

컴퓨터정보과 파이선 프로그래밍

01주차 강의 소개 및 개발환경 구축

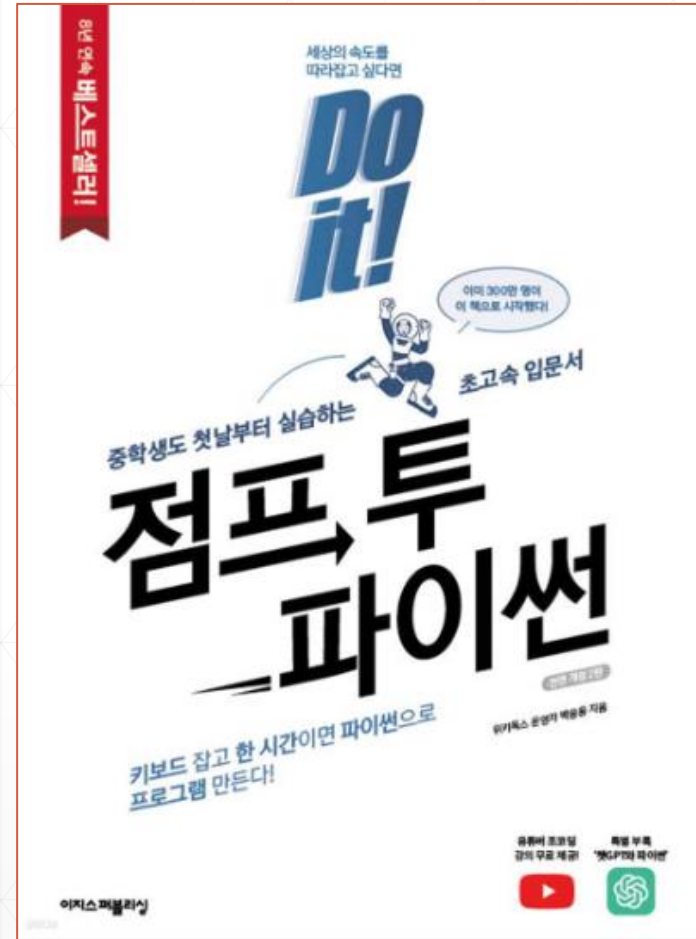
기본 정보

- 장은미
 - rfeater@gmail.com

- 강의실 4-403
 - A반 수 10-12교시
 - B반 수 2-4교시
 - C반 수 6-8교시

- 교재 – 점프 투 파이썬 (개정2판)
 - 박응용, 이지스퍼블리싱
 - YOUTUBE 강의 지원 교재

- 성적처리 (100점 기준)
 - 출석 20 (감점 : 결석(3) / 지각(1/시간당))
 - 과제 10
 - 중간 35 (지필)
 - 기말 35 (지필 또는 실습)

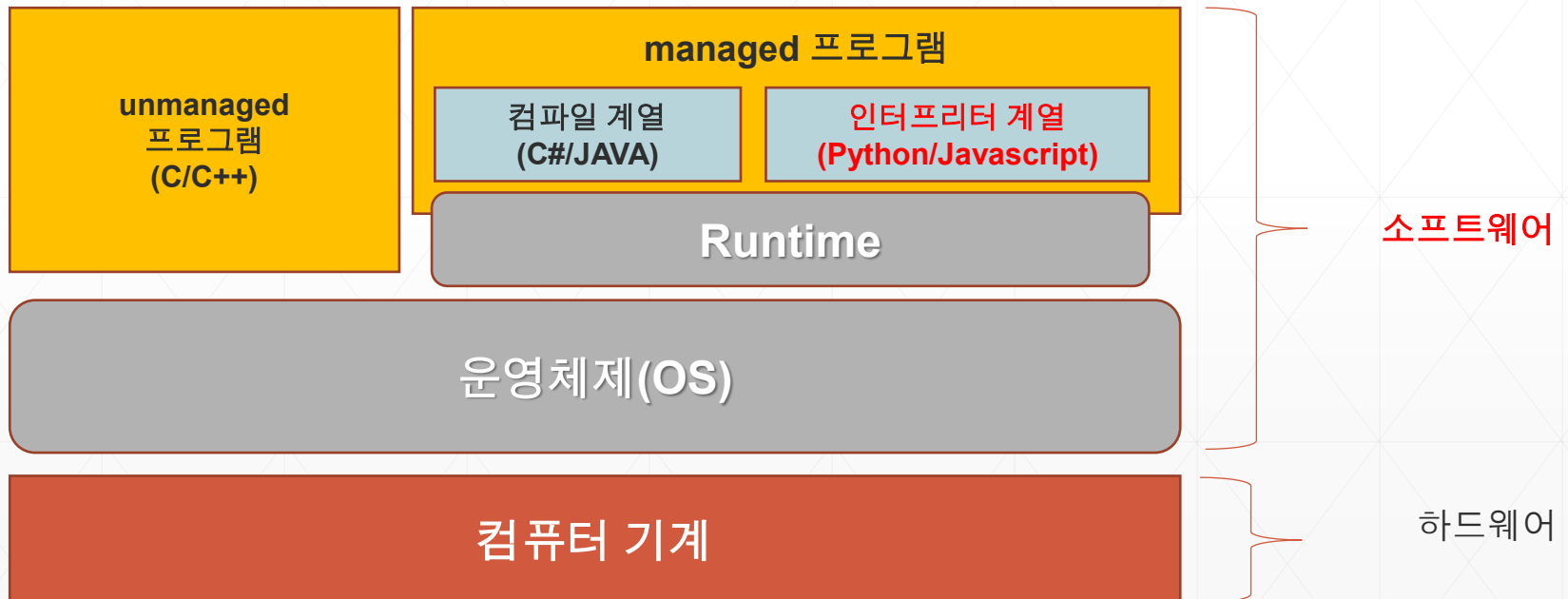


강의진행계획

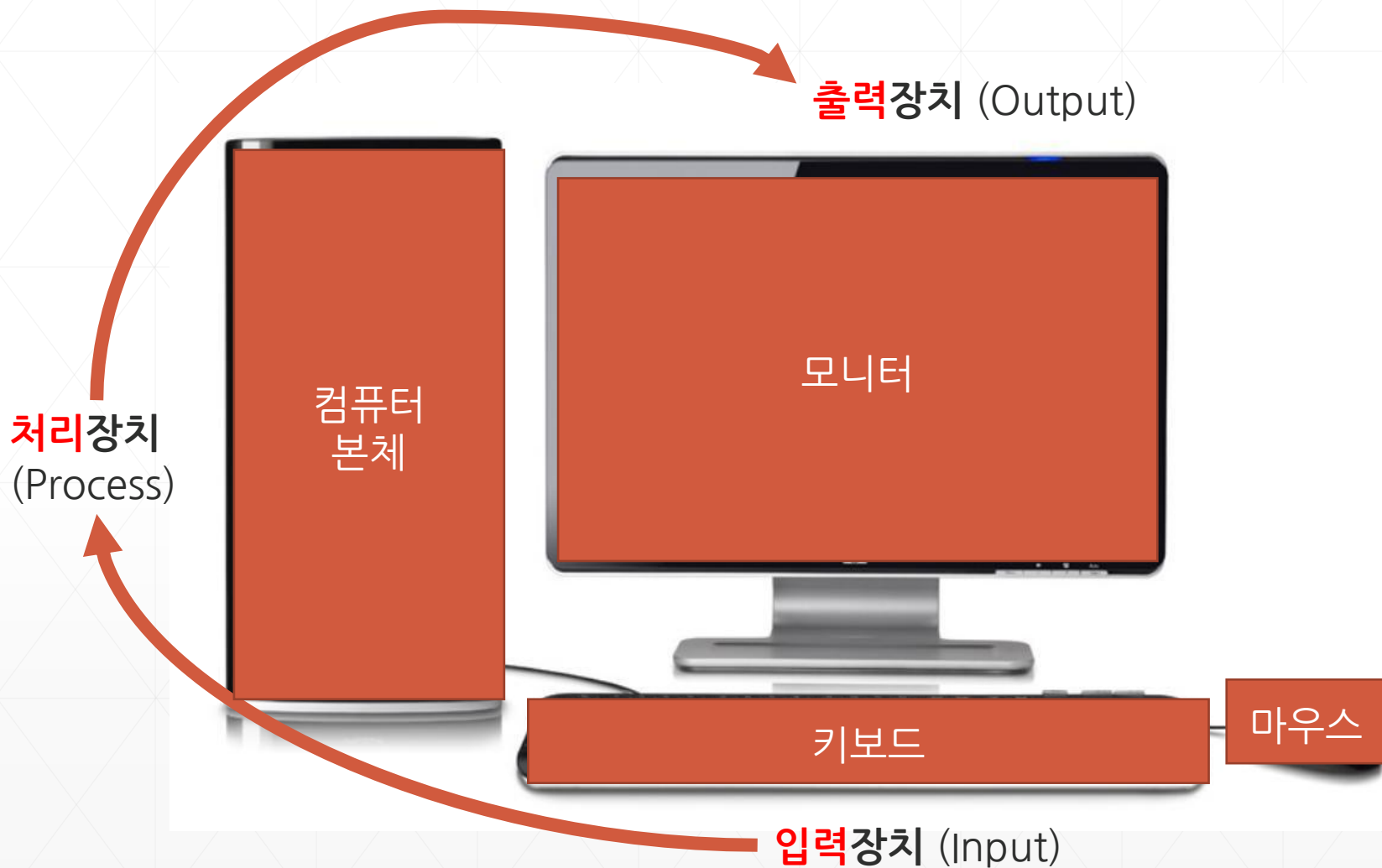
- 파이선을 통해 전반적인 프로그래밍의 기초지식을 습득하고, 논리적인 사고를 기를 수 있도록 이론과 실습을 진행한다.
 - 상세 내용은 강의계획서 참고
-

프로그래밍 개념

- 컴퓨터 = 하드웨어 + **소프트웨어**(프로그램)
 - 하드웨어
 - 중앙처리장치 \leftrightarrow 기억장치 \leftrightarrow 저장장치
 - 소프트웨어(프로그램) : 컴퓨터를 **범용적**으로 만드는 것
 - 컴퓨터는 프로그램만 바꾸어 주면 다양한 작업을 할 수 있다.



프로그램 실행 구조 (1)



프로그램 실행 구조 (2)

입력장치

KEYBOARD

2과 5를 더해서 모니터에 출력하세요.

정수 2을 저장한다.
정수 5를 저장한다.
저장한 정수 두 개를 덧셈 연산한다.
연산 결과를 저장한다.
저장한 결과를 화면에 출력한다.

처리장치

중앙처리장치
CPU

Register

기억장치

Memory
(RAM/ROM)

폰 노이만구조
프로그래밍 내장 방식

100	2
104	5
108	7
2010	100의 값과 104의 값을 더하세요.
2014	더한 값을 108에 넣어주세요.
2018	108의 값을 모니터로 출력하세요.

file 단위로 저장

보조기억장치

Storage
(HDD/SSD)

출력장치

MONITOR

프로그램에서 필요한 개념

- 데이터를 처리하여 원하는 정보를 생성한다.

I P O

- 값(데이터/정보)를 보관 : **I, O**
 - 메모리(RAM)에서 보관
 - 프로그램 내에서는 변수를 통해서 메모리에 데이터를 저장하고 접근함
- 처리 : **P**
 - 연산자와 제어문
 - 산술연산 및 기타 다양한 연산자
 - 선택문, 반복문 등...

컴파일러 VS 인터프리터

- 컴파일러 언어

- C LANGUAGE

- 원시(소스)코드 → 컴파일러 → 목적(기계, 머신)코드 → 실행
 - 통째로 실행가능한 코드로 만든 후 실행
 - 소스 코드 해석에 많은 시간이 소모
 - 빠른 실행 속도

- 인터프리터 (일종의 Runtime)

- PYTHON

- 원시(소스)코드 → 한 줄 실행 → 원시(소스)코드 → 한 줄 실행 → ...
 - 한 줄 씩 실행
 - 소스 코드 해석에 적은 시간
 - 느린 실행 속도

- ❖ 실행환경에 따른 분류

- Unmanaged
 - Managed

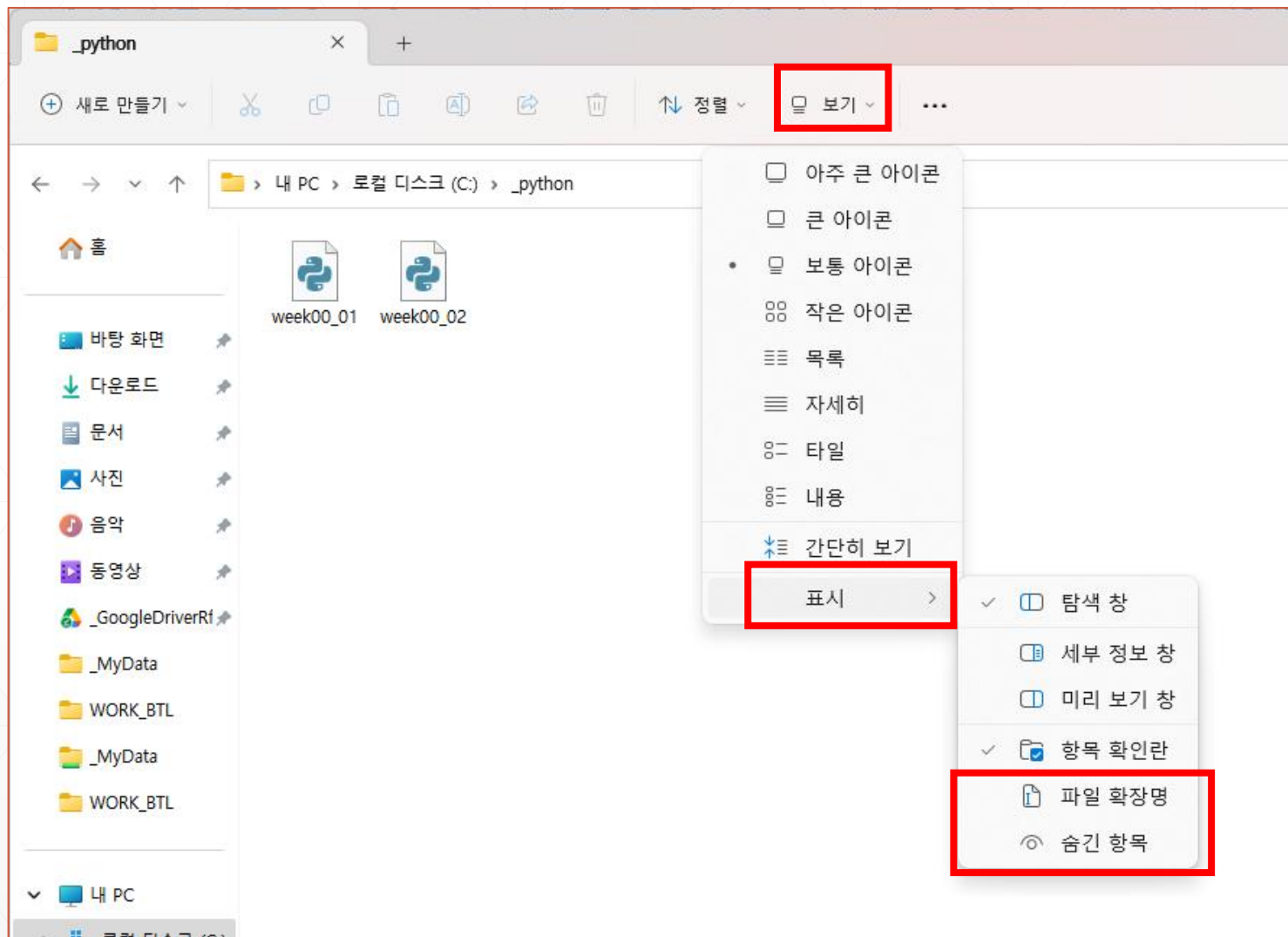
[개발환경 구축] 파이썬 인터프리터 및 편집 기 설치

(2025.08 기준)

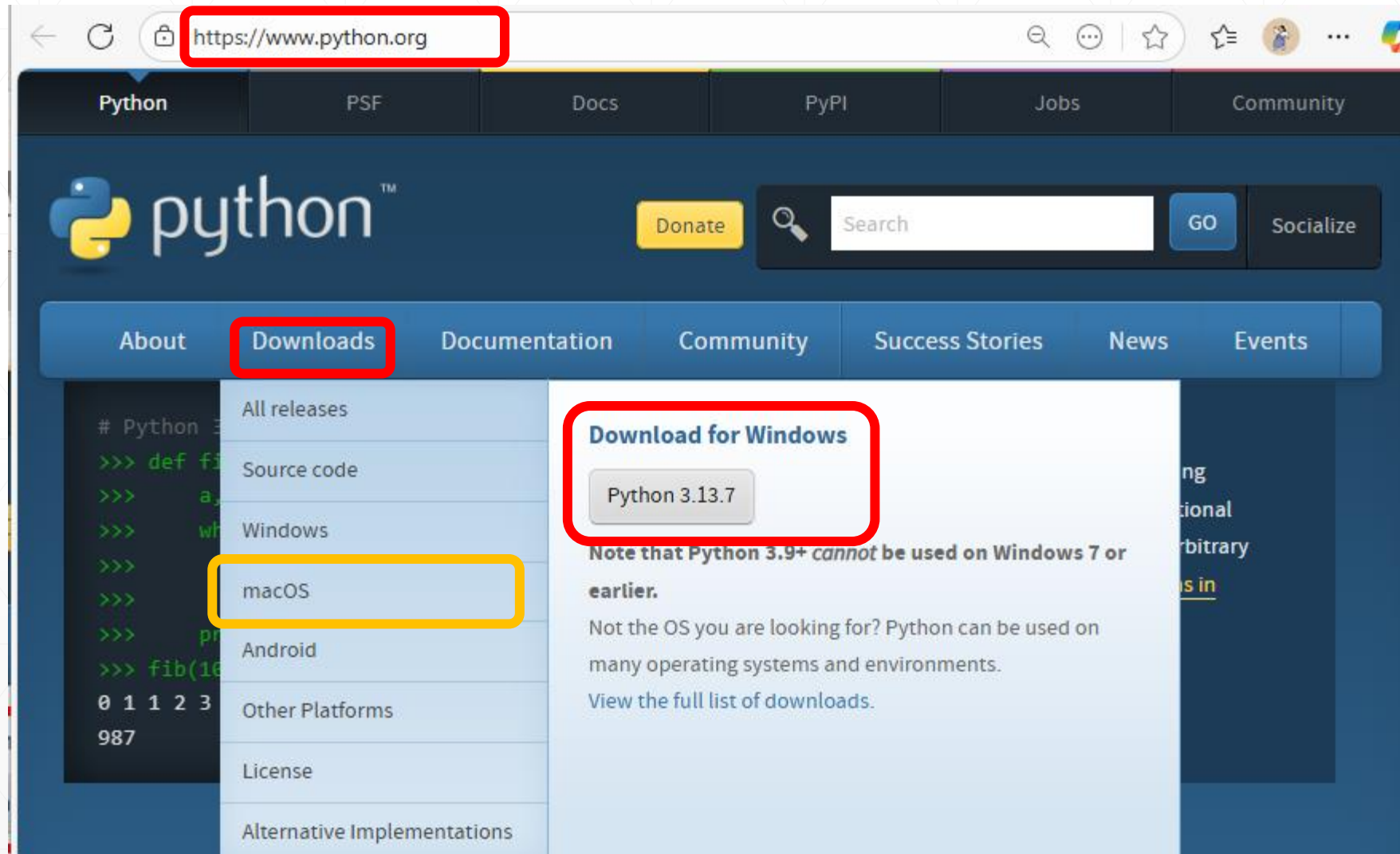
기본 수업 환경

- Windows 11 기준으로 진행
- 나머지 환경(LINUX, MAC OS, ..) 은 스스로 구축하기

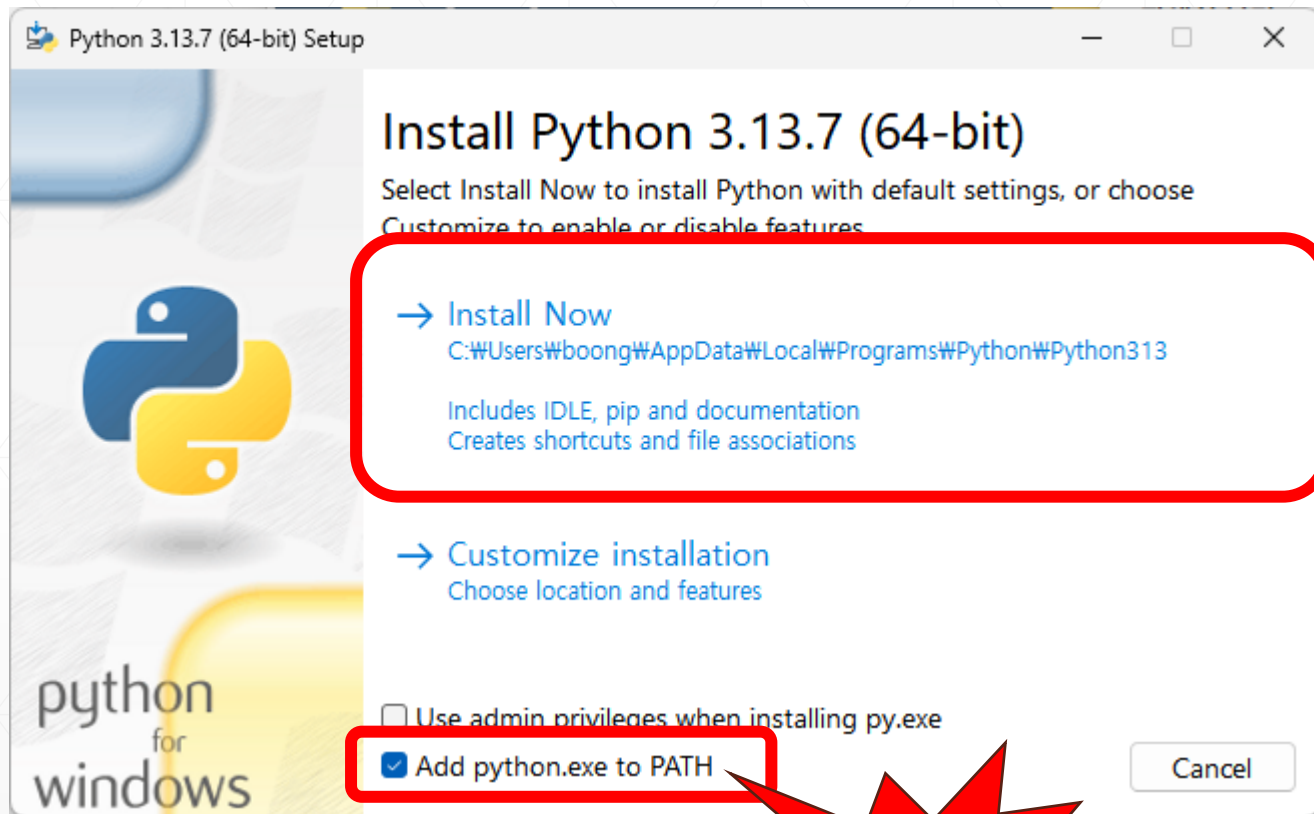
폴더설정



다운로드 : <https://www.python.org/>

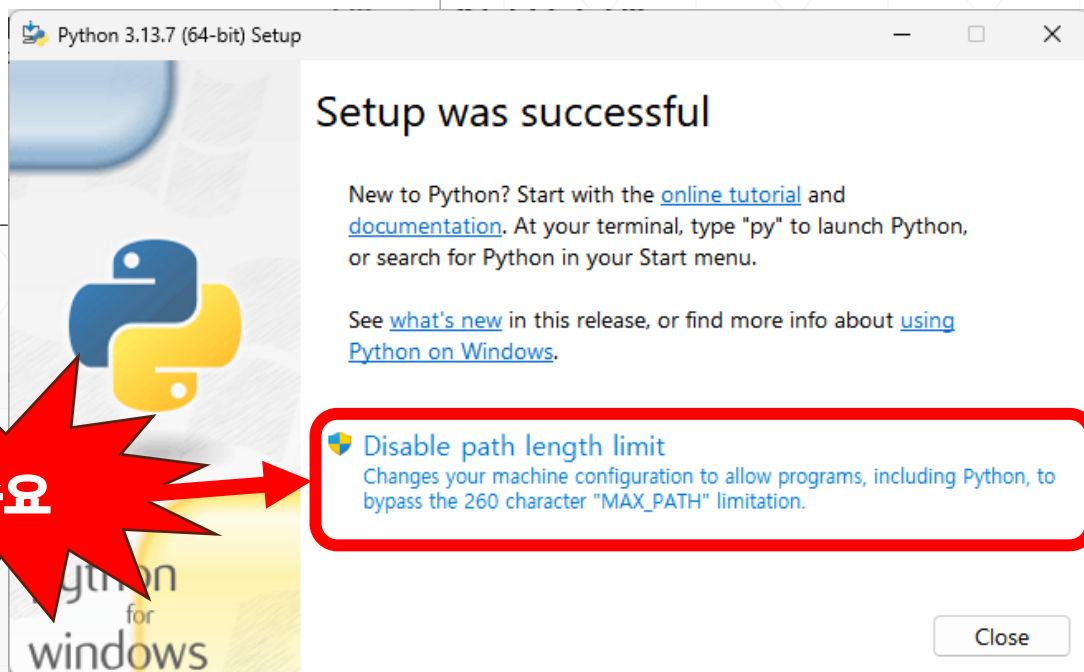
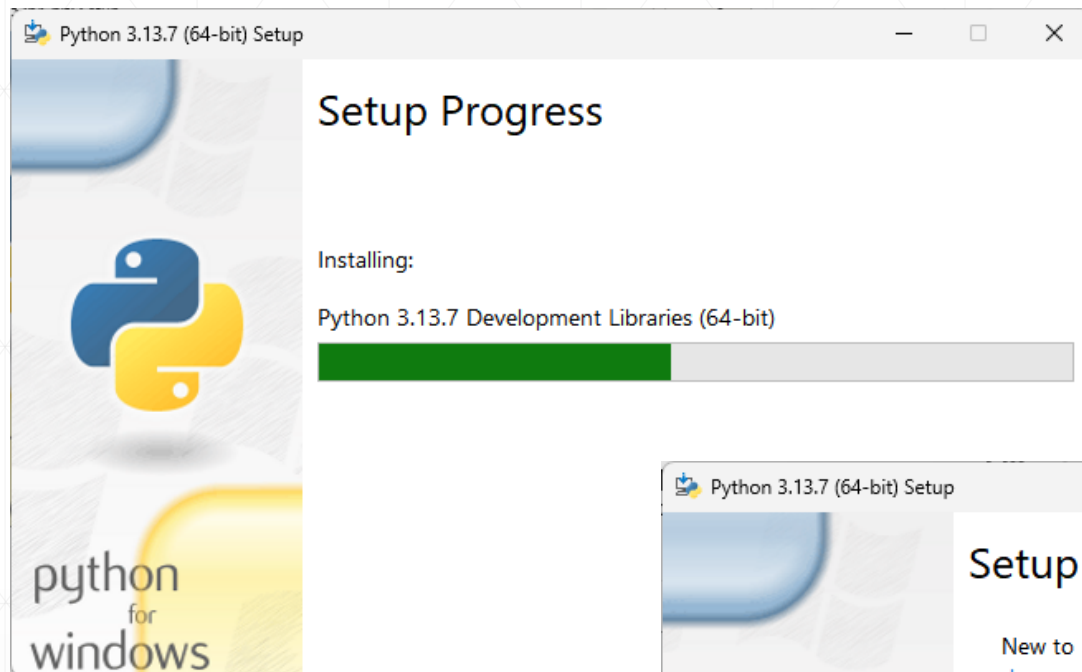


설치

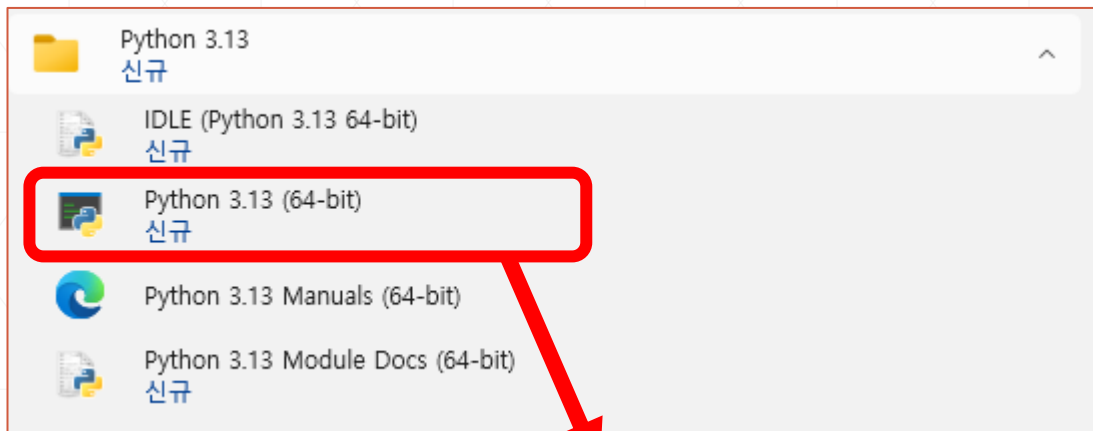


중요

설치



설치 완료 및 테스트



```
Python 3.13 (64-bit) × + v
Python 3.13.7 (tags/v3.13.7:bcee1c3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bi
Type "help", "copyright", "credits" or "license" for more information.
>>> |
```

설치 완료 및 테스트

 Python 3.13 (64-bit) X + v

```
Python 3.13.7 (tags/v3.13.7:bcee1c3, Aug 14 2025, 14:00:00) [AMD64]  
Type "help", "copyright", "credits" or "license()" for more
```

```
>>> a = 30 * 2
```

```
>>> print(a)
```

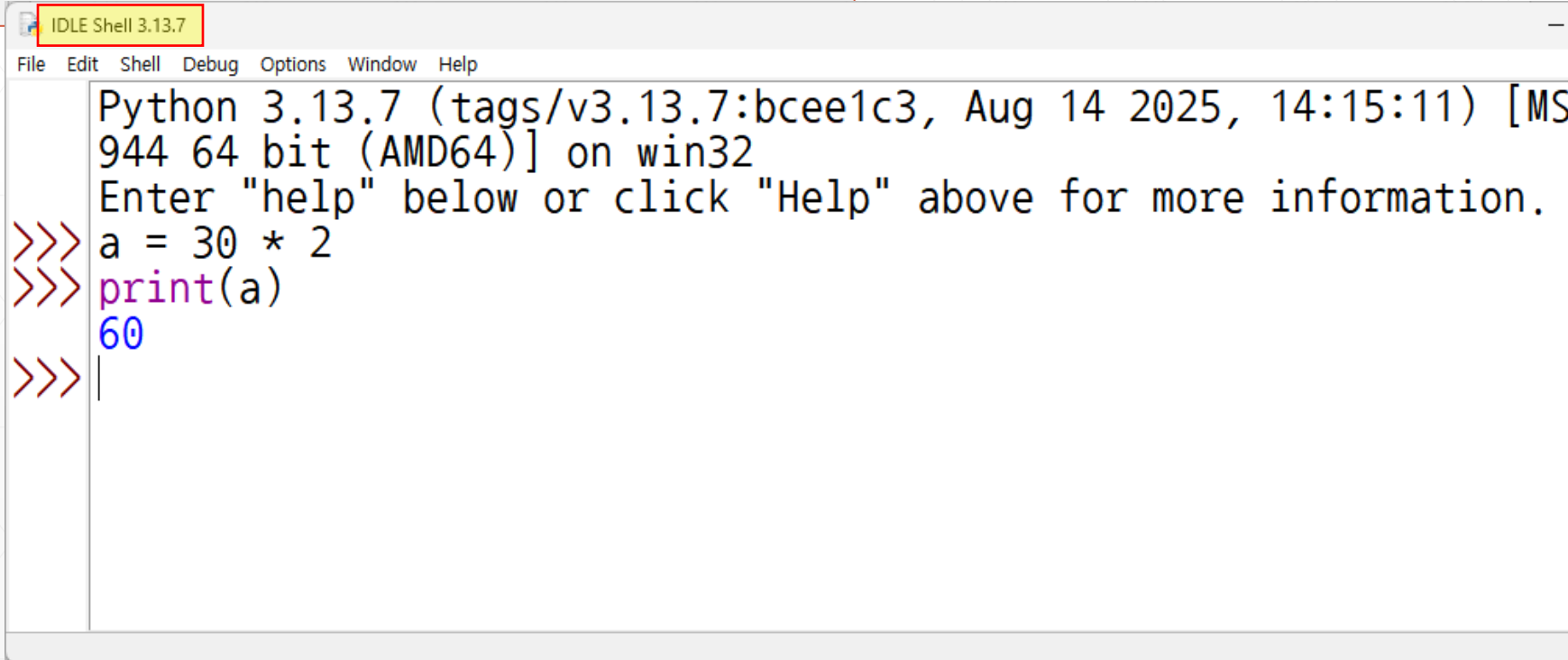
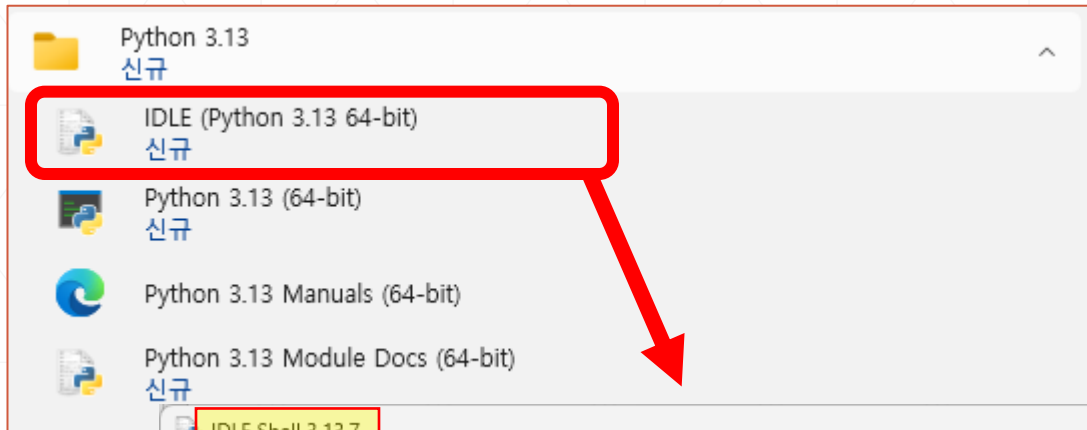
```
60
```

```
>>> |
```


관리 환경 정리

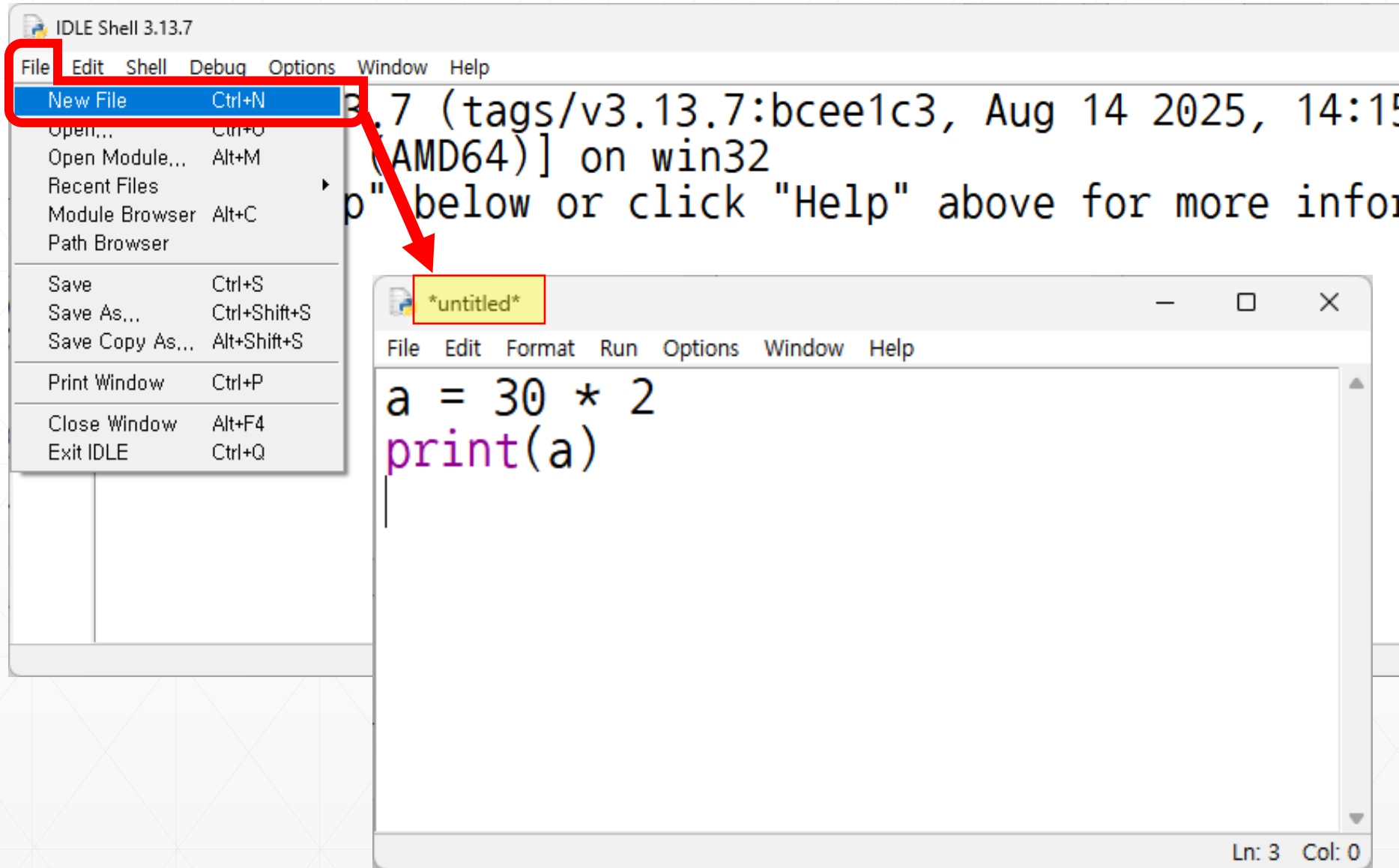
- 개인 컴퓨터나 학교 컴퓨터에 본인이 저장할 폴더를 만든다.
 - 예) c:\python_class
 - 예) d:\python_A_20240001
- 작업이 끝나면 개인 저장소에 저장한다.

IDLE 사용하기

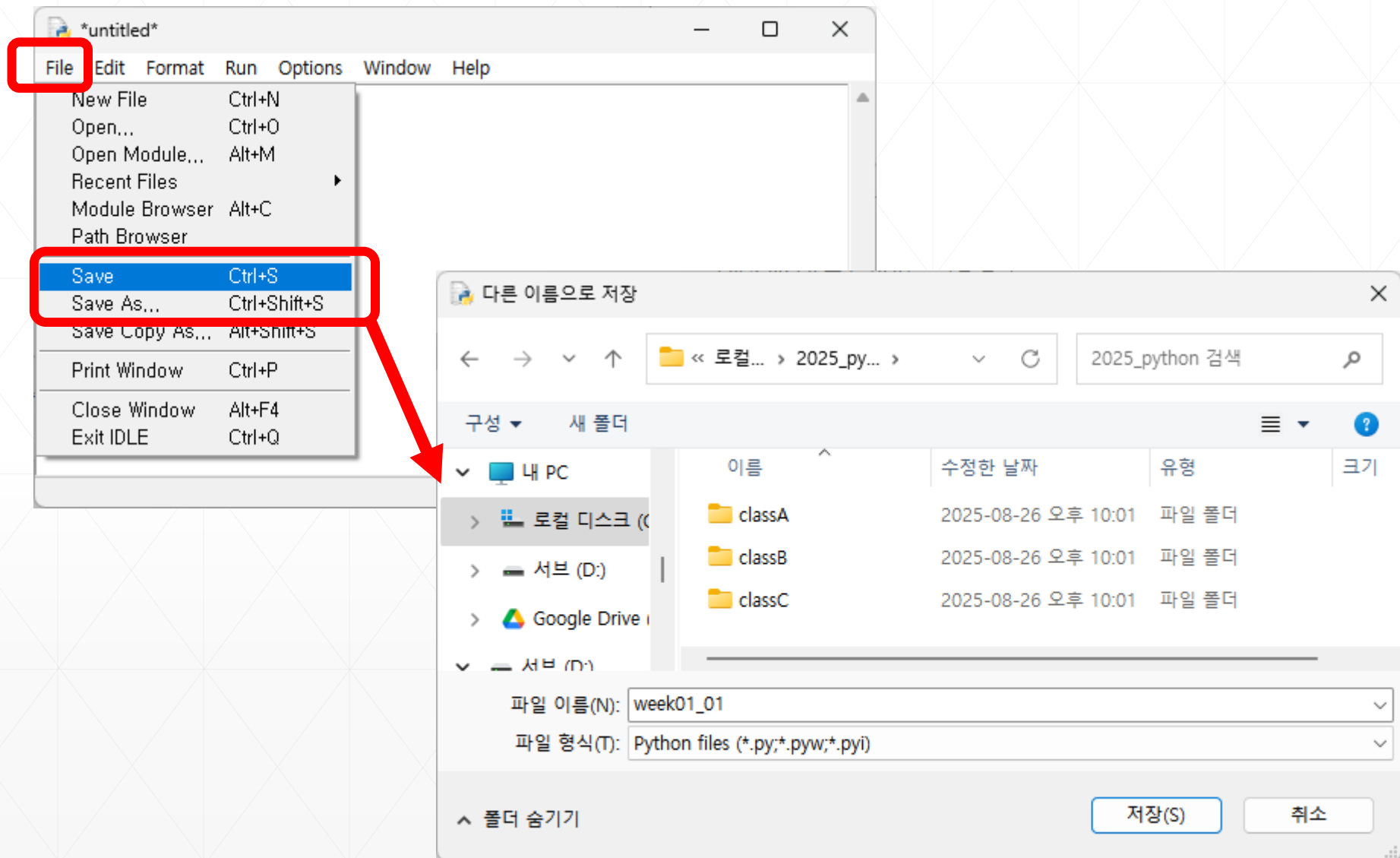


```
Python 3.13.7 (tags/v3.13.7:bcee1c3, Aug 14 2025, 14:15:11) [MS
944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> a = 30 * 2
>>> print(a)
60
>>> |
```

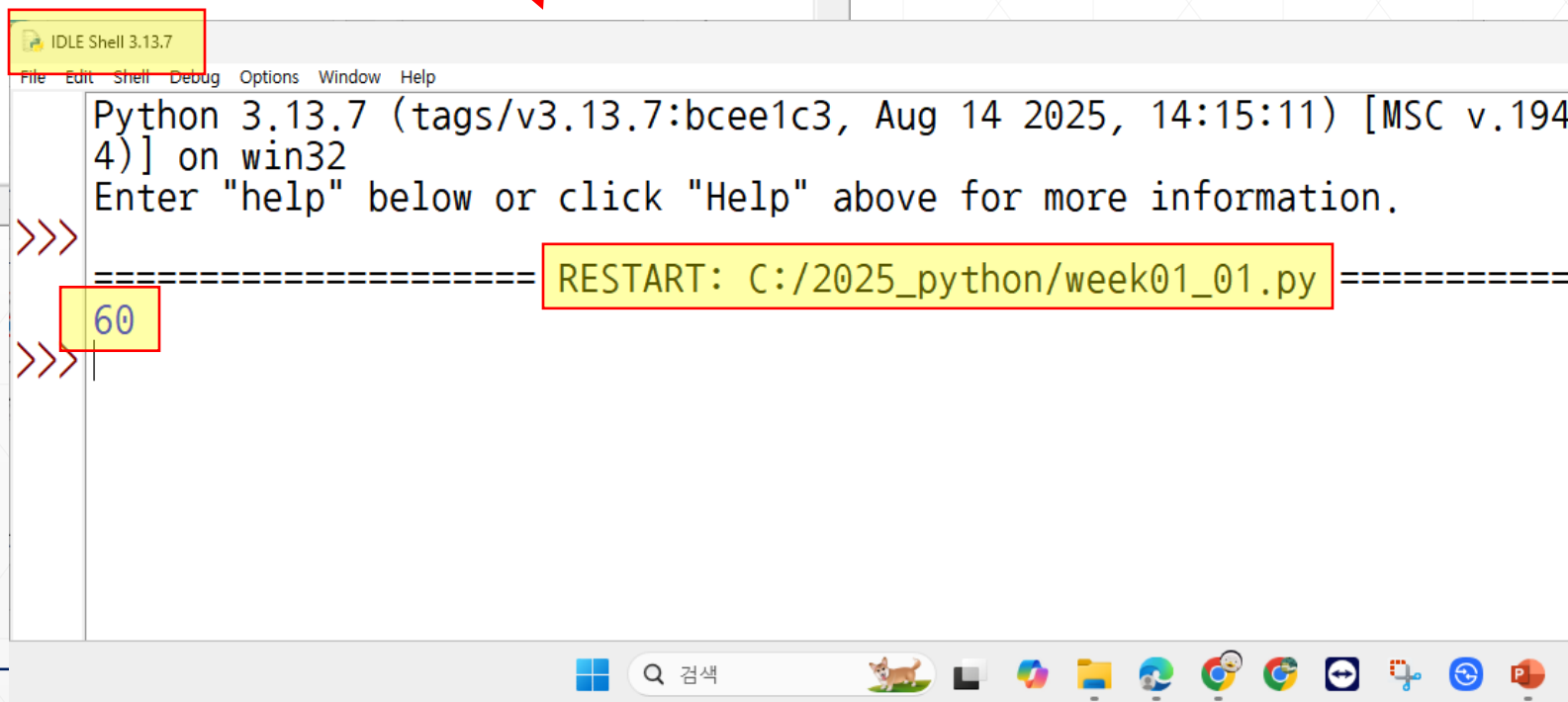
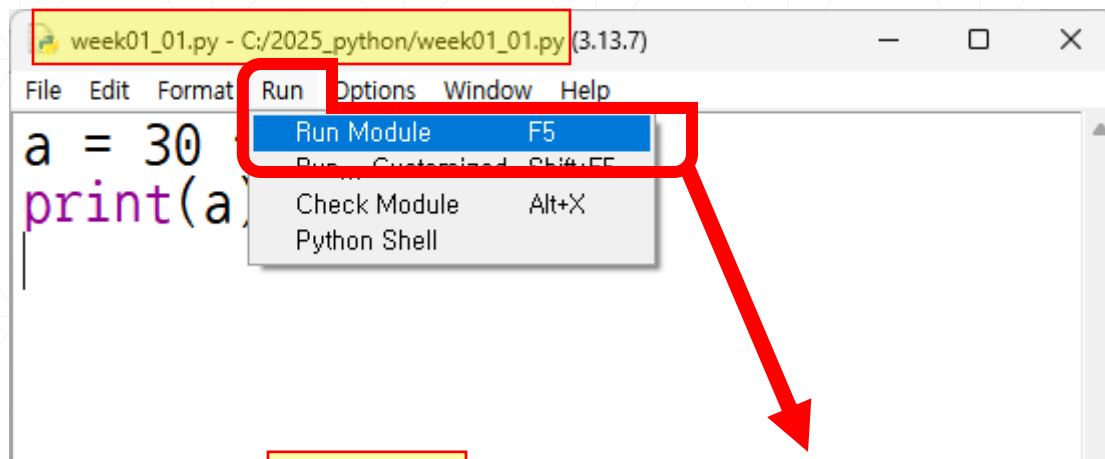
IDLE로 파일 만들어 실행하기



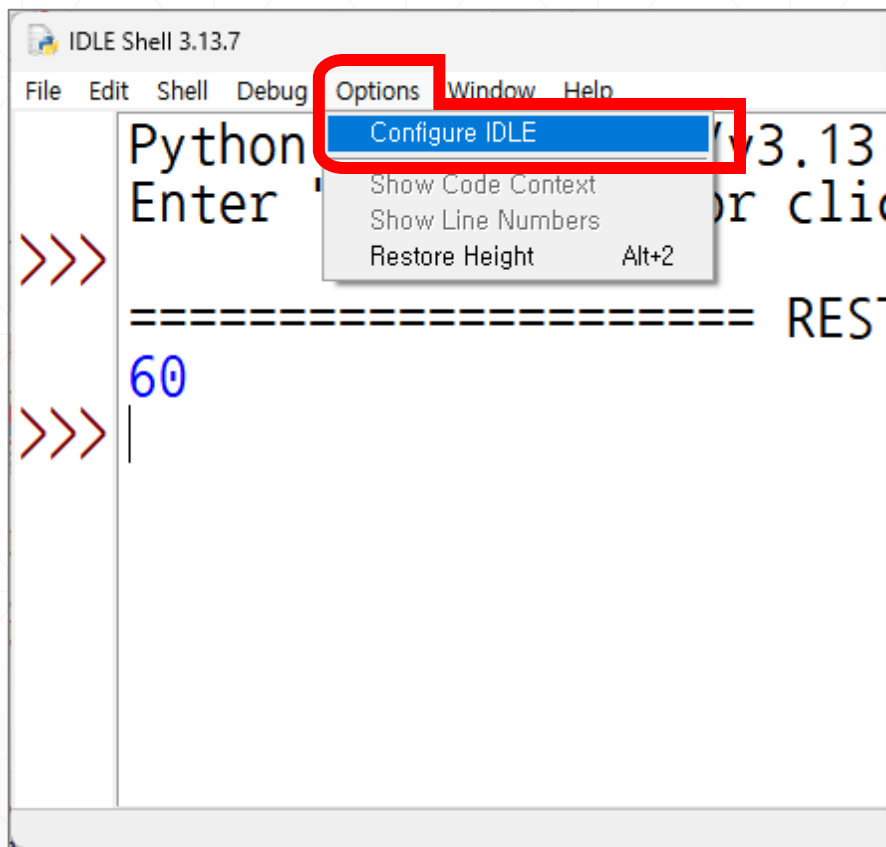
IDLE로 파일 만들어 실행하기



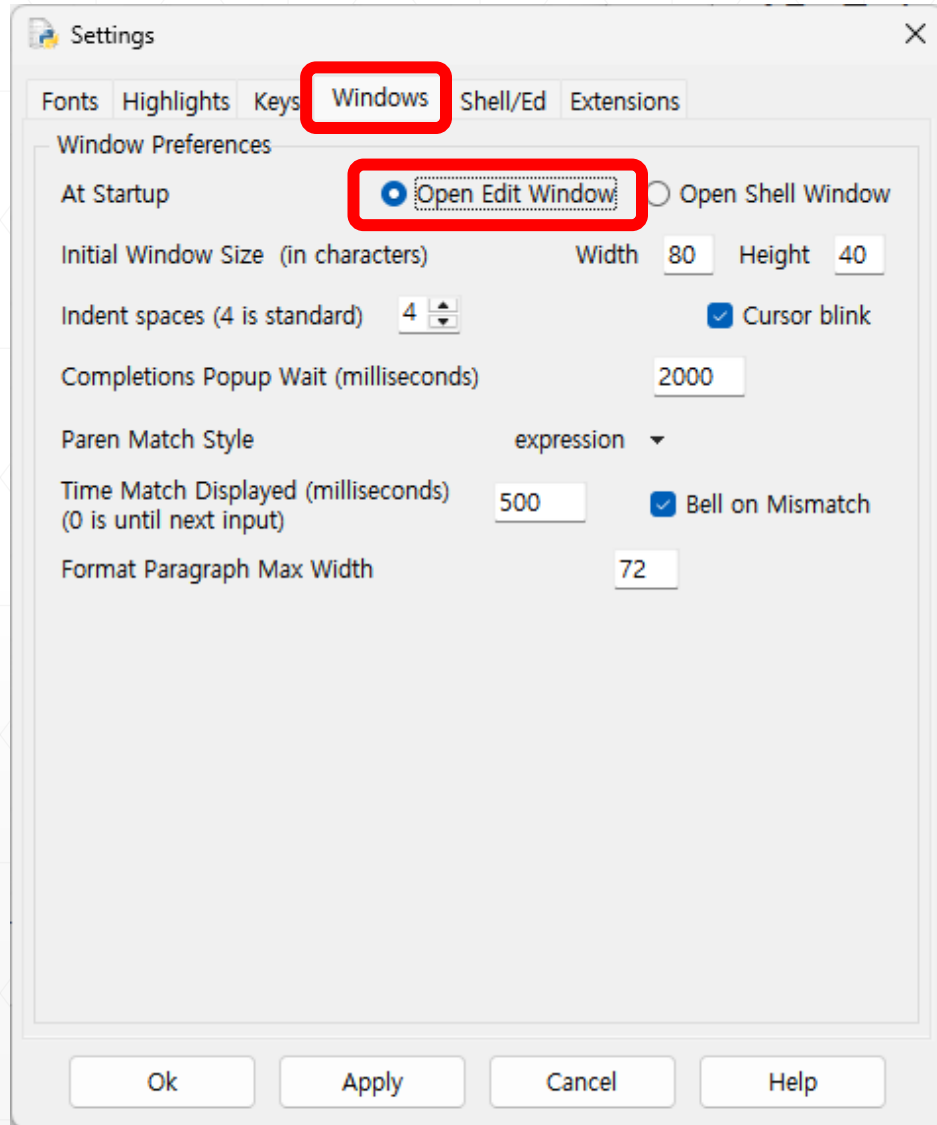
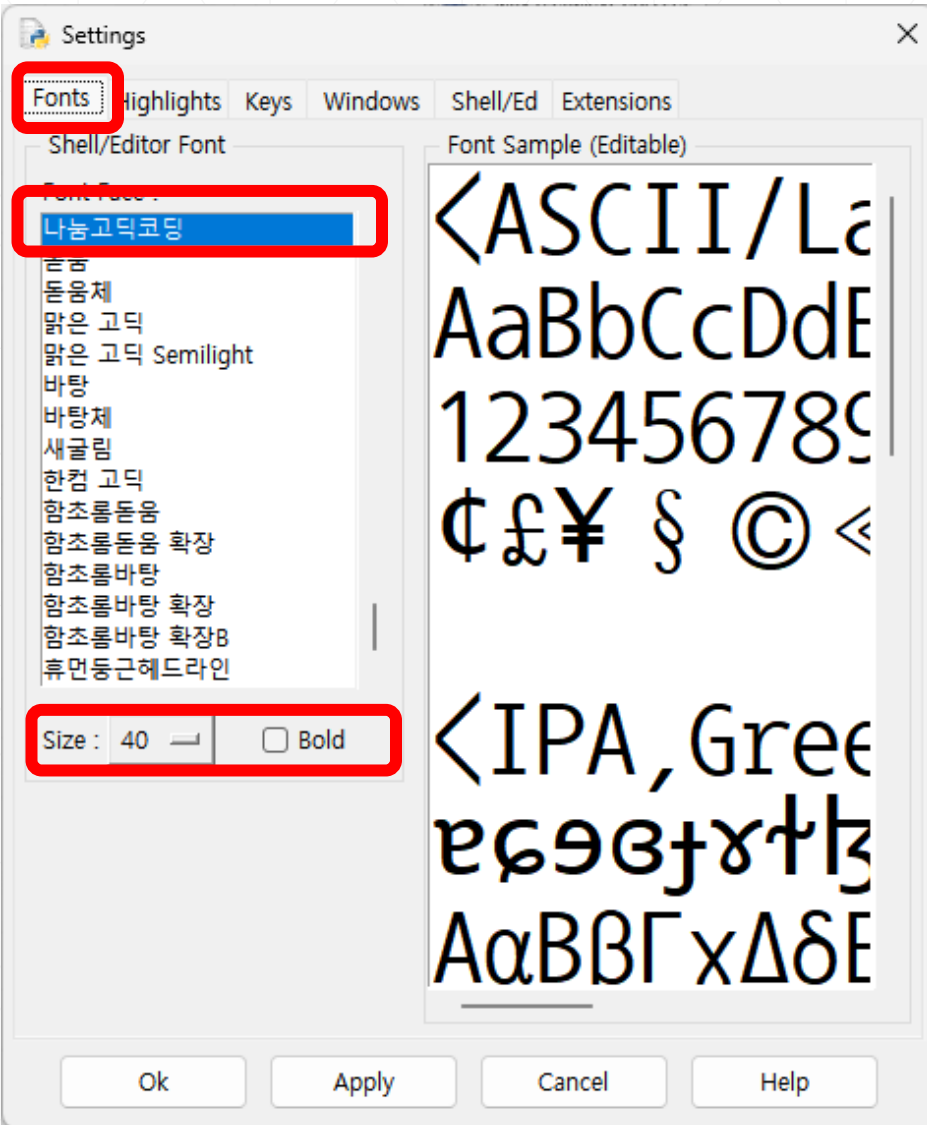
IDLE로 파일 만들어 실행하기



IDLE 설정



IDLE 설정



Interpreter

- 인터프리터 : 코드를 한 줄씩 읽어 내려가면서 해석하고 실행하는 방식
 - 비교: 컴파일러 - 모든 코드를 읽어서 문제가 없으면, 실행할 수 있도록 하는 방식
- Shell 실행방식
 - 프롬프트(>>>)에서 한 줄씩 입력하여 실행하고 결과 보여주는 방식
 - 대화형 인터프리터라고 함
- Editor를 이용한 실행방식
 - 여러 번 사용하거나 다량의 코드를 한꺼번에 작성하고 실행하기 위할 때 필요
 - 코드가 여러 줄이어도, 한 줄씩 읽으면서 해석하고 실행하는 방식임
 - 파일(.py)로 저장하여 처리

IDE (통합 개발 환경)

- 기본 IDE : IDLE
- 전문 IDE
 - PyCharm
 - Visual Studio Code
 - Jupyter Notebook

일반적인 프로그래밍 상식

- 정수 : 1
 - 00000000 00000000 00000000 00000001 (4 bytes)
- 실수 : 1.0
 - 0 011111111111
000 (8 bytes)
- 문자 : '1'
 - 00110001 (1 byte)
- 문자열 : "1"
 - 00110001 00000000 (2 bytes)

일반적인 프로그래밍 상식

- 정수 : 100000000
 - 00000000 10011000 10010110 10000000 (4 bytes)
- 실수 : 100000000.0
 - 0 10000010110
0010110101101000011111100011110101110000101000111101 (8 bytes)
- ~~문자 : '100000000'~~ (표현 불가)
- 문자열 : "100000000"
 - 00110001 00110000 00110000 00110000 00110000 00110000 00110000
00110000 00000000 (9 bytes)