

1주차  
파이썬 기초 프로그래밍



# 2020-2 파이썬 프로그래밍 강좌

- 강의 목표
  - 파이썬 언어로 작성된 프로그램을 읽고 이해한다.
  - 주어진 문제를 해결하는 파이썬 프로그램을 작성할 수 있다.
  - 파이썬 생태계에 수월하게 접근하기 위한 기초 실력 배양
- 교재
  - IT Cook book, 파이썬 for Beginner 2판, 우재남 지음, 한빛미디어
- 평가
  - 중간고사(20%), 기말고사(30%), 과제/미니 프로젝트(30%), 출석(20%)
- 수업 방식
  - 당분간 ZOOM을 이용하여 실시간 온라인 수업 운용
  - 상황에 따라 대면 수업/혼용 및 대면 수업 예정
  - 과제는 구글 클래스 룸 사용

# 강의계획

주	해당 장	주제
1	1장	프로그래밍 언어의 개념과 종류, 파이썬 소개, 파이썬 설치와 실행
2	2장	간단 계산기 프로그램 작성을 통한 프로그램 기본 작성법
3	3장	변수와 데이터형
4	4장	연산자
5	5장	조건문
6	6장	반복문
7	7장	리스트, 튜플, 딕셔너리
8		중간고사
9	8장	문자열
10	9장	함수와 모듈
11	10장	윈도 프로그래밍
12	11장	파일 입출력
13	12장	객체지향 프로그래밍
14	13장	데이터베이스
15	14장	미니 프로젝트(미니 편집기, 미니 포토샵, 슈팅 게임 등)
16		기말고사

# 소스 코드 공유, 과제 제출

- 수업 소스 코드 공유
  - <https://github.com/DIT-Python>
- 과제 제출 : 구글 클래스 룸
  - 2-1 :  
<https://classroom.google.com/c/MTMyMTgzNTc0ODA0?cjc=a6jkb56>
  - 2-2 :  
<https://classroom.google.com/c/MTMyMTgzNTc0ODc2?cjc=xauxk23>
  - 2-3 :  
<https://classroom.google.com/c/MTMyMTgzNTc0OTIx?cjc=yqn5rwk>

# 참고 학습 사이트 : Ask Django

- <https://www.askcompany.kr/vod/python/>

## 파이썬 차근차근 시작하기



본 코스는 **공개 VOD**로 제공됩니다. (공개 코스는 별도 로그인없이 내용을 확인하실 수 있습니다.)

보다 자연스러운 영어를 쓰기 위해서는, 영어 문법을 익히는 것 이외에도 영어권 문화를 익히는 것이 필요합니다.

파이썬 언어도 이와 마찬가지로 단순히 파이썬 문법을 익히는 것 이외에도 파이썬의 철학을 제대로 알 필요가 있습니다.

우리는 장고 웹프레임워크를 통해 웹서비스 개발을 해볼텐데요, 장고는 파이썬으로 구현된 웹프레임워크이니만큼 파이썬을 잘 이해할 수록 더욱 잘 쓰실 수 있습니다.

장고를 보다 효율적으로 사용하기 위한 목적으로서의 파이썬에 대해서 공부해봅시다.

1. 시작하기 (2017-03-31 발행)

NB PDF 13분 18초

2. 데이터타입 (2017-03-31 발행)

NB PDF 23분 27초

3. 기본 자료 구조 (2017-03-31 발행)

NB PDF 37분 4초

4. 블록문, 들여쓰기, 주석 (2017-03-31 발행)

NB PDF 16분 23초

5. 흐름 제어 (2017-03-31 발행)

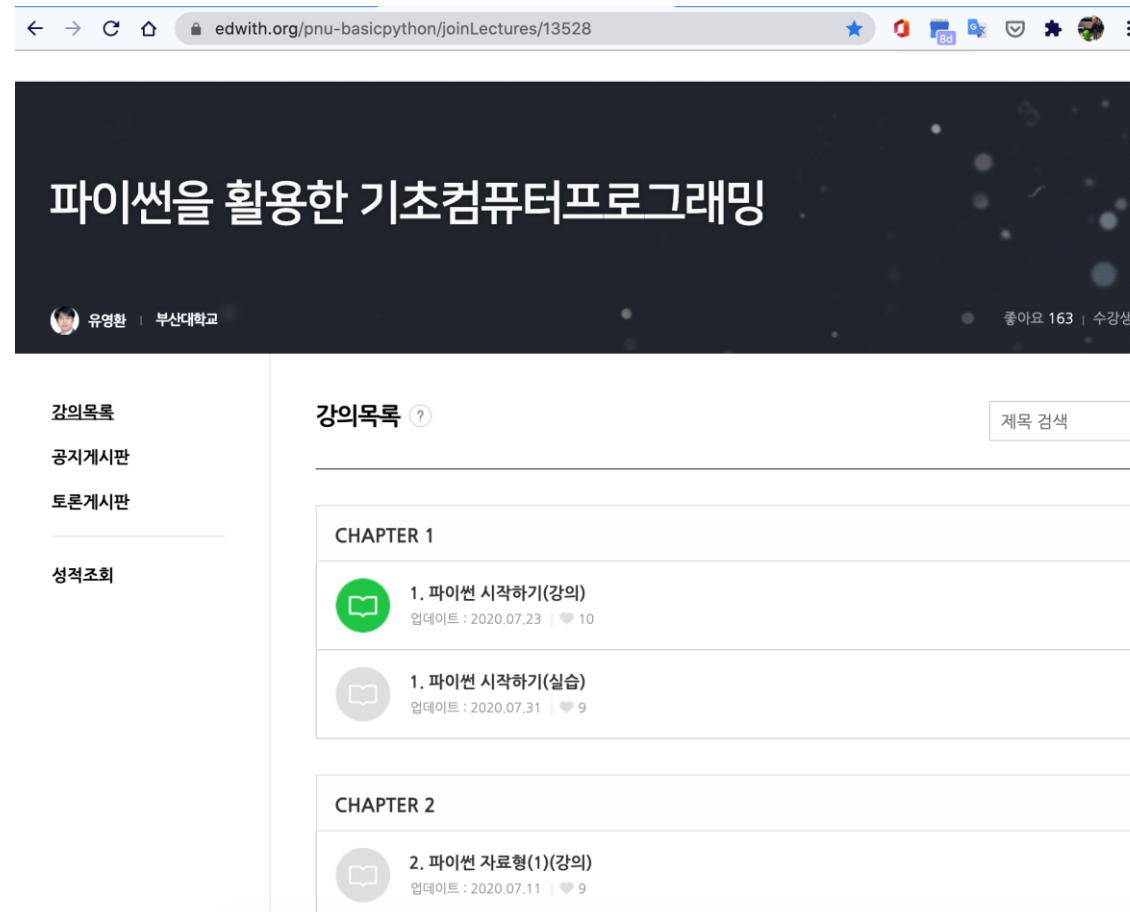
NB PDF 16분 57초

6. 함수 (2017-03-31 발행)

NB PDF 24분 25초

# 참고 학습 사이트 : edwith

- <https://www.edwith.org/pnu-basicpython/joinLectures/13528>



The screenshot shows a web browser window with the URL [edwith.org/pnu-basicpython/joinLectures/13528](https://www.edwith.org/pnu-basicpython/joinLectures/13528). The page title is "파이썬을 활용한 기초컴퓨터프로그래밍" (Basic Computer Programming Using Python). The instructor is "유영환" (Yoo Young-hwan) from "부산대학교" (Pusan National University). The page has 163 likes and 1 student. The left sidebar contains links: "강의목록" (Lecture List), "공지게시판" (Notice Board), "토론게시판" (Discussion Board), and "성적조회" (Grade Check). The main content area shows a "강의목록" (Lecture List) with a search bar. It lists two chapters: "CHAPTER 1" and "CHAPTER 2". Chapter 1 includes "1. 파이썬 시작하기(강의)" (1. Python Start (Lecture)) and "1. 파이썬 시작하기(실습)" (1. Python Start (Practice)). Chapter 2 includes "2. 파이썬 자료형(1)(강의)" (2. Python Data Type (1) (Lecture)).

← → ↺ ⌂ edwith.org/pnu-basicpython/joinLectures/13528 ★ 8d 6% 163 1

## 파이썬을 활용한 기초컴퓨터프로그래밍

유영환 | 부산대학교 좋아요 163 | 수강생 1

강의목록 공지게시판 토론게시판 성적조회

강의목록 ? 제목 검색

### CHAPTER 1

1. 파이썬 시작하기(강의)  
업데이트 : 2020.07.23 | 10

1. 파이썬 시작하기(실습)  
업데이트 : 2020.07.31 | 9

### CHAPTER 2

2. 파이썬 자료형(1)(강의)  
업데이트 : 2020.07.11 | 9

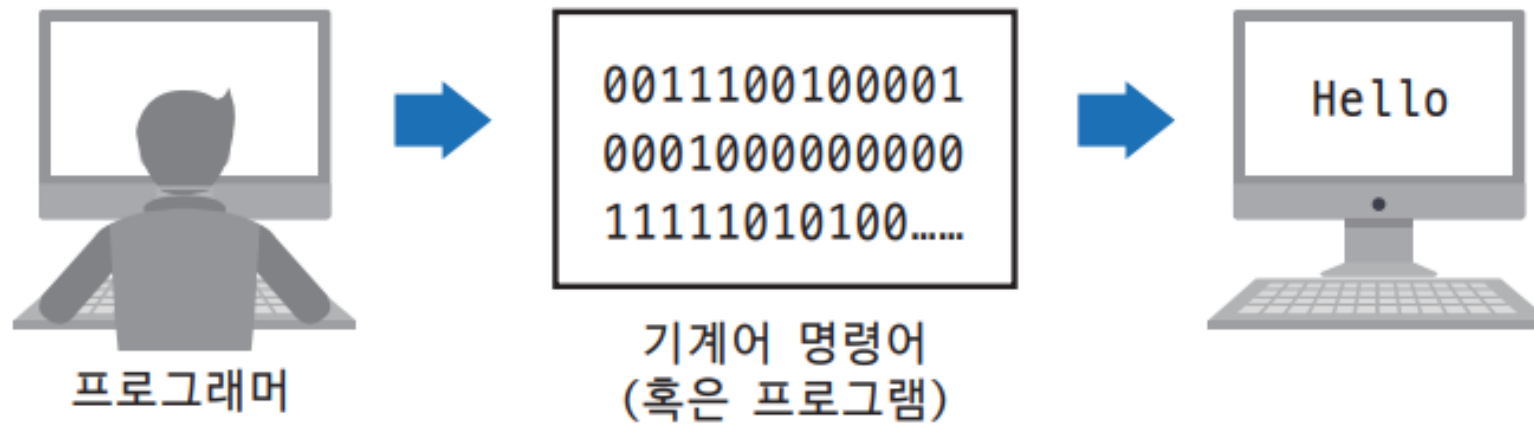
# 파이썬 소개 및 개발 환경 설정

## 학습목표

- 파이썬의 역사와 파이썬이 가진 특징을 이해한다.
- 컴파일러와 인터프리터 언어의 차이를 이해한다.
- 파이썬의 설치방법과 사용법을 이해한다.
- 간단한 파이썬 프로그램을 만들 수 있다.
- 파이썬 프로그램의 명령이 해석되고 실행되는 과정에 대해 이해한다.
- 파이썬 통합 개발환경의 필요성을 이해하고, 활용할 수 있다.
- 파이썬 프로그램을 통해 원하는 메시지를 출력할 수 있다.
- 대화형 모드를 이용하여 간단한 계산을 수행할 수 있다.

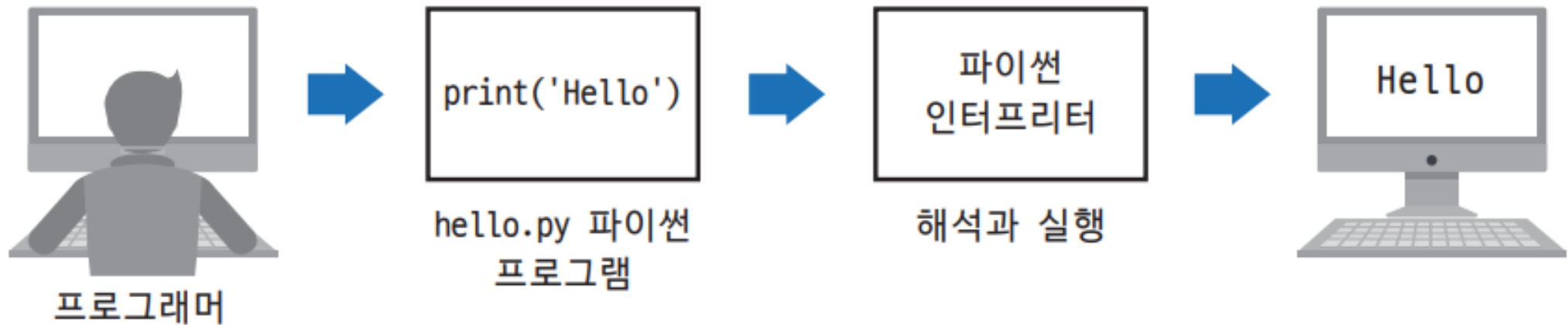


- 컴퓨터는 0과 1의 이진 값만을 이해하고 저장
- 컴퓨터가 수행하는 명령은 001110010001000... 의 값으로 되어 있음



- 01001000100...과 같은 명령은 사람이 이해하기 어렵고 작성 시간이 오래 걸리며 오류가 많고 수정이 힘들다.
- 사람이 이해할 수 있는 프로그래밍 언어를 이용하여 명령 입력

- 프로그래머는 hello.py라는 이름의 파이썬 명령어를 이용한 프로그램을 작성
- 인터프리터라는 프로그램이 hello.py라는 프로그램을 기계어 명령어로 변환하여 컴퓨터에서 실행



# 프로그래밍 언어

- 컴퓨터 시스템을 구동시키는 소프트웨어를 만들기 위한 **형식을 제대로 갖춘 언어**
- 컴퓨터가 수행할 수 있어야만 한다
- 자동차나 프린터 등의 미리 틀이 정해진 기계(**임베디드 시스템 Embedded System**)에서 제한된 목적과 용도로 사용되는 프로그래밍 언어는 수행 속도가 중요하므로 C 언어를 주로 사용
- 웹 서비스를 위해서는 HTML과 같은 마크업 언어와 JavaScript, PHP와 같은 언어가 많이 사용
- 그 중 파이썬은 강력하면서도 배우기 쉬움

# 파이썬 소개

- 파이썬
  - 1989년 귀도 반 로섬([Guido Van Rossum](#))에 의해 개발
  - 인터프리터 방식의 객체지향 프로그래밍 언어



[그림 1-5] 파이썬을 개발한 귀도 반 로섬  
(출처: 위키백과)



[그림 1-6] Python 로고

# 파이썬 소개

- 소스 코드 `source code`
  - 프로그래밍 언어로 작성된 명령어들의 목록

< 소스코드 예시 >

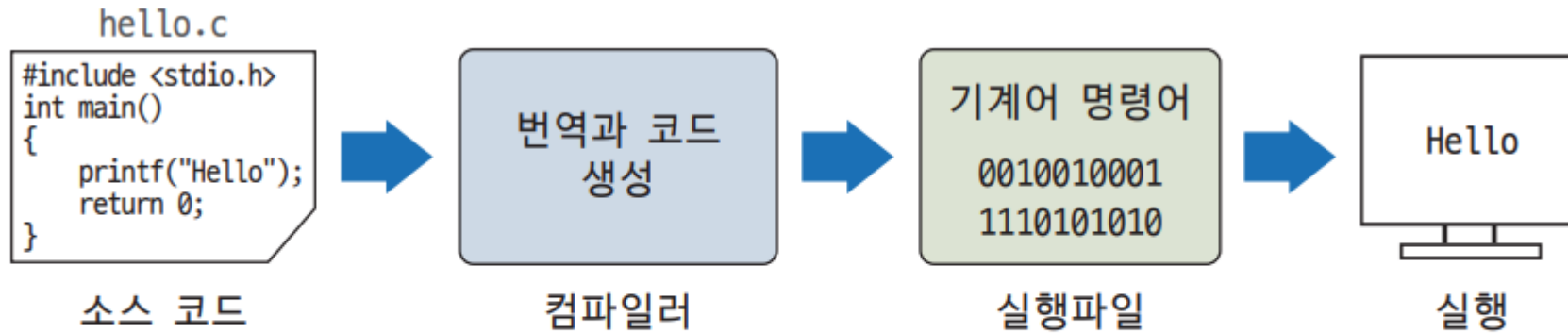
- 소스 파일 `source file`
  - 소스 코드가 저장된 파일

```
age = input('나이를 입력하세요')  
  
if int(age) < 19:  
    print('할인되었습니다.')  
else:  
    print('할인이 안 됩니다.')
```

정해진 문법에 맞게 명령을 입력하면  
컴퓨터는 이 코드를 실행하여 결과를 보여준다

- 컴파일 방식

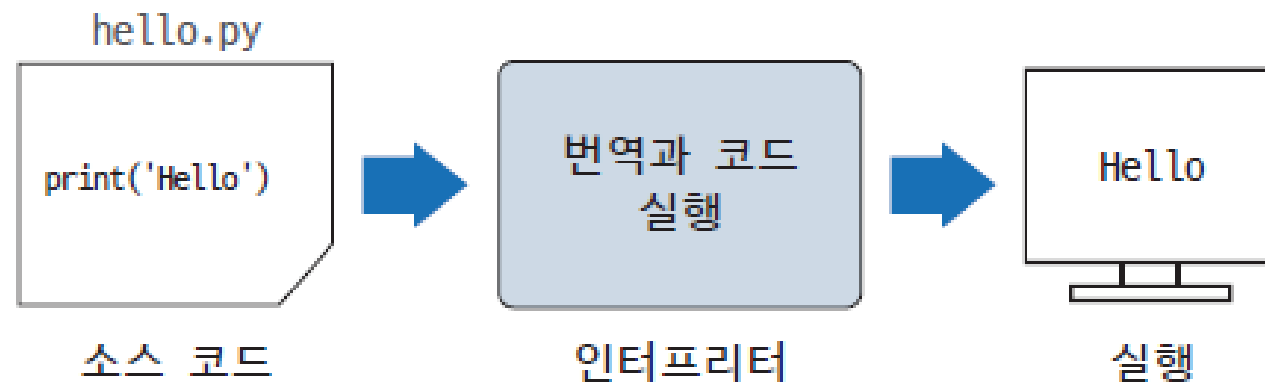
- 프로그램 명령어를 기계어로 번역한 후 이 기계어를 실행하는 방식
- C, C++, 파스칼 언어등이 있음



[그림 1-8] 컴파일 방식의 프로그램이 동작하는 원리: 컴파일러는 소스 코드를 번역하여 기계어 명령어를 생성한다. 이 기계어는 컴퓨터에서 실행되어 결과를 만든다.

- 인터프리터 방식

- 프로그램 명령어를 한 번에 한 줄씩 읽어 번역한 후 바로 실행
- 파이썬, BASIC 등의 언어가 있음



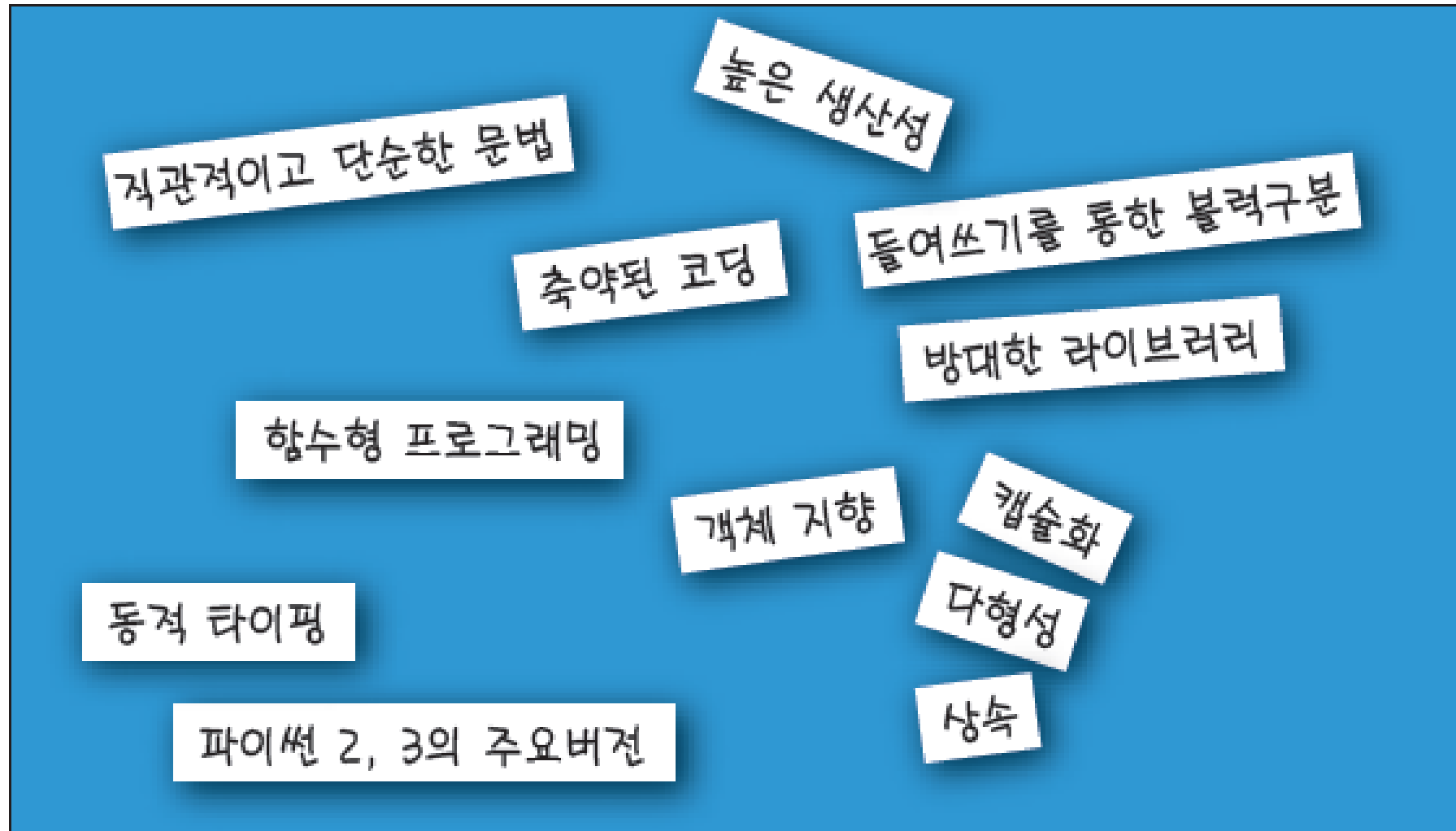
[그림 1-9] 인터프리터 방식의 프로그램이 동작하는 원리: 인터프리터는 소스 코드를 한 줄씩 읽어들이어서 번역한 후 이 코드를 실행하는 방식으로 동작한다.

# 인터프리터 방식과 컴파일 방식의 비교

	인터프리터 방식	컴파일 방식
정의	명령어들을 한 번에 한 줄씩 읽어 들여서 실행하는 방식이다.	명령어를 기계어로 번역하는 방식이다.
장점	컴파일 단계를 거칠 필요가 없다.	일반적인 경우 속도가 더 빠르다.
단점	실행 시간이 느리다.	원시 프로그램의 크기가 크다면 상당한 시간이 소요된다.
사용되는 언어	파이썬, BASIC 등	C, FORTRAN, PASCAL 등



# 파이썬의 특징



# 파이썬이 인기 있는 이유

- 1) 직관적이고 단순한 문법
- 2) 배우는데 시간이 비교적 적게 걸림
- 3) 축약된 코딩이 가능함
- 4) 짧은 코딩으로 많은 기능을 수행할 수 있음
- 5) 오픈소스 [Open source](#) 방식을 채택하여 방대한 라이브러리를 무료로 편리하게 이용할 수 있음
- 6) 객체지향 프로그래밍 언어의 특징을 가진다

# TIOBE 인덱스

Oct 2019	Oct 2018	Change	Programming Language	Ratings	Change
1	1		Java	16.884%	-0.92%
2	2		C	16.180%	+0.80%
3	4	▲	Python	9.089%	+1.93%
4	3	▼	C++	6.229%	-1.36%
5	6	▲	C#	3.860%	+0.97%
6	5	▼	Visual Basic .NET	3.745%	-2.14%
7	8	▲	JavaScript	2.076%	-0.20%

[그림 1-11] 프로그래밍 언어의 인기를 매기는 TIOBE 지수

# 파이썬 사용 환경 구축

- 방법 1 : 파이썬 설치
  - <https://www.python.org/>
- 방법 2 : Anaconda3 설치
  - <https://www.anaconda.com/downloads>
  - Jupyter Notebook 등 다양한 개발 도구 제공
  - 파이썬이 내장되어 있음
- 통합 개발 도구(IDE)
  - Pycharm
    - <https://www.jetbrains.com/ko-kr/pycharm/download/#section=windows>
  - VS Code
    - <https://code.visualstudio.com/>

# 파이썬 설치와 사용

- 홈페이지 접속
  - <http://www.python.org/>



- 다운로드 페이지 접속
  - 이 화면과 파이썬 버전은 최신 버전이 아닐 수 있습니다



[그림 1-13] 파이썬 다운로드 웹페이지

- 또는 Google.com에 접속하여 “python download”를 검색

The screenshot shows a Google search interface with the query "python download". The search results are filtered to "All" (전체). The top result is from python.org, titled "Download Python | Python.org". Below the title is the URL "https://www.python.org/downloads/" and a link to "이 페이지 번역하기" (Translate this page). The description states: "The official home of the Python Programming Language." Below this, there are four columns of links: "Python 3.7.0 released", "Python 3.7.1", "Windows", and "Mac OS X". Each link has a brief description. At the bottom, there is a link to "python.org 검색결과 더보기 >>" (View more search results for python.org). Below this, there is another result titled "Welcome to Python.org" with the URL "https://www.python.org/" and a link to "이 페이지 번역하기". The description for this result says: "The official home of the Python Programming Language. ... Download. Python source code and installers are available for download for all versions!"

Google

python download

전체 동영상 도서 뉴스 지도 더보기

설정 도구

검색결과 약 426,000,000개 (0.53초)

[Download Python | Python.org](#)  
<https://www.python.org/downloads/> ▼ [이 페이지 번역하기](#)  
The official home of the Python Programming Language.

[Python 3.7.0 released](#)  
Python 3.7.0 is the newest major release of the Python ...

[Python 2.7.15](#)  
Python 2.7.15 is a bugfix release in the Python 2.7 series.

[Python 3.7.1](#)  
Python 3.7.1 is the first maintenance release of Python ...

[Python 3.6.7](#)  
Python 3.6.7 is the seventh and current maintenance release ...

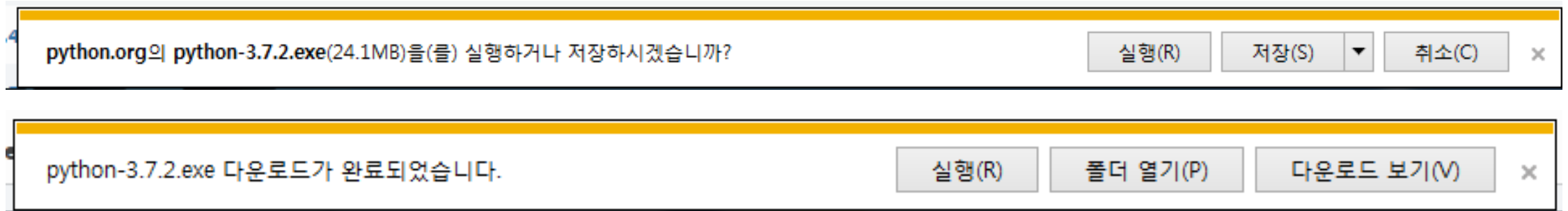
[Windows](#)  
Python Releases for Windows. Latest Python 3 Release ...

[Mac OS X](#)  
Python Releases for Mac OS X. Latest Python 3 Release ...

[python.org 검색결과 더보기 >>](#)

[Welcome to Python.org](#)  
<https://www.python.org/> ▼ [이 페이지 번역하기](#)  
The official home of the Python Programming Language. ... **Download.** Python source code and installers are available for **download** for all versions!

- 다운로드 페이지에서 "Download Python 3.7.2"를 선택
- 운영체제(macOS, Windows, Linux 등)를 선택 후 다운





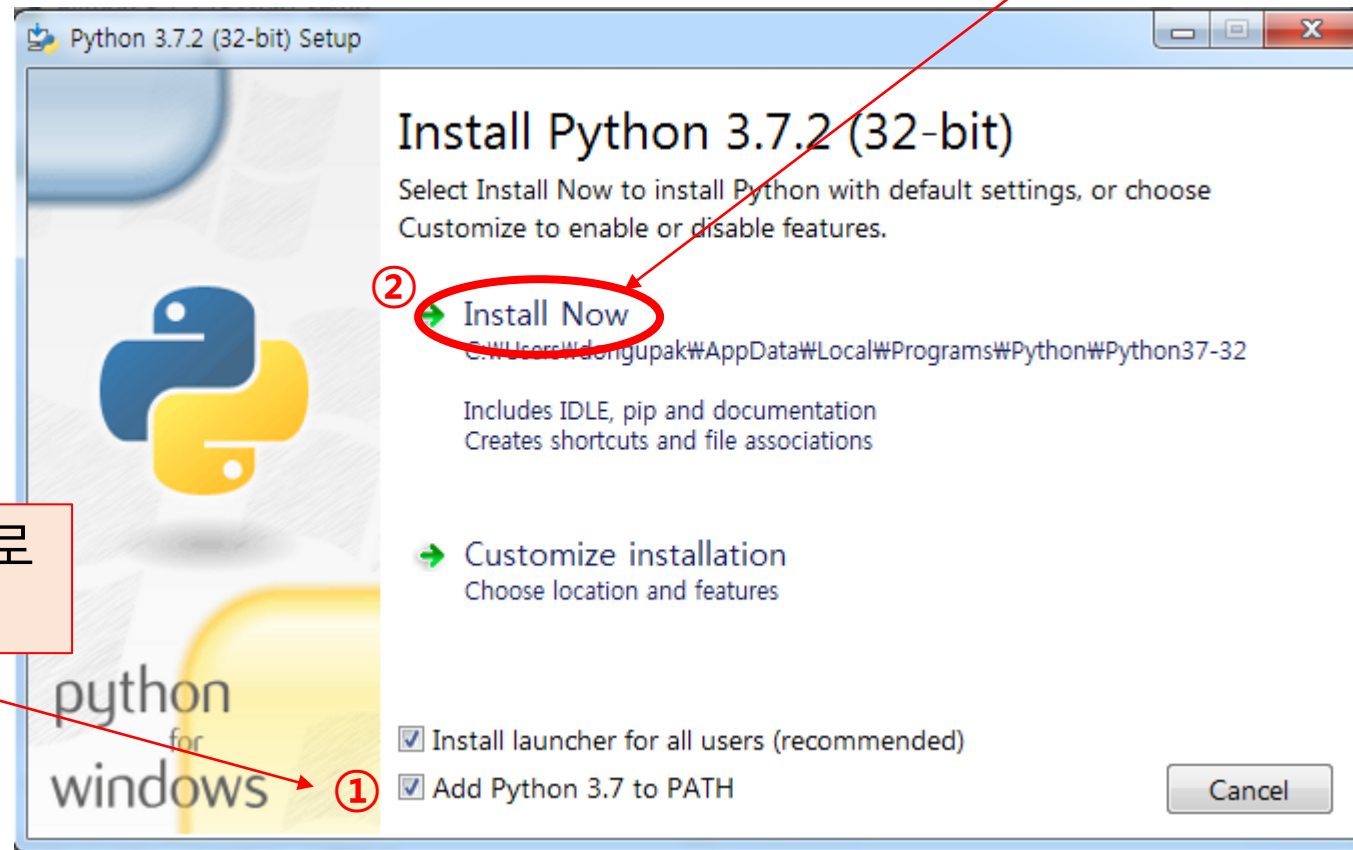
- python-3.7.2.exe 실행 후

“Install launcher for all users(recommended)”

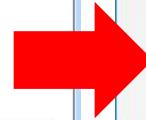
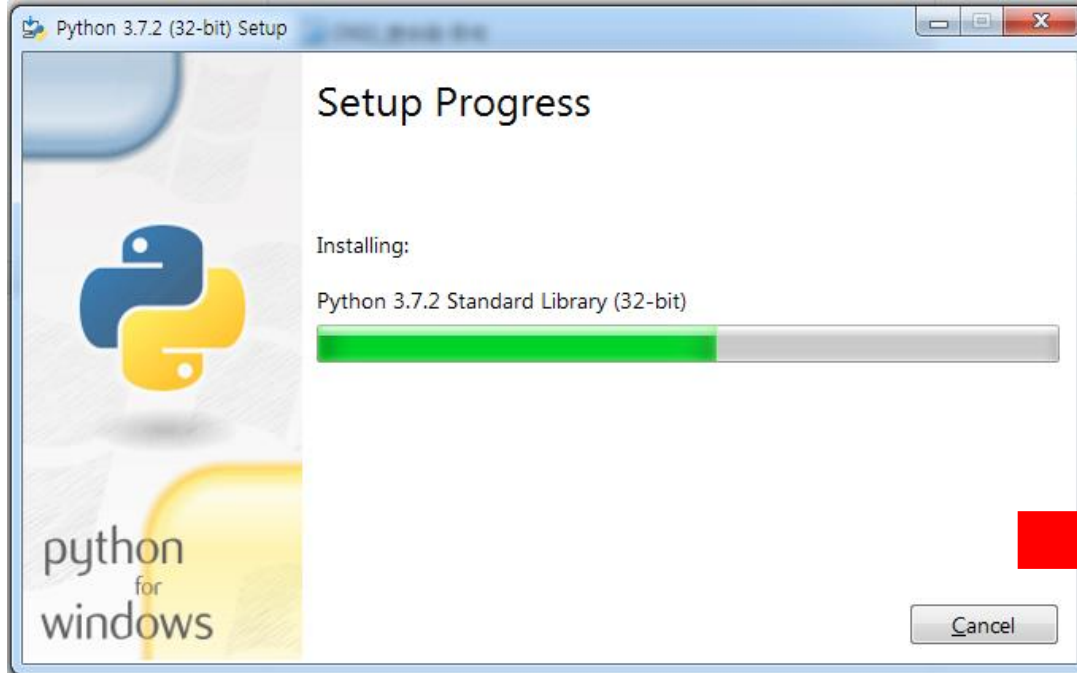
“Add Python 3.7 to PATH” 선택

Install Now를 선택

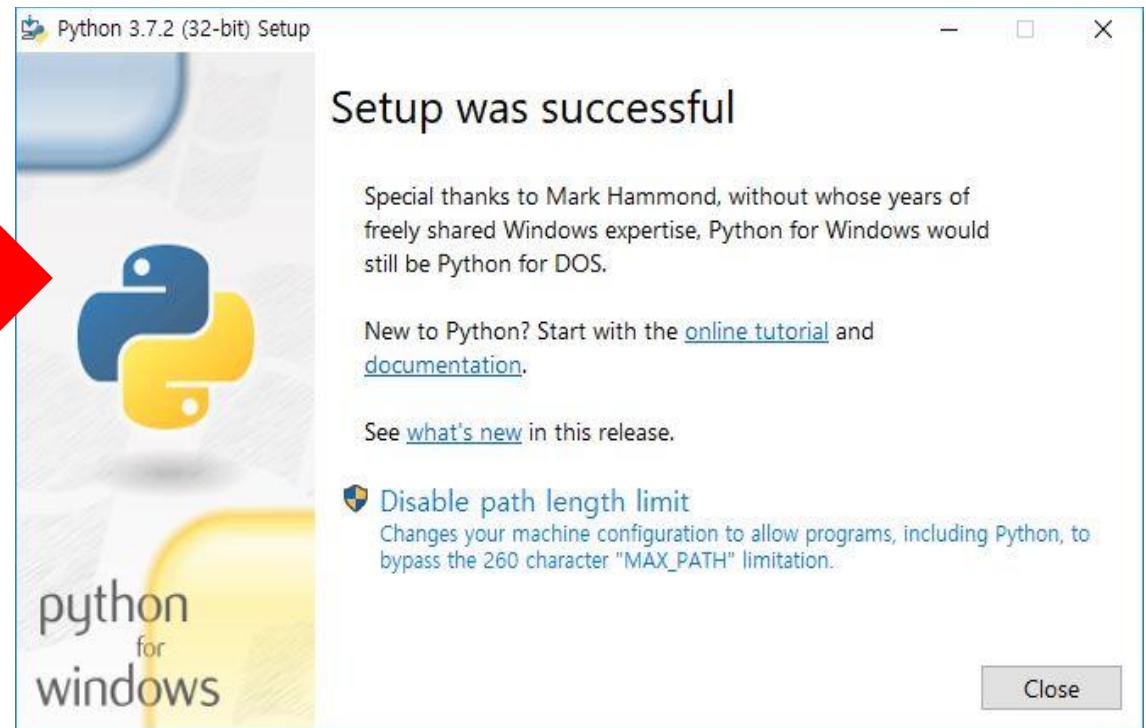
Python 3.7을 경로  
에 추가 함



- "Install Now"를 눌러 설치를 시작

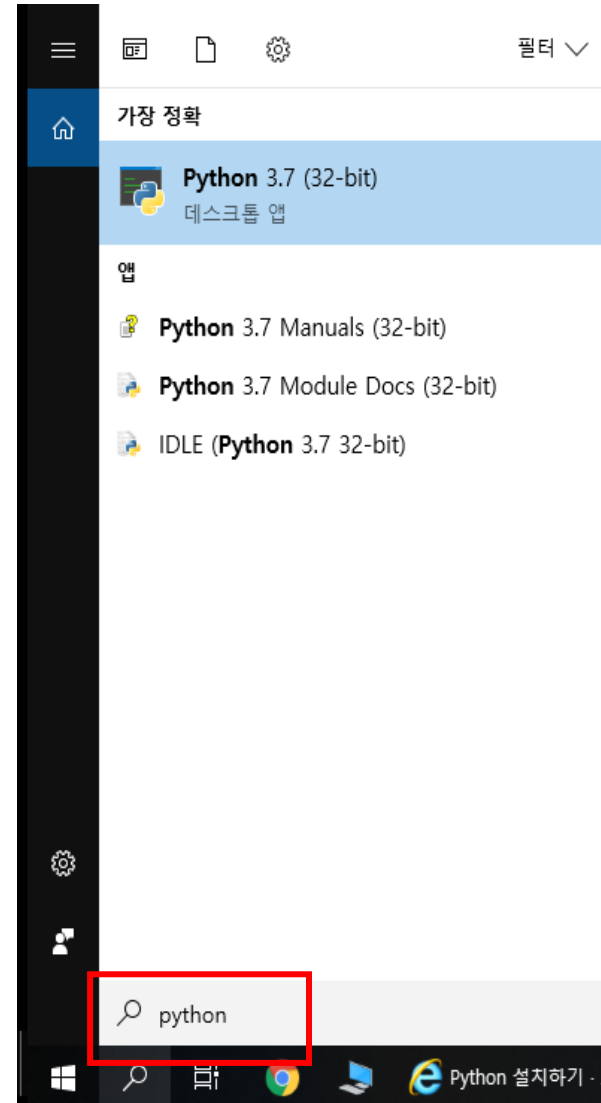


설치 완료 화면



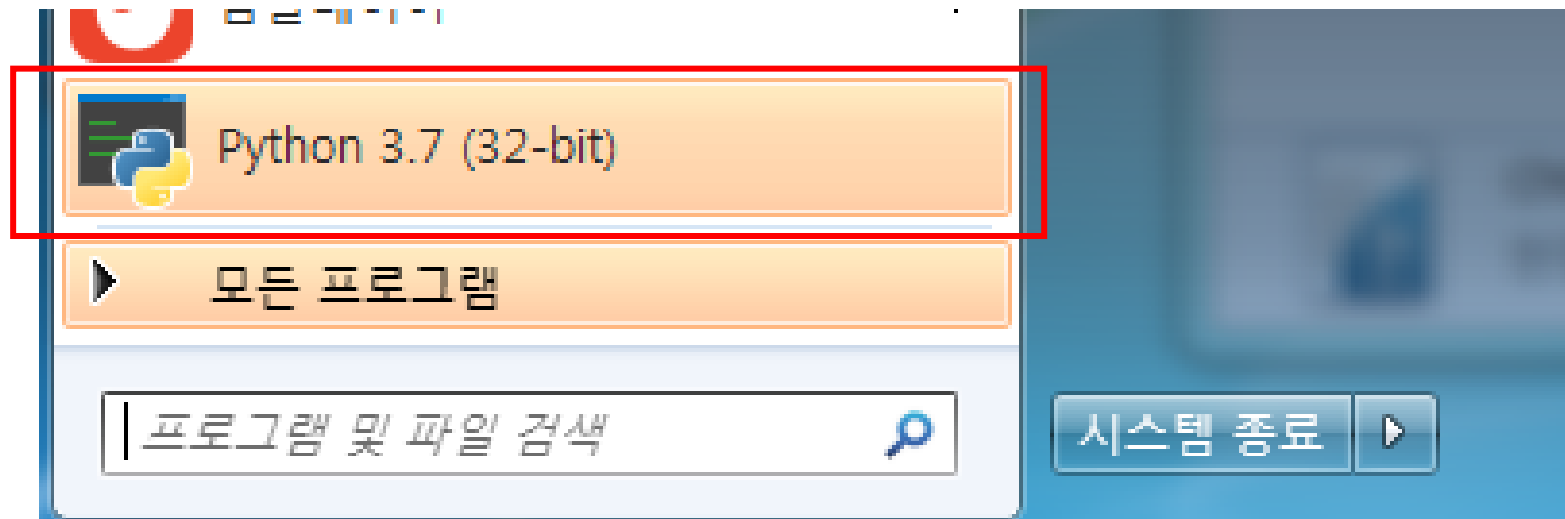
# 파이썬 대화창 사용하기

- 시작 버튼을 눌러 "python"을 검색
- Python 3.7을 눌러서 실행



# 윈도우 컴퓨터에서 파이썬 대화창 열기

- 시작-"Python 3.7 (32-bits)"을 선택



- 사용자의 입력을 받을 수 있는 프롬프트prompt에 파이썬 명령어를 준다.(커서가 깜박일 것이다)
- 프롬프트에 `print('Hello Python!!')` 을 입력 후 엔터키를 누르자
- 프롬프트 아래에 Hello Python!!이 출력된다

The image shows a screenshot of a Windows command prompt window titled "Python 3.7 (32-bit)". The window contains the following text:

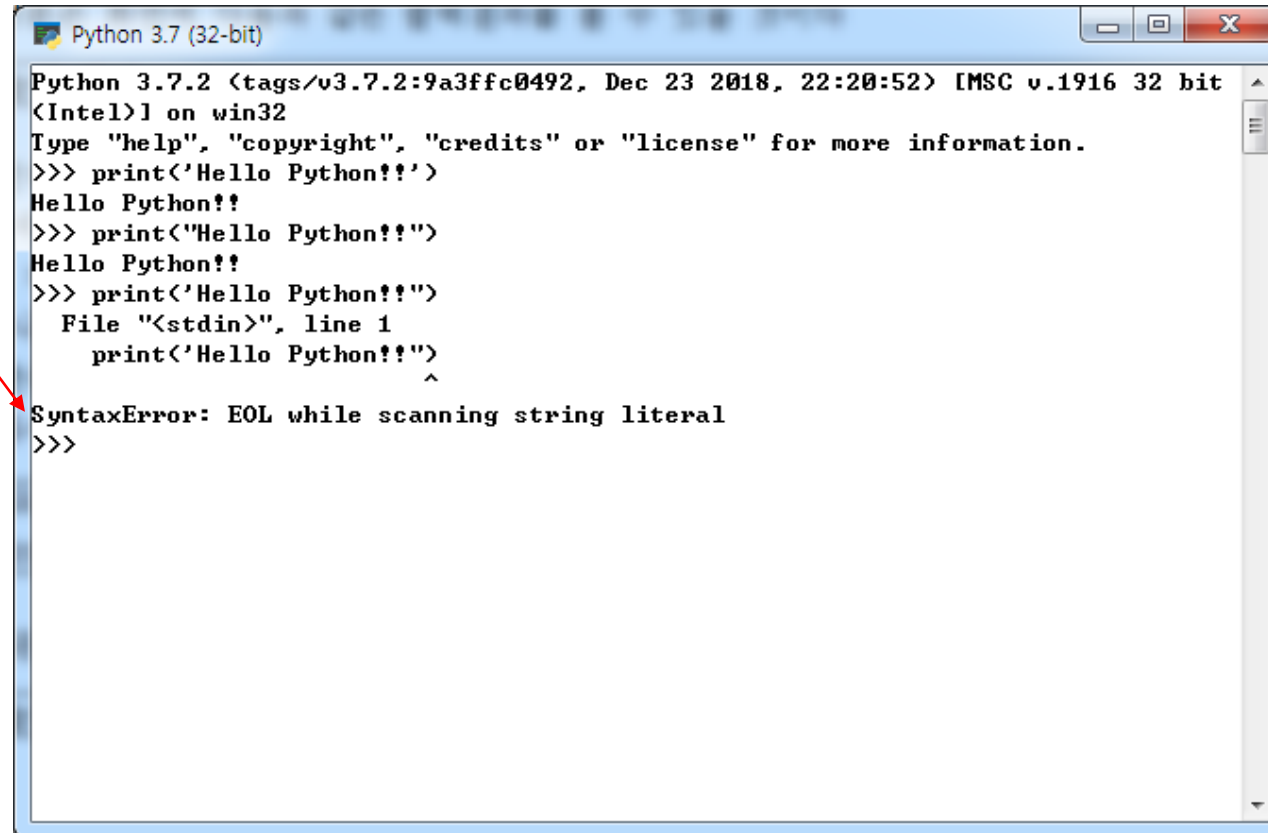
```
Python 3.7.2 (tags/v3.7.2:9a3fffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit  
<Intel>] on win32  
Type "help", "copyright", "credits" or "license" for more information.  
>>> print('Hello Python!!')  
Hello Python!!  
>>> _
```

Annotations with red arrows point to specific elements:

- A box labeled "파이썬 대화창" (Python dialog box) points to the window title bar.
- A box labeled ">>> 를 프롬프트라 한다" (Call >>> the prompt) points to the first prompt line.
- A box labeled "엔터키" (Enter key) points to the end of the command line.
- A box labeled "커서" (Cursor) points to the underscore character at the end of the last line.

- 오류가 날 경우 오류가 발생한 문장과 함께 오류의 내용을 출력
  - `print('Hello Python!!')` 와 같이 작은 따옴표로 시작해서 큰 따옴표로 끝나는 경우에는 파이썬이 잘못된 명령으로 인식해서 오류를 발생
  - 들여쓰기와 대소문자에도 민감한 특징이 있음
  - 대화창을 종료하고 싶으면 키보드의 Control 키와 'Z'키를 입력하거나(^Z 로 표기함) 또는 `exit()` 함수를 호출

오류 발생



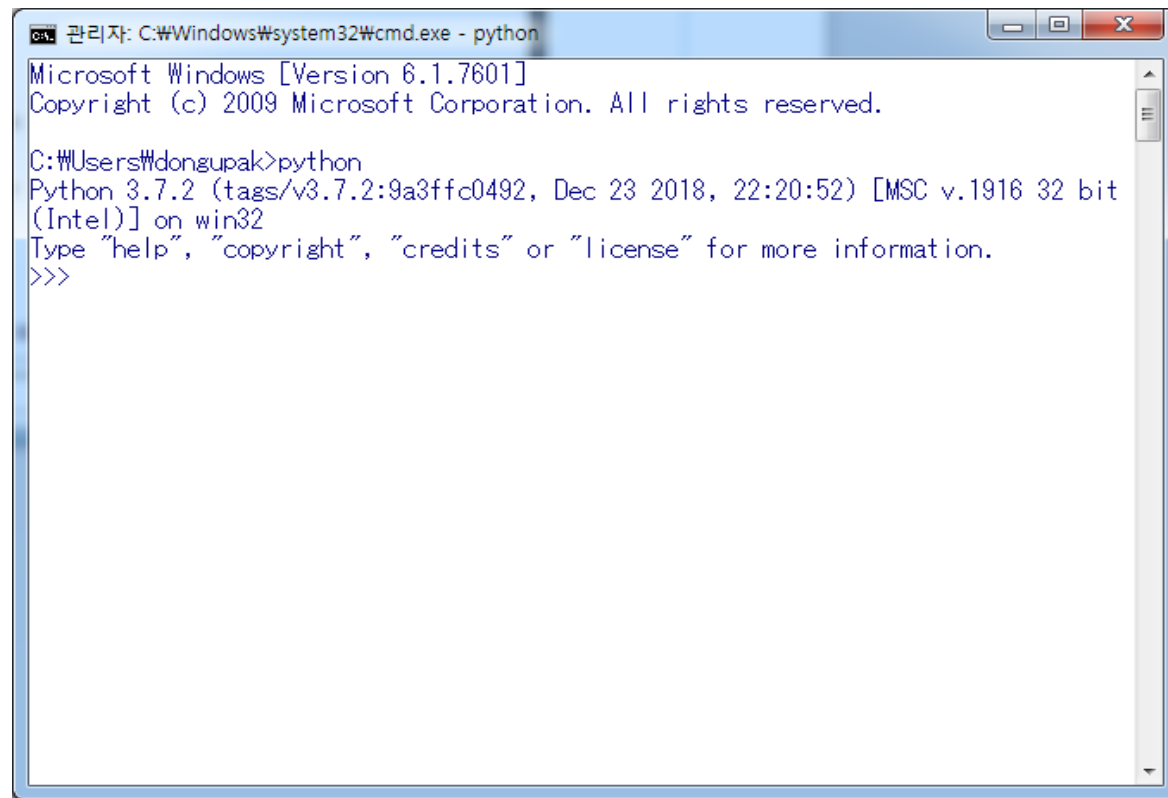
```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
<Intel>] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print('Hello Python!!')
Hello Python!!
>>> print("Hello Python!!")
Hello Python!!
>>> print('Hello Python!!')
File "<stdin>", line 1
    print('Hello Python!!')
    ^
SyntaxError: EOL while scanning string literal
>>>
```

# macOS와 윈도우에서 대화형 모드로 파이썬 실행하기

A screenshot of a macOS terminal window titled "student-lock — python3 — 80x24". The terminal shows the execution of the 'python' command, which starts the Python 3.6.1 interpreter. It displays the version, the provider (Continuum Analytics, Inc.), and the compiler (GCC 4.2.1). It then prompts for help, and the user enters 'print("Hello Python")', which outputs 'Hello Python'. The prompt '>>>' is visible at the bottom.

```
student-lock — python3 — 80x24
dongupak-MacBook:student-lock dongupak$ python
Python 3.6.1 |Continuum Analytics, Inc.| (default, May 11 2017, 13:04:09)
[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello Python")
Hello Python
>>>
```

macOS의 터미널과 파이썬 대화형 모드

A screenshot of a Windows command prompt window titled "관리자: C:\Windows\system32\cmd.exe - python". The window shows the execution of the 'python' command, which starts the Python 3.7.2 interpreter. It displays the version, the build number, and the architecture (MSC v.1916 32 bit). It then prompts for help, and the user enters 'print("Hello Python")', which outputs 'Hello Python'. The prompt '>>>' is visible at the bottom.

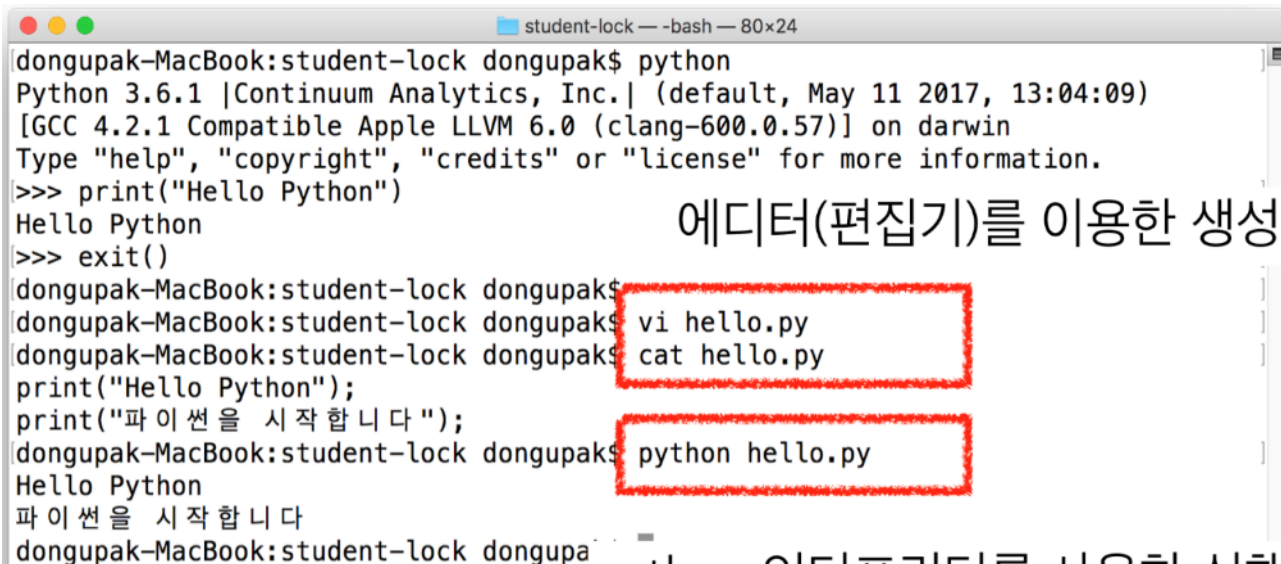
```
관리자: C:\Windows\system32\cmd.exe - python
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\dongupak>python
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

윈도우 운영체제의 명령처리기와 파이썬 대화형 모드

# macOS에서 편집기를 이용하여 파이썬 프로그램 작성하기

- 대화형 모드 : 명령 입력이 편리하나 명령문이 저장되지 않고 재사용이 불가능
  - > 텍스트 파일을 따로 저장시켜 이 파일을 실행



A terminal window titled 'student-lock — -bash — 80x24' showing the following commands and output:

```
dongupak-MacBook:student-lock dongupak$ python
Python 3.6.1 |Continuum Analytics, Inc.| (default, May 11 2017, 13:04:09)
[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello Python")
Hello Python
>>> exit()
dongupak-MacBook:student-lock dongupak$ vi hello.py
dongupak-MacBook:student-lock dongupak$ cat hello.py
print("Hello Python");
print("파이썬을 시작합니다");
dongupak-MacBook:student-lock dongupak$ python hello.py
Hello Python
파이썬을 시작합니다
dongupak-MacBook:student-lock dongupak$
```

에디터(편집기)를 이용한 생성

python 인터프리터를 사용한 실행




- 검색창에 cmd를 입력하여 Windows 명령 처리기를 열어서 파이썬 대화형 모드로 들어가는 것이 가능

자세한 결과 보기

시스템 종료 ▶

프로그램 (1)

 cmd



```
C:\Windows\system32\cmd.exe - python
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\dongupak>python
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
>>> type "help", "copyright", "credits" or "license" for more information.
>>>
```

코드 : hello.py 프로그램의 코드

hello.py (여러분이 idle에서 작성할 파일의 이름입니다)

```
print('Hello Python!')
```

실행결과

Hello Python!

- 편집 프로그램을 사용하여 hello.py 파일을 만들어 두면 파이썬 명령문 재사용이 가능
- 프로그램을 실행시키는 방법은 macOS나 리눅스의 터미널에서 다음과 같은 명령을 입력

```
$ python hello.py
```

Hello Python!

## 1.4.4 윈도우 컴퓨터에서 파이썬 프로그램하기

- C: 드라이브 아래에 workspace라는 이름의 폴더 생성
- 이 폴더 아래에 hello.py라는 프로그램을 만들기
- Windows 명령 처리기에서 다음 명령을 순차적으로 입력

```
C:\Users\me> c:
```

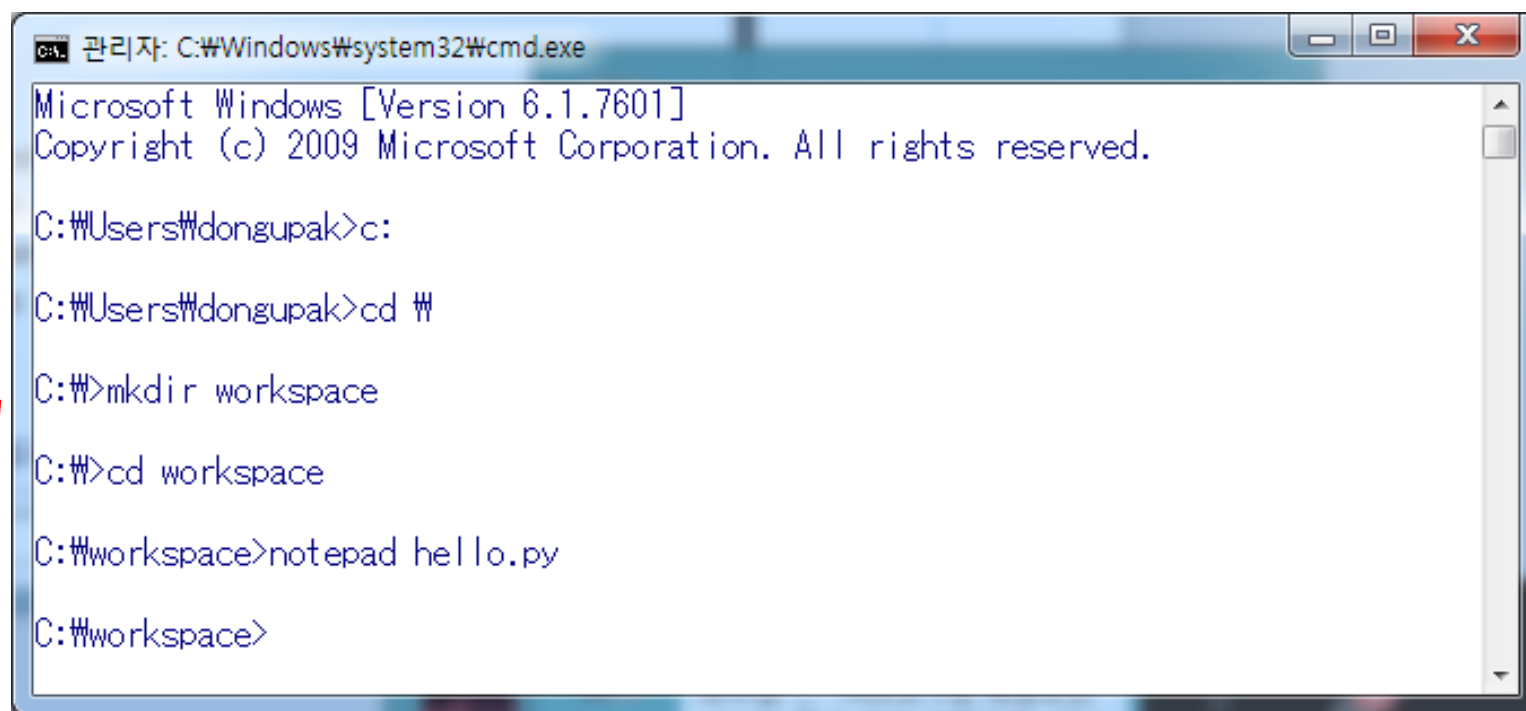
```
C:\Users\me> cd \ (주의 : 역슬래시 기호 \는 한글 윈도우에서 #로 나타남)
```

```
C:\> mkdir workspace
```

```
C:\> cd workspace
```

```
C:\workspace >
```

- 메모장을 열어서 hello.py 프로그램을 작성해 보기
- notepad hello.py를 입력

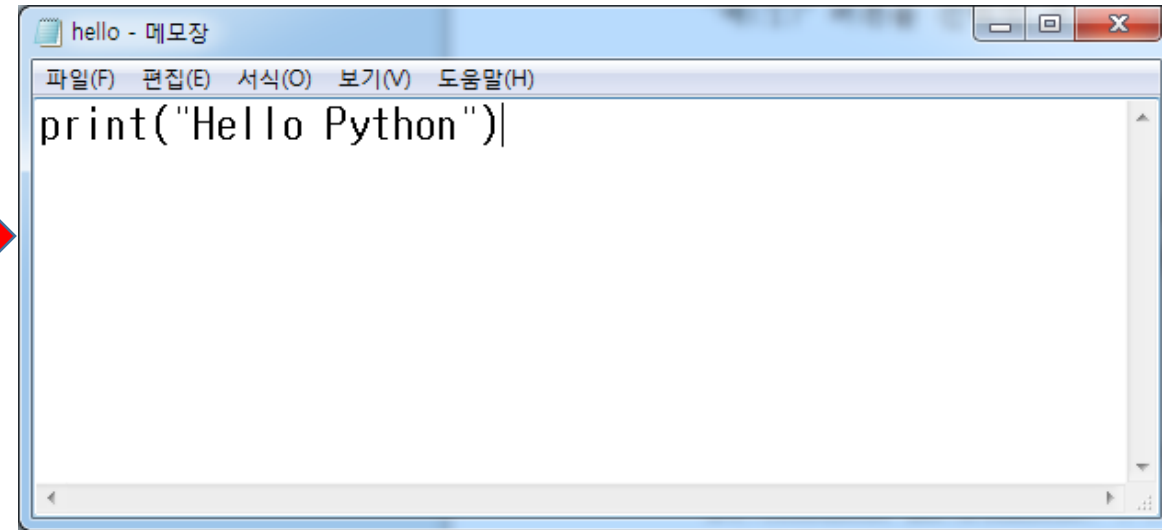
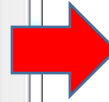
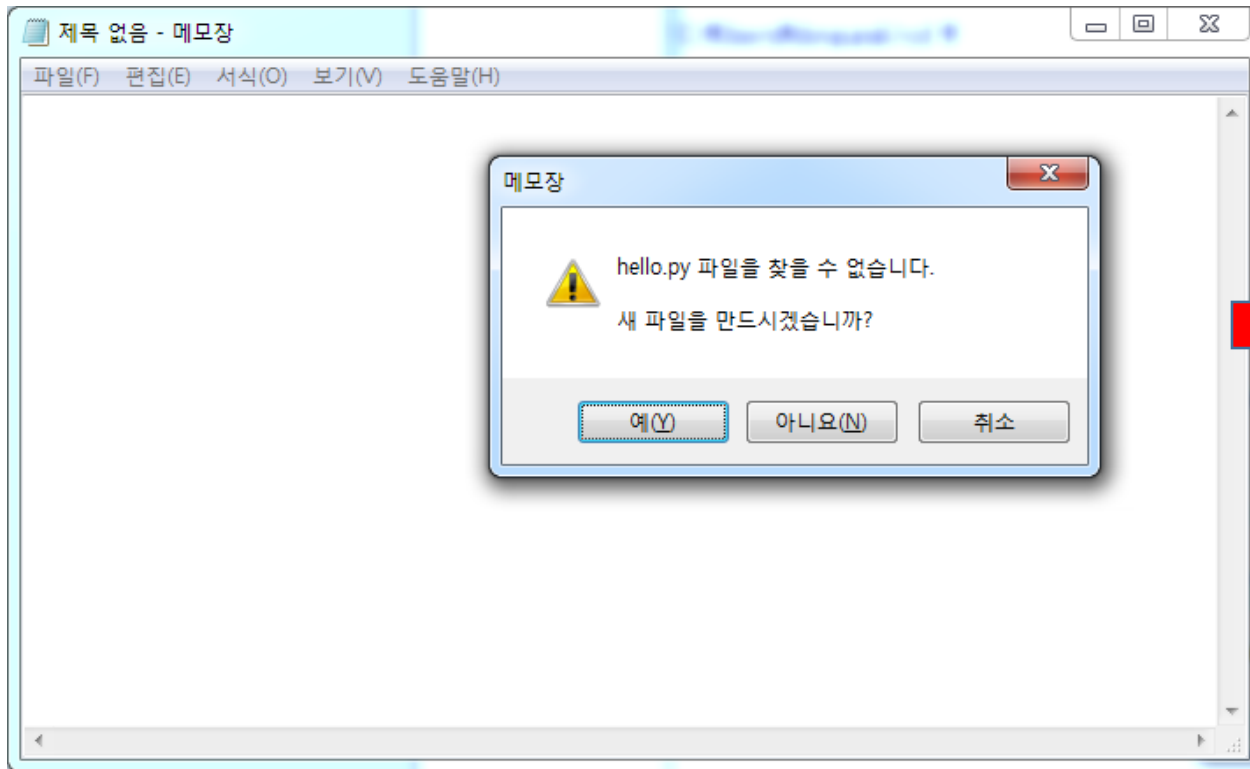


```
관리자: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

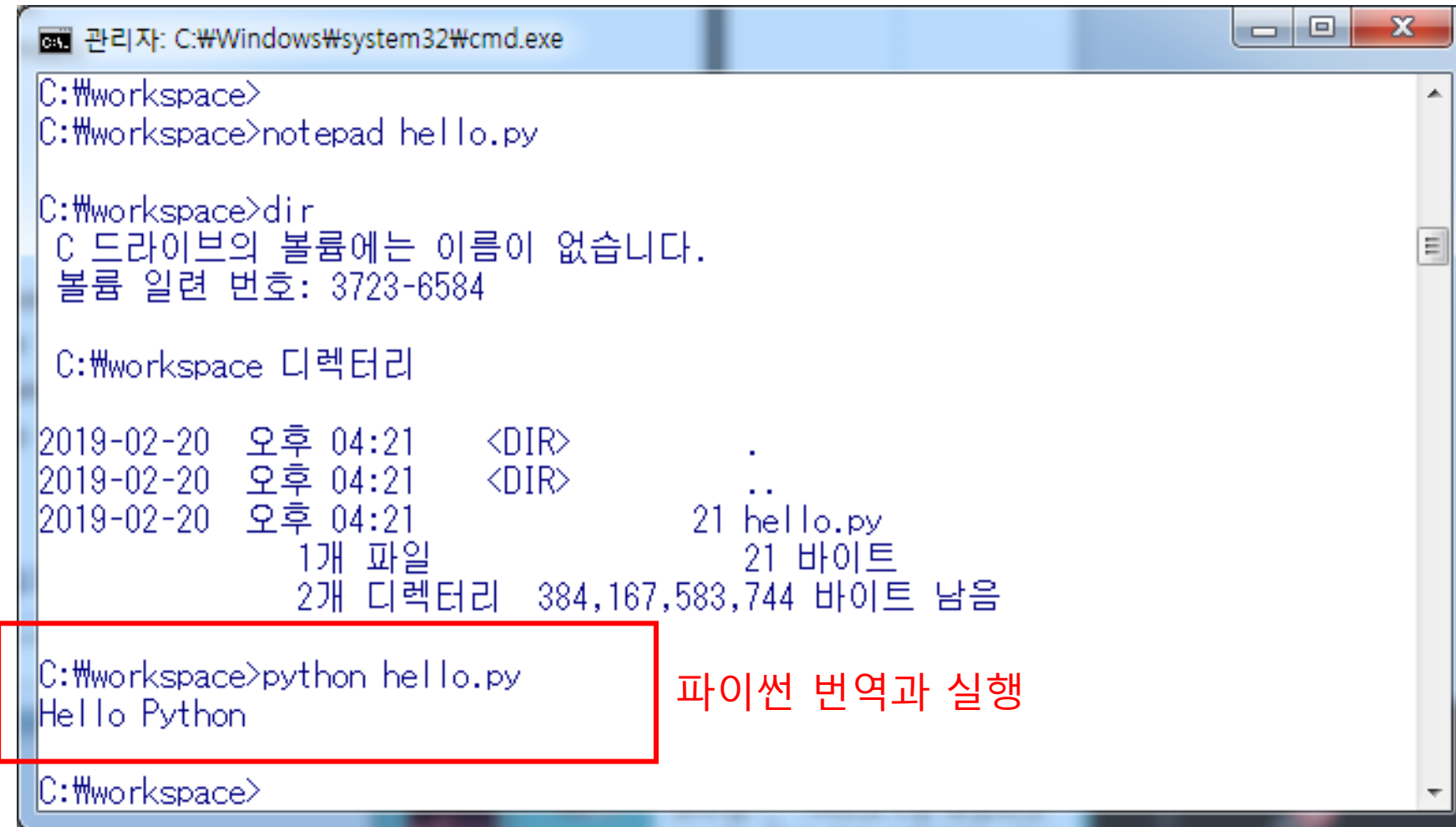
C:\Users\dongupak>c:
C:\Users\dongupak>cd \
C:\>mkdir workspace
C:\>cd workspace
C:\workspace>notepad hello.py
C:\workspace>
```

cd, mkdir 등은 명령행에서 입력할 수 있는 명령입니다.

- 메모장의 "hello.py" 파일을 찾을 수 없습니다. 새 파일을 만드시겠습니까?" 대화창의 "예(Y)" 버튼을 선택
- 메모장을 열어 다음과 같은 hello.py 작성



- dir 명령으로 현재 폴더에 hello.py 파일이 있는지 확인
- 인터프리터 실행을 위해 `python hello.py`를 입력



The screenshot shows a Windows Command Prompt window with the title bar '관리자: C:\Windows\system32\cmd.exe'. The command history is as follows:

```
C:\workspace>
C:\workspace>notepad hello.py
C:\workspace>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 3723-6584

C:\workspace 디렉터리

2019-02-20   오후 04:21   <DIR>          .
2019-02-20   오후 04:21   <DIR>          ..
2019-02-20   오후 04:21                21 hello.py
                        1개 파일                21 바이트
                        2개 디렉터리 384,167,583,744 바이트 남음

C:\workspace>python hello.py
Hello Python

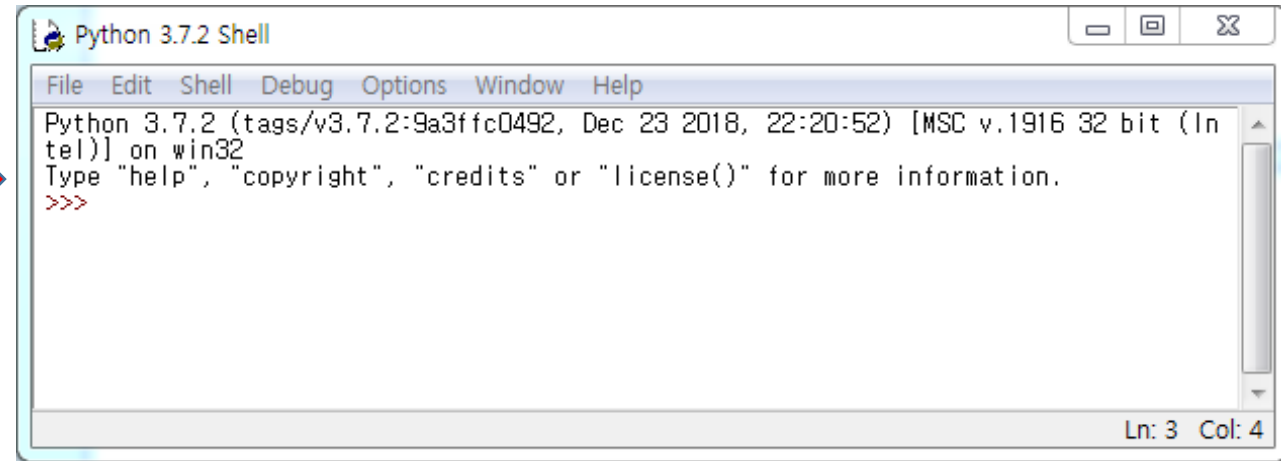
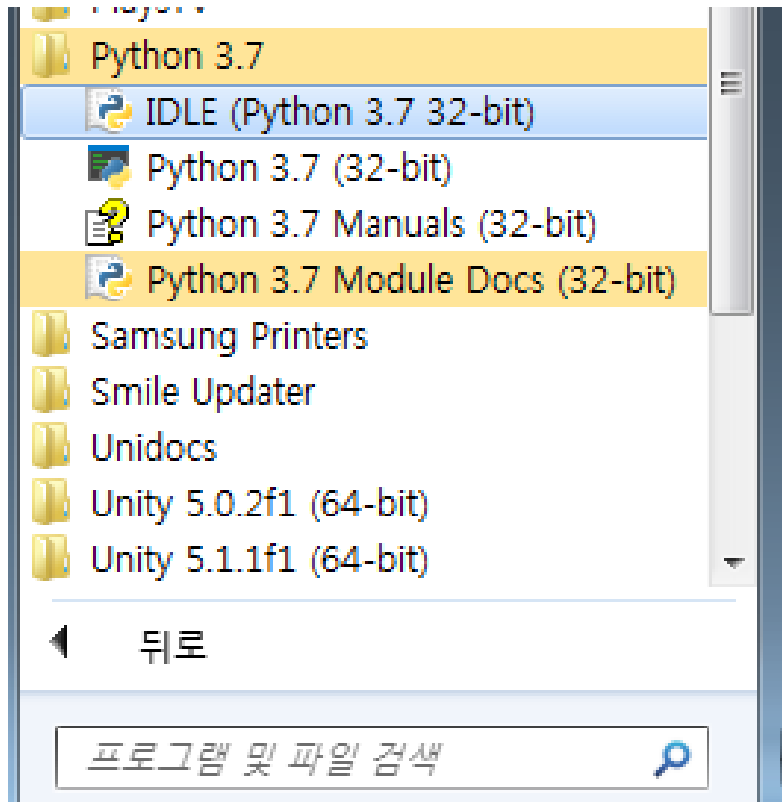
C:\workspace>
```

The command `python hello.py` and its output `Hello Python` are enclosed in a red rectangular box. To the right of this box, the text '파이썬 번역과 실행' (Python translation and execution) is written in red.

# 윈도우의 통합 개발환경(IDE) 사용하기

- IDLE란?
  - Integrated Development and Learning Environment의 약자
  - “통합 개발과 학습 환경”을 뜻함
- 파이썬 프로그램의 편집과 실행을 대화창에서 하는 것보다 쉽게 하도록 하기 위해 IDLE를 사용

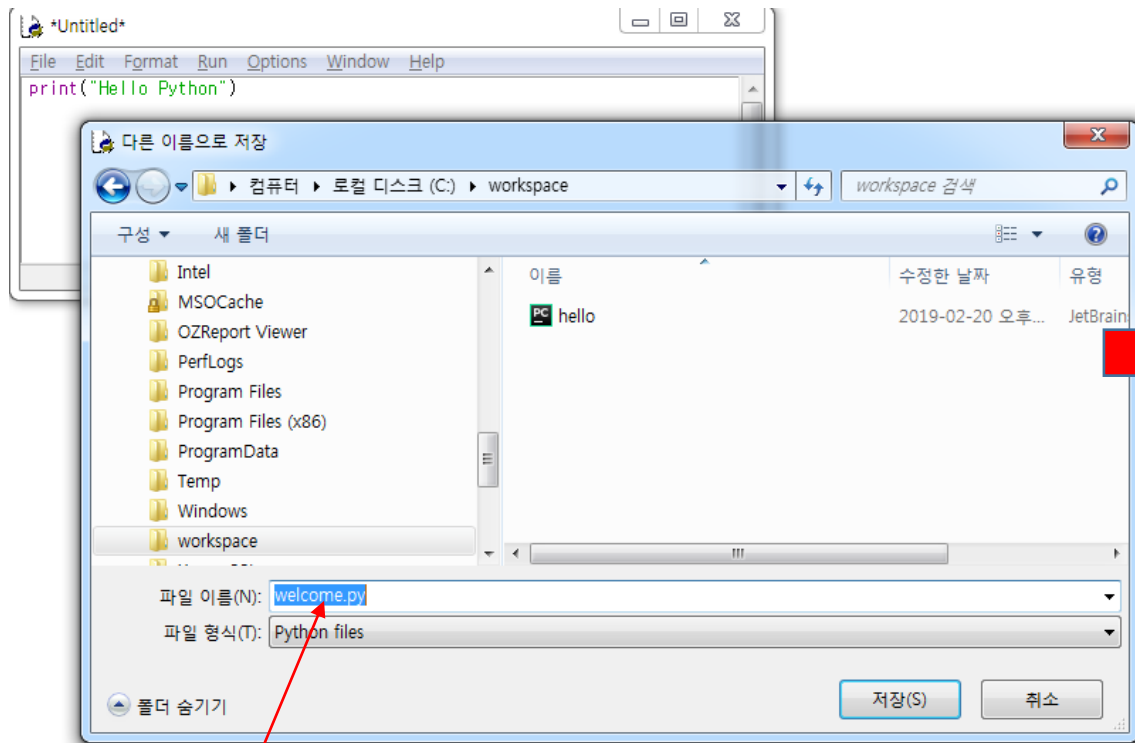
- 시작메뉴의 “모든 프로그램”에서 Python 3.7 검색 후 제일 상단의 IDLE (Python 3.7 32-bit)를 선택
- 이와 같은 파이썬 대화창을 **파이썬 셸** **python shell** 혹은 **파이썬 대화식 셸** **python interactive shell**이라고 한다.



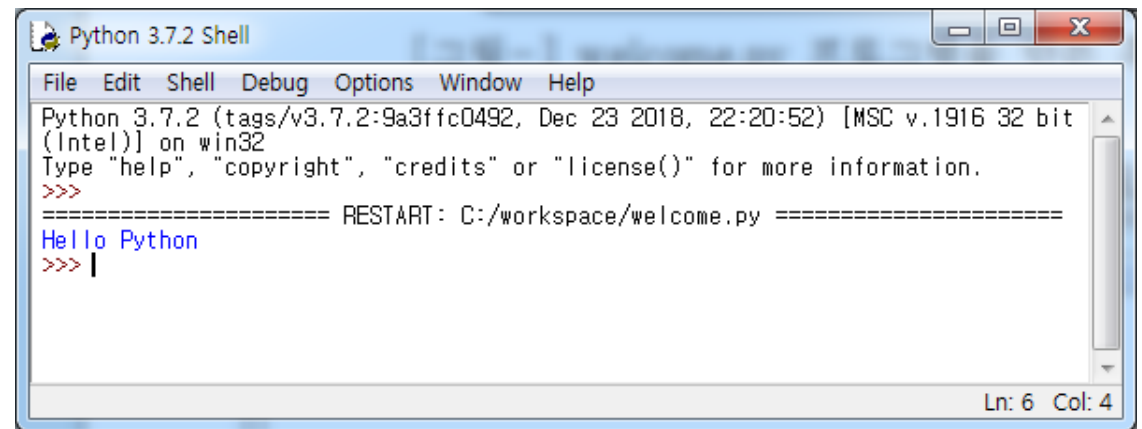


# "welcome.py"라는 이름의 파일 만들기

- 새로운 파일 만든 후 Run 메뉴의 "Run Module" 메뉴 실행 또는 단축 키 F5 입력



hello.py 가 있으므로 welcome.py라고 파일 이름을 짓고



실행 결과

Run 메뉴의 Run Module 실행

# 파이썬 2, 3의 차이

- print에서 print() 함수로 변화
- 표준 라이브러리의 재배치
- 스트링 포매팅 기능이 다양해짐
- 딕셔너리, 문자형에 구성요소 변경
- 함수 표기법의 변화
- 유니코드 지원등 많은 변화들
- 파이썬 2는 2020년 이후 지원되지 않음
- 새롭게 파이썬을 배운다면 파이썬 3으로 시작하기를 권함

# PyCharm 설치 및 실행하기

- <https://www.jetbrains.com/ko-kr/pycharm/download/#section=windows>
- Community Edition 다운로드


20  
years

도구 솔루션 언어 회사 지원 스토어

PyCharm

새로운 기능 기능 구매

다운로드



버전: 2020.2.1  
빌드: 202.6948.78  
2020년 8월 26일

시스템 요구 사항

설치 안내

기타 버전

## 다운로드 PyCharm

Windows Mac Linux

### Professional

과학 및 웹 Python 개발용. HTML, JS, SQL 지원.

다운로드


무료 평가판

### Community

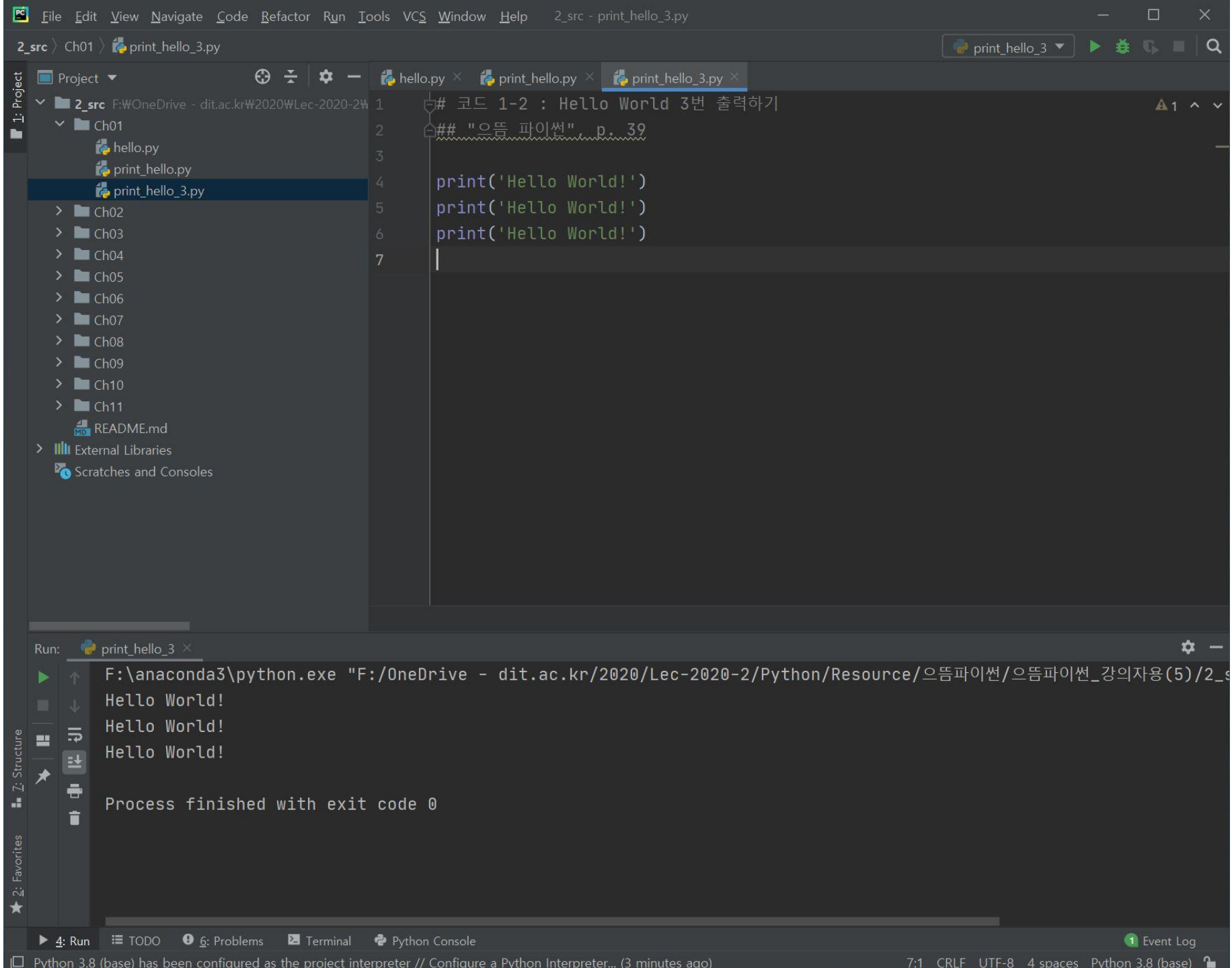
순수 Python 개발용

다운로드

무료, 오픈 소스

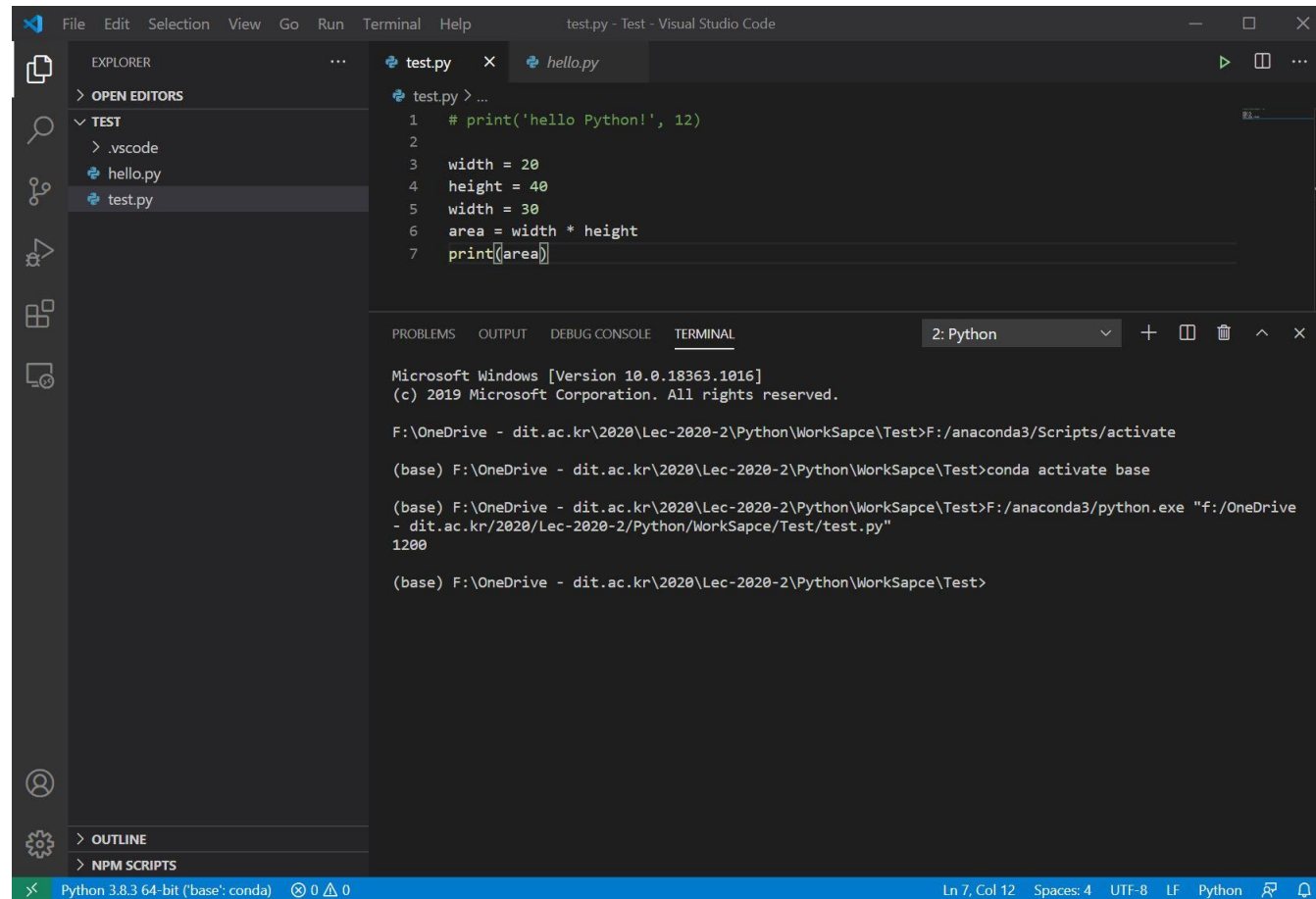


Toolbox App을 설치하여 PyCharm 및 향후 업데이트를 간편하게 다운로드하세요



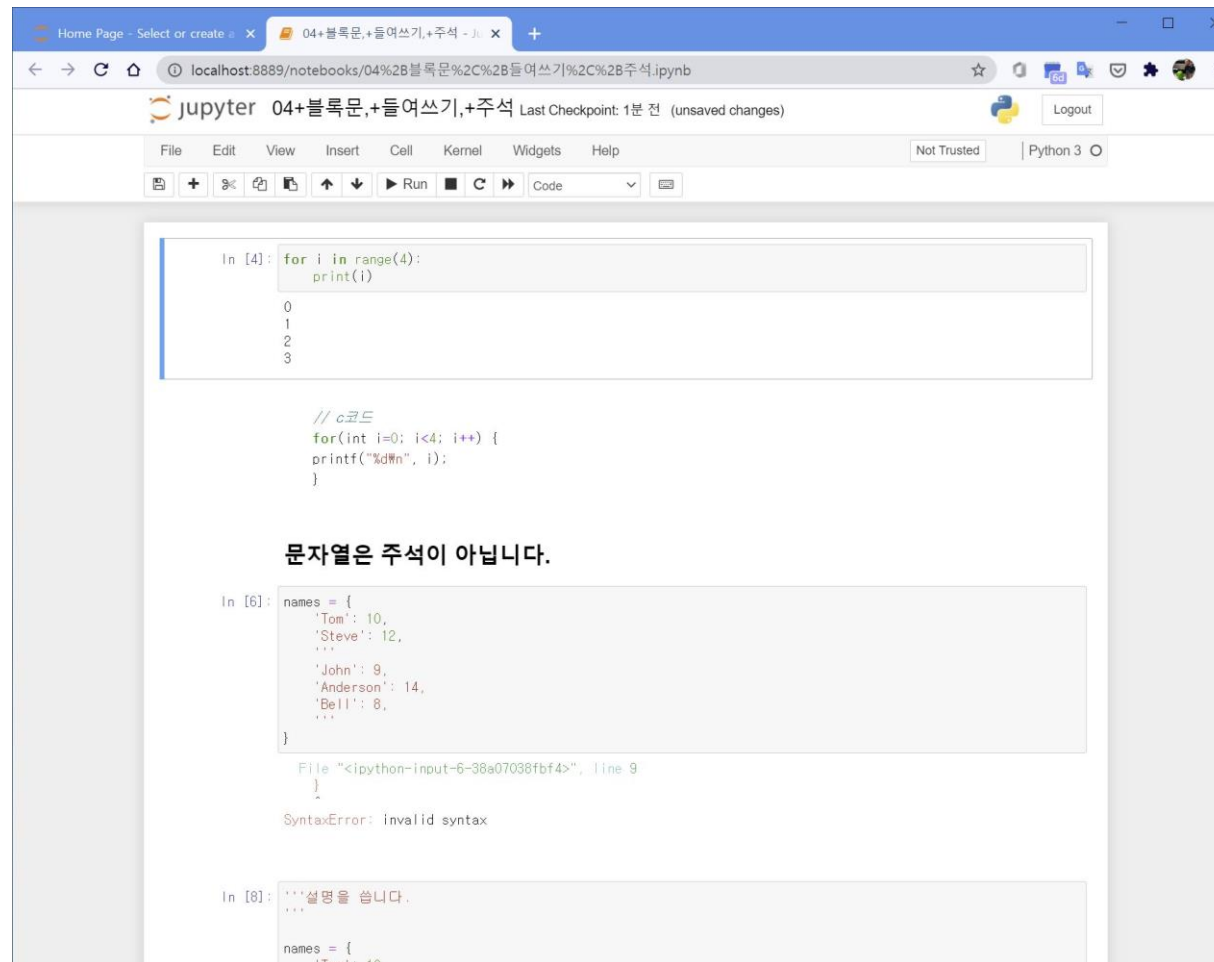
# Visual Studio Code

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



# Jupyter Notebook

- Anaconda3에 포함

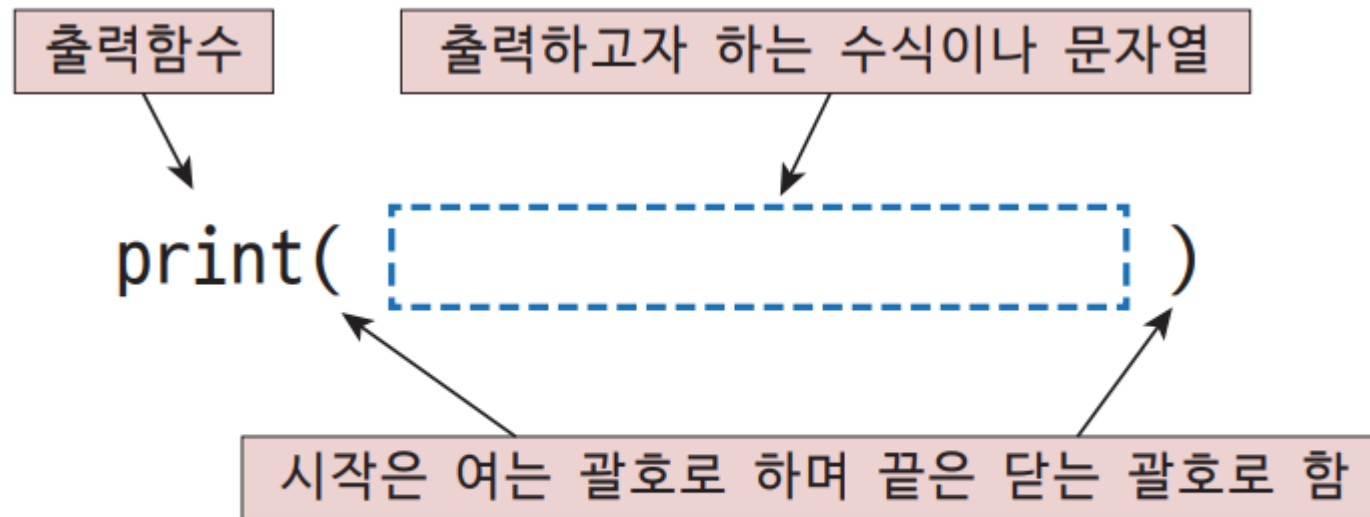


[실습]

1. Pycharm을 설치
2. 예제들을 코딩하고 실행하기

# Hello World 출력하기

- 문자열이나 수치값을 화면상에 출력하는 중요한 일을 하는 함수인 `print()`에 대해 알아봅시다



[그림 1-40] 파이썬의 출력함수인 `print()` 함수의 문법



- `print('Hello World')`와 같은 코드를 **표현문 expression statement** 혹은 **표현식**이라고 한다.
- 표현식은 간단하게 **문장statement**이라고 부르기도 한다.

코드 1-1 : Hello World 출력하기

`print_hello.py` (이 부분은 여러분이 만들 파일의 이름입니다)

```
print('Hello World!')
```

실행결과

Hello World!

# print\_hello\_3.py

- Hello World!를 세 줄에 걸쳐 반복시키기

코드 1-3 : Hello World 3번 출력하기

print\_hello\_3.py

```
print('Hello World!')  
print('Hello World!')  
print('Hello World!')
```

실행결과

```
Hello World!  
Hello World!  
Hello World!
```

# 문자열과 숫자 출력하기

대화창 실습 : 대화창에서 문자열과 숫자 출력하기

```
>>> print('Hello World!')
```

Hello World!

```
>>> print('Hello', 'World!')
```

Hello World!

```
>>> print('Hello'+'World!')
```

HelloWorld!

```
>>> print(100)
```

100

```
>>> print(2+4)
```

6

쉼표로 구분되어 있을 경우 공백을 삽입하고  
다음 문자열을 출력

덧셈기호로 문자열을 연결할 경우  
공백이 삽입되지 않음

2 + 4 연산의 결과가 출력됨

# 문자열과 숫자 출력하기

대화창 실습 : 대화창에서 간단한 숫자 출력하기와 연산

```
>>> 100
```

```
100
```

```
>>> 10 + 20
```

```
30
```

print() 함수를 사용하지 않을 경우에도 연산의 결과가 출력됨, 대화창을 간단한 계산기로 사용할 수 있음

```
>>> 10 - 20
```

```
-10
```

대화창에 입력한 10 + 20의 결과 30이 아래 줄에 출력됨

```
>>> 10 * 20
```

```
200
```

```
>>> 10 / 20
```

```
0.5
```

```
>>> 123 * 8765
```

```
1078095
```

```
>>> 10 + 20 * 30
```

```
610
```

# 변수 $x, y$ 를 사용하여 연산을 할 수 있다

대화창 실습 : 대화창에서 변수 선언하기와 연산하기

```
>>> x = 100
```

```
>>> y = 200
```

```
>>> x + y
```

```
300
```

```
>>> x - y
```

```
-100
```

```
>>> x * y
```

```
20000
```

```
>>> x / y
```

```
0.5
```

```
>>> x ** 2
```

```
10000
```

```
>>> n = 5
```

```
>>> n ** 2
```

```
25
```

변수는 특정한 값을 저장하는 기능을 한다

$x + y$  연산은  $x$  변수,  $y$  변수가 저장해둔 값을 불러와서 이 값 100, 200을 연산한다

새로운 변수의 생성과 값의 저장

$**$  는 제곱 연산을 수행하여 5의 2 제곱을 수행

# 계산기 프로그램 작성

- 결과 값은?

```
1: a = 100
2: b = 50
3: result = a + b
4: print(a, "+", b, "=", result)
5: result = a - b
6: print(a, "-", b, "=", result)
7: result = a * b
8: print(a, "*", b, "=", result)
9: result = a / b
10: print(a, "/", b, "=", result)
```

# 오류를 수정하시오.

- 오류가 발생하는 이유는?

```
1: a = input()
2: b = input()
3: result = a + b
4: print(a, "+", b, "=", result)
5: result = a - b
6: print(a, "-", b, "=", result)
7: result = a * b
8: print(a, "*", b, "=", result)
9: result = a / b
10: print(a, "/", b, "=", result)
```

# 도전 문제

- 위 계산기 프로그램은 한번만 실행이 된다. 이것을 루프(loop)를 사용하여 반복적으로 실행할 수 있도록 만드시오.
- 이때 a의 값이 9999가 입력되면 프로그램의 실행이 끝난다.

```
a 값을 입력하시오 :
2
b 값을 입력하시오 :
3
2 + 3 = 5
2 - 3 = -1
2 * 3 = 6
2 / 3 = 0.6666666666666666
*****
a 값을 입력하시오 :
1
b 값을 입력하시오 :
4
1 + 4 = 5
1 - 4 = -3
1 * 4 = 4
1 / 4 = 0.25
*****
a 값을 입력하시오 :
9999
program end!!!
```

Process finished with exit code 0