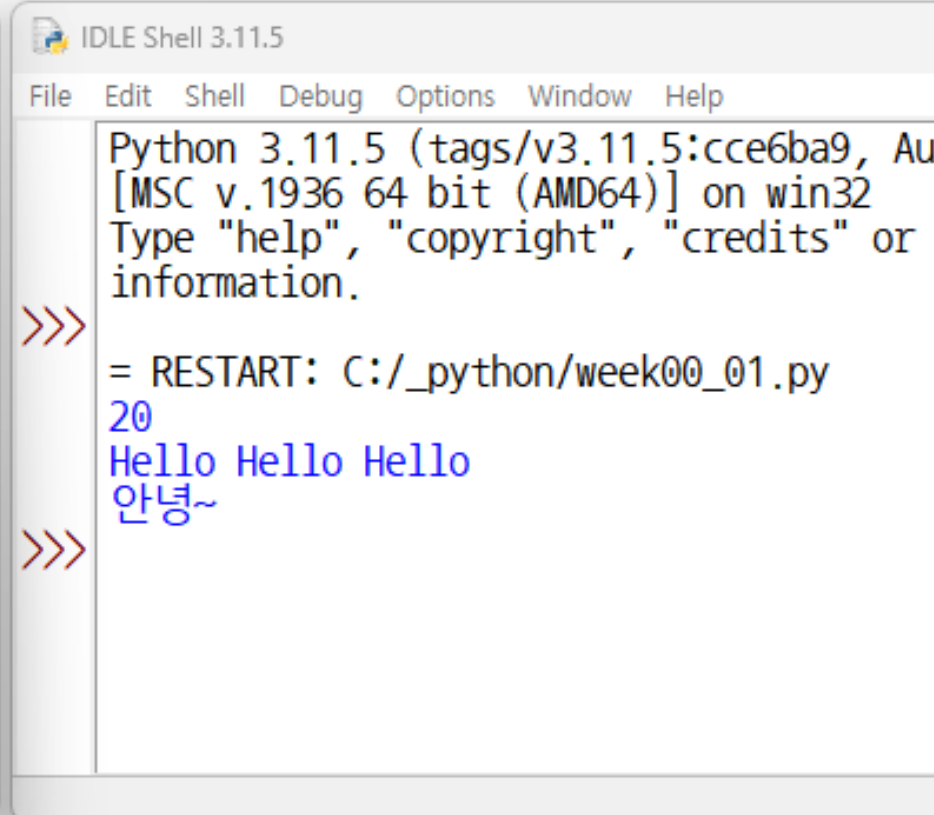
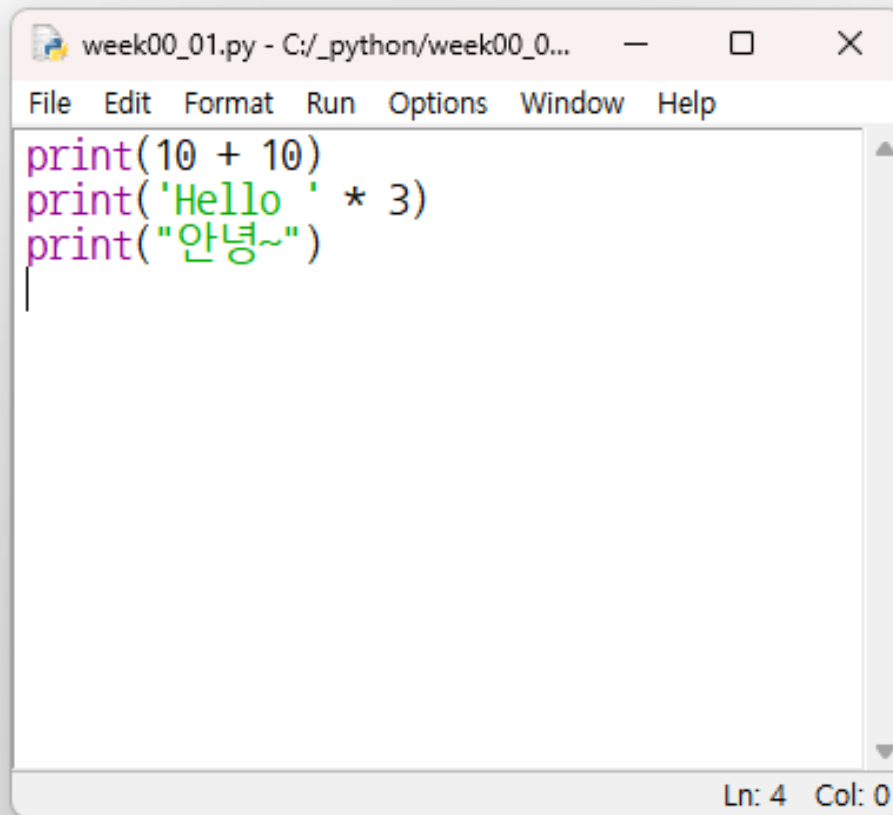
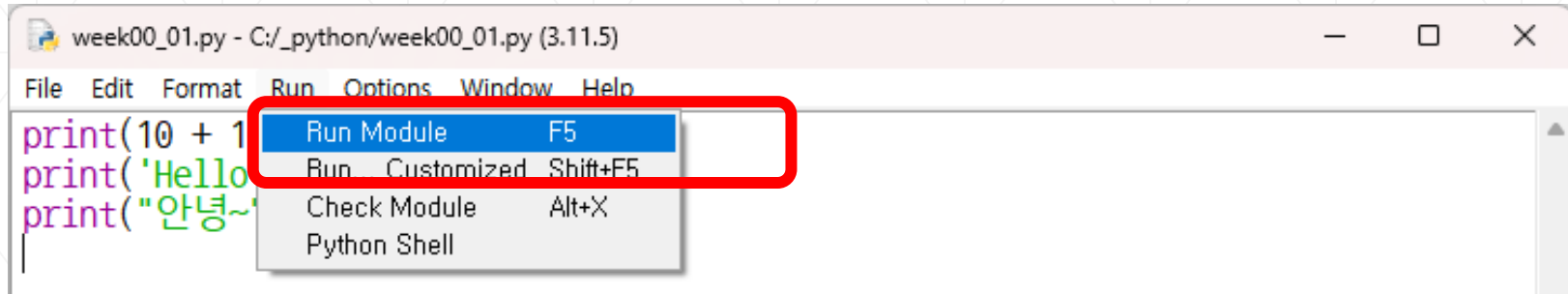
# IDLE로 파일 만들어 실행하기

The image illustrates the steps to create and save a Python file in IDLE. It shows the IDLE interface with the code:

```
print(10 + 10)
print('Hello ' * 3)
print("안녕~")
```

The **File** menu is open, and the **Save** option (Ctrl+S) is selected. A **Save As** dialog box is shown, with the file name set to `week00_01.py` and the file type set to `Python files`. The file is saved in the directory `C:/_python/`. The resulting file `week00_01.py` is shown in the IDLE interface, displaying the same code.

# IDLE로 파일 만들어 실행하기



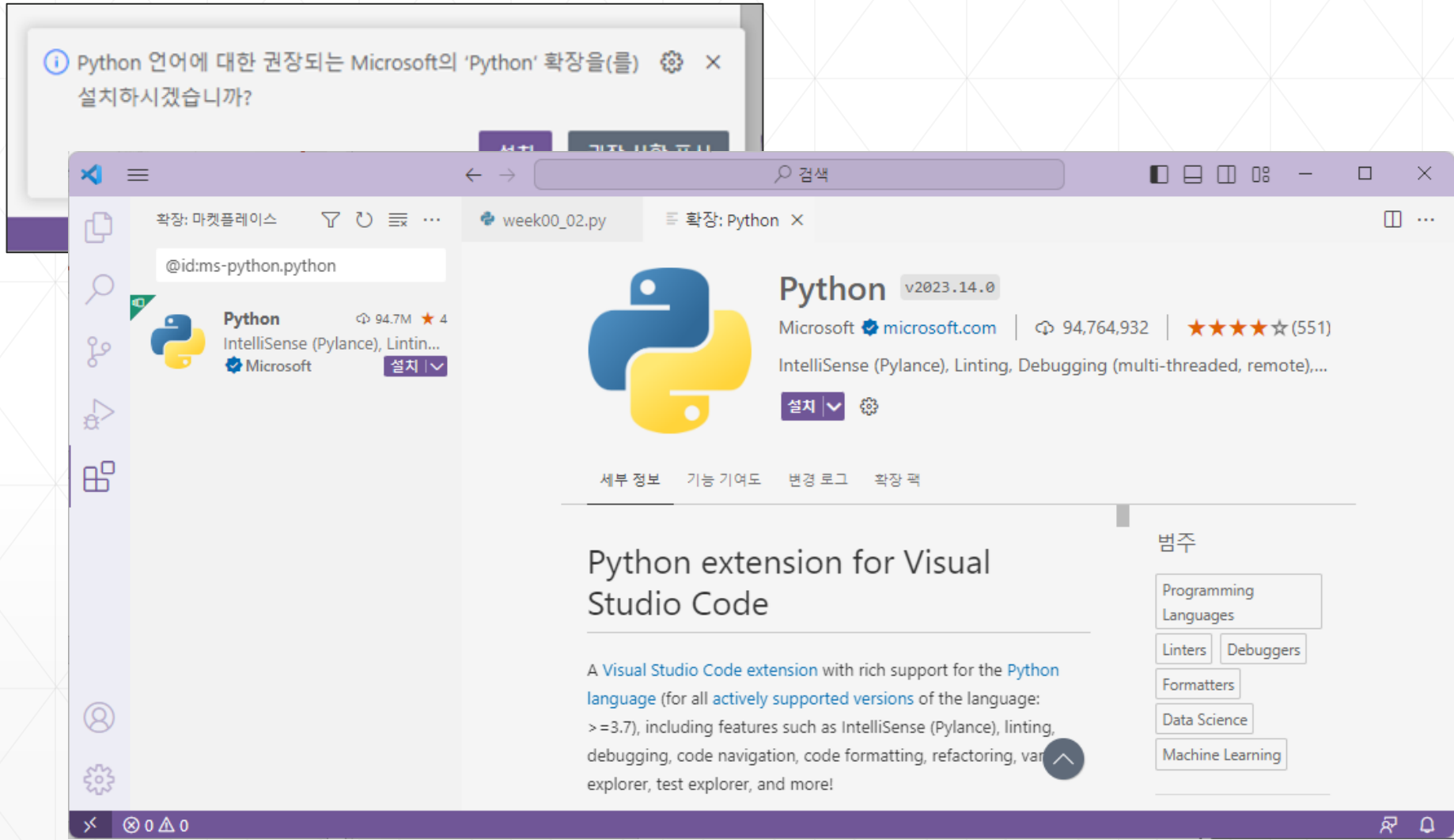
# Visual Studio Code

---

(1주차 참고)

반드시 앞의 Python Interpreter가 설치되어 있어야 함.

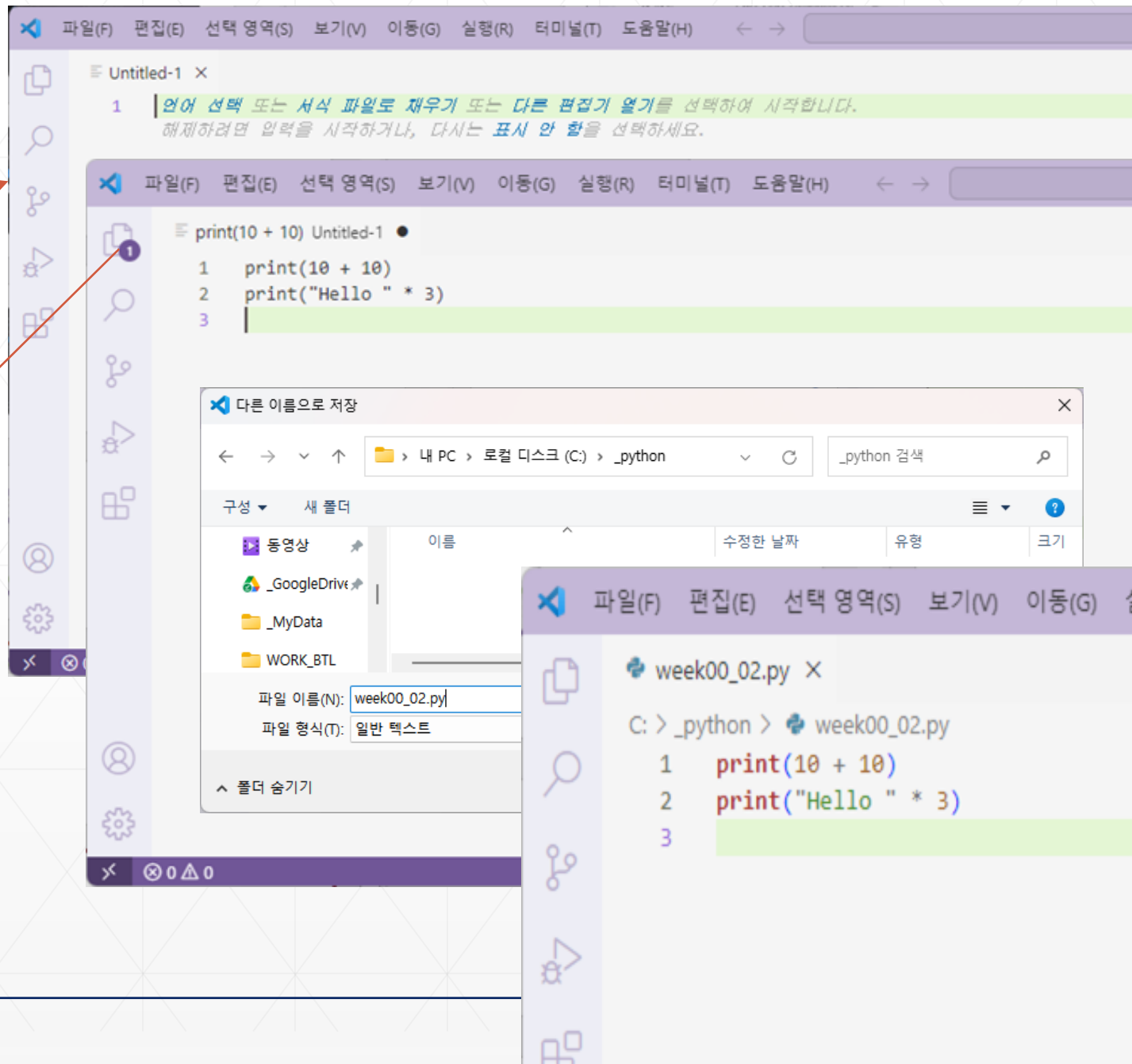
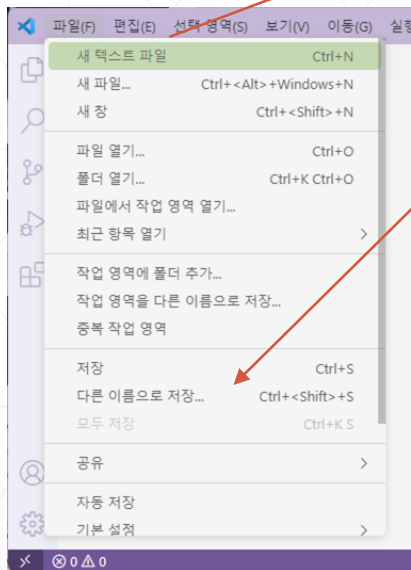
# 확장 설치 : Python (Microsoft)



# 실행

- 개인 컴퓨터나 학교 컴퓨터에 본인이 저장할 폴더를 만든다.
- 예) c:\python\_class

# 실행



# 1주차 핵심

- 설치환경 구성
- Compiler vs. Interpreter

# 2주차

---



# 목차 p.16

1. 파이썬 시작하기
  2. 자료형
  3. 조건문
  4. 반복문
  5. 함수
  6. 예외처리
  7. 모듈
  8. 클래스
- 파일처리
  - GUI 프로그램

# 목차 키워드

- 파이썬 시작하기
  - 파이썬이란
  - 실행환경
  - 파이썬 용어
    - 문장, 표현식, 키워드, 식별자, 주석, 연산자, 값(data)
    - print() 함수
- 자료형
  - 기본자료형
    - type() 함수
  - 문자열
    - 연산자
    - 인덱싱 & 슬라이싱
    - 기타
  - 숫자
    - 종류: 정수, 실수
    - 연산자 : 사칙연산자 , //, %, \*\*
  - 변수
    - 변수
    - 복합 대입 연산자
    - input() 함수
    - 형변환 ( 숫자 - 문자열 )
  - 문자열과 관련한 함수

- case (p.73)
  - PascalCase
  - snake\_case
- literal

# [추가 p.85] 기본 제공 자료형

그룹	자료형	생성자	리터럴 표현	내용
수치형 (Number)	<b>Integer</b>	<b>int()</b>	<b>1</b>	정수
	<b>Floating point number</b>	<b>float()</b>	<b>1.1</b>	실수
	Complex Number	complex()	2 + 3j	복소수
논리형 (Boolean)	Boolean	bool()	True	참(True)과 거짓(False)만 값을 가짐
군집형 (Collection)	<b>String</b>	<b>str()</b>	<b>"1" 혹은 '1'</b>	<b>문자열</b> (순서 O, 수정 X, 중복 O, 구성요소한정 O)
	List	list()	[1,2]	리스트 (순서 O, 수정 O, 중복 O, 구성요소한정 X)
	Tuple	tuple()	(1,2)	튜플 (순서 O, 수정 X, 중복 O, 구성요소한정 X)
	Set	set()	{1,2}	집합, 세트 (순서 X, 수정 O, 중복 X, 구성요소한정 X)
	Dictionary	dict()	{1:"1", 2:"2"}	사전, 딕셔너리 (순서 X, 수정 O, 중복 X, 구성요소한정 X) {key:value, ... , key <sub>n</sub> :value <sub>n</sub> }

모든 타입은 class로 만들어진 것