

세상의 속도를
따라잡고 싶다면



점프 투 파이썬

박응용 지음(위키독스 운영자)



파이썬이란 무엇인가?



01-1 파이썬이란?

01-2 파이썬의 특징

01-3 파이썬으로 무엇을 할 수 있을까?

01-4 파이썬 설치하기

01-5 파이썬 둘러보기

01-6 파이썬과 에디터



■ 파이썬(Python)

- 1990년 암스테르담의 귀도 반 로섬(Guido Van Rossum)이 개발한 인터프리터 언어
- 파이썬의 사전적 의미
 - 고대 신화에 나오는 파르나소스 산의 동굴에 살던 큰 뱀
- 구글에서 만든 소프트웨어의 50% 이상이 파이썬으로 작성됨
- 인스타그램(Instagram), 넷플릭스(Netflix), 아마존(Amazon) 등에서 사용
- 공동 작업과 유지 보수가 매우 쉽고 편리함



- 파이썬은 인간다운 언어이다

- 파이썬은 사람이 생각하는 방식을 그대로 표현할 수 있는 언어

```
if 4 in [1, 2, 3, 4]: print("4가 있습니다")
```

||

만약 4가 1, 2, 3, 4 중에 있으면 '4가 있습니다'를 출력한다.

■ 문법이 쉬워 빠르게 배울 수 있다

- 문법 자체가 아주 쉽고 간결하며 사람의 사고 체계와 매우 닮아 있음
- 유명한 프로그래머인 에릭 레이먼드(Eric Raymond)는 공부한 지 단 하루 만에 자신이 원하는 프로그램을 작성!

■ 무료이지만 강력하다

- 오픈 소스 → 무료로 언제 어디서든 파이썬을 다운로드하여 사용 가능
- 파이썬과 C는 찰떡 궁합
 - 프로그램의 전반적인 뼈대는 파이썬으로 만들고,
빠른 실행 속도가 필요한 부분은 C로 만들어서 파이썬 프로그램 안에 포함

■ 간결하다

펄(Perl)	파이썬
100가지 방법으로 1가지 일 처리	가장 좋은 방법 1가지만 사용

- 프로그램이 실행되게 하려면
줄(들여쓰기)을 반드시 맞추어야 함
 - 가독성 ↑

```
languages = ['python', 'perl', 'c', 'java']

for lang in languages:
    if lang in ['python', 'perl']:
        print("%6s need interpreter" % lang)
    elif lang in ['c', 'java']:
        print("%6s need compiler" % lang)
    else:
        print("should not reach here")
```

- 프로그래밍을 즐기게 해 준다
 - 프로그래머가 만들고자 하는 기능에만 집중할 수 있음
- 개발 속도가 빠르다

“Life is too short, You need Python.”

인생은 너무 짧으니 파이썬이 필요해.

- 파이썬의 엄청나게 빠른 개발 속도를 두고 유행처럼 퍼진 말

■ 파이썬으로 할 수 있는 일

■ 웹 프로그래밍

- 파이썬은 웹 프로그램을 만들기에 매우 적합한 도구
- 파이썬으로 제작한 웹 사이트는 셀 수 없이 많을 정도

■ 인공지능과 머신러닝

- 자연어 처리, 음성 인식, 이미지 인식과 같은 인공지능 기술 구현
- 인공지능과 머신러닝 프로그래밍을 쉽게 할 수 있도록
scikit-learn, TensorFlow, PyTorch, Keras 등과 같은 다양한 라이브러리 제공

■ 수치 연산 프로그래밍

- C로 작성된 넘파이(Numpy) 수치 연산 모듈을 통해 빠른 수치 연산 가능



■ 파이썬으로 할 수 있는 일

■ 데이터 분석

- NumPy, Pandas, Matplotlib 등의 라이브러리를 활용해 데이터 처리, 통계 분석, 시각화를 손쉽게 수행
- 판다스가 등장한 이래 데이터 분석에 R보다 파이썬을 사용하는 사례 증가



■ 데이터베이스 프로그래밍

- Sybase, Infomix, Oracle, MySQL, PostgreSQL 등의 데이터베이스에 접근하기 위한 도구 제공
- 자료를 변형 없이 그대로 파일에 저장하고 불러오는 파이썬 모듈 피클(pickle)

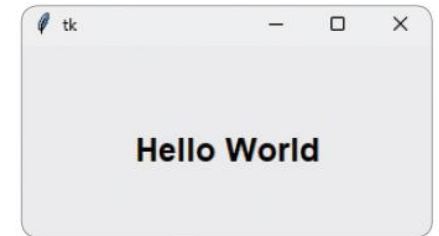
■ 시스템 유틸리티 제작하기

- 운영체제(윈도우, 리눅스 등)의 시스템 명령어를 사용하는 도구를 통한 시스템 유틸리티 제작

■ 파이썬으로 할 수 있는 일

■ GUI(Graphic User Interface) 프로그래밍

- 화면에 윈도우 창을 만들고 프로그램을 동작시킬 수 있는 메뉴나 버튼, 그림 등을 추가하는 것
- GUI 프로그래밍을 위한 도구들을 갖추고 있어, GUI 프로그램을 만들기 쉬움 (예 - Tkinter(티케이인터))

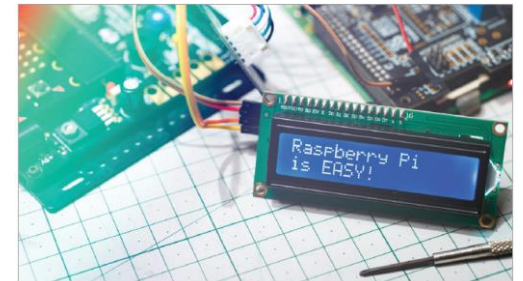


■ C/C++와 결합하기

- C나 C++로 만든 프로그램을 파이썬에서, 파이썬으로 만든 프로그램을 C나 C++에서 사용 가능

■ 사물 인터넷

- 라즈베리파이를 제어하며 사물 인터넷 구현



- 파이썬으로 할 수 없는 일

- 시스템과 밀접한 프로그래밍 영역

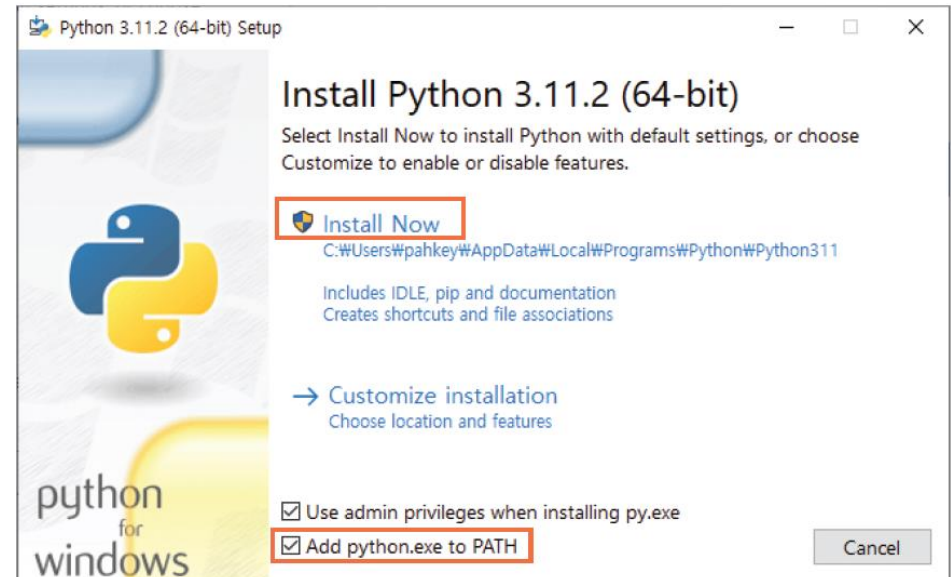
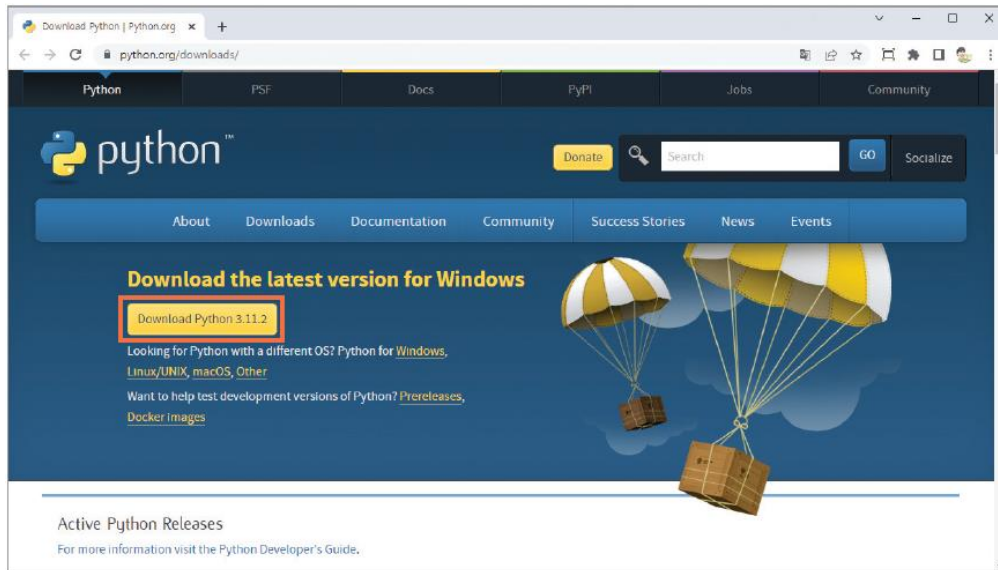
- 운영체제, 엄청난 횟수의 반복과 연산이 필요한 프로그램 등
매우 빠른 속도를 요구하거나 하드웨어를 직접 건드릴어야 하는 프로그램에는 적합하지 않음

- 모바일 프로그래밍

- 안드로이드 네이티브 앱(android native app) 개발에는 아직 역부족
 - 아이폰 앱 개발은 할 수 없음

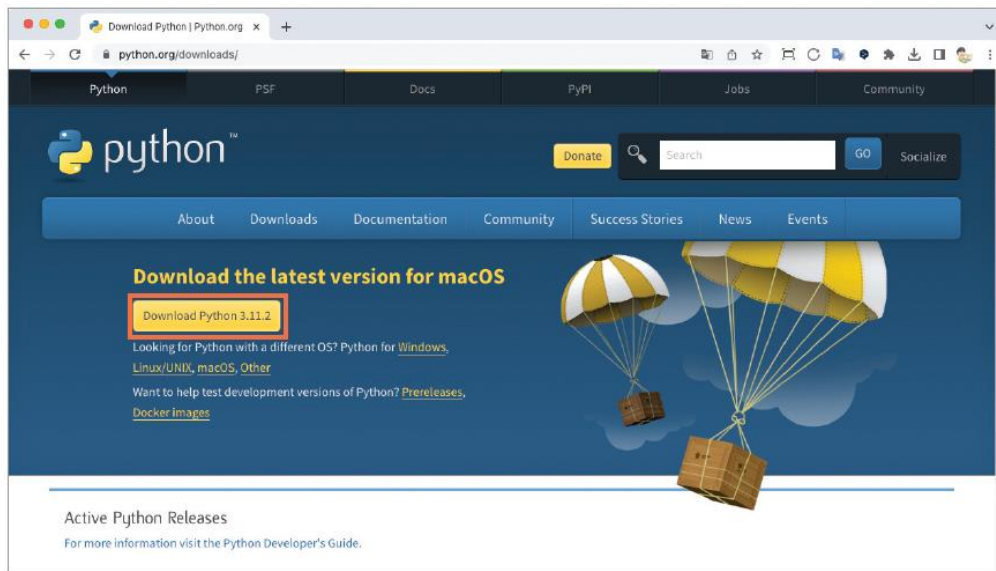
■ 윈도우에서 파이썬 설치하기

1. 파이썬 공식 홈페이지(www.python.org/downloads)에서 Python 3.x 최신 버전 다운로드
2. 파이썬이 어느 곳에서든지 실행될 수 있도록 '**Add python.exe to PATH**' 옵션 선택



■ 맥에서 파이썬 설치하기

1. 파이썬 공식 홈페이지(www.python.org)의 [Downloads] 메뉴에서 맥용 설치 파일 다운로드
2. 내려받은 **python-3.11.x-macos1.pkg** 파일을 실행하여 설치
3. 터미널에서 명령을 입력해 자신의 맥에 설치된 파이썬 버전 확인



pahkey@mymac ~ % **python3 -V** 직접 입력해 보자.
Python 3.11.2

```
-zsh
Last login: Sat Apr 15 10:13:55 on ttys000
pahkey@mymac ~ % python3 -V
Python 3.11.2
pahkey@mymac ~ %
```

■ 파이썬 기초 실습 준비하기

■ 파이썬 대화형 인터프리터 실행

- [시작] → 'python'을 검색 → 'Python 3.11(64-bit)'을 찾아 실행
※ 맥 사용자는 터미널에서 'python3'을 입력



■ 파이썬 대화형 인터프리터(파이썬 셸)

- 사용자가 입력한 소스 코드를 실행하는 환경

```
Python 3.11.2 (tags/v3.11.2:878ead1, Feb 7 2023, 16:38:35) [MSC v.1934 64 bit  
(AMD64)] on win32
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>>
```

프롬프트(>>>)가 있는 곳이 입력하는 부분

■ 파이썬 기초 실습 준비하기

■ 수식 입력

- 입력에 따른 결과값이 바로 출력됨

```
>>> 1 + 1 ← 수식 입력
2 ← 결과값 바로 출력
>>>
```

■ 대화형 인터프리터 종료

- 1) Ctrl + Z → Enter
- 2) 내장함수 quit()이나 exit()를 사용
- 3) sys 모듈 사용

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

■ 파이썬 기초 문법 따라 해 보기

■ 사칙 연산

- 덧셈(+), 뺄셈(-), 나눗셈(/), 곱셈(*)

```
>>> 1 + 2  
3
```

```
>>> 3 / 2.4  
1.25  
>>> 3 * 9  
27
```

■ 변수에 숫자 대입하고 계산하기

```
>>> a = 1  
>>> b = 2  
>>> a + b  
3
```

■ 변수에 문자 대입하고 출력하기

```
>>> a = "Python"  
>>> print(a)  
Python
```

```
>>> a = "Python"  
>>> a  
'Python'
```

■ 조건문 if

```
>>> a = 3  
>>> if a > 1:  
...     print("a는 1보다 큼니다.")  
...  
a는 1보다 큼니다.
```

들여쓰기((Spacebar) 4칸)
Enter 입력

■ 파이썬 기초 문법 따라 해 보기

■ 반복문 for

- 대괄호([]) 사이에 있는 값을 하나씩 출력

```
>>> for a in [1, 2, 3]:  
...     print(a) ↓  
... ↓  
1  
2  
3
```




■ 반복문 while

- 일정한 조건이 만족하는 동안 문장을 반복 수행

```
>>> i = 0  
>>> while i < 3:  
...     i = i + 1 ↓  
...     print(i) ↓  
... ↓  
1  
2  
3
```

■ 파이썬 기초 문법 따라 해 보기

■ 함수

```
>>> def add(a, b):  
...      return a + b   
...       
>>> add(3, 4)  
7
```

- 파이썬에서 함수를 정의할 때 사용하는 예약어 def
※ 예약어: 프로그래밍 언어에서 이미 문법적인 용도로 사용하고 있는 단어
- return은 함수에서 값을 변환할 때 사용

01-6 파이썬과 에디터

■ IDLE 에디터로 파이썬 프로그램 작성하기

■ 파이썬 IDLE 실행

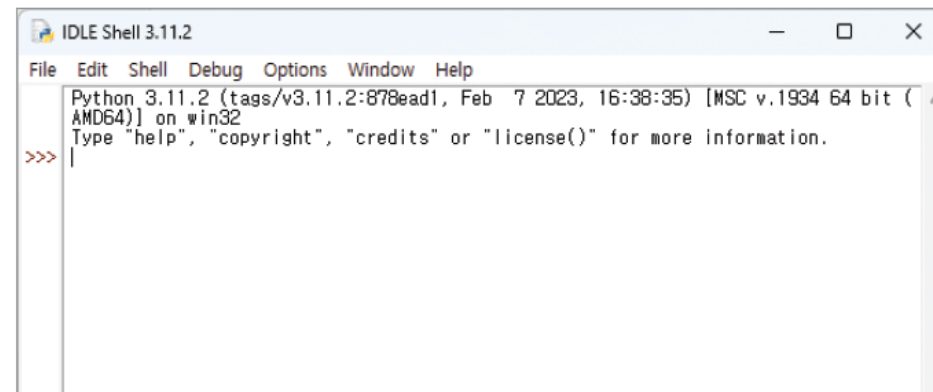
- IDLE(Integrated Development and Learning Environment)
 - 파이썬 프로그램 작성을 도와주는 통합 개발 환경

■ [시작] → 'idle' 검색



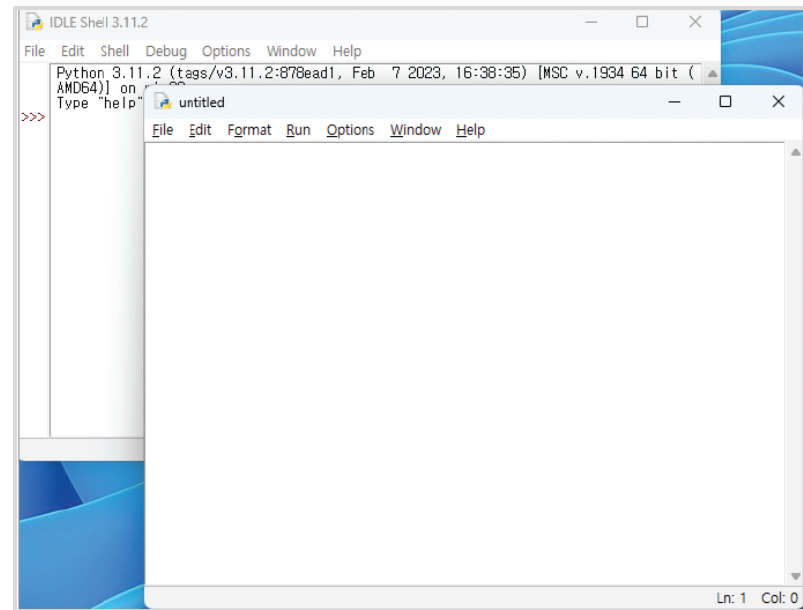
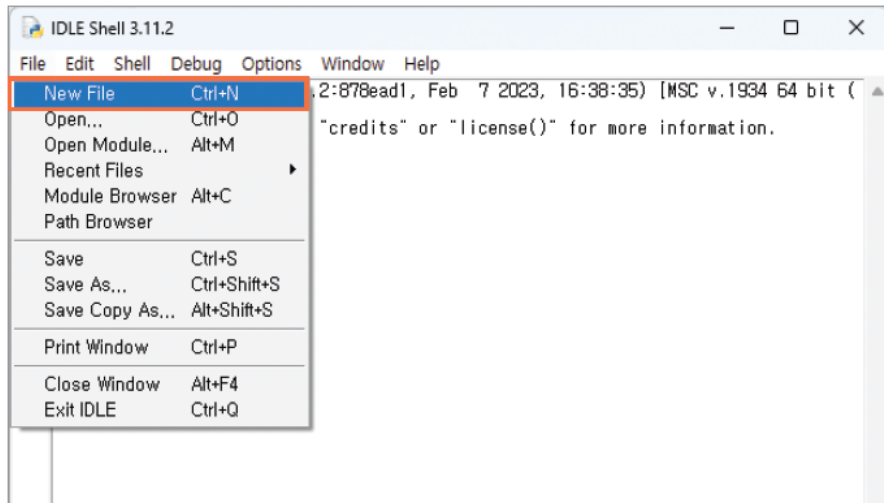
■ IDLE 셸(Shell) 창

- IDLE 에디터로 작성한 프로그램의 실행 결과를 표시하는 용도로만 사용



01-6 파이썬과 에디터

- IDLE 에디터로 파이썬 프로그램 작성하기
 - IDLE 에디터(Editor) 실행



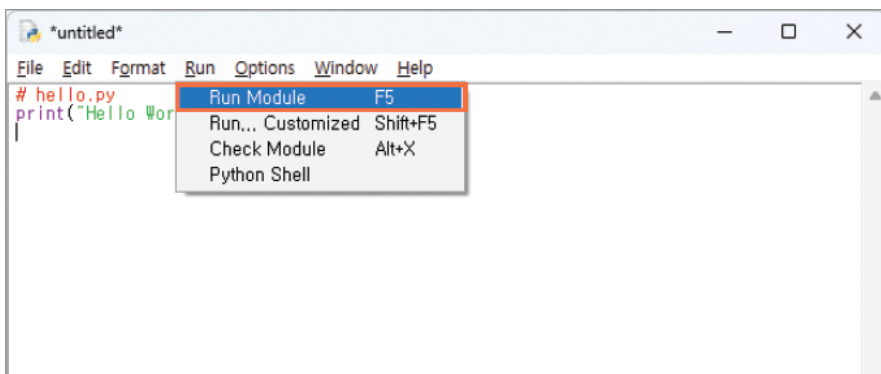
01-6 파이썬과 에디터

■ IDLE 에디터로 파이썬 프로그램 작성하기

1) 파이썬 프로그램 작성

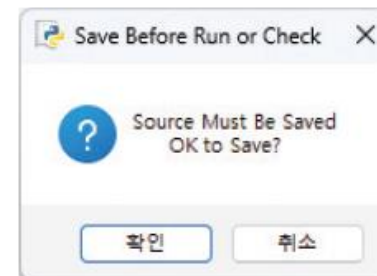
```
# hello.py  
print("Hello World")
```

2) 작성한 프로그램 실행(F5)



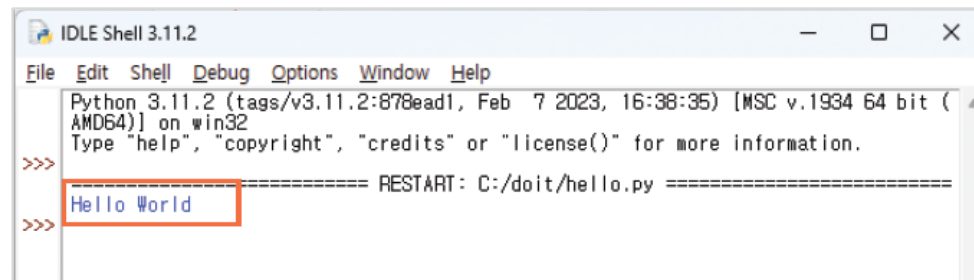
3) 파일 저장

- 경로) C:\doit\hello.py



4) 실행 결과

- IDLE 셸 창에 표시됨

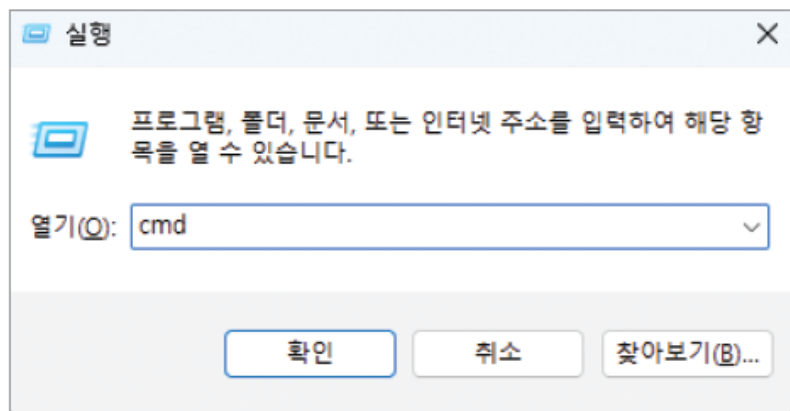


01-6 파이썬과 에디터

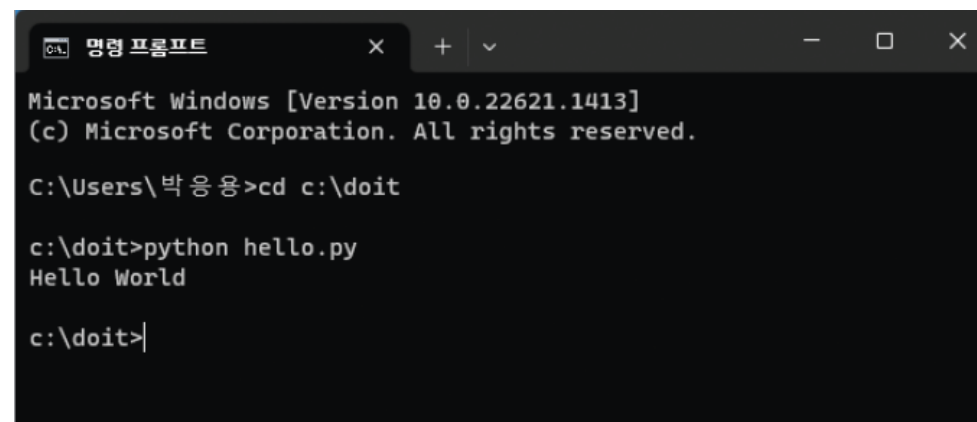
■ 명령 프롬프트 창에서 파이썬 프로그램 실행하기

1) 명령 프롬프트 창(command prompt) 열기

- 윈도우 키 + R → 'cmd' 입력 후 Enter



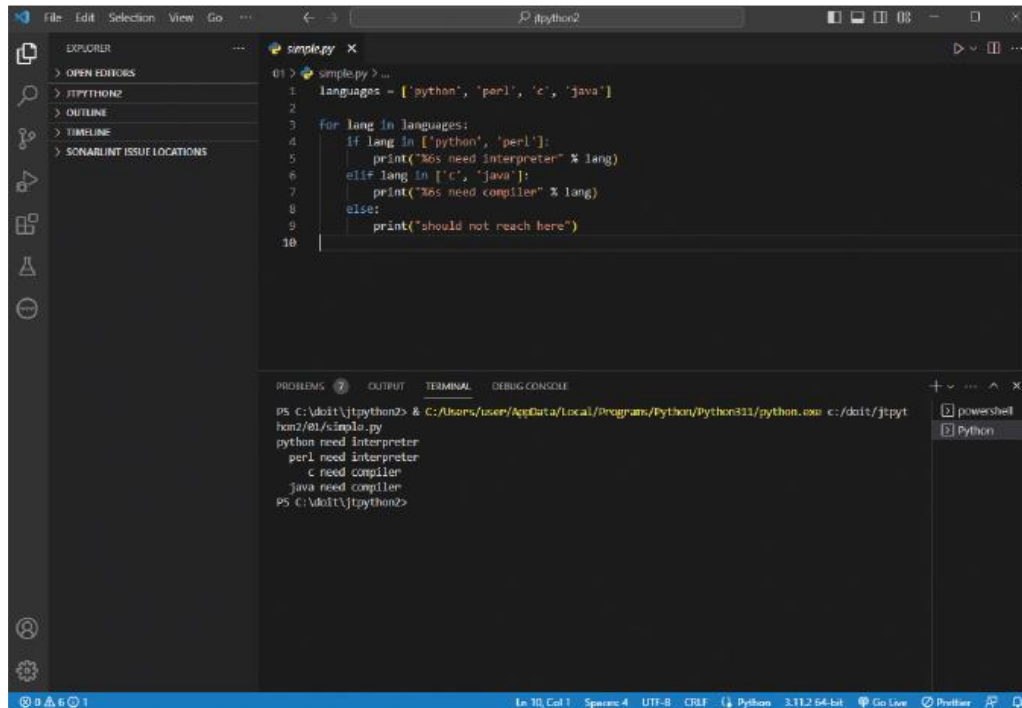
2) hello.py 저장한 경로로 이동하여 실행



01-6 파이썬과 에디터

■ 추천하는 에디터

- 비주얼 스튜디오 코드(code.visualstudio.com)



- 파이참
(www.jetbrains.com/pycharm/download)



*"Life is too short,
You need Python!"*

인생은 너무 짧으니,
파이썬이 필요해!

감사합니다.