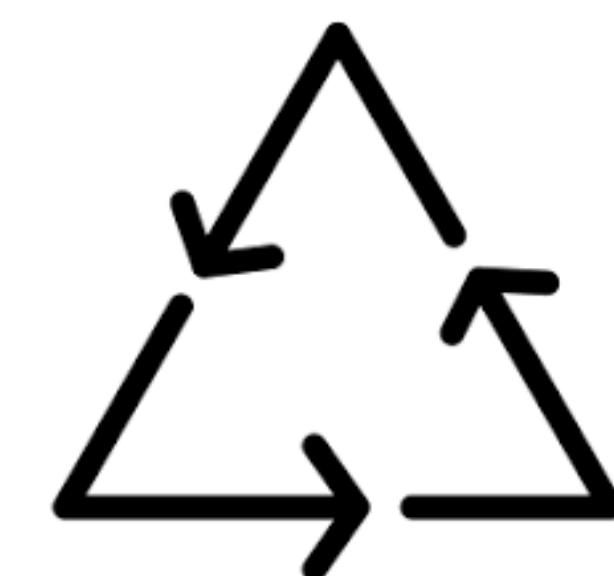


CHAPTER 2

파이썬 프로그래밍 환경

Python Programming Environment



박진수 교수
서울대학교·경영대학
jinsoo@snu.ac.kr



학습 목차

- 파이썬 개발 환경 설치하기
- 프로그래밍 툴
- Lab : IDLE 둘러보기
- 명령어 셀 실행 환경
- Lab : 셀 명령어 기초

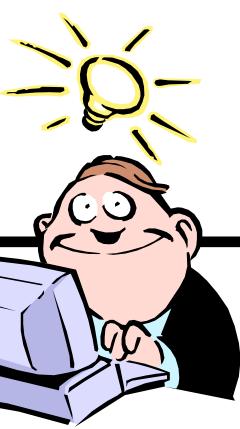
파이썬 개발 환경 설치하기

Python Setup





운영체제 비트 확인



● 윈도우 10의 경우

● [설정] → [시스템] → [정보]

The screenshot shows the Windows 10 Settings interface. On the left, the navigation pane is visible with options like Home, Settings search, System, Battery, Storage Space, Tablet Mode, Multi-tasking, PC Screen Mirroring, Shared Experience, and Desktop Experience. The 'Information' button at the bottom of the pane is highlighted with a red box. The main content area is titled '정보' (Information) and shows '디바이스 사양' (Device specifications). It lists the device as 'SAMSUNG PC' with the device name 'LAPTOP-5GJN5JSH'. The processor is listed as 'Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz'. RAM is '16.0GB'. The device ID is 'F6C80D77-17AC-4E4C-B71F-6E56D07A44A7'. The product ID is '00325-95800-00000-AAOEM'. The 'System Type' entry, which reads '64비트 운영 체제, x64 기반 프로세서' (64-bit operating system, based on x64 processor), is also highlighted with a red box. Below it, 'Pen & Touch' information states that pen or touch input is not supported. A button at the bottom right says '이 PC의 이름 바꾸기' (Change this PC's name).



운영체제 비트 확인



모든 윈도우 버전

- 화면 하단 [Windows 검색] 아이콘 선택 → '제어판' 입력 → [제어판] → [시스템 및 보안] 또는 [모든 제어판 항목] → [시스템]

The screenshot shows the Windows 10 System window. On the left, there's a sidebar with links like '장치 관리자', '원격 설정', '시스템 보호', and '고급 시스템 설정'. The main area has a title '컴퓨터에 대한 기본 정보 보기'. It shows 'Windows 버전' as 'Windows 10 Home' and the copyright notice '© 2018 Microsoft Corporation. All rights reserved.' Below that is a large 'Windows 10' logo. The '시스템' section lists the processor as 'Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz', RAM as '16.0GB', and the system type as '64비트 운영 체제, x64 기반 프로세서'. The 'SAMSUNG' logo is visible in the bottom right corner.

프로세서:	Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz
설치된 메모리(RAM):	16.0GB
시스템 종류:	64비트 운영 체제, x64 기반 프로세서
펜 및 터치:	이 디스플레이에 사용할 수 있는 펜 또는 터치식 입력이 없습니다.



32비트 운영체제 파일 설치 파일 내려받기



- 본 강좌의 실습을 위해 파이썬 최신 버전을 설치

- <https://www.python.org/downloads/>

The screenshot shows the Python.org homepage with a focus on the Windows download section. The top navigation bar includes links for Python, PSF, Docs, PyPI, Jobs, and Community. Below the navigation is the Python logo and a search bar with buttons for 'Donate', 'Search', 'GO', and 'Socialize'. A secondary navigation bar below the main one offers links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a large yellow button with the text 'Download the latest version for Windows' and a sub-button 'Download Python 3.9.6' which is highlighted with a red border. To the right of the text is a graphic of two parachutes descending from clouds, each carrying a wooden crate.

Download the latest version for Windows

Download Python 3.9.6

Looking for Python with a different OS? Python for [Windows](#), [Linux/UNIX](#), [Mac OS X](#), [Other](#)

Want to help test development versions of Python? [Pre-releases](#)

Looking for Python 2.7? See below for specific releases

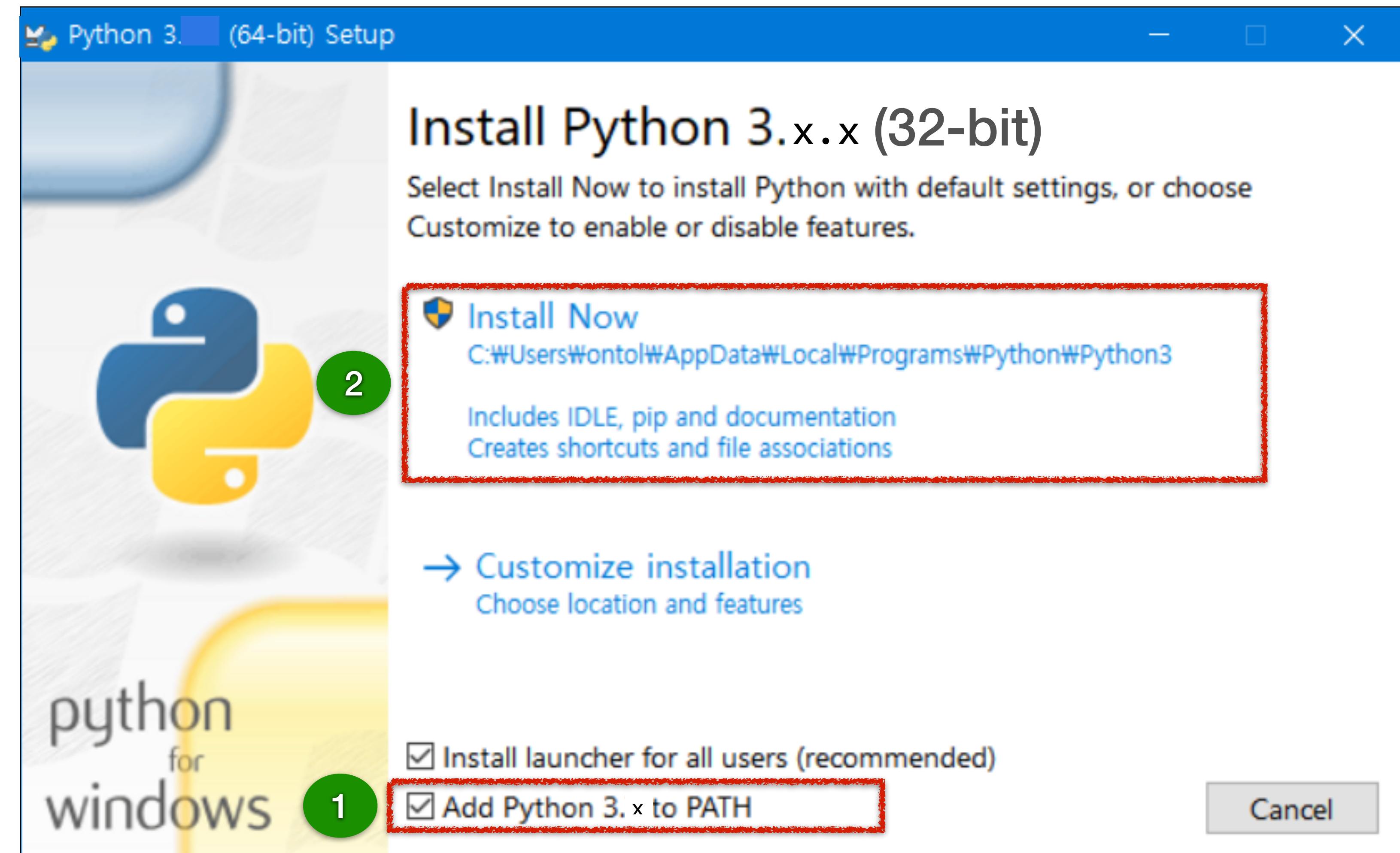


Windows

파이썬 설치하기 : 32비트 운영체제



- [Add Python 3.x to PATH]를 클릭해 체크하고 [Install Now]를 눌러 설치 시작
- 파이썬 설치 경로를 잘 기억해 둔다





64비트 운영체제 파일 설치 파일 내려받기



- 본 강좌의 실습을 위해 파이썬 최신 버전을 설치

- <https://www.python.org/downloads/>

The screenshot shows the Python.org homepage with a focus on the Windows download section. The top navigation bar includes links for Python, PSF, Docs, PyPI, Jobs, and Community. Below the navigation is the Python logo and a search bar with buttons for 'Donate', 'Search', 'GO', and 'Socialize'. A secondary navigation bar below the main one offers links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a large yellow call-to-action button labeled 'Download Python 3.9.6'. To its right, text says 'Looking for Python with a different OS? Python for Windows, Linux/UNIX, Mac OS X, Other'. Further down, it asks if you want to 'help test development versions of Python?' and provides a link to 'Pre-releases'. At the bottom, it mentions 'Looking for Python 2.7? See below for specific releases'. The background of the page features a cartoon illustration of two boxes descending from the sky on parachutes.



Windows

64비트 운영체제 파일 설치 파일 내려받기



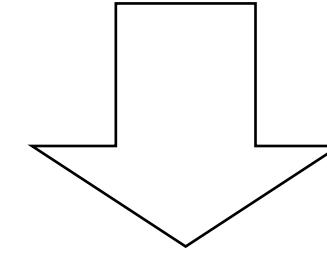
- Download [Windows embeddable package \(32-bit\)](#)
- Download [Windows embeddable package \(64-bit\)](#)
- Download [Windows help file](#)
- Download [Windows installer \(32-bit\)](#)
- [**Download Windows installer \(64-bit\)**](#)



64비트 운영체제 파이썬 설치하기

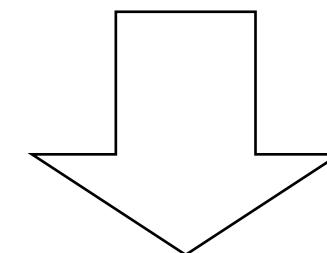


셋업 메뉴에서 하단 [Customize installation] 클릭



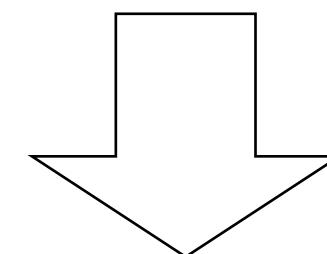
[Option Features]

- (1) 옵션을 모두 선택한 후
- (2) [Next] 버튼 클릭



[Advanced Options]

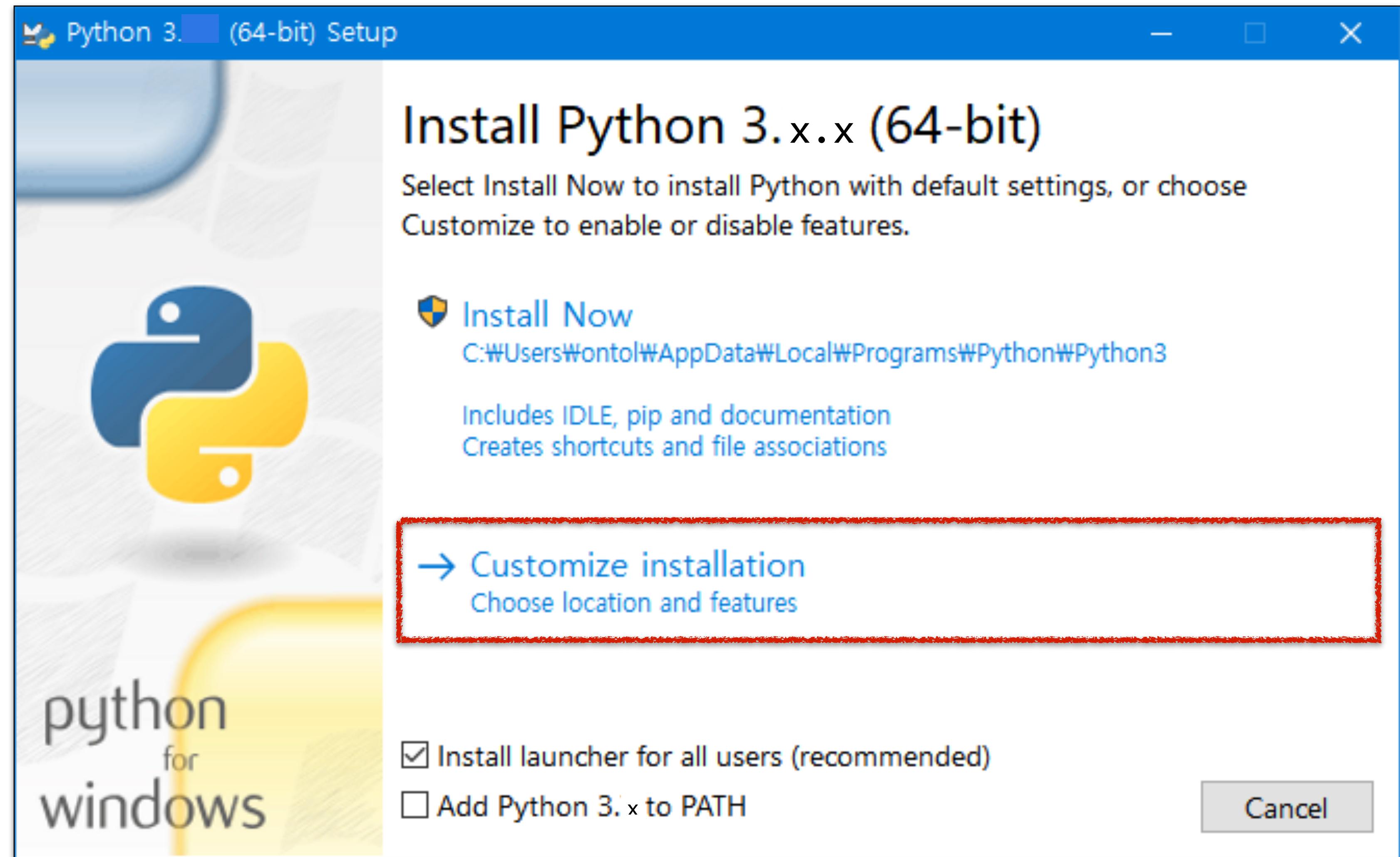
- (1) 옵션을 **모두 선택**하고
- (2) 하단 'Customize install location'에서 폴더 위치를 'C:\Python38'로 변경한 후
- (3) [Install] 버튼 클릭



파이썬 설치가 완료되면 [Close] 버튼을 클릭해서 설치 마법사 프로그램을 종료



- Step 1: [Customize installation]을 클릭



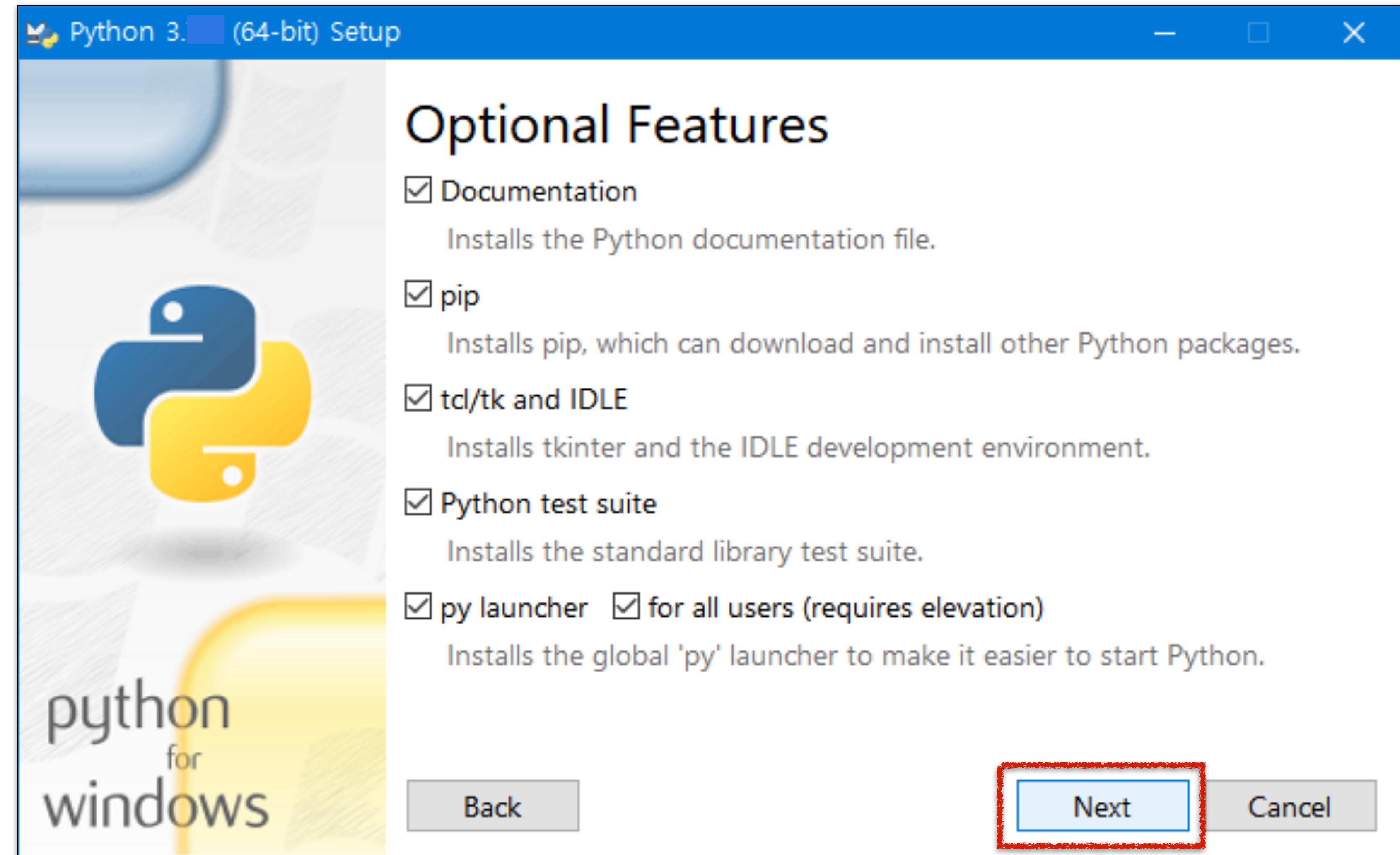


Windows

파이썬 설치하기

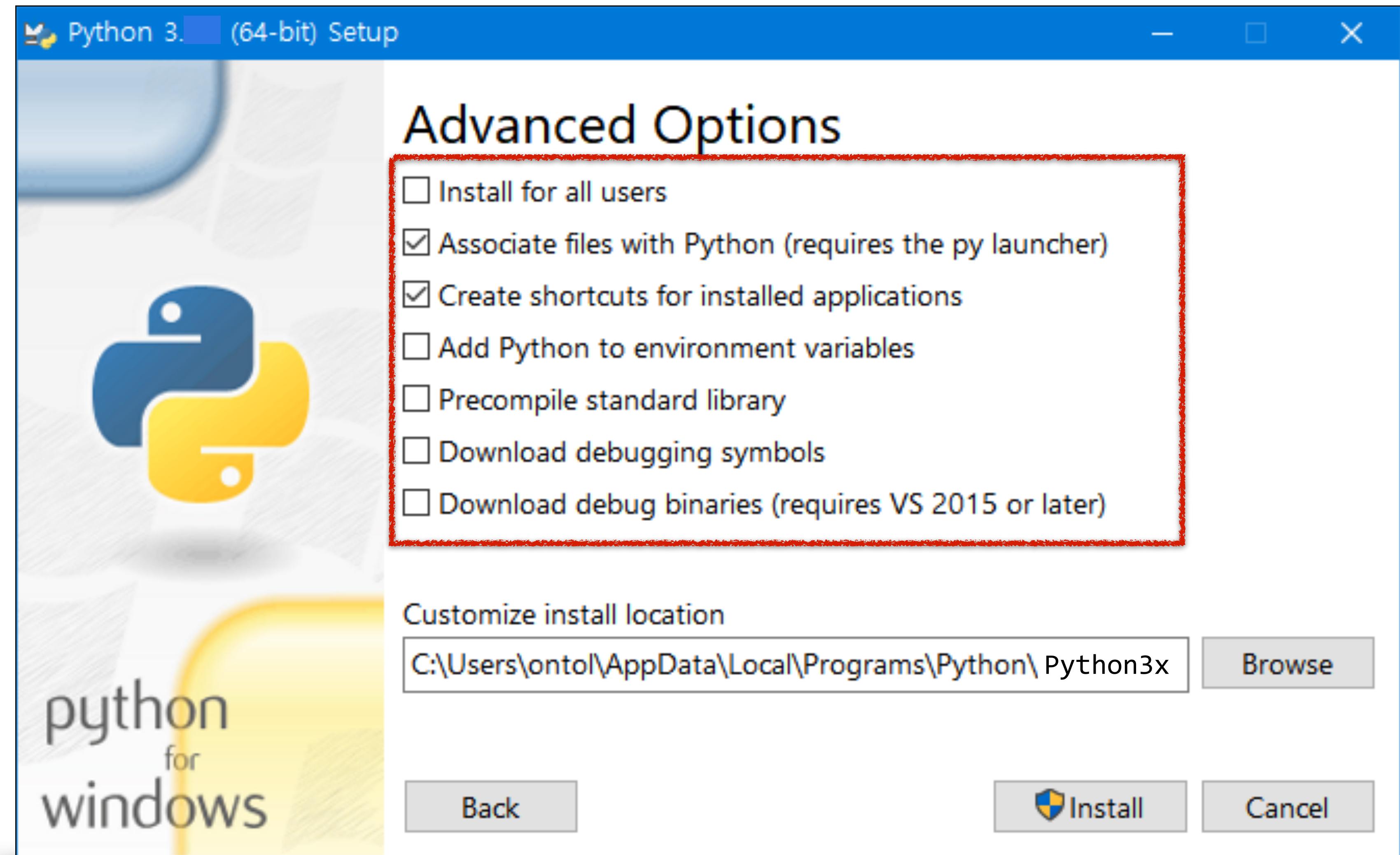


- Step 2: 옵션을 모두 선택되어져 있기 때문에 그냥 [Next] 버튼을 클릭



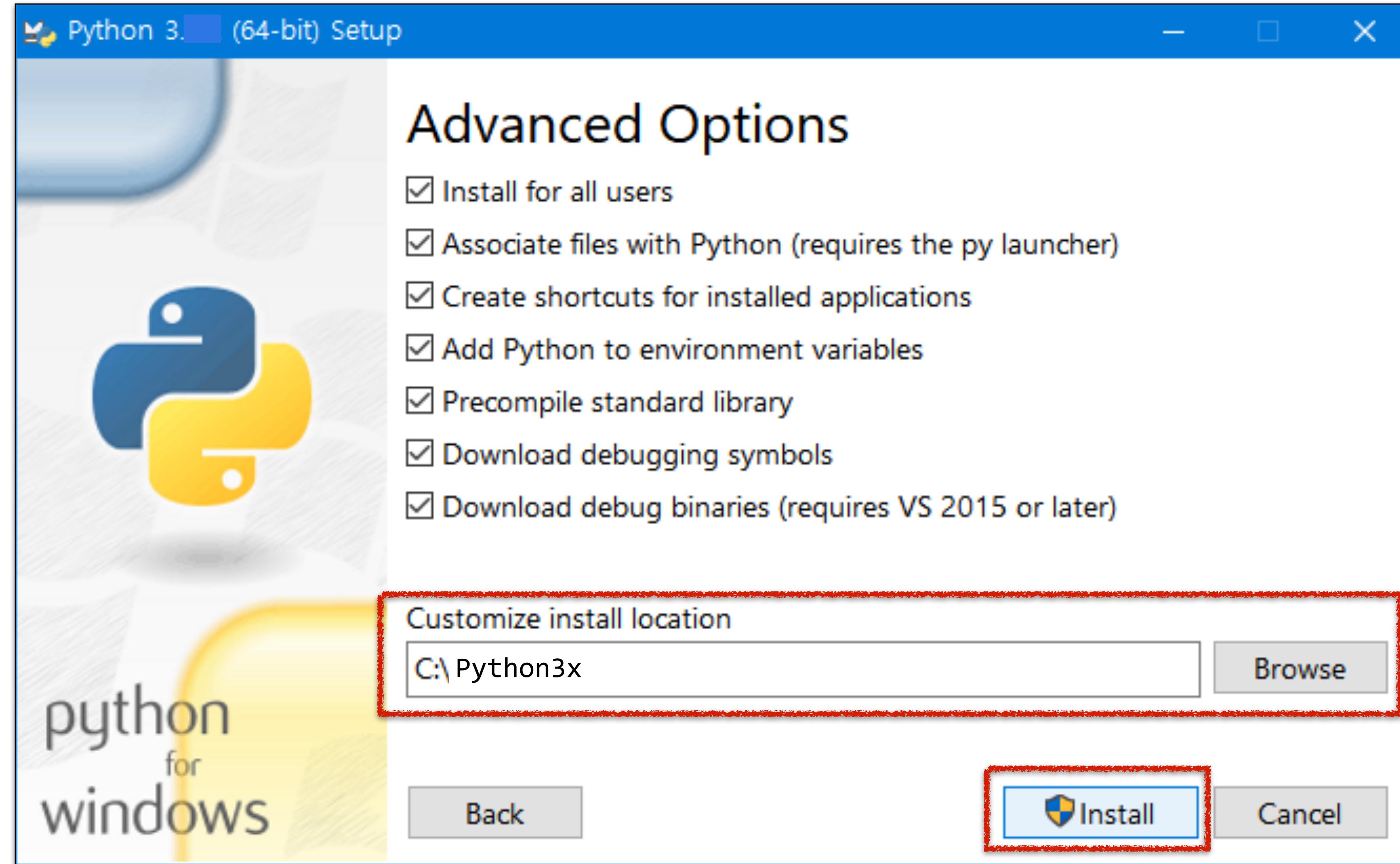


● Step 3: 옵션을 모두 선택





- Step 4: 옵션을 모두 선택하면 [Customize install location]의 설치 위치가 자동으로 'C:\Program Files\Python3x'로 변경됨
-> 저장한 폴더는 변경 가능함 (예를 들어, 'C:\Python3x')

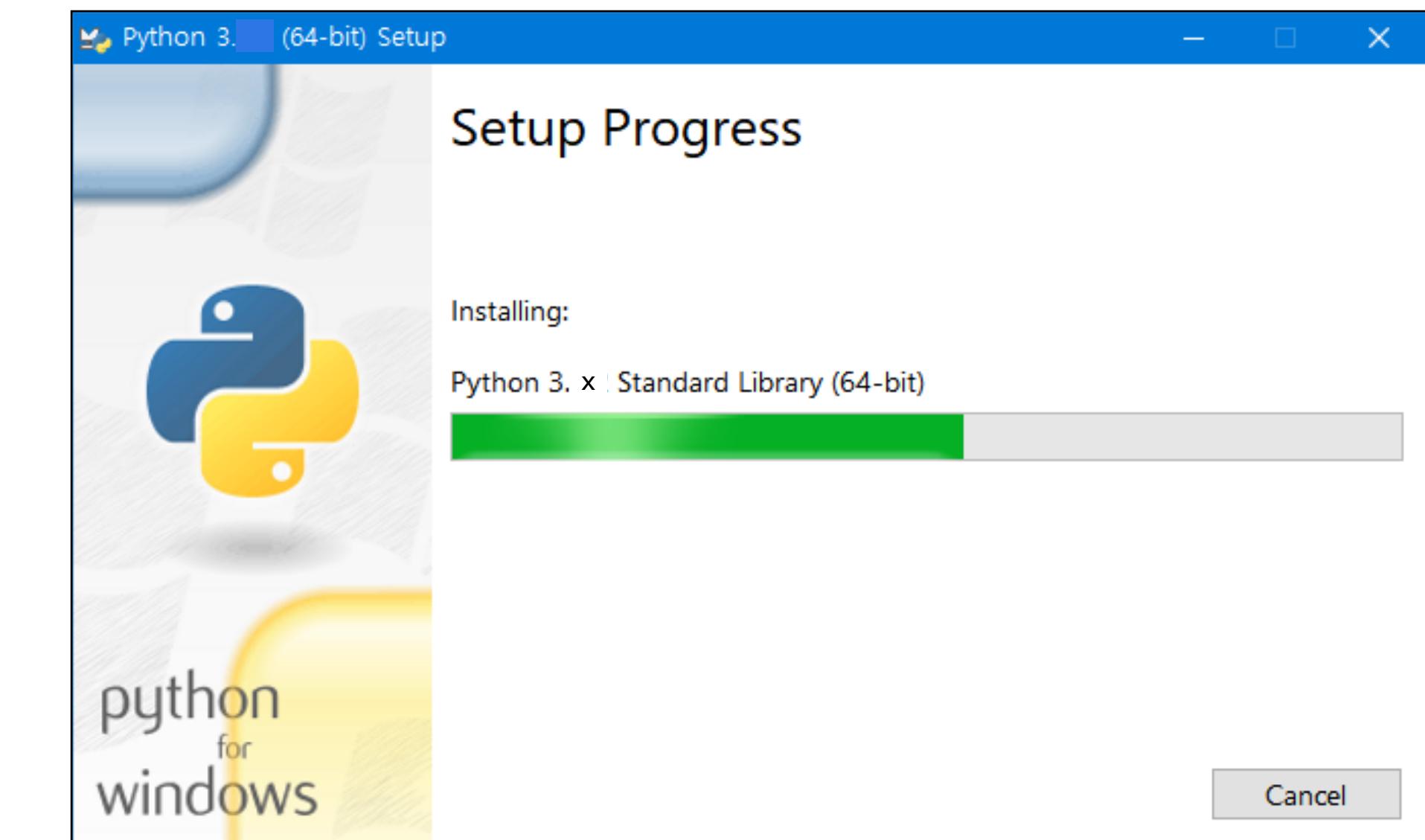




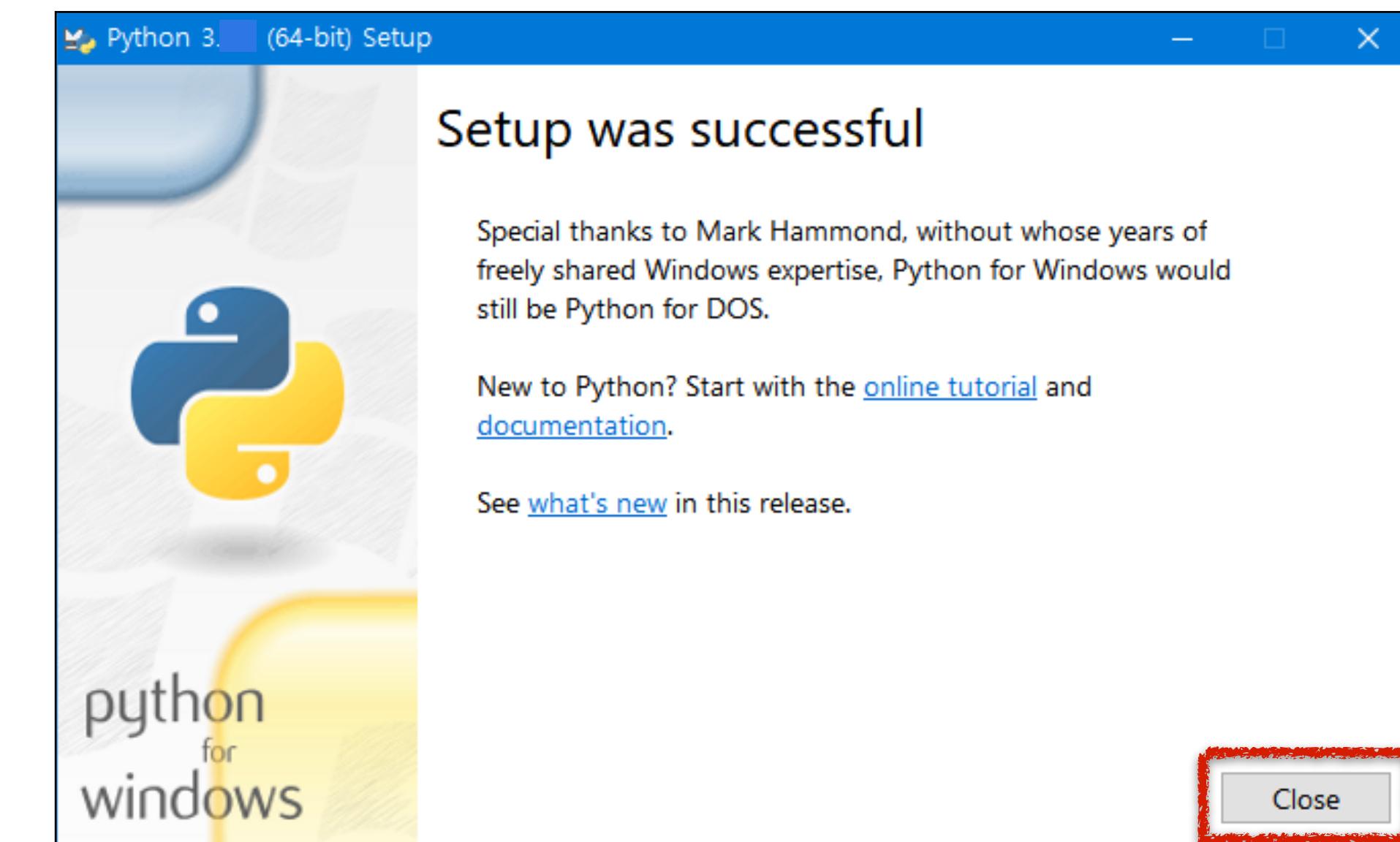
파이썬 설치하기



- Step 5: 우측 화면은 현재 파이썬을 설치하고 있는 중이라는 뜻



- Step 6: 파이썬 설치가 완료되면 [Close] 버튼을 클릭해서 설치 마법사 프로그램을 종료





파이썬 설치하기



- 본 강좌의 실습을 위해 파이썬 최신 버전을 설치
 - <https://www.python.org/downloads/>
- 설치 프로그램을 실행하면 파이썬 설치 마법사가 실행됨
 - [계속] 버튼을 클릭해서 파이썬 설치를 시작

The screenshot shows the Python.org homepage. At the top, there's a navigation bar with tabs for Python, PSF, Docs, PyPI, Jobs, and Community. Below the navigation bar is a large Python logo. To the right of the logo are buttons for Donate, Search, and GO, followed by Socialize. A secondary navigation bar below the main one has tabs for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a large yellow banner with the text "Download the latest version for Mac OS X". Inside this banner is a yellow button with the text "Download Python 3.9.6". Below the banner, there's text for users looking for other operating systems and links for pre-releases and specific Python 2.7 releases. To the right of the text is a cartoon illustration of two boxes descending from the sky on parachutes.



설치 확인

Checking Python Setup





● IDLE에서 확인하기

- 바탕화면 하단 [Windows 검색] -> 'IDLE' 입력 -> 검색 결과에서 [IDLE (Python 3.x 64-bit)] 항목을 클릭
- IDLE 창이 나타나면 파이썬 설치는 성공

● 명령 프롬프트 셸에서 확인하기

- 바탕화면 하단 [Windows 검색] -> 'cmd' 입력 -> 검색 결과에서 [명령 프롬프트] 항목을 클릭
- 명령 프롬프트 창에서 'python'을 입력하고 [Enter] 키를 눌러 파이썬이 실행되면 파이썬 설치는 성공



● IDLE에서 확인하기

- [command](또는 [control]) + [space bar] → [Spotlight 검색] 창에서 'IDLE' 입력 → [IDLE] 항목을 더블 클릭
- IDLE 창이 나타나면 파이썬 설치는 성공

● 터미널 셸에서 확인하기

- [command](또는 [control]) + [space bar] → [Spotlight 검색] 창에서 '터미널' 입력 → [터미널] 항목을 더블 클릭
- 터미널 창이 나타나면 'python3'을 입력하고 [return] 키를 눌러 파이썬이 실행되면 파이썬 설치는 성공

프로그래밍 툴

Programming Tools





웹 브라우저에서 파이썬 프로그래밍 : PythonAnywhere

- <https://www.pythonanywhere.com>

The screenshot shows a PythonAnywhere console interface. At the top, there are icons for file operations and a menu, followed by the text "Python3.7 console 7600219". On the right, there is a "Share with others" button. The main area is a terminal window with the following content:

```
Python 3.7 (default, [REDACTED], 00:00:00)
[GCC [REDACTED]] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> name = input('이름을 입력하세요...: ')
이름을 입력하세요...: 박진수
>>> print('반갑습니다.', name + '님!')
반갑습니다. 박진수님!
>>> 
```



웹 브라우저에서 파이썬 프로그래밍 : replit.com

- <https://replit.com/languages/python3>

The screenshot shows the Replit web-based Python development environment. On the left, the sidebar includes icons for Files, Share, Lock, Settings, Database, and Checkmark. The main interface shows a file named 'main.py' with the following code:

```
1 name = input('이름을 입력하세요...: ')
2 print('반갑습니다', name + '님!')
```

A large green 'Run ▶' button is positioned above the code editor. To the right, the 'Console' tab is active, displaying the output of the program's execution:

```
이름을 입력하세요...: Jinsoo Park
반갑습니다 Jinsoo Park님!
```

The 'Shell' tab is also visible in the console area.



웹 브라우저에서 파이썬 프로그래밍 : Google Colaboratory

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

The screenshot shows the Google Colaboratory interface. At the top, there's a navigation bar with a 'CO' logo, a 'python' icon, and a star icon. The menu bar includes '파일', '수정', '보기', '삽입', '런타임', '도구', '도움말', and '모든...'. On the right side of the menu are icons for '댓글', '댓글', '공유', and a gear for settings. A circular profile picture of a person is also present.

Below the menu, there are buttons for '+ 코드' and '+ 텍스트'. To the right, there are status indicators for 'RAM' (green checkmark) and '디스크' (grey bar). A pencil icon indicates '수정 가능'.

The main workspace displays a code cell with two lines of Python code:

```
[ 1 ] 1 name = input('이름을 입력하세요....: ')
2 print('반갑습니다', name + '님!')
```

Below the code cell, the output is shown:

```
이름을 입력하세요....: 박진수
반갑습니다 박진수님!
```



텍스트 편집기에서 파이썬 프로그래밍

● 프로그래밍을 위한 텍스트 편집기(text editors)

- ASCII 파일이나 유니코드 텍스트 파일을 편집할 수 있는 소프트웨어
- 윈도우의 ‘메모장’과 비슷한 기능을 하지만 프로그래밍이 용이하도록 다양한 기능을 제공하는 편집기도 많으니 장단점을 잘 비교해 자신에게 맞는 편집기를 사용

● Visual Studio Code

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



● Sublime Text

- <http://www.sublimetext.com>



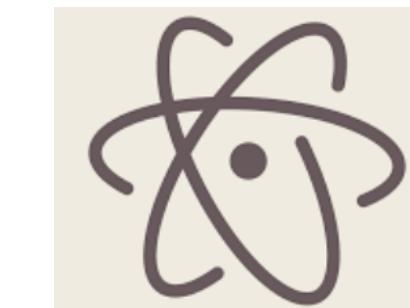
● Notepad++

- <https://notepad-plus-plus.org>



● Atom

- <https://atom.io/>





● 통합개발환경(IDE, integrated development environment)

- ⦿ 편집기, 컴파일러 등 프로그램 개발에 필요한 대부분의 도구들을 담고 있는 소프트웨어

● IDE의 장점

- ⦿ 함수명, 명령어, 매개변수 등 다양한 코드 구성 요소들을 색깔로 쉽게 구별하는 코드구문강조(syntax highlighting) 기능
- ⦿ 문법 오류가 있을 경우 실시간으로 확인 가능
- ⦿ 명령어 자동완성 기능 제공
- ⦿ 다양한 디버깅 툴을 제공하기 때문에 오류 발견이 용이

● IDE의 단점

- ⦿ Sublime Text 같은 단순한 문서편집기보다 용량을 훨씬 더 많이 차지함

● PyCharm

- ⦿ 가장 평가가 좋은 IDE로 Heavy Developer가 주로 사용
- ⦿ 무료 버전은 ‘Community Edition’
- ⦿ <https://www.jetbrains.com/pycharm/>

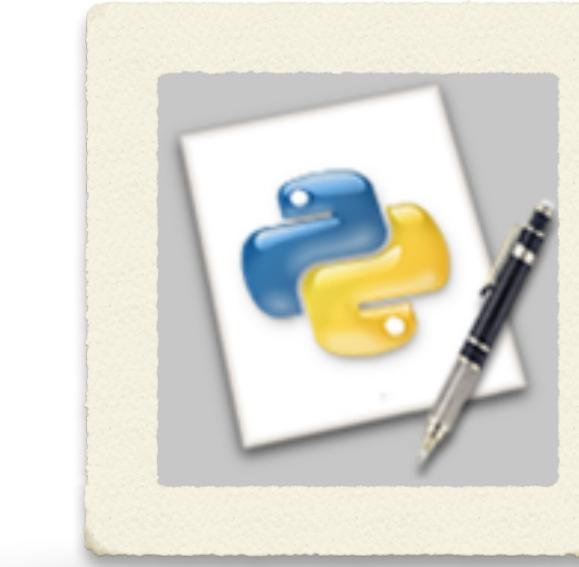




- **IDLE** : Integrated Development and Learning Environment

- 내장(built-in) 프로그램

- 파이썬 설치 시 기본적으로 함께 설치되는 프로그램
 - TkInter 그래픽 사용자 인터페이스(GUI) 라이브러리를 사용



- 3가지 핵심 기능

- 내장된 코드 편집기를 통해 자동 들여쓰기, 색깔을 통해 명령어를 구분하는 코드구문강조(syntax highlighting) 기능, 명령어 자동 완성 기능 등을 지원
 - 디버거가 장착되어 있어 각 코드 별로 한줄씩 한줄씩 오류를 파악하고 수정하는 것이 가능
 - 대화형 모드와 인터프리터 모드 둘 다 사용 가능
 - 대화형 모드(interactive mode)
 - 파이썬 셸(shell)에서 명령어(코드)를 입력받고 그 결과를 바로 출력
 - 인터프리터 모드(interpreter mode)
 - 파이썬 프로그램 편집기에서 프로그램을 작성한 후 파이썬 파일을 실행

IDLE 둘러보기

Lab Exercises

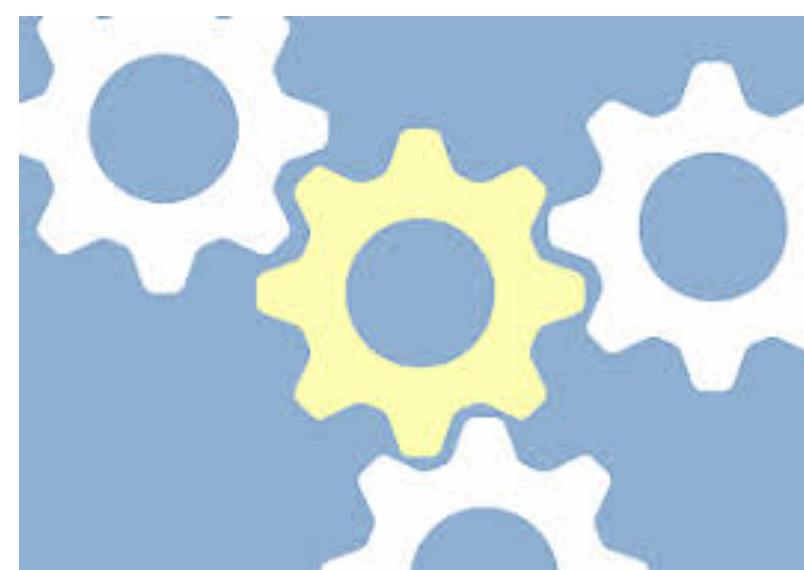




IDLE 기능 둘러보기



Configuring IDLE





IDLE 실행하기



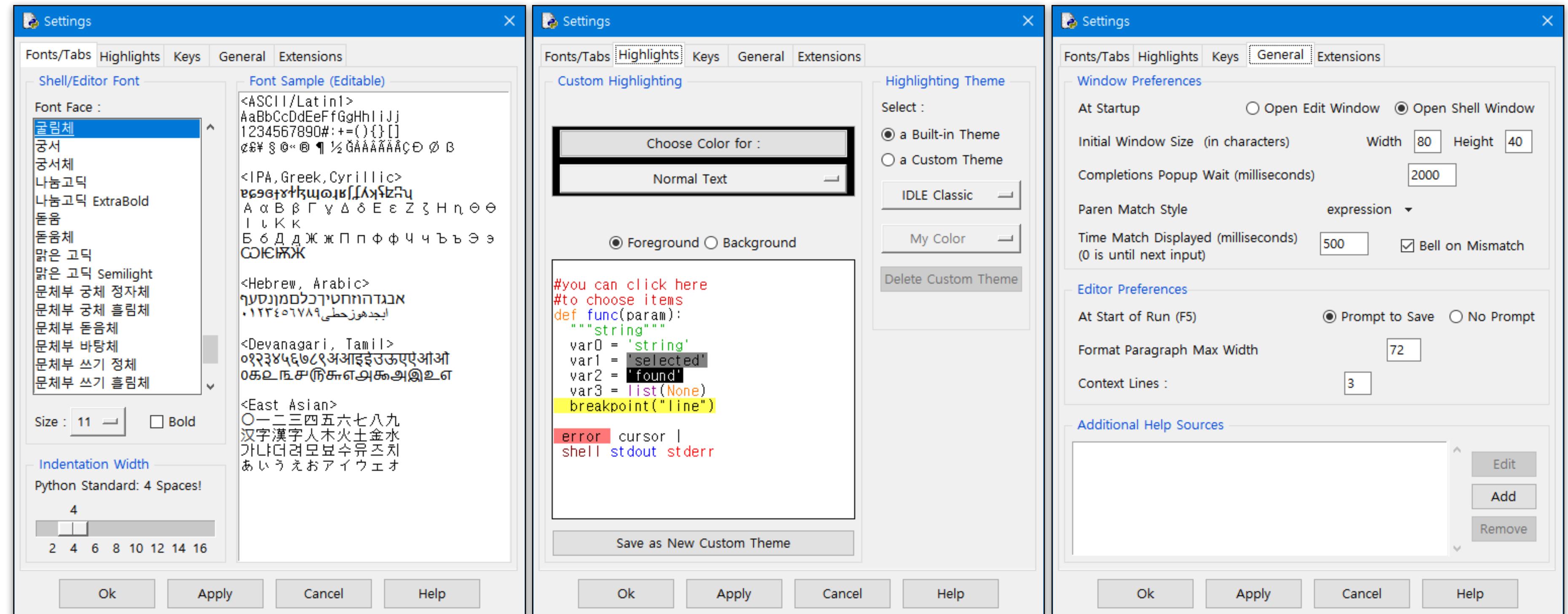
● IDLE 실행하기

- [Windows 검색]에서 ‘IDLE’ 입력 → [IDLE (Python 3.x 64-bit)] 항목 선택

● 옵션 메뉴에서 환경설정을 통한 IDLE 개인화

- [Options] → [Configure IDLE]

- [Settings] 창



● 도움말

- [Help] → [IDLE Help]



IDLE

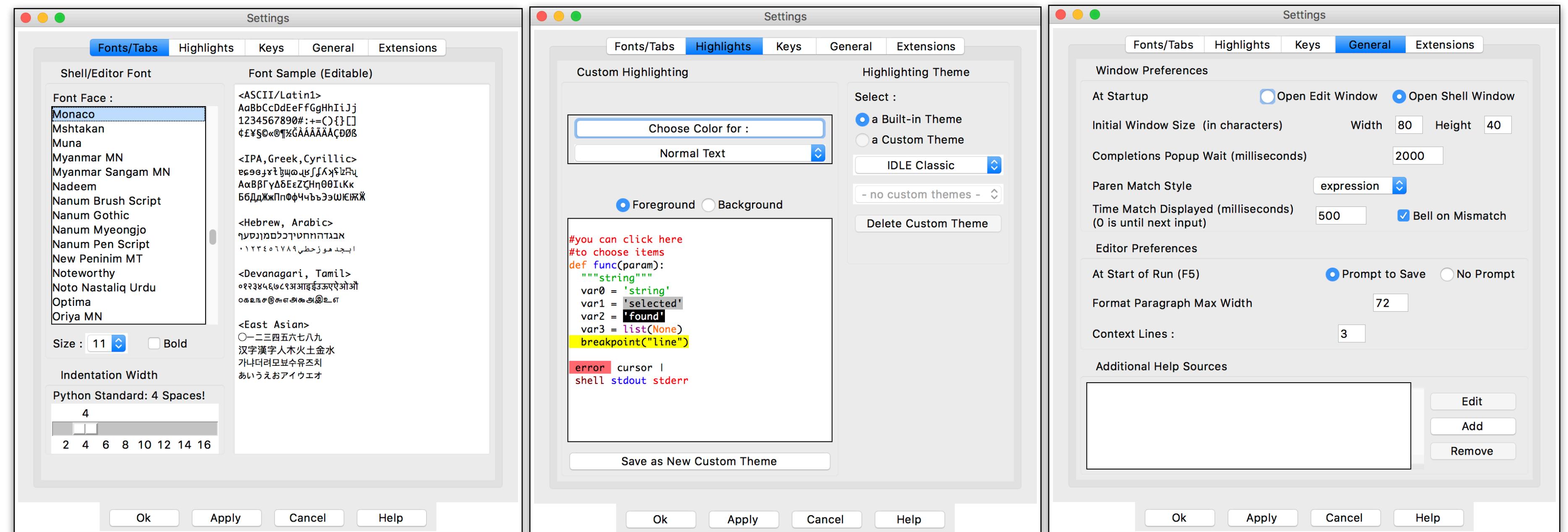


● IDLE 실행하기

- [command](또는 [control]) + [space bar]
- [Spotlight 검색] 창에서 ‘IDLE’ 입력 → [IDLE] 항목 더블 클릭

● 메뉴에서 환경설정을 통한 IDLE 개인화

- [IDLE] → [Preferences...]
- [Settings] 창



● 도움말

- [Help] → [IDLE Help]



IDLE 샘플 코드 맛보기



Tasting IDLE Turtle Demo



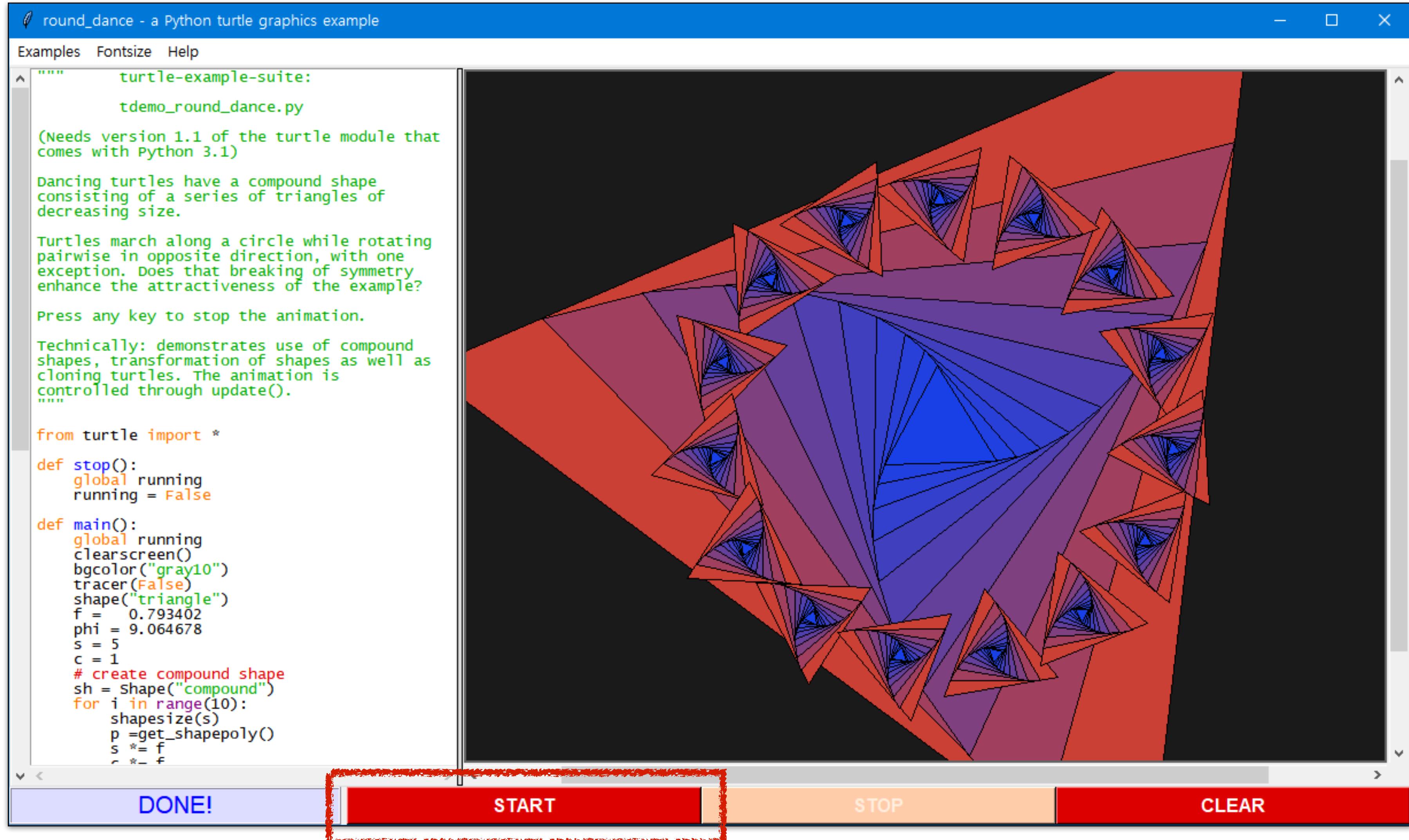


Windows

Turtle 데모 예시 파일 열기



- [Help] → [Turtle Demo] → [Examples] → [round_dance] → [START] 버튼 클릭 → 실행





- [Help] → [Turtle Demo] → [Examples] → [round_dance] → [START] 버튼 클릭 → 실행

round_dance - a Python turtle graphics example

""" turtle-example-suite:
tdemo_round_dance.py

(Needs version 1.1 of the turtle module that
comes with Python 3.1)

Dancing turtles have a compound shape
consisting of a series of triangles of
decreasing size.

Turtles march along a circle while rotating
pairwise in opposite direction, with one
exception. Does that breaking of symmetry
enhance the attractiveness of the example?

Press any key to stop the animation.

Technically: demonstrates use of compound
shapes, transformation of shapes as well as
cloning turtles. The animation is
controlled through update().
"""

```
from turtle import *  
  
def stop():  
    global running  
    running = False  
  
def main():  
    global running  
    clearscreen()  
    bgcolor("gray10")  
    tracer(False)  
    shape("triangle")  
    f = 0.793402  
    phi = 9.064678  
    s = 5  
    c = 1  
    # create compound shape  
    sh = Shape("compound")  
    for i in range(10):  
        shapesize(s)  
        p = get_shapepoly()  
        s *= f  
    update()  
    onkey(stop, "Escape")  
    done()
```

demo running... START CLEAR



IDLE로 파이썬 실행하기



Running Python with IDLE



IDLE 대화형 모드

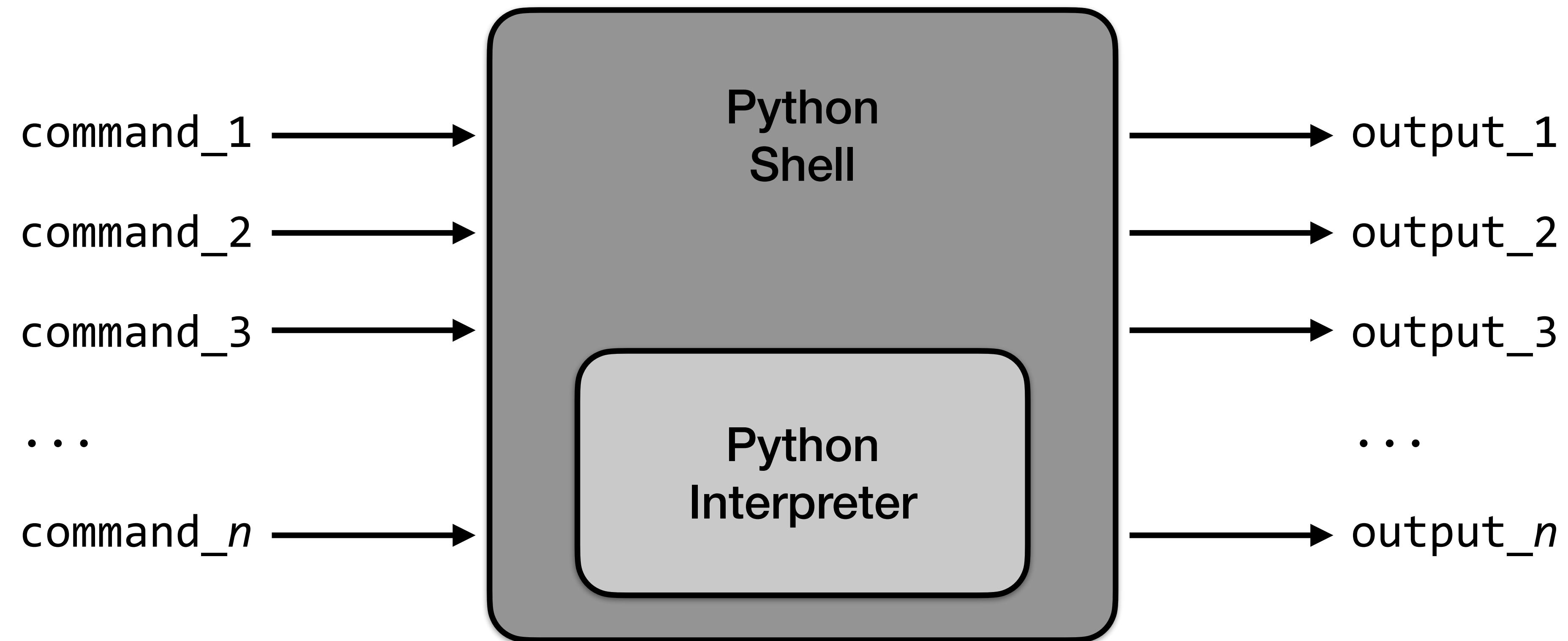
IDLE Interactive Mode





● Interactive Mode

- 파이썬 셸(shell) 실행 후, 명령어(코드)를 하나씩 실행
- 명령어 별 결과를 바로 확인 가능





● Auto-complete>Show completions

- [파이썬 셀] 창과 [파이썬 프로그램] 창 모두 적용됨

● 'pr'만 입력 → [tab] 키 → [print] 선택

- 만약 'pri'까지 입력하고 [tab] 키를 누르면 드롭다운 메뉴를 보여주지 않고 바로 'print'가 완성됨

```
>>> pr|  
print  
property  
quit  
range  
repr  
reversed  
round  
set  
setattr  
slice
```

● print 다음에 '('를 입력

```
>>> print(|  
print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)
```



- 아래 코드 3줄을 입력

```
>>> for i in range(1, 10):  
    for j in range(1, 10):  
        print(f'{i} x {j} = {i * j}')
```

```
1 x 1 = 1  
1 x 2 = 2  
1 x 3 = 3  
1 x 4 = 4  
1 x 5 = 5
```

- IDLE의 [파이썬 셀] 대화형 모드에서 첫 줄 작성 후 [Enter] 또는 [return] 키를 누르면 자동으로 다음 줄로 넘어간 후 4칸을 들여쓰기를 해줌
- 커서가 이동한 위치에서 둘째 줄의 코드를 입력하고 [Enter] 또는 [return] 키를 누르면 자동으로 다음 줄로 넘어간 후 4칸을 들여쓰기를 해줌
- 커서가 이동한 위치에서 둘째 줄의 코드를 입력하고 [Enter] 또는 [return] 키를 연속 두 번 누르면 코드가 실행됨

IDLE 인터프리터 모드

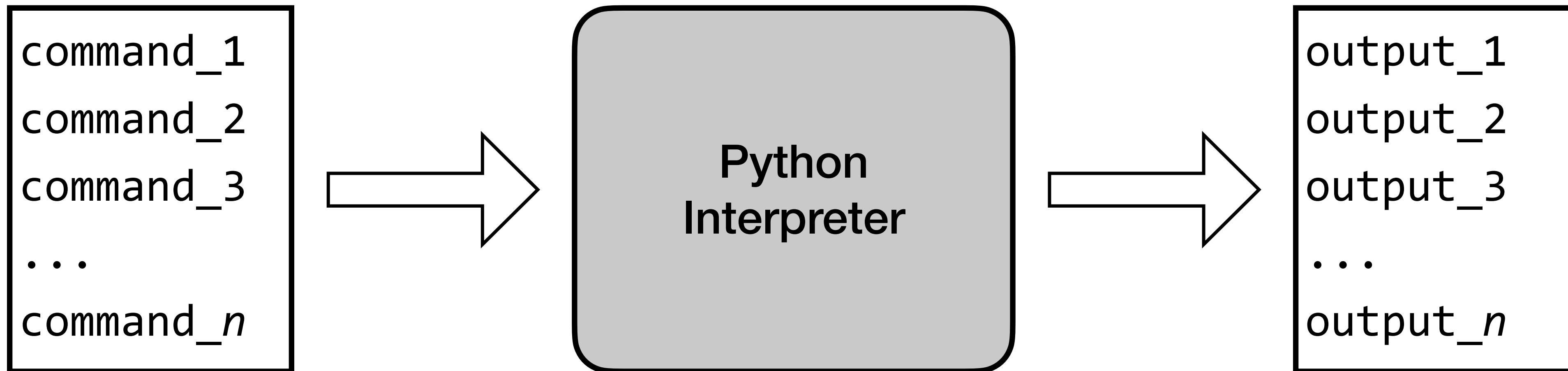
IDLE Interpreter Mode





● Interpreter Mode

- 텍스트 편집기를 사용하여 프로그램을 작성한 후 한번에 실행
- 인터프리터가 프로그램 전체의 문법 오류 확인 후, 명령어 하나 씩 실행



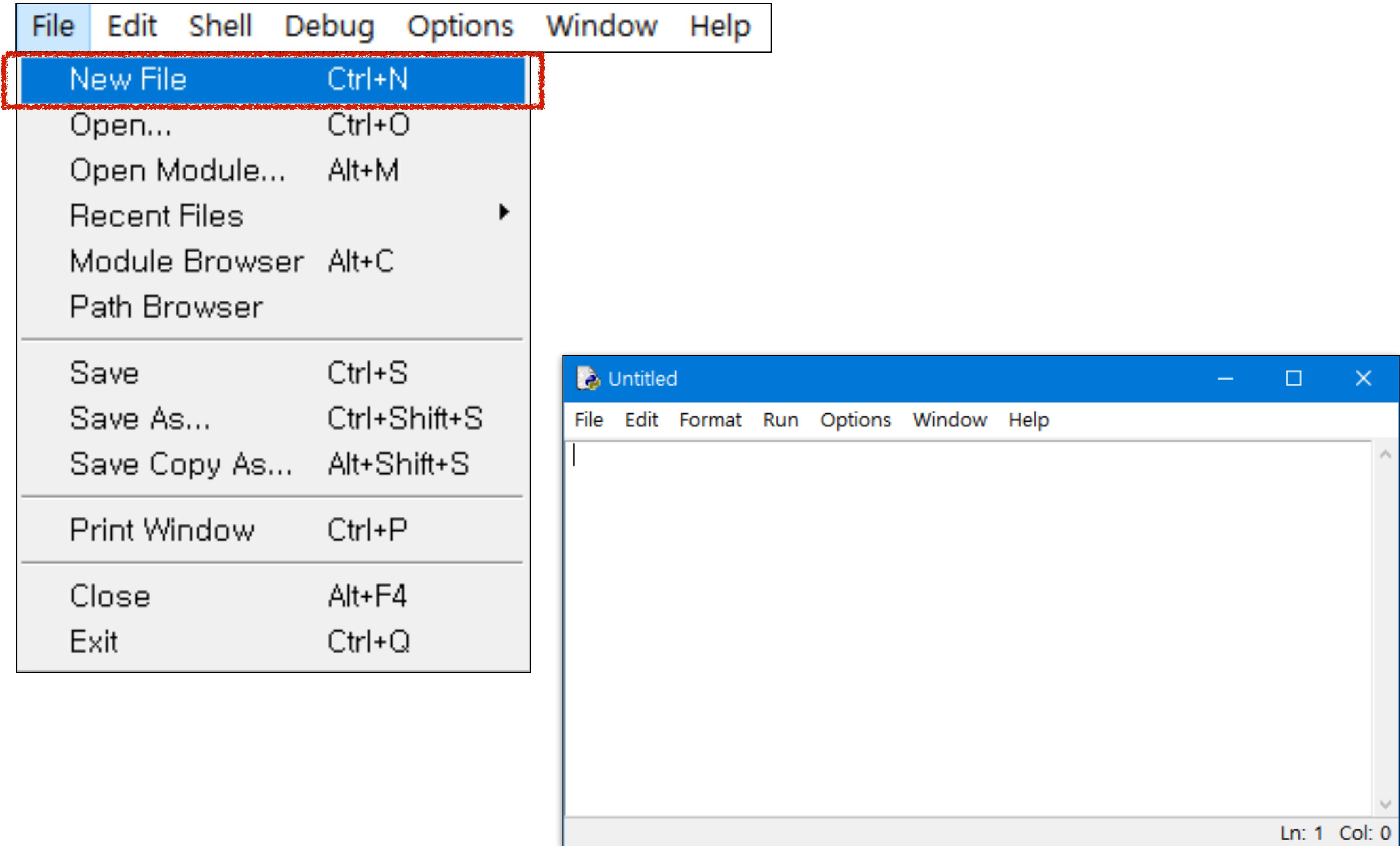


Windows

프로그램 작성을 위해 새 파일 창 열기



- [File] → [New File]



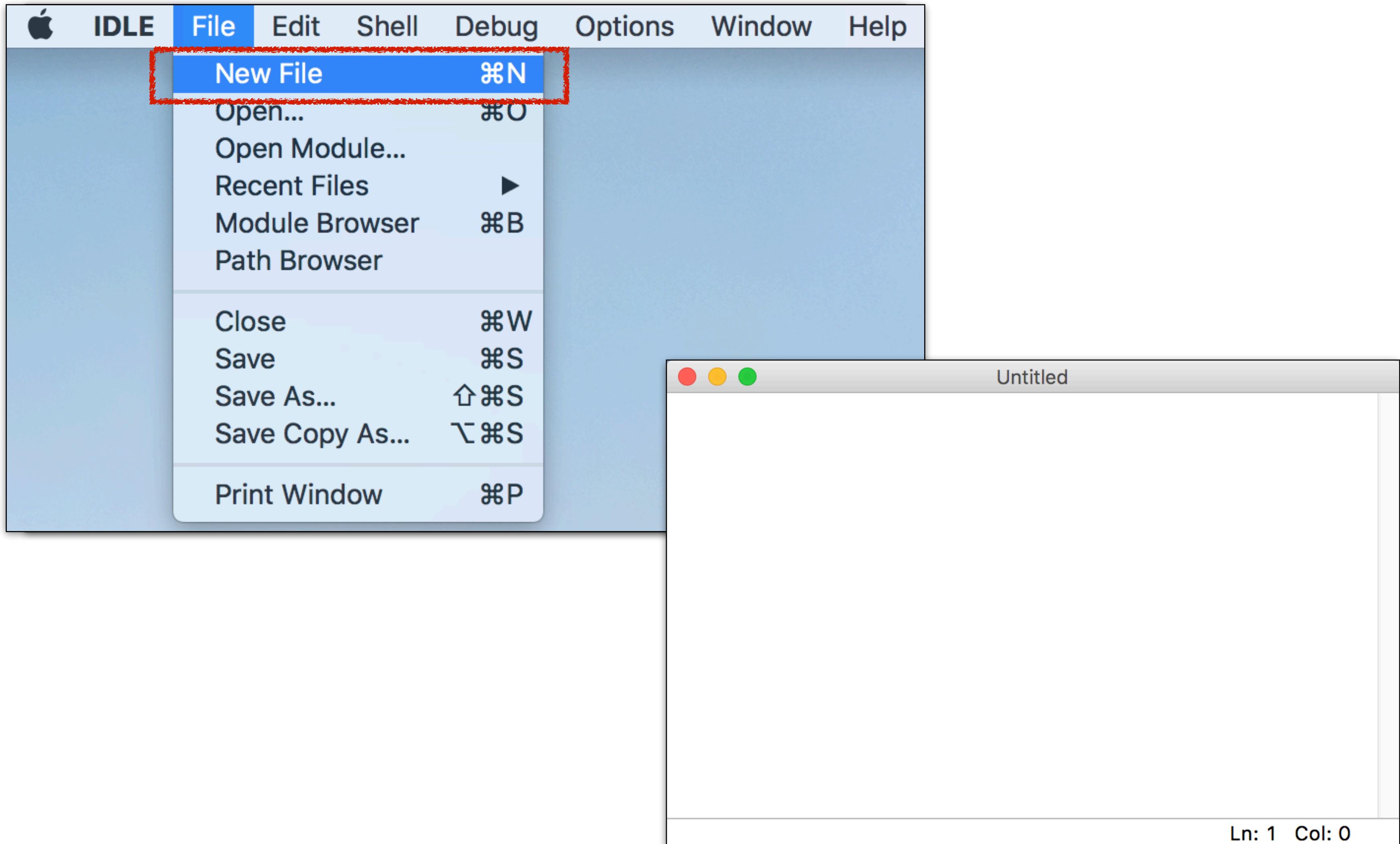


macOS®

프로그램 작성을 위해 새 파일 창 열기



- [File] → [New File]





- Copy & Paste 또는 입력
 - 명령문만 복사

The screenshot shows a code editor window with the title bar "*untitled*". The code in the editor is:

```
for i in range(1, 10):
    for j in range(1, 10):
        print(f'{i} x {j} = {i * j}')
```

In the bottom right corner of the editor window, the text "Ln: 3 Col: 8" is displayed.

- [File] → [Save]
- [Run] → [Run Module] 또는 단축 키 [F5]
- 실행 결과 확인

디버깅

Debugging





- 디버깅(debugging)

- 컴퓨터 프로그램의 문법적 오류(오타 등)나 논리적 오류인 버그(bug)를 찾아내기 위해 테스트하고 수정하는 과정

- [파이썬 셸] 창 선택 → 아래 내용 입력 → [Enter] 또는 [return] 키

```
>>> print('안녕 파이썬')
```

- 오류 발생!!!

```
>>> print('안녕 파이썬')
```

```
SyntaxError: EOL while scanning string literal
```

```
>>>
```

- 오류 수정

```
>>> print('안녕 파이썬')
```

- 실행

```
>>> print('안녕 파이썬')
```

```
안녕 파이썬
```

```
>>>
```



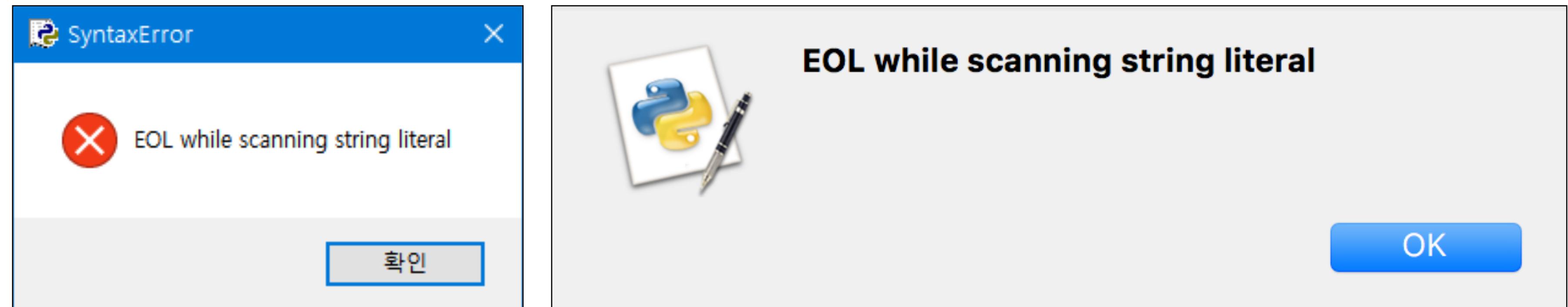
IDLE 인터프리터 모드



- [File] → [New File] → 아래 내용 입력

```
print('안녕 파이썬')
```

- [File] → [Save] → [F5] → 오류 발생!!!



- 디버깅 → 저장 → 실행



● 네이버나 구글 검색창에 오류 메시지 Copy & Paste

NAVER SyntaxError: EOL while scanning string literal

통합검색 블로그 카페 지식iN 이미지 동영상 어학사전 뉴스 더보기 ▾

정렬 ▾ 기간 ▾ 영역 ▾ 옵션유지 깨짐 켜짐 | 상세검색 ▾

블로그

Error: SyntaxError: EOL while scanning string literal 2015.05.21.
Return to main page Error: SyntaxError: EOL while scanning string literal <error_note>"EOL" stands for "end of line". An EOL error means that Python...
엠지님의 블로그~! blog.naver.com/swwwwaa?Redirect=Log&logNo... | 블로그 내 검색

SyntaxError 2017.11.26.
SyntaxError SyntaxError: EOL while scanning string literal 문자열의 끝에서 줄마음표 찍지 않았을 경우에 발생
시간을 들여서 노... blog.naver.com/h2odra?Redirect=Log&lo...

파이썬 문자열 선언 - Python String 2010.01.21.
hello = "test ^ SyntaxError: EOL while scanning string literal """(쌍따옴표 3개)를 사용한 녀석은 아래와 같이 """ 가 나올 때 까지 계속 입력을 받고...
구차니의 잡동사니 ... minimonk.net/1250 | 블로그 내 검색

[블로그 더보기 >](#)

카페

파이썬 기초 코딩 2017.09.04.
에러>>> "반갑습니다" "반갑습니다">>> '안녕' SyntaxError: EOL while scanning string literal>>> print("문자를 입력할때는 ", "로 짹을 맞춰야 함")문자를 입력할때는...
양주종의 코딩스쿨 ► C언어 · C++· 파... cafe.naver.com/funcc/4374... | 카페 내 검색

파이썬(Python) 기초 – 파이썬입출력,파이썬자료형 2017.06.14.
함>>> 'korea"SyntaxError: EOL while scanning string literal>>> "korea'SyntaxError: EOL while scanning string literal>>> "korea" "korea'>>> 'korea"korea'>>> print(30,50,60,70)30 50 60 70>>> a=100...
양주종의 코딩스쿨 ► C언어 · C++· 파... cafe.naver.com/funcc/4238... | 카페 내 검색

Google SyntaxError: EOL while scanning string literal

All Images Videos News More Settings Tools

About 12,900 results (0.38 seconds)

[python: SyntaxError: EOL while scanning string literal - Stack ...](#)
<https://stackoverflow.com/questions/3561691/python-syntaxerror-eol-while-scanning-string-literal>
I have the above mentioned error in s1="some very long string....." Anyone know what i am doing wrong?

8 answers

Best Answer 102 votes
You are not putting a " before the end of the line. Use "" if you want to do this: "" a very long stringthat can span multiple lines ""

Answer 2 of 8 49 votes
I had this problem - I eventually worked out that the reason was that I'd included \ characters in the string. If you have any of these, "escape" them with \ a...

Answer 3 of 8 12 votes
(Assuming you don't h... line breaks in your str... long is this string really suspect there is a limit long a line read from a

[What's an EOL error and how do I fix it? | Codecademy](#)
https://www.codecademy.com/en/forum_questions/52c0b0977c82ca009d006037 ▾
Jan 23, 2014 - 2 posts - 1 author
Hello. I'm trying to finish Multiline Comments (lesson 9/16) and have run into a problem. When I try to submit my code I get an error pointing to the end of line one "SyntaxError: EOL while scanning string literal". What does that mean? What should be at the end of line one?

5/9 SyntaxError: EOL while scanning string literal 2 posts 13 Jan 2016
SyntaxError: EOF while scanning triple-quoted string literal ... 6 posts 19 Jan 2014
Error when writing sentence over 2 lines 3 posts 14 Jan 2013
More results from www.codecademy.com

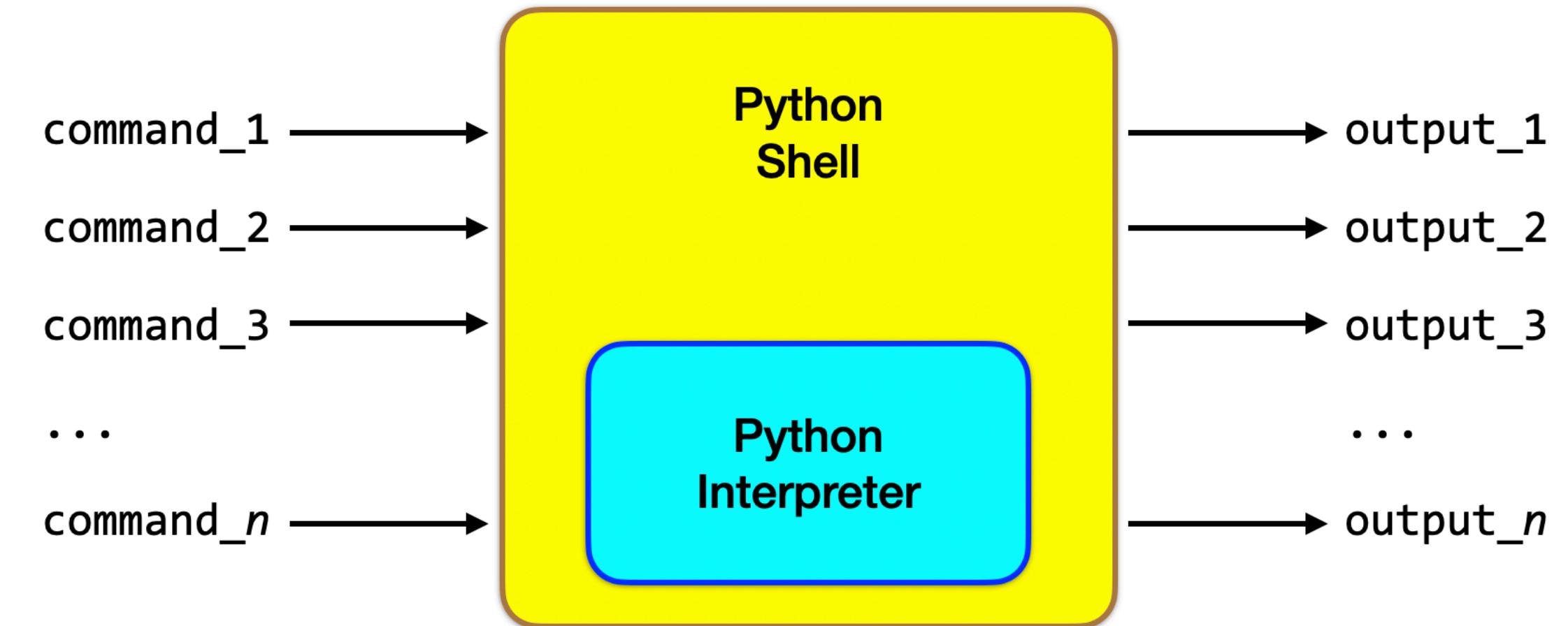
[Getting "EOL while scanning string literal" error, and what happened ...](#)
<https://teamtreehouse.com/.../getting-eol-while-scanning-string-literal-error-and-what-...> ▾
May 21, 2015 - Hi: I am going over projects and code challenges and adding them to my portfolio of work. When I did Shopping List Take Three, I noticed that I got a "EOL while scanning string literal" error. I decided to try and make the input statement all one line. One problem: when I did, the other functions acted odd.



정리 : 두 가지 모드에서 파이썬 프로그램 실행

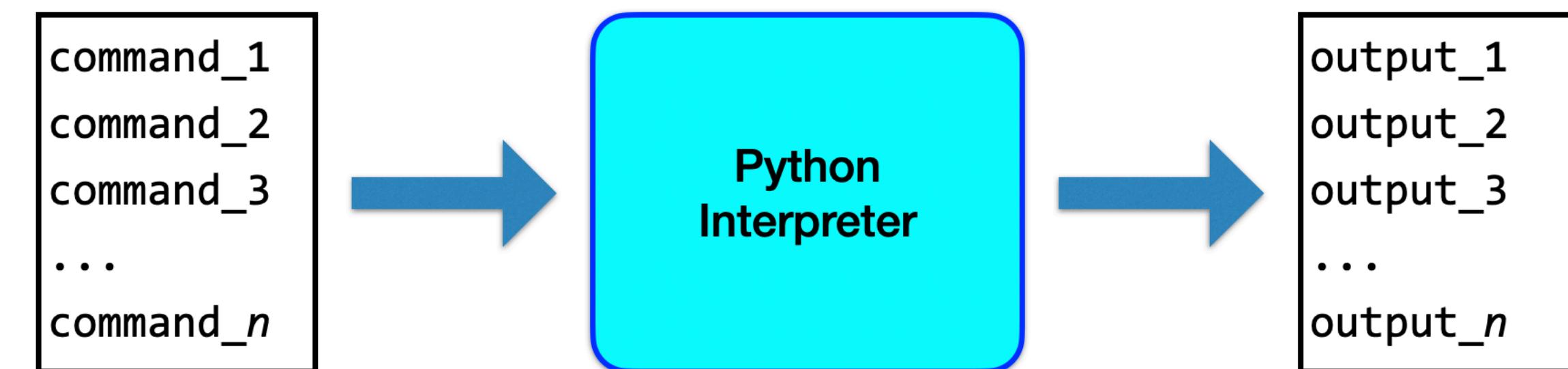
● 대화형 모드(interactive mode)

- 대화 창(파이썬 셀)에서 실시간으로 명령어를 입력
- 대화형 모드를 실행하면 프롬프트('>>>' 표시)가 나타남
 - 프롬프트는 사용자로부터 명령어를 입력받을 준비가 되었다는 뜻
 - 명령어가 처리되면 다음 명령어를 기다리는 프롬프트가 다시 나타남
 - 단, 잘못된 명령어를 입력해서 실행할 경우 오류 메시지가 출력됨
- 파이썬의 다양한 기능을 테스트해 볼 때 편리함



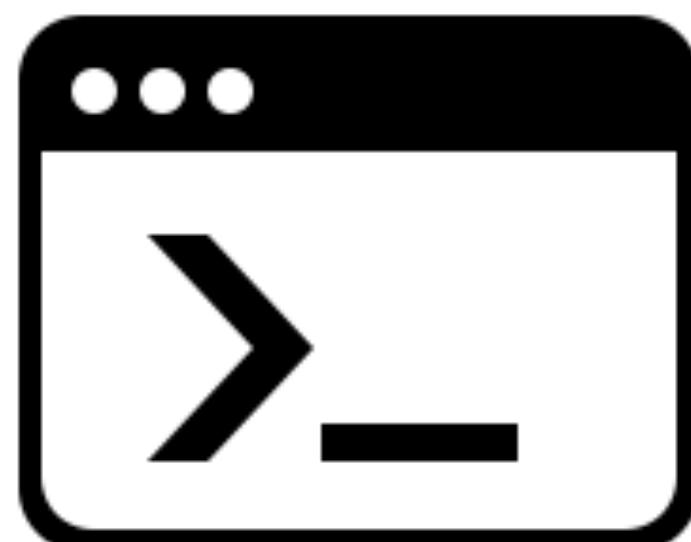
● 인터프리터 모드(interpreter mode)

- 파이썬 명령어들을 문서 형태로 저장(즉, 코드 파일을 생성한다는 뜻)
 - 프로그램 창에서 작성한 코드를 파일 형태로 저장
 - 참고 : 대화형 모드에서는 입력한 명령어들이 저장되지 않음
 - 반드시 파일 확장자를 .py로 설정
- 프로그램 파일을 실행하기 위해 메뉴에서
 - [Run] → [Run Module]
 - 또는
 - [F5]



명령어 셸 실행 환경

Command Shell Environment





● 셸(shell)이란?

- ⦿ 커널(kernel)이라는 운영체제의 내부 핵심과 사용자 사이의 인터페이스
- ⦿ 사용자의 명령을 해석해서 운영체제에 전달하고 그 처리 결과를 사용자에게 보여주는 시스템 프로그램
- ⦿ 사용자는 다양한 셸(shell) 환경에서 프로그램을 실행할 수 있음

● 셸의 종류

⦿ 그래픽 셸(graphic shell)

- ⦿ 그래픽 사용자 인터페이스(GUI, graphic user interface)
- ⦿ 윈도우(Windows)
- ⦿ 맥 OS(macOS)
- ⦿ 유닉스(UINX)와 리눅스(LINUX)의 X 윈도우 시스템(X Window System, X11)

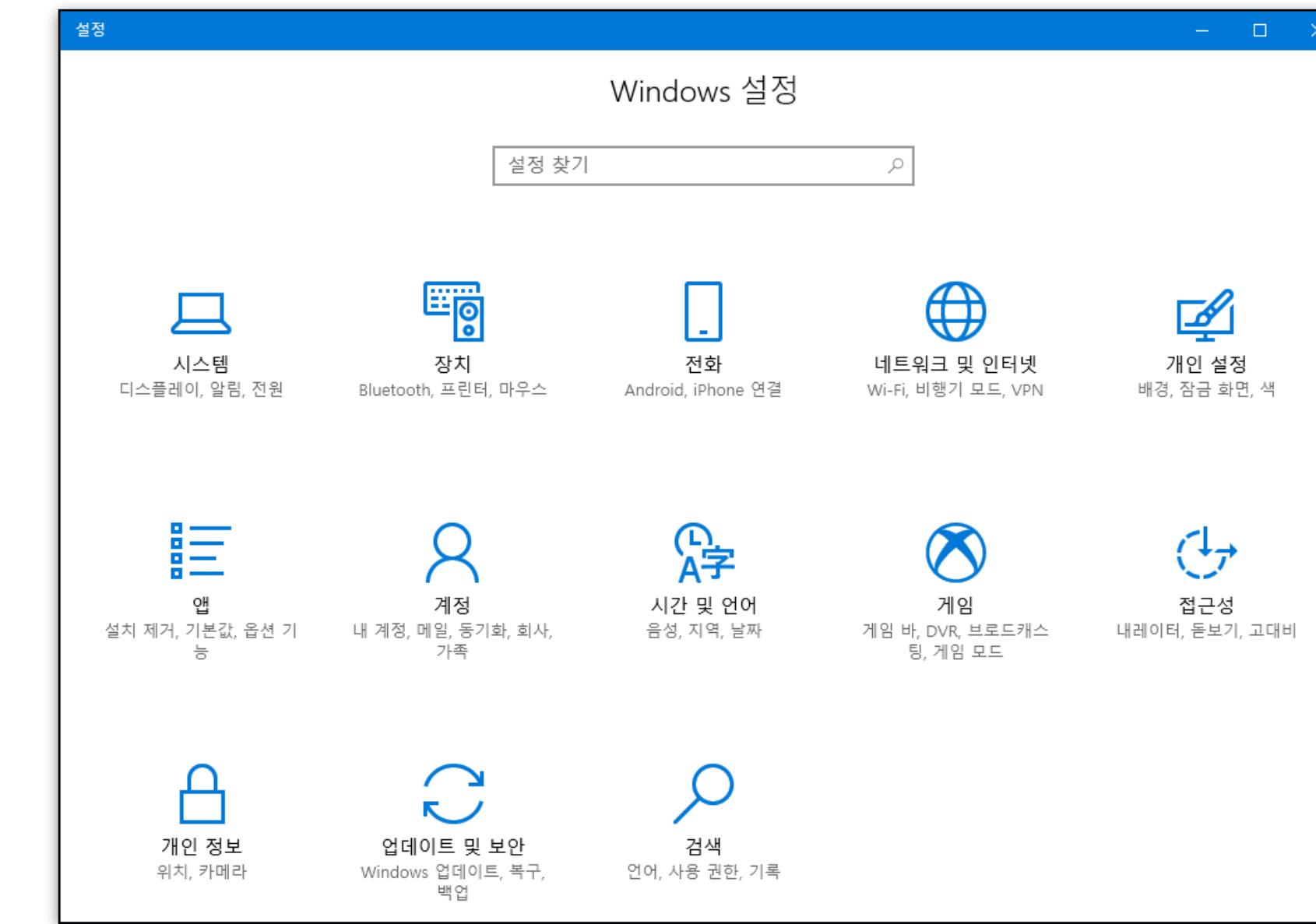
⦿ 명령어 셸(command/character shell)

- ⦿ 명령어 인터페이스 또는 명령 줄 인터페이스(CLI, command line interface)
 - 사용자가 컴퓨터 키보드 등을 통해 문자열 형태로 입력을 하며, 컴퓨터로부터의 출력 또한 문자열로 이루어짐
- ⦿ 윈도우(Windows) 운영체제 : 도스(DOS) 셸 환경의 명령 프롬프트
- ⦿ 맥 OS(macOS) 운영체제 : bash 셸 환경의 터미널



예시 : 그래픽 셀

● 윈도우(Windows)



● 맥 OS(macOS)



- 원도우(Windows) : 도스(DOS) 셸 환경의 명령 프롬프트

```
C:\Users\montol>dir
C 드라이브의 볼륨: ontology
볼륨 일련 번호: 3EOA-201C

C:\Users\montol 디렉터리

2017-12-28 오전 11:53 <DIR>
2017-12-28 오전 11:53 <DIR>
2017-12-15 오후 08:36 <DIR>
2017-12-18 오후 08:44 <DIR>
2017-12-18 오후 08:44 <DIR>
2017-12-18 오후 08:44 <DIR>
2017-12-18 오후 08:44 <DIR>
2017-12-11 오후 06:28 <DIR>
2018-01-04 오후 07:00 <DIR>
2017-12-18 오후 08:48 <DIR>
2017-12-28 오전 11:53 <DIR>
2017-12-18 오후 08:44 <DIR>
2017-12-18 오후 08:44 <DIR>
2018-01-02 오후 12:04 <DIR>
2017-12-11 오후 06:28 <DIR>
2017-05-11 오전 11:27 <DIR>
2017-12-18 오후 08:44 <DIR>
2017-12-18 오후 08:44 <DIR>
2017-12-18 오후 08:44 <DIR>

.                                .
..                               .idlerc
3D Objects
Contacts
Desktop
Documents
Downloads
Favorites
Intel
Links
Music
OneDrive
Pictures
Roaming
Saved Games
Searches
Videos

0개 파일          0 바이트
18개 디렉터리   156,936,695,808 바이트 남음

C:\Users\montol>
```

- 맥 OS(macOS) : bash 셸 환경의 터미널

```
[> ls -l
total 8
-rw-r--r--  1 root  staff  558 12  3 22:36 AT.postflight.1381
drwx-----@ 3 jsp   staff   96 12  1 22:44 Applications
drwx-----+ 4 jsp   staff  128 12  3 21:48 Desktop
drwx-----+ 3 jsp   staff   96 12  1 22:27 Documents
drwx-----+ 4 jsp   staff  128  1  5 00:24 Downloads
drwx-----@ 12 jsp  staff  384 12  1 23:06 Dropbox
drwx-----@ 14 jsp  staff  448  1  6 22:10 Google 드 라 이 브
drwx-----@ 66 jsp  staff 2112  1  2 23:07 Library
drwx-----+ 3 jsp   staff   96 12  1 22:27 Movies
drwx-----+ 4 jsp   staff  128  1  2 22:40 Music
drwx-----+ 3 jsp   staff   96  1  4 23:54 Pictures
drwxr-xr-x+ 5 jsp   staff  160 12  1 22:27 Public
drwxr-xr-x  10 jsp  staff  320  1  4 23:47 software-downloads
> ]
```



● 명령어 셸에서 파이썬 셸 실행

- 실행 파일 이름 : **python** 또는 **python3**
- 파이썬 셸이 시작 메시지인 파이썬 버전 및 저작권 안내를 출력한 후 프롬프트가 나타남
 - 대화형 모드 기본 프롬프트 : '**>>>**'
 - 한 명령문 내에서 줄 바꿈이 있을 경우 '**...**' 프롬프트가 표시됨

```
> python
Python 3.x.x ...
Type "help", "copyright", "credits" or "license" for more information.
>>> 1 + 1
2
>>> for i in range(5):
...     print(i)
...
0
1
2
3
4
>>>
```



- 방법 1

```
>>> quit()
```

- 방법 2

```
>>> import sys  
>>> sys.exit()
```

- 방법 3

- 창 닫기 아이콘 클릭

- 방법 4 : macOS 경우 단축 키 사용

- [control] + Z
 - [control] + D



만약 같은 파이썬 코드를 여러 번 반복해서 실행해야 한다면?



인터프리터 모드



부록 1 : 셸 명령어 기초

- 프로그램 파일은 파이썬 인터프리터를 통해 실행됨
- 주로 명령어 셸(CLI)에서 이루어짐
- 윈도우(Windows) 명령 프롬프트에서 실행

파이썬 환경설정에서 'PATH'에 파이썬 설치경로가
지정되지 않았을 경우 여기서 에러가 발생

C:\Users\montol>python hello.py
안녕 파이썬
C:\Users\montol>

A screenshot of a Windows Command Prompt window titled "명령 프롬프트". The command "python hello.py" is entered, followed by the Korean text "안녕 파이썬". A red arrow points from the text "파이썬" to the error message in the callout box.

- 유닉스 기반 시스템(e.g., Unix, Linux, 또는 macOS)

course-examples — -bash — 49x5
\$ python hello.py
안녕 파이썬
\$

A screenshot of a Mac OS X terminal window titled "course-examples — -bash — 49x5". The command "python hello.py" is entered, followed by the Korean text "안녕 파이썬". A red arrow points from the text "파이썬" to the error message in the callout box.



● <인터프리터 모드>로 프로그램을 <실행>하고자 할 때도 IDLE을 사용할 수 있나요?

- 가능은 합니다만 본 수업에서는 권장하지 않습니다. 명령어 셸(CLI), 즉 ‘명령 프롬프트’(윈도우) 또는 ‘터미널’(맥)에서 실행하길 권장합니다.
- 만약 ‘명령 프롬프트’(윈도우) 또는 ‘터미널’(맥) 사용이 익숙하지 않을 경우 본 수업의 실습 <셸 명령어 기초>에서 기본 내용을 다루니 학습하면 됩니다.
- 추가 정보가 필요할 경우 구글에서 ‘Command Line Crash Course’를 검색하거나 네이버에서 ‘명령 프롬프트 사용법’ 또는 ‘터미널 사용법’을 검색하면 관련 정보를 얻을 수 있습니다.

● 프로그램 실행 시 ‘No such file or directory’ 라는 메시지가 출력되면서 프로그램이 실행되지 않습니다

- ‘명령 프롬프트’(윈도우) 또는 ‘터미널’(맥)에서 실행할 경우 실행하고자 하는 파일이 있는 폴더까지 들어가야 합니다.
- ‘명령 프롬프트’(윈도우) 또는 ‘터미널’(맥)에서 위치를 이동하는 방법(특정 폴더에 들어가거나 나오는 방법)은 ‘cd’ 명령어를 이용하면 됩니다.
 - 가령 만약 파일이 저장된 위치가 `c:\prg\hello.py`이고 현재 사용자가 위치한 곳이 `C` 드라이버라면 ‘`cd \prg`’ 명령어를 입력함으로써 `prg` 폴더에 들어갈 수 있으며, ‘`python hello.py`’ 명령어를 입력하면 `hello.py` 파일을 실행할 수 있습니다.
 - 상위 폴더로 빠져나가는 명령어는 ‘`cd ..`’입니다. 즉, ‘`cd \폴더명`’(또는 맥의 경우 ‘`cd /폴더명`’)은 해당 폴더로 들어가는 것이고, ‘`cd ..`’은 현재 폴더에서 상위 폴드로 빠져나오는 것입니다.
- 아직 잘 이해가 되지 않는다면 구글에서 ‘Command Line Crash Course’를 검색하거나 네이버에서 ‘명령 프롬프트 명령어’ 또는 ‘터미널 명령어’를 검색하시면 관련 정보를 얻을 수 있습니다.



셸 명령어 기초



Lab Exercises

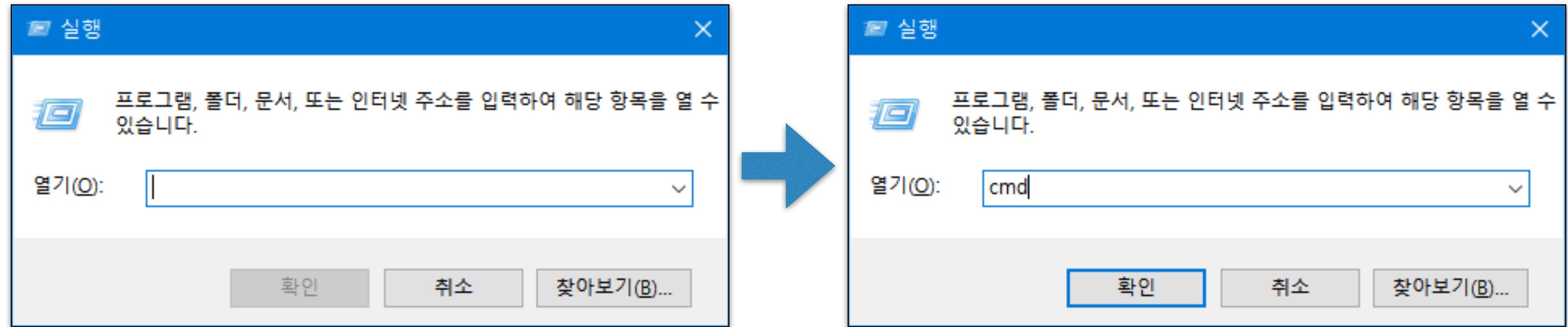




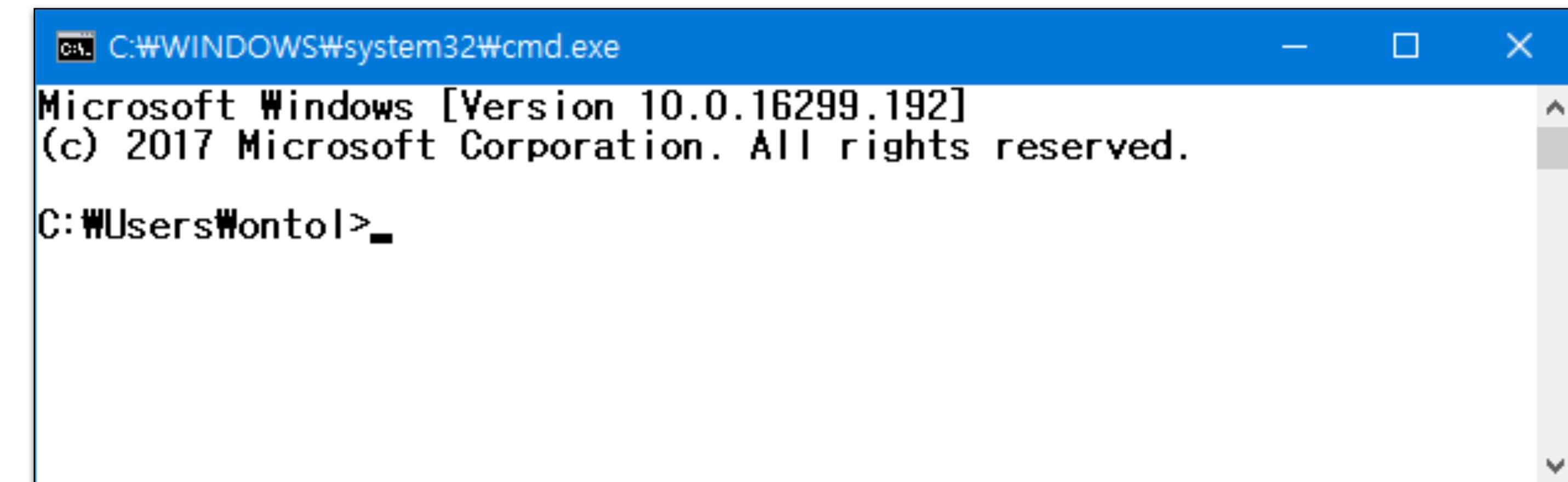
명령 프롬프트 열기



- 키보드에서 [윈도우 시작버튼(창문 아이콘)] + [R]을 누름
- 실행 창이 나타나면 ‘cmd’라고 입력한 후 [Enter] 키를 누름 → cmd.exe 파일이 실행됨



- 그러면 아래와 같이 ‘명령 프롬프트’ 창이 열림



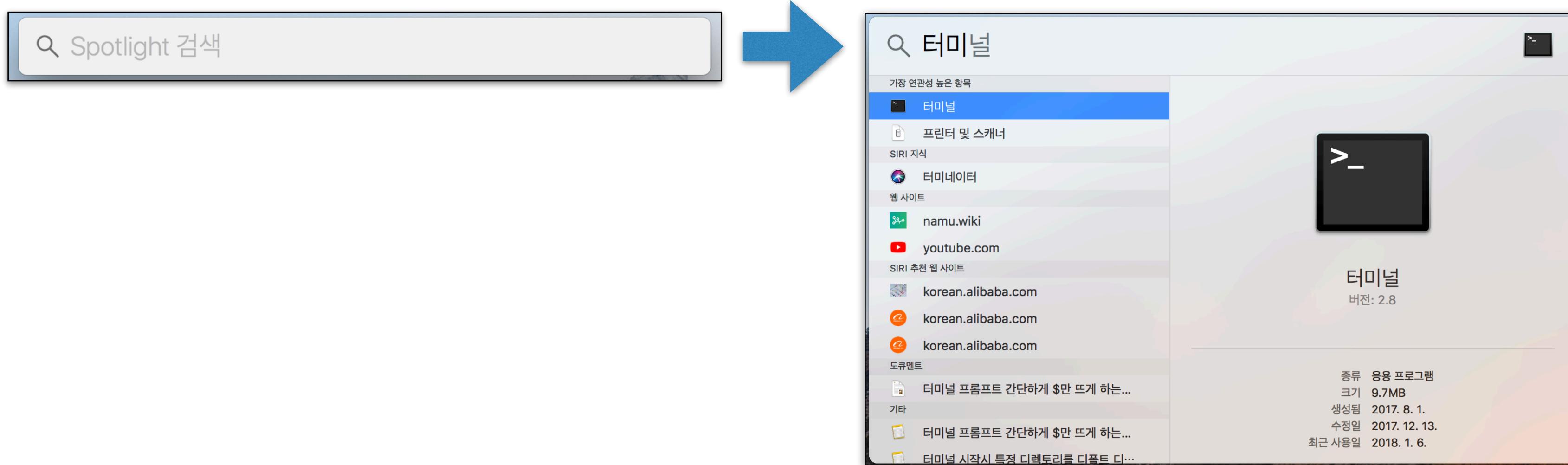


macOS®

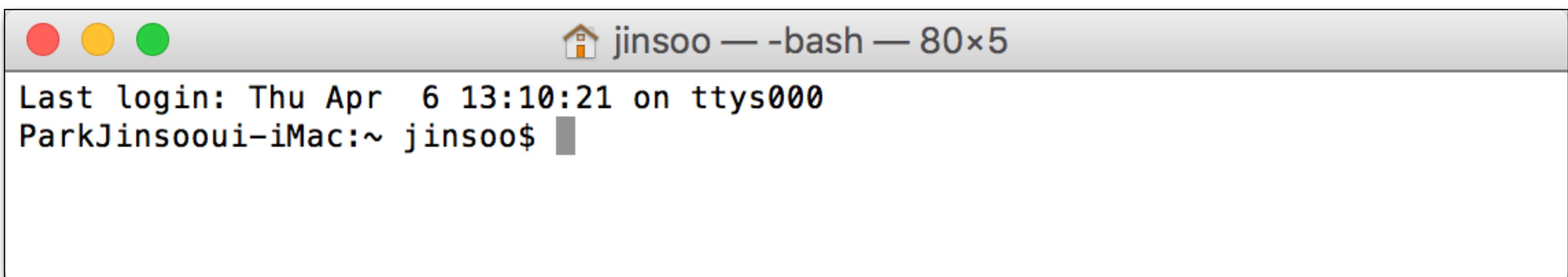
터미널 열기



- 키보드에서 [command](혹은 [control]) + [스페이스 바]를 누름
- [스포트라이트(Spotlight) 검색] 창이 나타나면 ‘터미널’이라고 입력한 후 [터미널] 항목을 더블 클릭



- 그러면 아래와 같이 ‘터미널’ 창이 열림





- **help**

- ‘명령 프롬프트’에서 제공하는 명령어 목록을 볼 수 있음

- **help [명령어]** 또는 **[명령어] /?**

- 특정 명령어에 대한 자세한 내용 도움 요청

- **cls**

- 화면에 나타난 모든 출력을 지움

- **exit**

- ‘명령 프롬프트’ 종료

```
명령 프롬프트
C:\Users\montol>help cls
화면을 지웁니다.

CLS

C:\Users\montol>cls /?
화면을 지웁니다.

CLS

C:\Users\montol>exit
```



macOS®

도우미 명령어



● man [명령어]

- 특정 명령어에 대한 자세한 내용 도움 요청

● clear

- 화면에 나타난 모든 출력을 지움

● exit

- ‘터미널’ 프로세스를 종료

The screenshot shows two terminal windows. The top window is titled 'jinsoo — bash — 80x5' and displays the command 'ParkJinsooui-iMac:~ jinsoo\$ man clear'. The bottom window is titled 'jinsoo — less ▶ man clear — 80x22' and displays the man page for 'clear(1)'. The man page includes sections for NAME, SYNOPSIS, DESCRIPTION, and SEE ALSO, along with a note about curses version 5.7.

```
jinsoo — bash — 80x5
ParkJinsooui-iMac:~ jinsoo$ man clear

jinsoo — less ▶ man clear — 80x22
clear(1)                                         clear(1)

NAME
    clear — clear the terminal screen

SYNOPSIS
    clear

DESCRIPTION
    clear clears your screen if this is possible. It looks in the environment for the terminal type and then in the terminfo database to figure out how to clear the screen.

    clear ignores any command-line parameters that may be present.

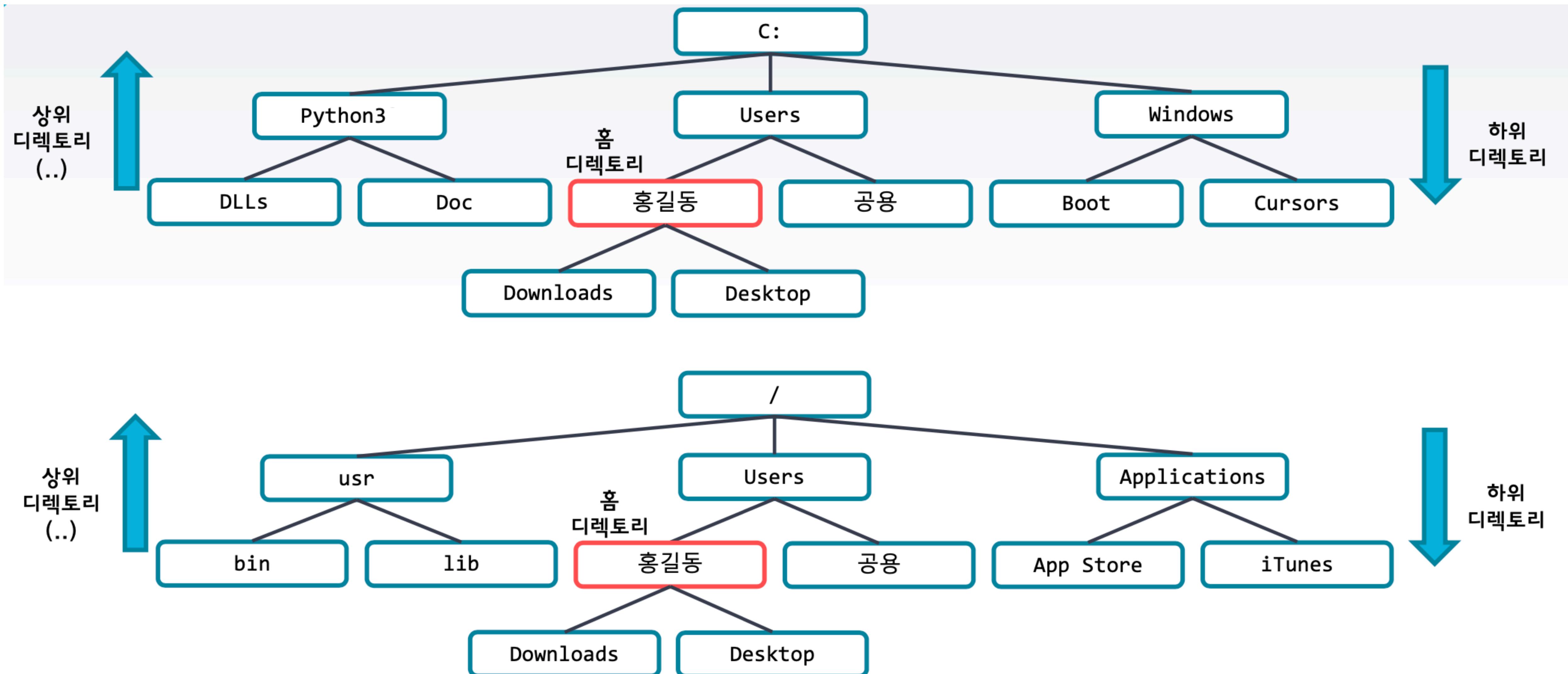
SEE ALSO
    tput(1), terminfo(5)

This describes ncurses version 5.7 (patch 20081102).

(END)
```



디렉토리 계층 구조



[출처: 이일주(2021)]



현재 작업 디렉토리 위치 출력 명령어

- 현재 작업이 진행 중인 디렉토리의 위치를 알고 싶다면 아래와 같이 입력

- ⌚ Windows의 경우

```
> cd
```

또는

```
> chdir
```

- ⌚ macOS의 경우

```
$ pwd
```

- 디렉토리(directory)
 - ▶ 운영체제 파일 시스템의 한 부분으로 파일과 다른 디렉토리를 가지고 있으며 폴더(folder)와 같은 뜻으로 사용
- 경로(path)
 - 파일 시스템에서 특정한 위치에 있는 파일이나 디렉토리를 구분자(' / ', '\ ', '₩' 등)로 분리해서 문자열로 표현한 디렉토리 트리 계층(tree hierarchy) 형식
 - e.g., Windows의 경우 'C:\Users\홍길동', macOS의 경우 '/Users/홍길동'



- 실행 예시

- Windows의 경우

```
> cd  
C:\Users\ontology
```

- macOS의 경우

```
$ pwd  
/Users/ontology
```



새로운 디렉토리 생성 명령어

- 현재 경로 아래에 새로운 디렉토리를 만들기 위해서는 아래 명령어를 입력

- 현재 작업 디렉토리에서 새로운 디렉토리를 만들면 이를 ‘하위 디렉토리’라고 함

- Windows의 경우

```
> mkdir [디렉토리 이름]
```

- macOS의 경우

```
$ mkdir [디렉토리 이름]
```

새로운 디렉토리 생성 예시



● 실행 예시

앞으로 작업할 파이썬 프로그램을 저장하기 위해 사용자 홈 디렉토리에 ‘`pyprg`’라는 이름을 가진 디렉토리를 생성

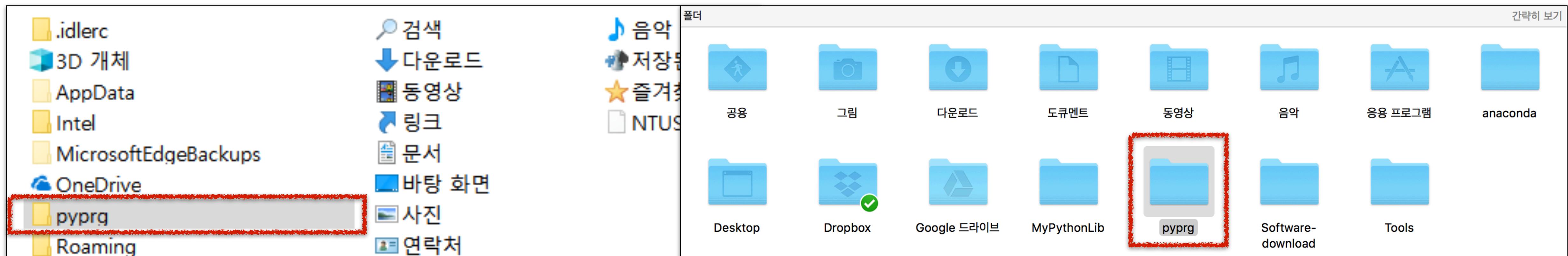
Windows의 경우

```
> mkdir pyprg
```

macOS의 경우

```
$ mkdir pyprg
```

● 사용자 홈 디렉토리 아래에 새로운 디렉토리(폴더)가 생성된 것을 확인할 수 있음





- 현재 경로에서 다른 디렉토리로 작업 디렉토리를 변경하고 싶으면 아래 명령어를 사용

- Windows의 경우

```
> cd [경로]
```

- macOS의 경우

```
$ cd [경로]
```

- 경로

- 절대경로(absolute path)

- 현재 작업 디렉토리와 상관없이 최상위(루트) 디렉토리부터 시작하는 고유한 경로
 - Windows 예 : C:\Users\ontology\pyprg\homework
 - macOS 예 : /Users/ontology/pyprg/homework

- 상대경로(relative path)

- 현재 작업 디렉토리를 기준으로 상대적인 위치에 있는 경로
 - Windows 예 : ./pyprg\homework
 - macOS 예 : ./pyprg/homework



하위 디렉토리로 경로 변경

- 하위 디렉토리로 이동하고 싶으면 아래 명령어를 사용

- Windows의 경우

```
> cd [디렉토리 이름]
```

- macOS의 경우

```
$ cd [디렉토리 이름]
```



- 실행 예시

- Windows의 경우

```
> cd pyprg  
> cd  
C:\Users\ontology\pyprg
```

- macOS의 경우

```
$ cd pyprg  
$ pwd  
/Users/ontology/pyprg
```



- 상위 디렉토리로 이동하고 싶으면 아래 명령어를 사용

- Windows의 경우

```
> cd ..
```

- macOS의 경우

```
$ cd ..
```



- 실행 예시

- Windows의 경우

```
> cd  
C:\Users\ontology\pyprg  
> cd ..  
> cd  
C:\Users\ontology
```

- macOS의 경우

```
$ pwd  
/Users/ontology/pyprg  
$ cd ..  
$ pwd  
/Users/ontology
```



기타 다른 디렉토리로 경로 변경 예시



- 실행 예시 : 사용자 홈 디렉토리에서 시작

- Windows의 경우 : ‘Windows’ 디렉토리 아래 있는 ‘System’ 디렉토리로 이동

```
> cd C:\Windows\System  
> cd  
C:\Windows\System
```

절대경로 사용

```
> cd ..\..\Windows\System  
> cd  
C:\Windows\System
```

상대경로 사용

- macOS의 경우 : ‘usr’ 디렉토리 아래 있는 ‘local’ 디렉토리의 하위 디렉토리인 ‘bin’으로 이동

```
$ cd /usr/local/bin  
$ pwd  
/usr/local/bin
```

절대경로 사용

```
$ cd ../../usr/local/bin  
$ pwd  
/usr/local/bin
```

상대경로 사용



홈 디렉토리로 복귀

- 현재 경로에서 사용자 홈 디렉토리로 돌아가기 위해서는 아래 명령어를 입력

- Windows의 경우

```
> cd %homepath% (또는 %HOMEPATH%)
```

- macOS의 경우

```
> cd ~
```

- **홈 디렉토리(home directory)**
 - ▶ 시스템에 계정을 가진 사용자가 로그인하면 그 사용자에게 할당된 개인 작업 영역의 최상위 디렉토리



- 실행 예시

- Windows의 경우

```
> cd %homepath%
> cd
C:\Users\ontology
```

- macOS의 경우

```
> cd ~
> pwd
/Users/ontology
```



디렉토리 열람 명령어

- 현재 디렉토리에 있는 파일과 하위 디렉토리 목록을 열람하기 위해서는 아래 명령어를 사용

- Windows의 경우

```
> dir
```

- macOS의 경우

```
$ ls
```



● 실행 예시

⦿ Windows의 경우

```
> dir  
C 드라이브의 볼륨: ontology  
볼륨 일련 번호: 3E0A-201C  
  
C:\Users\ontology 디렉터리  
  
2018-01-08 오후 08:51 <DIR> .  
2018-01-08 오후 08:51 <DIR> ..  
2018-01-08 오후 12:18 <DIR> Contacts  
2018-01-08 오후 12:18 <DIR> Desktop  
2017-12-11 오후 06:28 <DIR> Documents  
2018-01-08 오후 12:18 <DIR> Downloads  
2018-01-10 오전 11:47 <DIR> Favorites
```

⦿ macOS의 경우

```
$ ls  
Applications Documents Library Music Public pyprg  
Desktop Downloads Movies Pictures Tools
```



파일 복사 명령어

- 파일을 복사하기 위해서는 아래 명령어를 사용

- Windows의 경우

```
> copy [원본 (경로)파일 이름] [대상 (경로)파일 이름]
```

- macOS의 경우

```
$ cp [원본 (경로)파일 이름] [대상 (경로)파일 이름]
```

- 파일을 복사할 때 다른 이름으로 복사 가능

복사하려는 파일이 경로에 존재하기 않으면 에러 메시지가 나타남

같은 파일 이름이 대상 디렉토리에 있을 경우 기존 파일을 덮어씀



● 실행 예시

- ‘test.txt’라는 파일이 현재 작업 디렉토리에 있다고 가정

Windows의 경우

```
> copy test.txt test2.txt  
> dir /w  
...      test.txt      test2.txt  
> copy test.txt C:\Temp\test.txt  
> dir /w C:\Temp  
...      test.txt
```

macOS의 경우

```
$ cp test.txt test2.txt  
$ ls  
...      test.txt      test2.txt  
$ cp test.txt /tmp/test.txt  
$ ls /tmp  
...      test.txt
```



파이썬 프로그램 실행 명령어 : 인터프리터 모드

- 파이썬 파일을 실행하려면 아래 명령어를 사용

- Windows의 경우

```
> python [(경로)파일 이름].py
```

- macOS의 경우

```
$ python3 [(경로)파일 이름].py
```



macOS의 경우 환경 설정에 따라 **python**은 파이썬 2이고 **python3**이 파이썬 3일 수 있음

- 본 수업에서는 **python**이 파이썬 3이라 가정
- 만약 사용자 환경에서 **python3**이 파이썬 3일 경우는 **python3**을 사용해야 함



● 실행 예시

- ▣ 우측과 같은 ‘hello.py’ 파일이 현재 작업 디렉토리에 있다고 가정

▣ Windows의 경우

```
> python hello.py  
안녕 파이썬
```

```
hello.py - C:\Users\Wontol\Wp... File Edit Format Run Options Window Help print('안녕 파이썬') Ln: 1 Col: 0
```

▣ macOS의 경우

```
$ python hello.py  
안녕 파이썬
```

```
hello.py - /Users/jinso... print('안녕 파이썬') Ln: 1 Col: 0
```



● 코드 작성 및 실행

- 아래 코드를 작성한 후 'print_error.py'라는 파일명으로 저장

```
print("Hello", "Python~~~!")
print('Hello Python~~~!')
print('I love Python.')
print("So I'll master Python.")
```

- 저장한 파일 실행 예시 → 오류 발생

```
C:\Users\montol>python print_error.py
File "print_error.py", line 21
    print('I love Python.')
SyntaxError: EOL while scanning string
C:\Users\montol>
```

```
[\$ python print_error.py
File "print_error.py", line 21
    print('I love Python.')
SyntaxError: EOL while scanning string literal
\$]
```



Lab : 명령어 셸에서 파이썬 코드 실행하기



- 디렉토리 만들기

- 사용자 홈 디렉토리 아래 ‘`pyprg`’ 디렉토리를 만들고 그 아래 ‘`lab`’ 디렉토리를 생성

- 파이썬 프로그램 작성하기

- ‘`lab`’ 디렉토리에 아래 ‘`helloworld.py`’ 파일을 만들어 아래 내용을 입력한 후 저장

```
x = 'Hello '
y = 'World!'
z = x + y
print(z)
```

- 파이썬 프로그램 실행하기

- ‘`helloworld.py`’를 실행하면 아래와 같은 결과가 나와야 함

```
> python helloworld.py
Hello World!
```