

1장. 파이썬 개발 환경 준비

파이썬



Contents

- ❖ 목차
 - 1. 파이썬이란
 - 2. 파이썬 설치하기
 - 3. 텍스트 에디터 설치하기
 - 4. 파일 작성 및 실행

0. 학습목표

- ❖ 파이썬에 대하여 설명할 수 있다.
- ❖ 파이썬으로 할 수 있는 일과 할 수 없는 일에 대하여 설명할 수 있다.
- ❖ 파이썬 개발 환경을 설정할 수 있다.

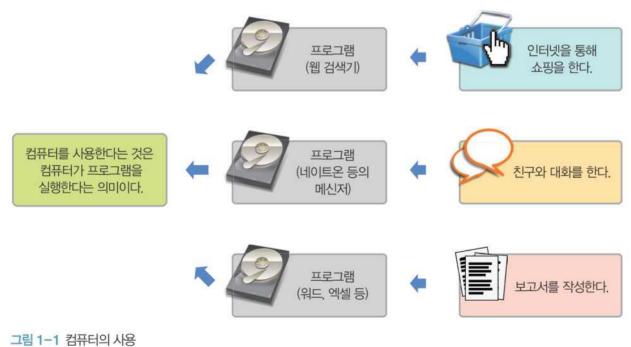


컴퓨터는 무엇을 실행하는가

컴퓨터의 용도

-01

- 다른 사람과 대화를 합니다
- 인터넷 쇼핑을 합니다
- 문서를 작성합니다



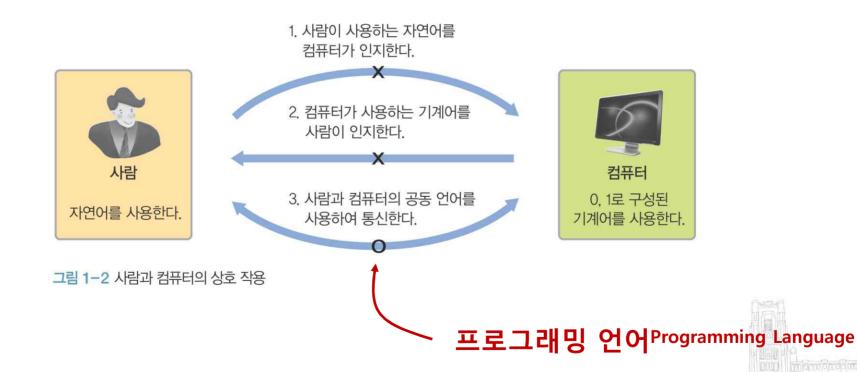


프로그램은 무엇으로 만드는가

■ 컴퓨터와 사람의 상호 작용

•

- 컴퓨터는 아직까지 2진 법만 기억
- 사람은 자연어를 사용. 컴퓨터와 사람은 3가지 상호 작용 형태가 가능



프로그램은 무엇으로 만드는가

■ 프로그래밍 언어란?

0

- 문제를 해결하기 위한 알고리즘을 컴퓨터로 하여금 실행시키도록 하는 형식 언어
- 프로그래밍한 결과물은 프로그램 또는 소프트웨어

```
content: ":
                                                   blockquote p { margin-bottom: 10px }
   content: none:
                                                   strong, b [ font-weight: bold ]
                                                   em, i, cite (
table (
                                                      font-style: normal;
   border-collapse: collapse;
                                                      font-family: arial:
  border-spacing: 0;
                                                   small { font-size: 100% }
button, input, select, textarea [ margin: 0 ]
                                                   figure ( margin: 10px 0 )
:focus { outline: 0 }
                                                   code, pre (
a:link ( -webkit-tap-highlight-color: #FF5E99 )
                                                      font-family: monospace,consolas,sans-serif;
img, video, object, embed (
                                                      font-weight: normal:
  max-width: 100%:
                                                      font-style: normal;
  height: auto!important;
iframe [ max-width: 100% ]
                                                      margin: 5px 0 20px 0;
blockquote (
                                                      line-height: 1.3em;
  font-style: italic:
                                                      padding: 8px 10px;
  font-weight: normal;
                                                      overflow: auto:
   font-family: Georgia, Serif;
   font-size: 15px;
   padding: 0 10px 20px 27px;
                                                            ng: 0 8px;
  position: relative:저급언어
                                                            pight: 1.5;고급언어
   margin-top 25au
blockquote:af
                                              </>
                기계어, 어셈블리어
                                                                C++, Java, Python
   position: a
   content:
                                                                           인터프리터
```

프로그램은 무엇으로 만드는가

■ 프로그래밍 언어의 발전 단계

•

■ 1950년대에 컴퓨터가 개발된 이후 프로그래밍 언어는 느리지만 꾸준히 발전



그림 1-3 프로그래밍 언어의 발전 단계에 따른 세대별 구분



•

■ 프로그램을 작성하기 위해서는 설계과정이 필요

 설계과정이 생략되면 오류가 발생하게 되고, 이러한 오류를 고치는 데 드는 비용은 처음에 설계를 좀 더 명확하게 하는 데 드는 비용보다 훨씬 더 많이 소요







■ 프로그램 작성 단계(작은 규모)

1단계: 입출력 정의



2단계: 절차 정의

- ① 문제를 명확하게 정의한다.
- ② 문제를 작게 분해한다
- ③ 분해된 각각의 작은 문제를 순서, 선택, 반복의 논리로 절차를 정의하고 연결한다.

절차는 알고리즘이나 의사 코드, 순서도로 작성한다.



3단계: 프로그램 코드 작성

그림 1-5 프로그램 작성 단계



■ 1단계 : 입출력을 정의합니다.

•

■ 입출력이 없는 프로그램은 의미가 없는 프로그램

[문제] 1부터 100까지의 합을 구하는 프로그램

1부터 100까지의 합을 구하는 프로그램은 입력이 존재하지 않습니다. 문제에서 1과 100이 지정되었기 때문입니다. 출력은 1부터 100까지 더한 결과입니다.

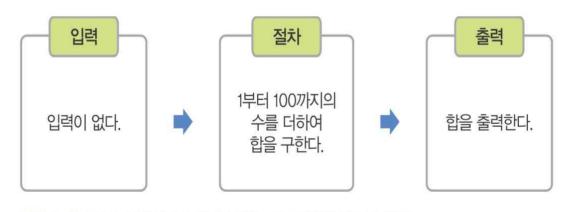


그림 1-6 1부터 100까지의 합을 구하는 프로그램의 입출력 정의

- 2단계 : 절차를 정의합니다.
 - 프로그램 작성의 핵심
 - 단계
 - 1) 문제를 명확하게 정의합니다.
 - 2) 문제를 작게 분해합니다.
 - 3) 분해된 각각의 작은 문제-를 순서, 선택, 반복의 논리로 절차를 정의 하고 연결합니다.

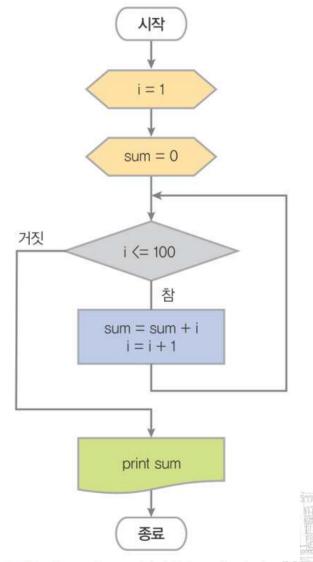


그림 1-7 1부터 100까지의 합을 구하는 순서도

3단계 : 프로그램 코드를 작성합니다.

```
01 #include <stdio.h>
02
03 main()
04 {
05    int i=1;
06   int sum=0;
07    while (i <= 100)
08    {
09         sum=sum+i;
10         i=i+1;
11    }
12    printf("1부터 100까지의 합: %d", sum);
13 }
```

- 0

```
01 TEST
            START
            BALR
                      BASE 0
03
            USING
                      BEGIN+2 BASE
04
                      4,4 --- 4번 레지스터의 내용을 0으로 설정
05
                      2,=F'1' -- 2번 레지스터에 1을 넣음
06
                      3,=F'108' - 3번 레지스터에 100을 넣음
07 LOOP
08
09
10
11
            LTORG
12 BASE
            EOU
                      15
13
            END
```

```
01 10 SUM = 0
02 20 FOR I=1 TO 100 --- I가 1부터 100이 될 때까지 4번 문장까지를 반복
03 30 SUM = SUM + I --- SUM에 I 값을 더함
04 40 NEXT I --- 반복의 범위
05 50 PRINT SUM --- SUM 출력
06 60 END
```

```
01 ISUM=0
02 D0 100 I=1,100,1 --- 1부터 100까지 1씩 증가시키면서 100(4번 문장)번까지 반복
03 ISUM=ISUM+I --- 합계에 더함
04 100 CONTINUE
05 WRITE (6,200) ISUM
06 200 FORMAT(8X,I5)
07 STOP
08 END
```



■ 주요 언어로 작성된 프로그램 : 어셈블리 프로그램

01 TEST	START	0
02 BEGIN	BALR	BASE,0
03	USING	BEGIN+2,BASE
04	SR	4,4 4번 레지스터의 내용을 0 으로 설정
05	L	2,=F'1' 2번 레지스터에 1을 넣음
06	L	3,=F'100' 3번 레지스터에 100을 넣음
07 LOOP	AR	4,3 4번과 3번의 내용을 더함
08	SR	3,2 3 번에서 2 번의 내용을 뺌
09	BCT	3,*-4 7 번 문장으로 분기
10	BR	14
11	LTORG	
12 BASE	EQU	15
13	END	

■ 주요 언어로 작성된 프로그램 : 포트란 프로그램

```
01 ISUM=0
02 DO 100 I=1,100,1 --- 1부터 100까지 1씩 증가시키면서 100(4번 문장)번까지 반복
03 ISUM=ISUM+I --- 합계에 더함
04 100 CONTINUE
05 WRITE (6,200) ISUM
06 200 FORMAT(8X,I5)
07 STOP
08 END
```



■ 주요 언어로 작성된 프로그램 : 코볼 프로그램

```
01 IDENTIFICATION
                        DIVISION.
02 PROGRAM-ID.
                        HAP.
03 ENVIRONMENT
                        DIVISION.
04 CONFIGURATION
                       SECTION.
05 SOURCE-COMPUTER, IBM-PC.
06 OBJECT-COMPUTER IBM-PC.
07 INPUT-OUTPUT
                        SECTION.
08 FILE-CONTROL.
                       SELECT PR-F ASSIGN TO OUTPUT ":HAP".
09 DATA
             DIVISION.
10 FILE
             SECTION.
11 FD PR-F LABEL
                        RECORD OMITTED.
12 01 PR-R PIC
                       X(80).
13 WORKING-STORAGE
                        SECTION.
14 01 LIST.
15 03 N PIC 9(3).
16 03 HAP PIC 9(4).
17 PROCEDURE
                        DIVISION.
18 AA.
                        PERFORM BB UNTIL N = 100.
                                                        BB 블록 반복 수행
19
                        DISPLAY "***1부터 100까지의 합***"
                                                        합계 출력
20
                        DISPLAY "HAP = " HAP.
21
                        STOP RUN.
                                                      --- 프로그램 종료
22 BB.
                        ADD 1 TO N
                                                      --- N 값을 1 증가시킴
23
                        ADD N TO HAP.
                                                      -- N 값을 HAP에 더함
```



주요 언어로 작성된 프로그램 : 베이직 프로그램

```
01 10 SUM = 0
02 20 FOR I=1 TO 100 --- I가 1부터 100이 될 때까지 4번 문장까지를 반복
03 30 SUM = SUM + I --- SUM에 I 값을 더함
04 40 NEXT I --- 반복의 범위
05 50 PRINT SUM --- SUM 출력
06 60 END
```



■ 주요 언어로 작성된 프로그램 : 파스칼 프로그램

```
01 program hap;
02 uses Crt;
03 var
                                                 변수 선언
04
        i,sum : integer;
05 begin
        ClrScr;
06
07
    sum := 0;
                                              --- 변수를 0으로 설정
08
        for i := 1 to 100 do
                                                 1부터 100까지 반복하며 sum에 값을
                                                 더함
09
                   sum := sum + i;
10
        writeln('1부터 100까지의 합은 ===> ',sum); --- 값을 출력
11 end.
```



주요 언어로 작성된 프로그램 : 자바 프로그램

```
01 public class SumTest {
      public static void main(String[] args) {
02
03
         int i, sum=0;
                                              --- 변수를 선언하고 sum을 0으로 설정
04
         for (i = 1; i \le 100; i = i + 1)
                                                 1부터 100까지 반복하며 sum에 값을
05
                  sum = sum + i;
06
07
         System.out.println("1부터 100까지의 합은 " +sum); -- 값을 출력
08
09 }
```



■ 주요 언어로 작성된 프로그램 : C 프로그램



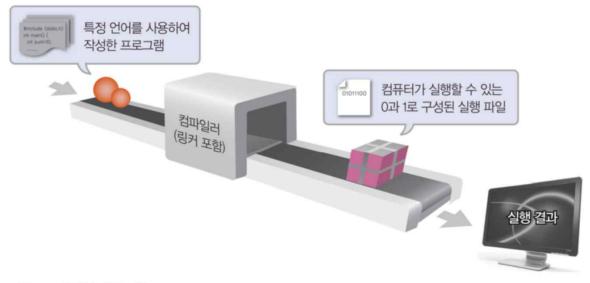
주요 언어로 작성된 프로그램 : 파이썬 프로그램



- 작성된 프로그램은 컴퓨터가 알아 들을 수 있는 0과 1로 번역되어 실행되어야 합니다.
 - 프로그램을 0과 1로 번역하여 실행하는 방법
 - 1) 컴파일 기법
 - 2) 인터프리트 기법
 - 3) 하이브리드 기법



■ 컴파일 기법 : 컴파일러에 의해 0과 1로 구성된 2진 파일로, 번역된 다음 실행하는 방법







■ 인터프리트 기법 : 프로그램을 직접 한 줄씩 번역한 다음 바로 실행하여 결과를 나타내는 방법

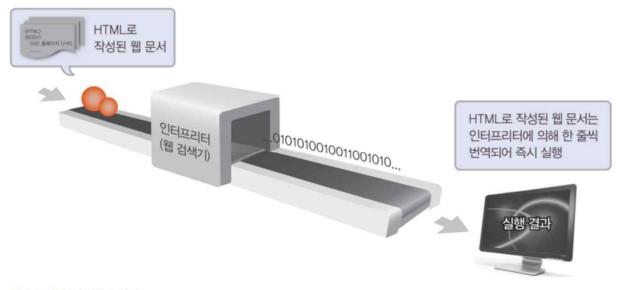


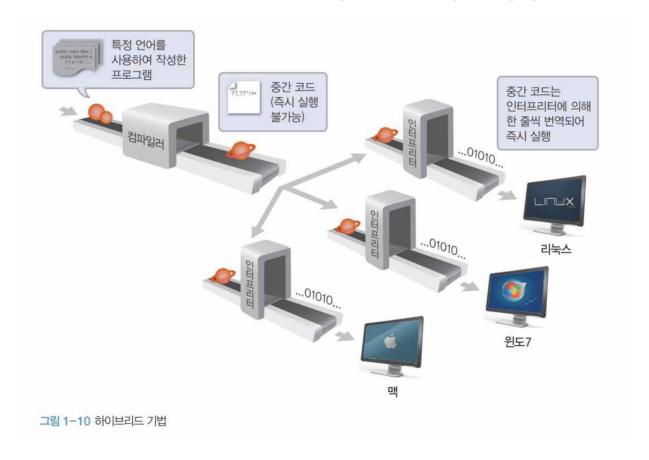
그림 1-9 인터프리트 기법

•



•

하이브리드 기법 : 컴파일 기법과 인터프리트 기법을 모두 사용
 하는 방식. 프로그램을 중간코드로 번역한 다음,
 중간 코드를 직접 한 줄씩 번역하여 실행





소프트웨어는 무엇으로 만드는가?

- ❖ 프로그래밍 언어(Programming Language)
 - 프로그램을 작성하기 위해 만들어진 인공 언어 체계
- ❖ 컴파일 방식과 인터프리트 방식 프로그래밍 언어

	언어	실행 속도	이식성
컴파일 방식	C, C++, 파스칼, 에이다	CPU가 실행할 수 있는 기계 코드로 컴파일되므로 실행 속도 빠름 .	원래 애플리케이션이 개발된 CPU/운영체제와 다른 CPU/운영체제용으로 옮기는 경우, 대체로 코드를 그대로 사용할수 없음. 이식성 낮음. ※ C#, 자바 등 가상머신 기반의 언어는 이식성 높음
인터프리트 방식	베이직, 파이썬, 루비, PHP, 펄		인터프리터만 대상 CPU/운영체제를 지원한다면 애플리케이션 코드는 변경할 필요 없이 어떤 환경에서나 실행 가능. 이식성 높음.

■ 파이썬의 등장

• 파이썬(Python)은 귀도 반 로섬(Guido Van Rossum)이 1991년에 개발한 언어로, 처음에는 C 언어 기반으로 개발되었는데 이후 다양한 기능이 개발되어 추가되었다.





[파이썬 로고]

귀도 반 로섬]



■ 파이썬의 특징

- 1. 플랫폼 독립적인 언어: 어떤 운영체제든 상관없이 사용할 수 있는 언어를 말한다.
- 인터프리터 언어: 컴파일러 언어와 달리, 소스코드 자체가 바로 실행되는 특징이 있는 언어이다.
 이로 인해 속도는 느리지만, 굉장히 간편하게 사용할 수 있다.
- 3. **객체 지향 언어 :** 해당 프로그램이 해결해야 할 문제의 구성요소를 요소별로 정의한 뒤, 각 요소의 기능(메서드)과 정보(속성)를 정의하여 요소들을 결합하고, 프로그램을 작성하는 방식이다.
- 4. 동적 타이핑 언어: 프로그램의 실행 시점에서 각 프로그램 변수의 타입을 결정하는 언어이다.



■ 파이썬을 배우는 이유

- 1. 쉽고 간단한 프로그래밍 언어
- 화면에 'Hello World!'라는 텍스트를 출력하는 프로그램을 작성한다고 가정하자.
 파이썬과 자바로 코드를 작성하면 아래와 같다.

```
파이썬
```

```
print("Hello World!")
```

자바(Java)

```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

■ 파이썬을 배우는 이유

- 1. 쉽고 간단한 프로그래밍 언어
- 파이썬과 자바로 화면에 1부터 10까지 출력하는 프로그램을 코드로 작성하면 다음처럼 두 언어를 비교할 수 있다.

파이썬

```
for i in range(1, 11):

print (i)
```

자바(Java)

```
for (i = 1; i <11; i++){
    System.out.println (i)
}</pre>
```

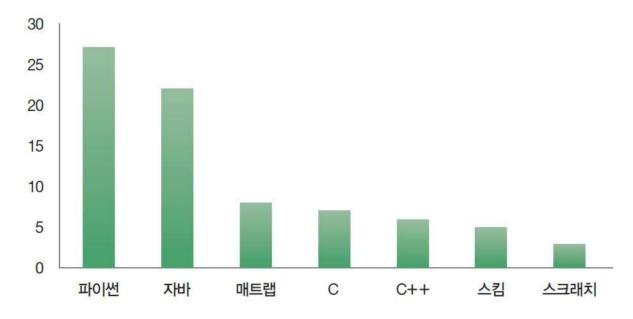
• 이처럼 파이썬은 프로그래밍을 처음 배우는 초보자도 다른 프로그래밍 언어보다 훨씬 간단하고 이해하기 쉽다는 장점이 있다.

- 파이썬을 배우는 이유
- 2. 다양한 라이브러리 제공
- 파이썬은 다양한 라이브러리를 제공하여 활용 범위가 넓다.
- 개발자들이 만들어 놓은 많은 모듈과 패키지를 쉽게 사용할 수 있다.



■ 파이썬을 배우는 이유

- 3. 대중적인 프로그래밍 언어
- 대표적인 컴퓨터 공학 학회지인「ACMAssociation for Computing Machinery」은 2014년 조사를 통해 미국 39개 대학 중 파이썬을 기초 프로그래밍 언어로 선택한 대학이 가장 많다고 밝혔다.

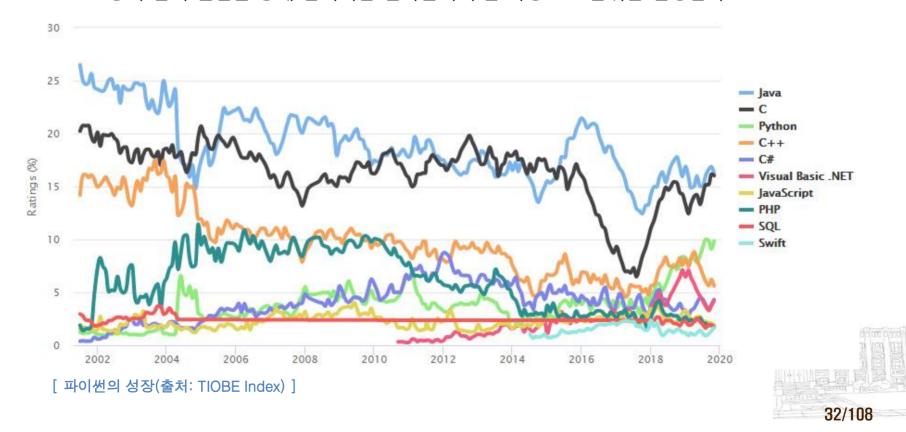


[미국 39개 대학의 기초 프로그래밍 언어 강의 선택 현황(출처: ACM)]



■ 파이썬을 배우는 이유

- 3. 대중적인 프로그래밍 언어
- TIOBE Index는 해당 프로그래밍 언어를 이용하는 엔지니어 및 업체의 수와 Google, Bing,
 Yahoo 등의 검색 엔진을 통해 검색되는 결과물의 수를 바탕으로 순위를 결정한다



■ 파이썬을 배우는 이유

3. 대중적인 프로그래밍 언어

Nov 2019	Nov 2018	Change	Programming Language	Ratings	Change
1	Ä		Java	16.246%	-0.50%
2	2		С	16.037%	+1.64%
3	4	^	Python	9.842%	+2.16%
4	3	~	C++	5.605%	-2.68%
5	6	^	C#	4.316%	+0.36%
6	ā	~	Visual Basic ,NET	4.229%	-2.26%
7	7		JavaScript	1.929%	-0.73%
8	8		PHP	1.720%	-0.66%
9	9		SQL	1.690%	-0.15%
10	12	^	Swift	1.653%	+0.20%

- 파이썬을 배우는 이유
- 3. 2019년 엔지니어들이 가장 선호하는 프로그래밍 언어

Rank	Language	Type		Score
1	Python	•	₽ @	100.0
2	Java	9 0	0	96.3
3	С	0	D ⊕	94.4
4	C++	0	□ •	87.5
5	Я		Q.	81.5
6	JavaScript	•		79.4
7	C#	• -	P 0	74.5
8	Matlab		Ģ	70.6
9	Swift) 🖸	69.1
10	Go	•	Ç.	68.0



2. 파이썬 개발 환경

- ❖ 텍스트 에디터 → Visual Studio Code
 - 프로그래밍 언어 입력, 작성하는 도구 → 메모장, 노트패드++, Editplus, 파이참 등
- ❖ 파이썬 인터프리터
 - 프로그래밍 언어 해석하는 도구

```
int;=0;

zwhile(;<Psc_(*unt)i

form_s(RPSC,PSCRC;),">c**);

while(!feof(PSC));

form(Psc);

if(st>cmp)==

n else!forts(BSC);

form(PSC);

form(PSC);

infc!=se(BSC);

il

ib
```

EULE MILIE

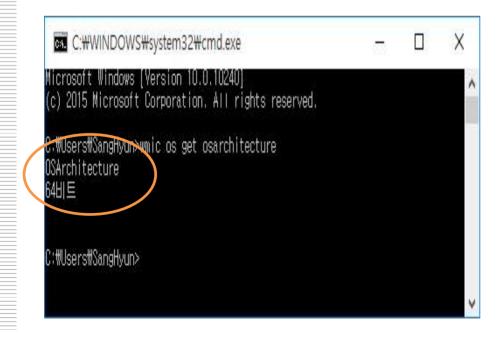


파이번 해석/실행기



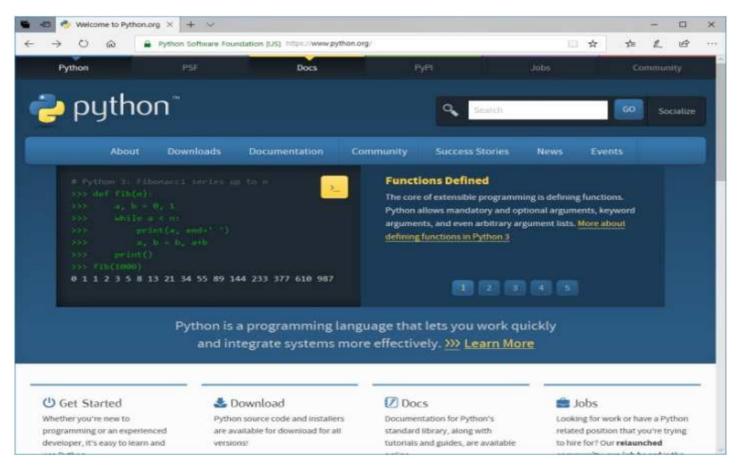
2. 파이썬 설치하기

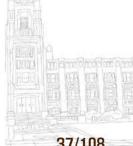
- ❖ 윈도우용 파이썬 설치 파일은 32비트/64비트용이 따로 배포됨
 - 어떤 버전을 사용할지 판단하기 위해서는 PC에 탑재되어 있는 윈도우의 아키텍쳐를 파악하는 것이 필요
- ❖ 윈도우의 아키텍쳐 알아내기
 - [윈도우]+[R] 키를 눌러 <실행> 창 표시
 - <실행> 창이 나타나면 [열기] 입력 상자에 "cmd"를 입력하고 [확인] 버튼을 클릭
 - 명령행 프롬프트에서 "wmic os get osarchitecture" 을 입력
 - 또는 [내 컴퓨터] 마우스 우클릭 > [속성]



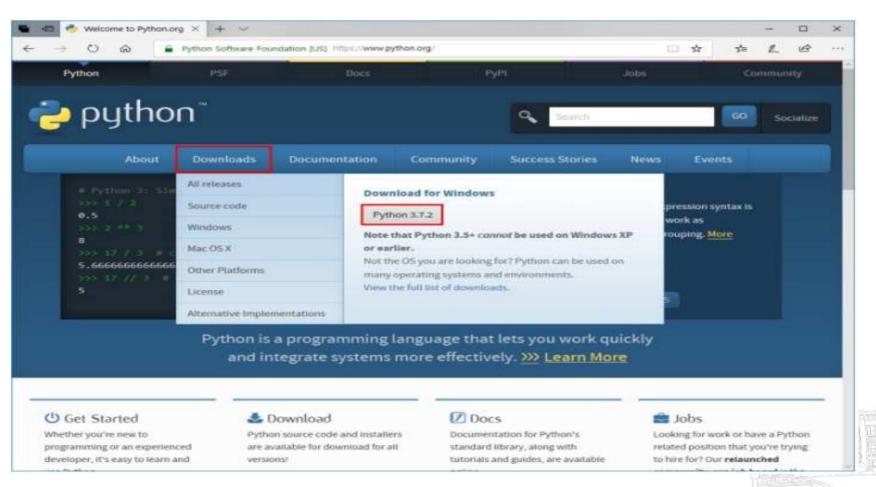


- ❖ 파이썬 공식 홈페이지에서 다운로드
 - https://www.python.org
 - [Downloads] → [Download Python 3.7.5] 메뉴 클릭

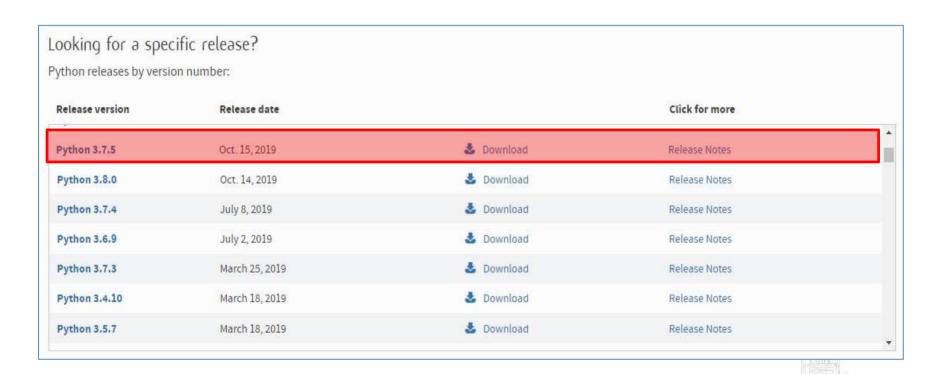




- ❖ 파이썬 공식 홈페이지에서 다운로드
 - https://www.python.org
 - [Downloads] 메뉴 클릭 후 아래 부분으로 이동



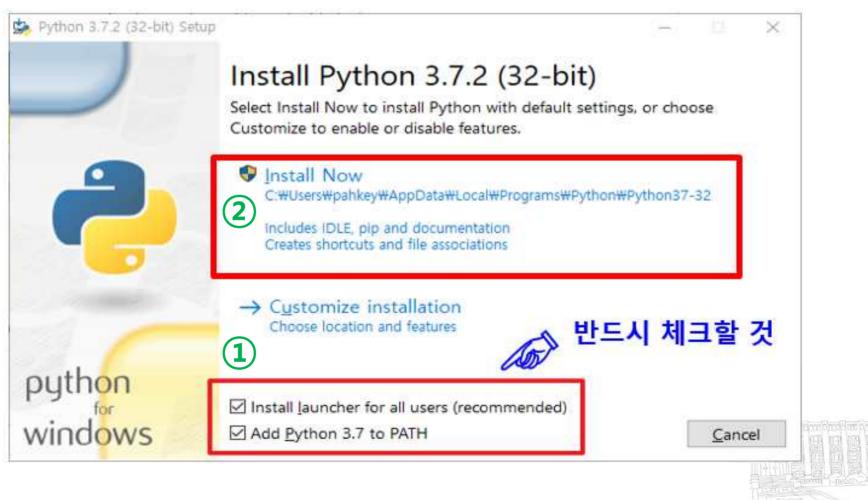
- ❖ 파이썬 공식 홈페이지에서 다운로드
 - https://www.python.org
 - 메뉴에서 [Python 3.7.5] 버전 클릭 후 아래 부분으로 이동



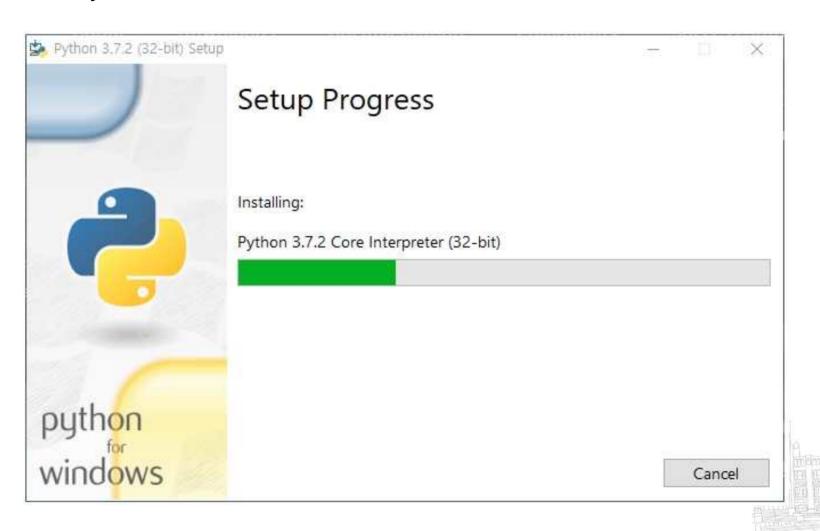
- ❖ 파이썬 공식 홈페이지에서 다운로드
 - https://www.python.org
 - 메뉴에서 [Windows x86_64 executeable installer] 클릭 후 다운로드 함.

Files					
Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		1cd071f78ff6d9c7524c95303a3057aa	23126230	SIG
XZ compressed source tarball	Source release		08ed8030b1183107c48f2092e79a87e2	17236432	SIG
macOS 64-bit/32-bit installer	Mac OS X	(Deprecated) for Mac OS X 10.6 and later	cd503606638c8e6948a591a9229446e4	35020778	SIG
macOS 64-bit installer	Mac OS X	for macOS 10.9 and later	20d9540e88c6aaba1d2bc1ad5d069359	28198752	SIG
Windows help file	Windows		608cafa250f8baa11a69bbfcb842c0e0	8141193	SIG
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	436b0f803d2a0b393590030b1cd59853	7500597	SIG
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	697f7a884e80ccaa9dff3a77e979b0f8	26777448	SIG
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	b8b6e5ce8c27c20bfd28f1366ddf8a2f	1363032	SIG
Windows x86 embeddable zip file	Windows		726877d1a1f5a7dc68f6a4fa48964cd1	6745126	SIG
Windows x86 executable installer	Windows		cfe9a828af6111d5951b74093d70ee89	25766192	SIG
Windows x86 web-based installer	Windows		ea946f4b76ce63d366d6ed0e32c11370	1324872	SIG
	1			THE PERSON NAMED IN COLUMN	

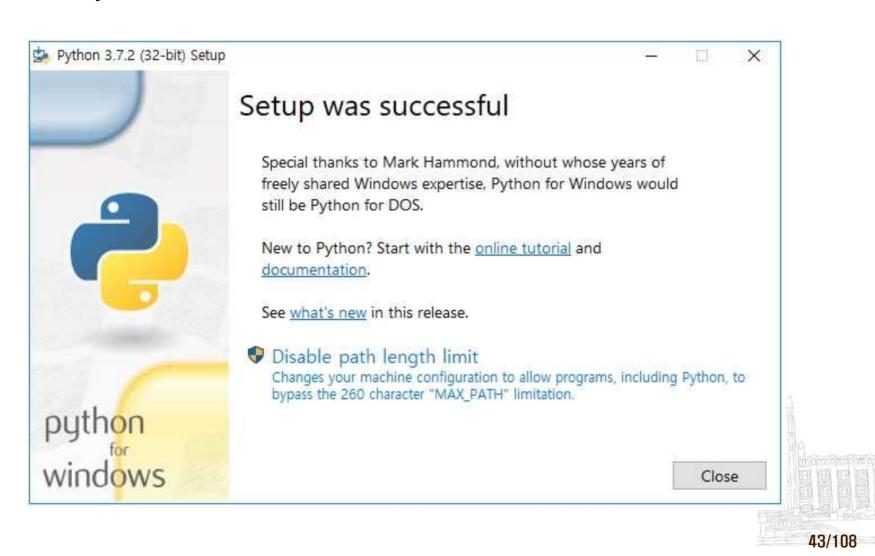
- ❖ 설치하기
 - [Add Python 3.7.5 to PATH] 설정 → [Install Now] 클릭



- ❖ 설치하기
 - [Add Python 3.7.5 to PATH] 설정 → [Install Now] 클릭



- ❖ 설치하기
 - [Add Python 3.7.5 to PATH] 설정 → [Install Now] 클릭



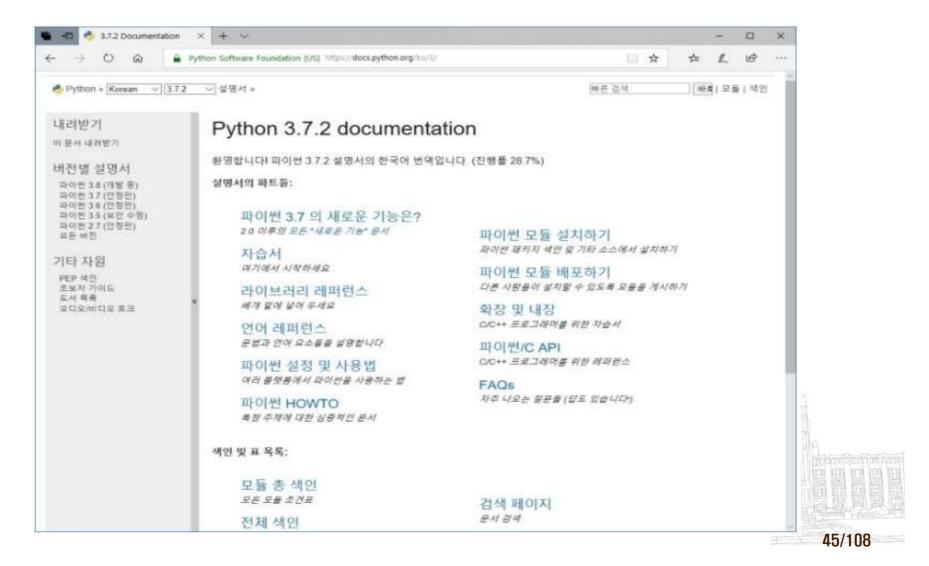
❖ 설치하기

Online tutorial: https://docs.python.org/3.7/tutorial/index.html



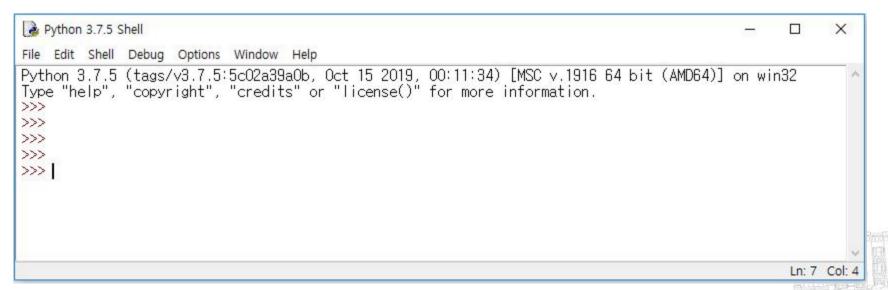
❖ 설치하기

documentation : https://docs.python.org/3.7/index.html



❖ 설치 확인

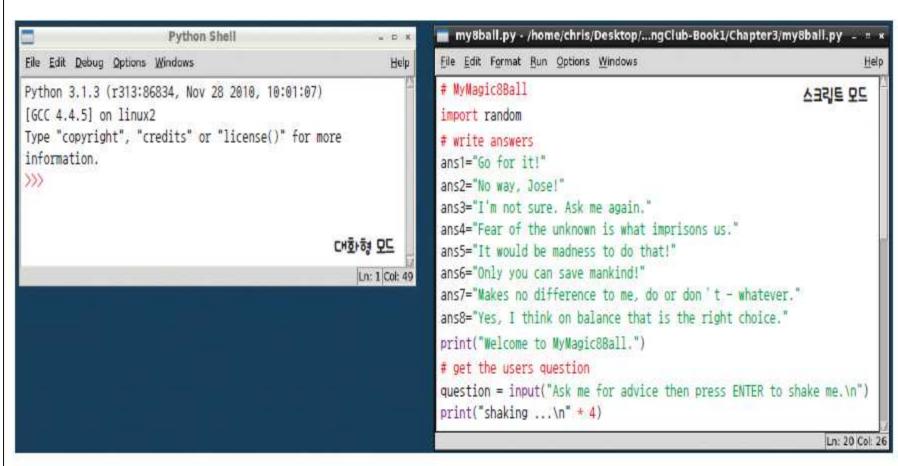




2. IDLE의 두 가지 모드

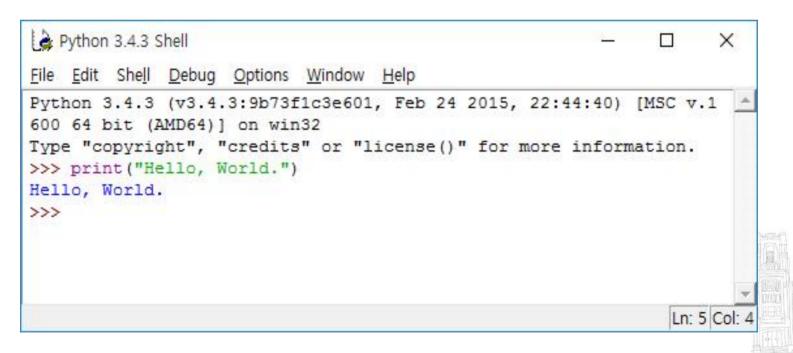
IDLE의 두 가지 모드

그림 3.1 IDLE - 왼쪽은 대화형 모드, 오른쪽은 스크립트 모드



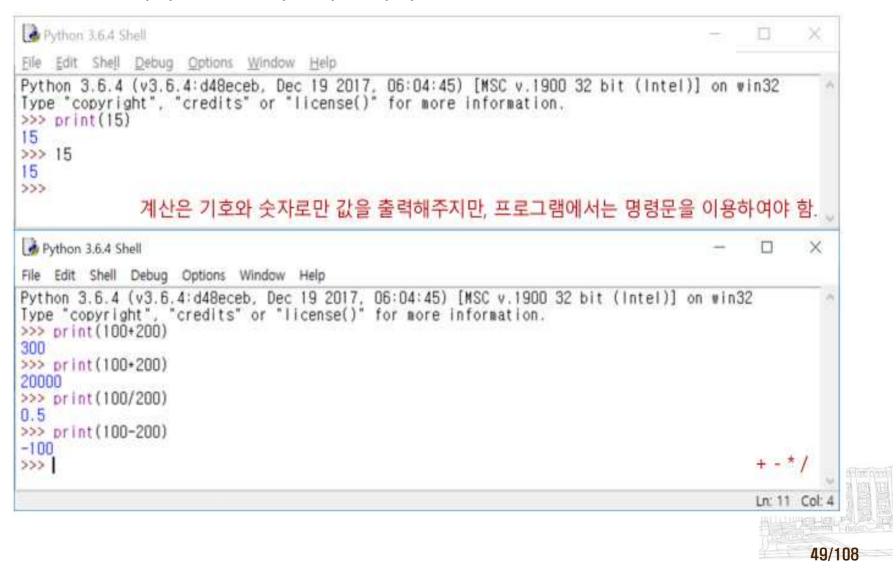
2. IDLE의 두 가지 모드 - 파이썬 쉘로 코딩하기

- ❖ IDLE은 대화형 파이썬 쉘(Python Shell)과 편집기 모드로 사용 가능
 - 짧고 간단한 예제 코드는 파이썬 쉘을 이용
 - 길고 복잡한 예제 코드는 코드 편집기를 이용
- ❖ 파이썬 쉘로 코딩하기
 - [Python 3.X]→[IDLE(Python GUI)] 항목을 찾아 클릭하여 파이썬 쉘 실행
 - 다음과 같이 print("Hello, World.") 입력하고 엔터키 입력



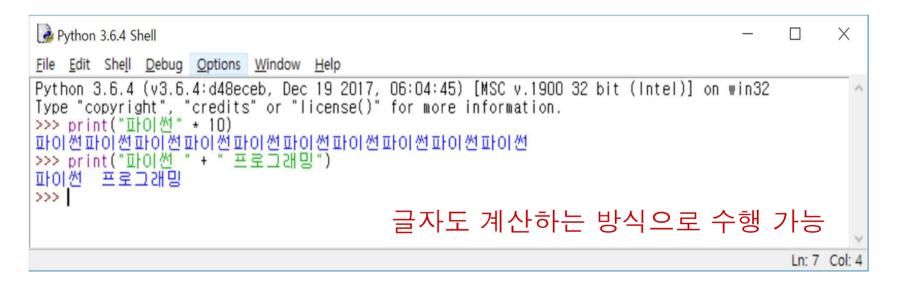
2. IDLE의 두 가지 모드 - 파이썬 쉘로 코딩하기

- ❖ 파이썬 쉘로 코딩하기
 - 프롬프트에서 간단한 계산 수행하기



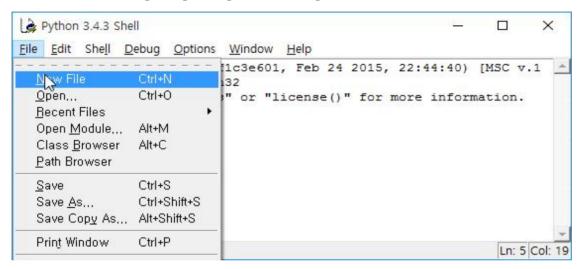
2. IDLE의 두 가지 모드 - 파이썬 쉘로 코딩하기

- ❖ 파이썬 쉘로 코딩하기
 - 프롬프트에서 간단한 계산 수행하기

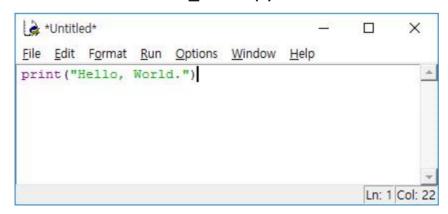




- ❖ 코드 편집기로 코딩하기 1
 - IDLE 창에서 [File] → [New File] 메뉴 항목을 클릭하여 코드 편집기 실행

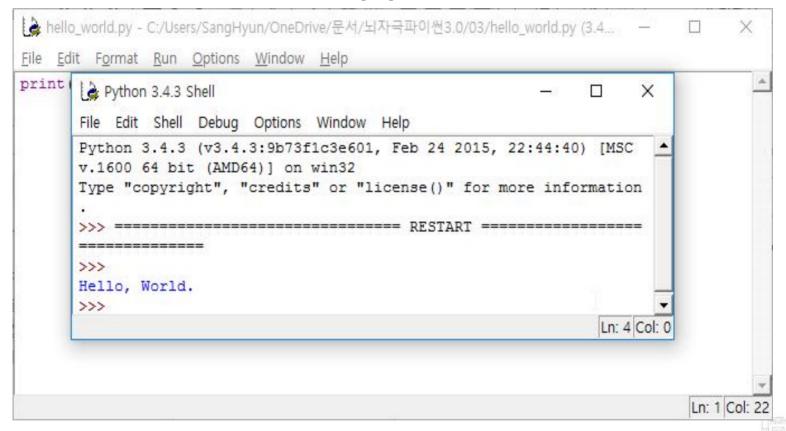


- 다음과 같이 코드를 입력하고 [Ctrl]+[S]키를 눌러서 파일 저장.
- 파일 이름은 hello_world.py

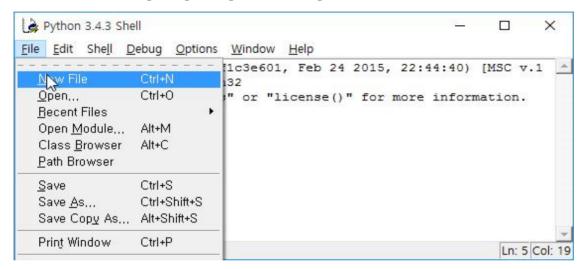




- ❖ 코드 편집기로 코딩하기 2
 - 코드 편집기가 활성화된 상태에서 [F5] 버튼을 클릭



- ❖ 코드 편집기로 코딩하기 3
 - IDLE 창에서 [File] → [New File] 메뉴 항목을 클릭하여 코드 편집기 실행

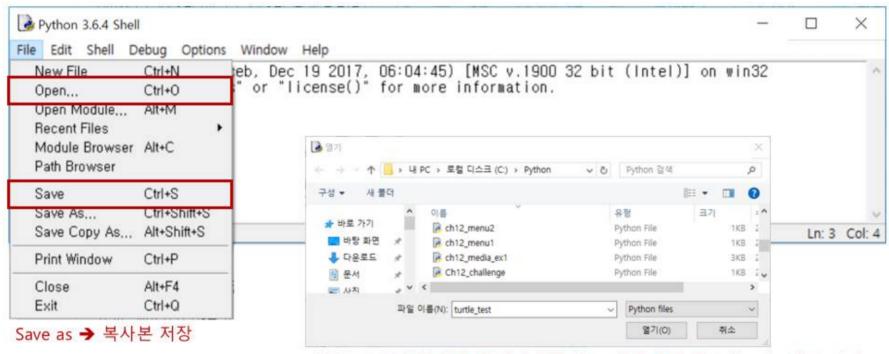


- 다음과 같이 코드를 입력하고 [Ctrl]+[S]키를 눌러서 파일 저장.
- 파일 이름은 cal01.py 로 저장

```
# file : cal01.py
import calendar
calendar.setfirstweekday(6) # 일요일을 첫 요일로..
calendar.prmonth(2019, 2)
```



- ❖ 코드 편집기로 코딩하기 3
 - IDLE 창에서 [File] → [New File] 메뉴 항목을 클릭하여 코드 편집기 실행



기본으로 설치한 경우 위치가 복잡함 → 설치 폴더 간소화 또는 바로 가기

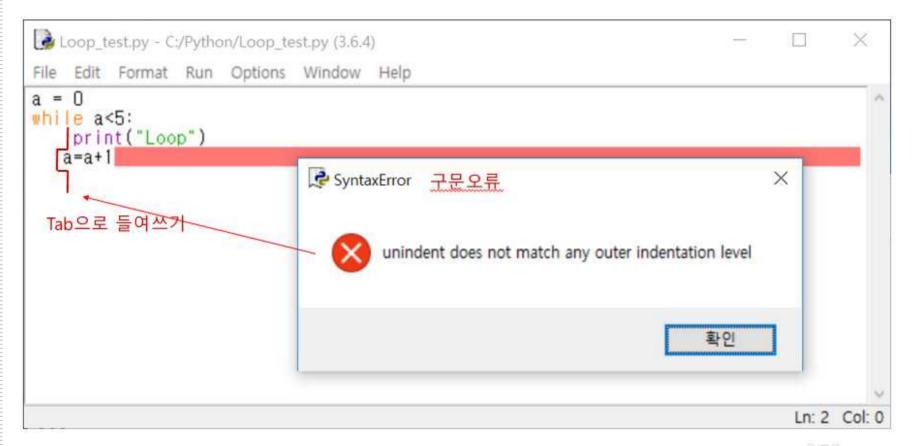


❖ 프로그래밍 오류 > 버그 / 수정 > 디버깅

```
Python 3.6.4 Shell
                                                                                                       X
File Edit Shell Debug Options Window Help
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, O6:04:45) [MSC v.1900 32 bit (Intel)] on win32 Type "copyright", "credits" or "license()" for more information.
>>> pritn(41)
Traceback (most recent call last):
 File "<pyshell#0>", line 1, in <module>
    pritn(41)
NameError: name 'pritn' is not defined
>>> <u>pritn(41)</u>
Traceback (most recent call last):
                                                           → 오류 함수 역 추적 시작
                                                           → 파일 "<pyshell#0>"의 라인1번째에
  File "<pyshell#0>", line 1, in <module>
    pritn(41)
                                                           → pritn(41)이 오류
NameError: name 'pritn' is not defined
                                                           → pritn은 정의되지 않은 이름
```



❖ 프로그래밍 오류 > 버그 / 수정 > 디버깅





- ❖ 파이썬 철학
 - IDLE 창에서 아래의 명령 실행

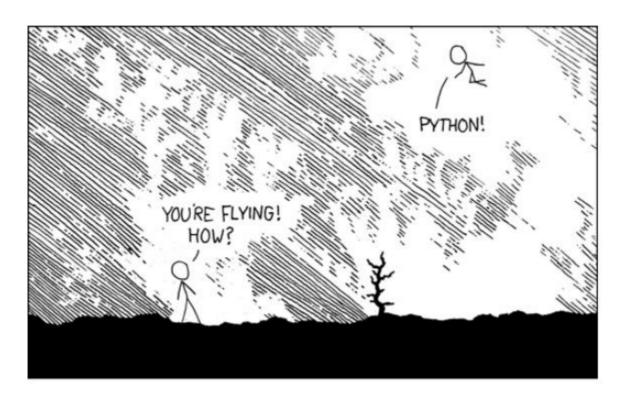
>>> import this

아름다운 것이 추한 것보다 낫다.(Beautiful is better than ugly) 명시적인 것이 암시적인 것보다 낫다. (Explicit is better than implicit) 간결한 것이 복잡한 것보다 낫다. (Simple is better than complex)



- ❖ 파이썬 철학
 - IDLE 창에서 아래의 명령 실행

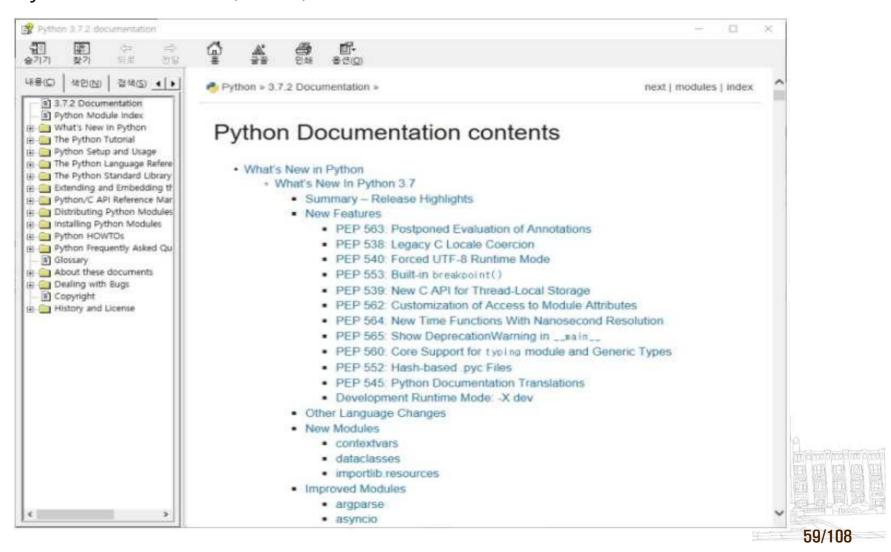
>>> import antigravity



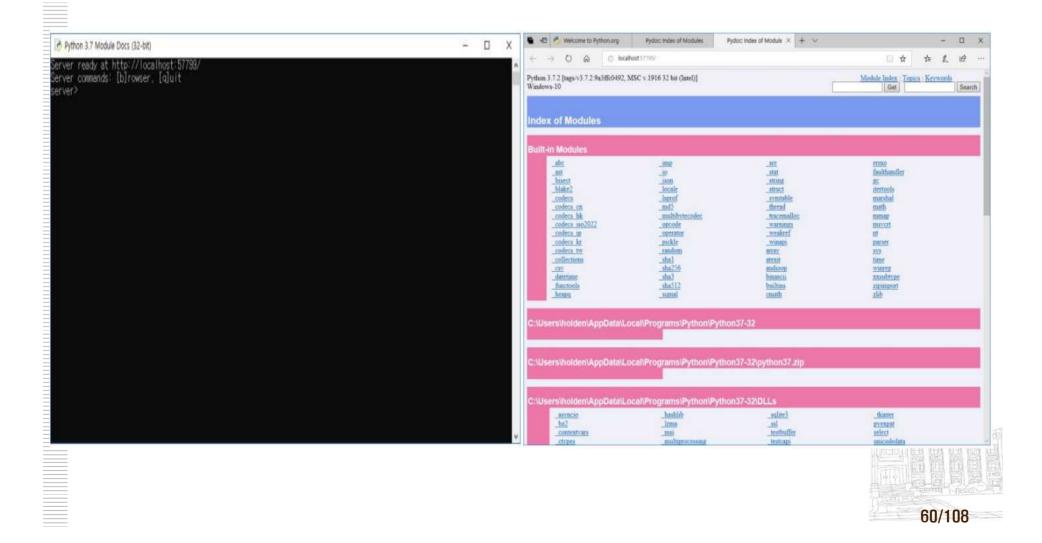


❖ 설치 확인

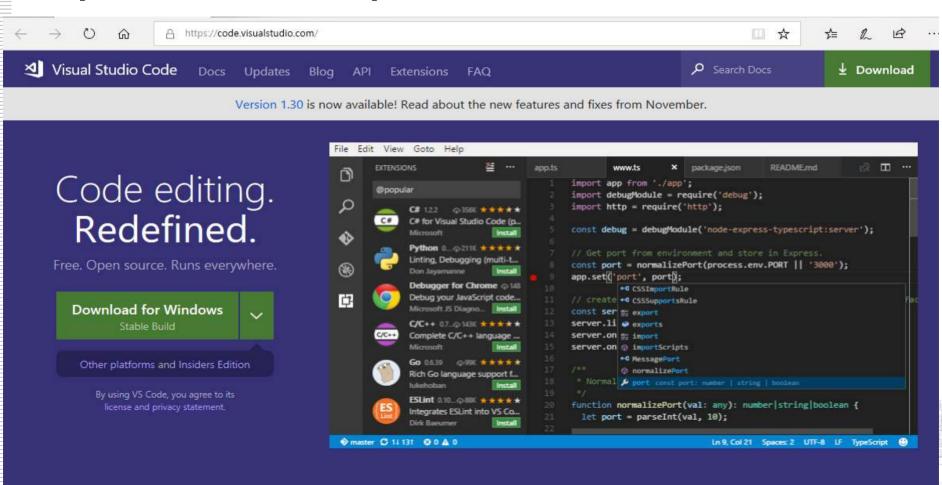
Python 3.7 Manual (32-bit) 실행 화면



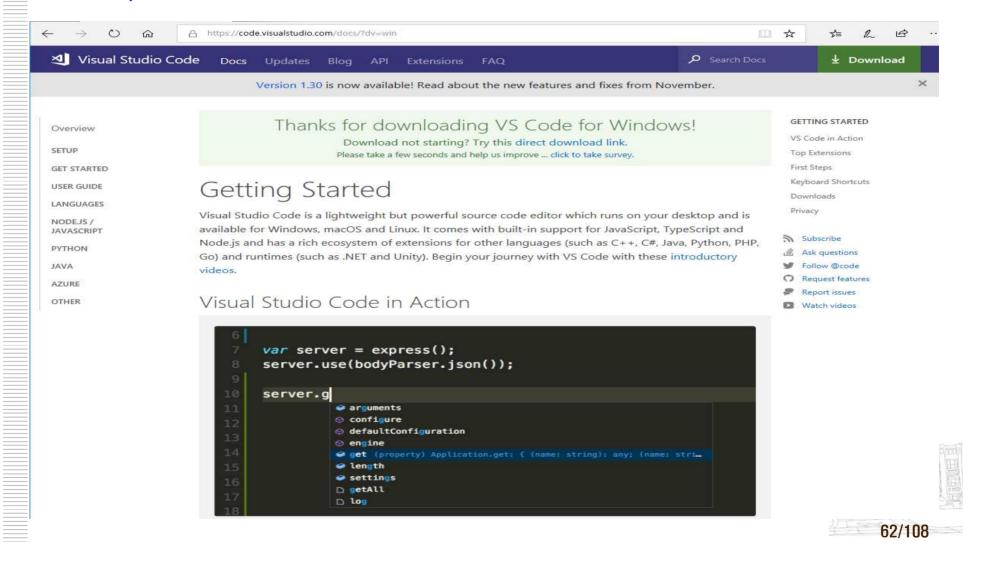
- ❖ 설치 확인
 - Python 3.7 Module Docs (32-bit) 실행 화면



- ❖ Visual Studio Code 홈페이지에서 다운로드
 - http://code.visualstudio.com
 - [Download for Windows]

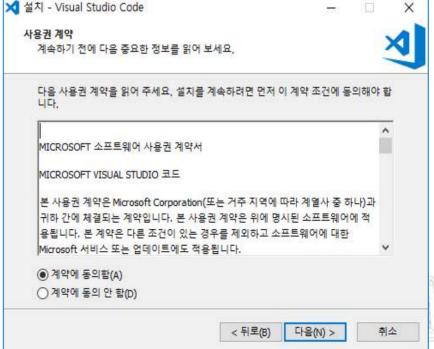


- ❖ Visual Studio Code 홈페이지에서 다운로드
 - http://code.visualstudio.com



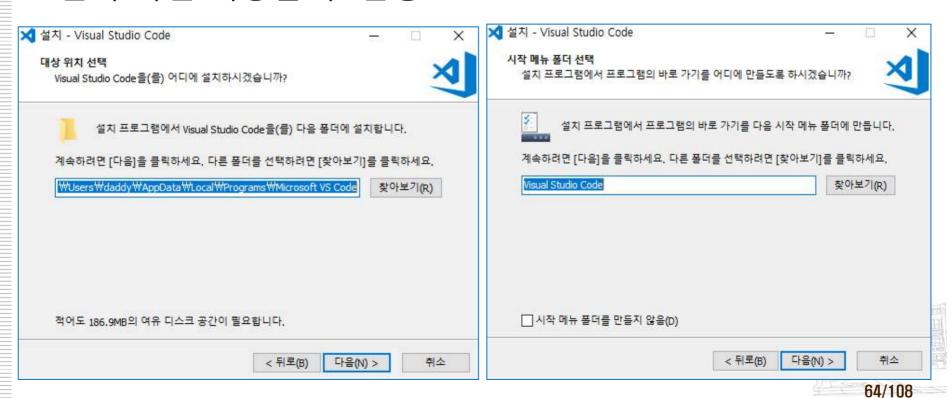
- ❖ Visual Studio Code 홈페이지에서 다운로드
 - http://code.visualstudio.com
 - [Download for Windows]
- ❖ 설치 파일 저장한 후 실행



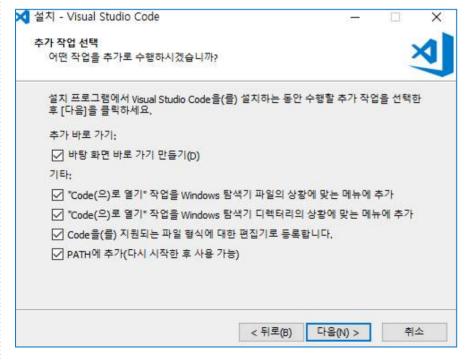


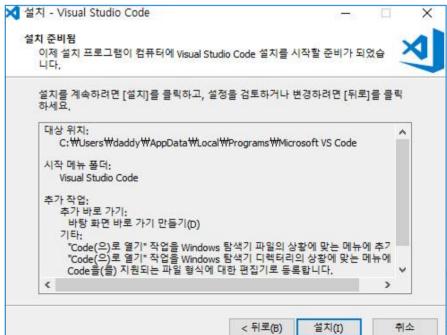
63/108

- ❖ Visual Studio Code 홈페이지에서 다운로드
 - http://code.visualstudio.com
 - [Download for Windows]
- ❖ 설치 파일 저장한 후 실행

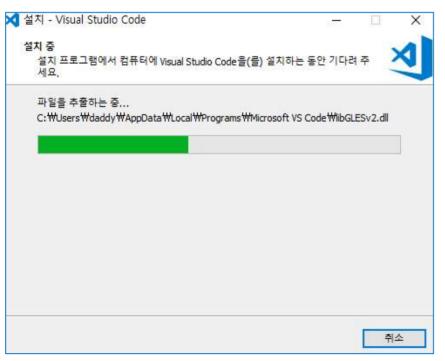


- ❖ Visual Studio Code 홈페이지에서 다운로드
 - http://code.visualstudio.com
 - [Download for Windows]
- ❖ 설치 파일 저장한 후 실행





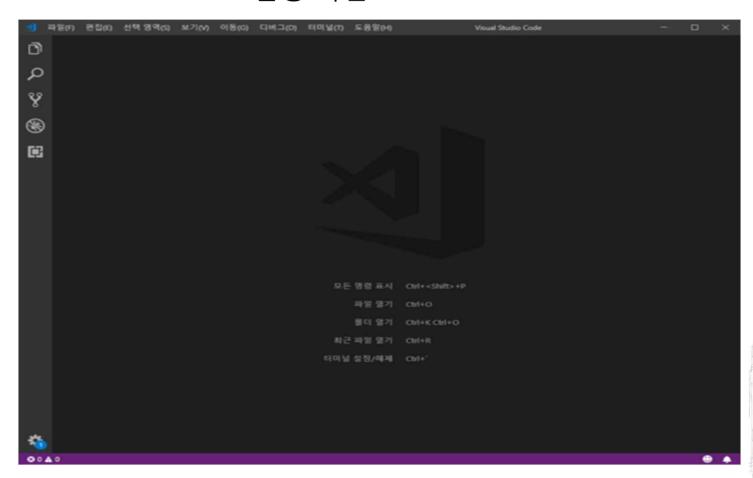
- ❖ Visual Studio Code 홈페이지에서 다운로드
 - http://code.visualstudio.com
 - [Download for Windows]
- ❖ 설치 파일 저장한 후 실행



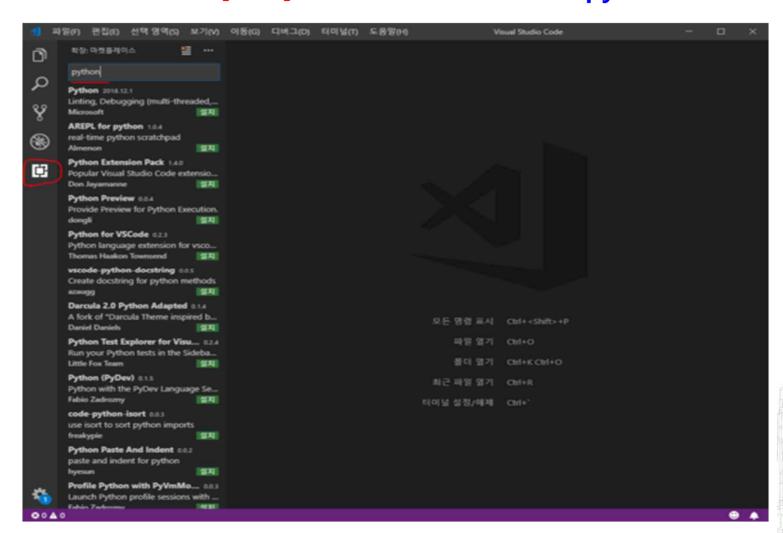


66/108

- ❖ Visual Studio Code 환경 설정
 - 단축키 (https://demun.github.io/vscode-tutorial/shortcuts/)
 - Visual Studio Code 실행 화면



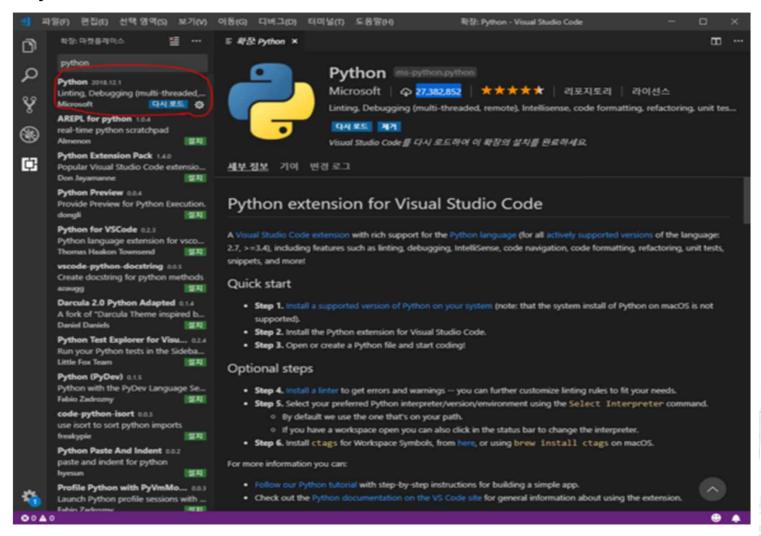
- ❖ Visual Studio Code 환경 설정: CTRL + SHIFT + x
 - 왼쪽 메뉴탭의 [확장]버튼 클릭 -> 검색창에 "python" 입력



68/108

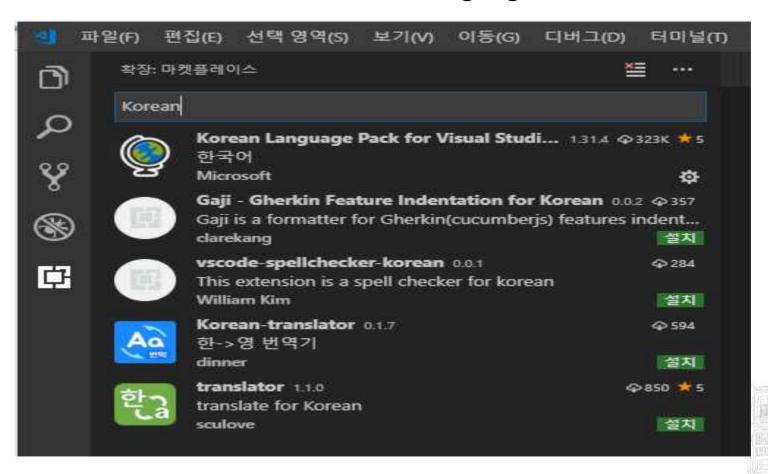
❖ Visual Studio Code 환경 설정

■ Python "설치" 버튼을 눌러 VS code에서 파이썬을 확장.



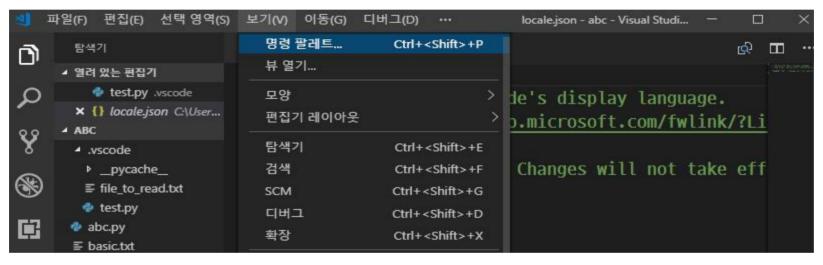
3. 코드 작성 및 실행

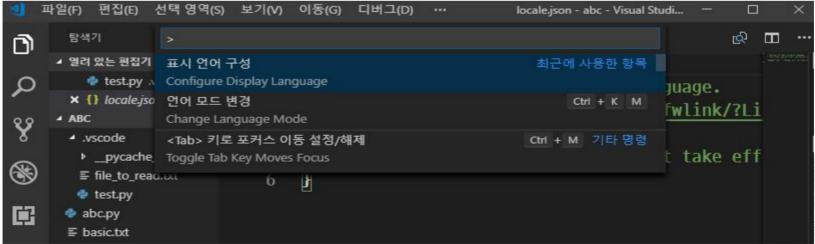
- ❖ Visual Studio Code 한글 확장 프로그램 설치
 - [확장] [korean] 검색 후 "Korean Language Pack for......" 설치



3. 코드 작성 및 실행

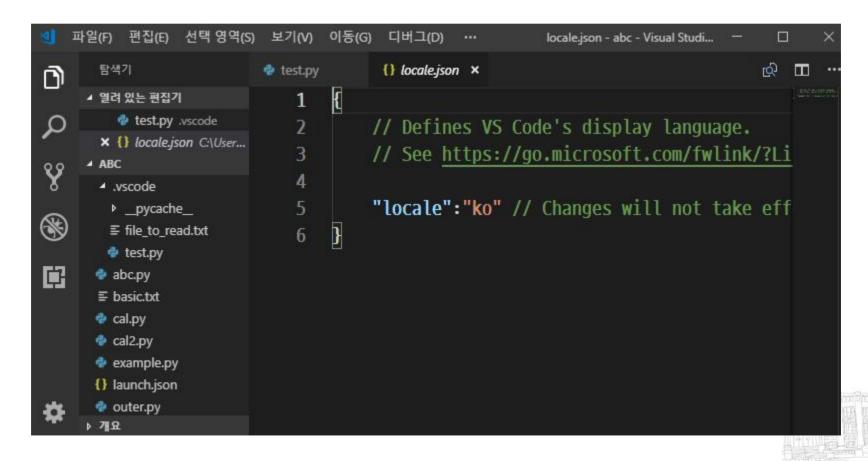
- ❖ Visual Studio Code 한글 확장 프로그램 설치
 - [보기] [명령 팔레트] [표시 언어 구성] 검색 후 선택



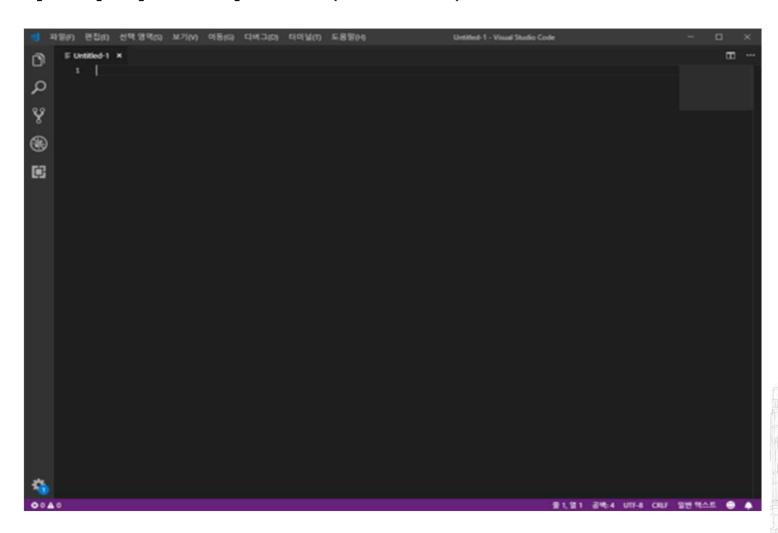


3. 코드 작성 및 실행

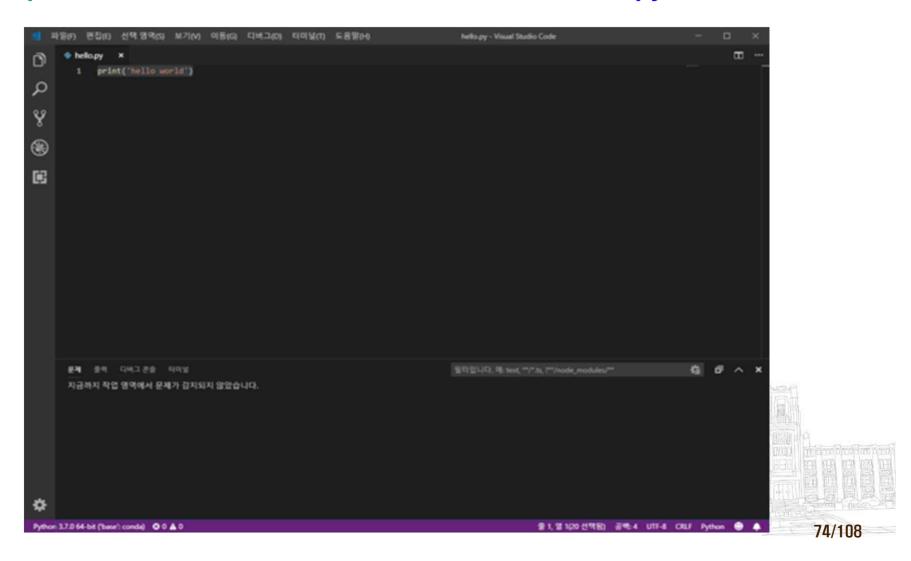
- ❖ Visual Studio Code 한글에서 영어로
 - [보기] [명령팔레트] [표시 언어 구성] 선택
 - "locale": "ko" -> "locale": "en-us" 수정 후 프로그램 재실행



- ❖ Visual Studio Code 환경 설정
 - [파일] [새 파일] 클릭 : (CTRL + N)



- ❖ Visual Studio Code 환경 설정
 - print("hello world") 입력 후 CTRL + S -> 파일명.py 로 저장



- ❖ Visual Studio Code 환경 설정
 - 다음과 같은 에러가 발생할 경우 파이썬 인터프리터가 선택되지 않아 발생하는 경우이다.

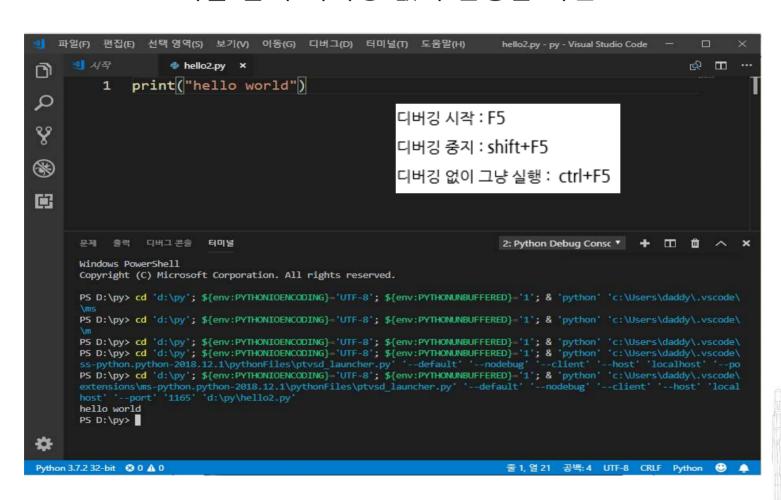
[Select Python Interpreter]를 클릭하여 설치해준다.

No Python interpreter is selected. You need to select a Python ☆ × interpreter to enable features such as IntelliSense, linting, and debugging.
 소스: Python (확장)

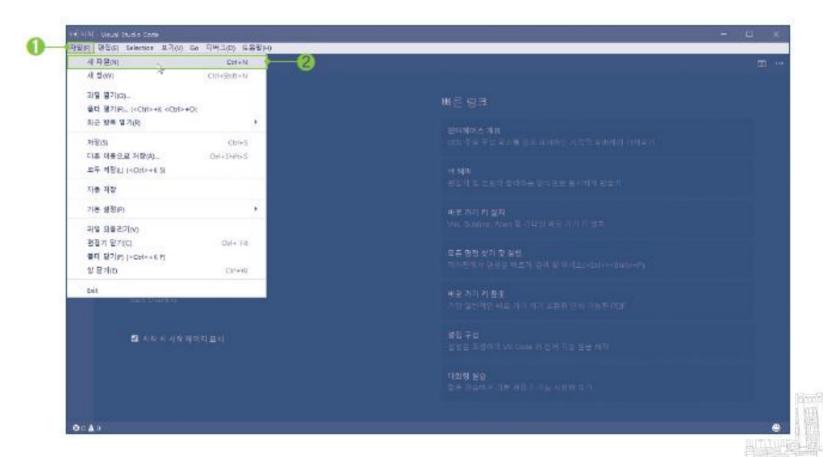
Select Python Interpreter



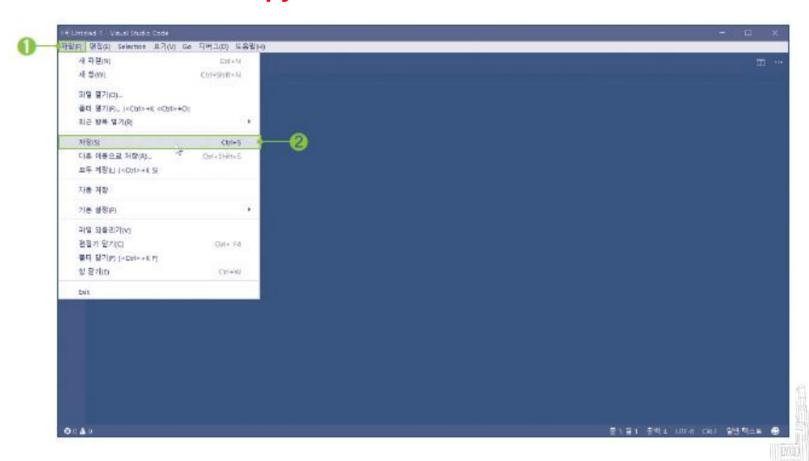
- ❖ Visual Studio Code 환경 설정
 - CTRL + F5 키를 눌러 디버깅 없이 실행한 화면



- ❖ 윈도우에서 파일 만들기
 - Visual Studio Code 실행
 - [파일] [새 파일(N)] 메뉴 클릭 (단축키: Ctrl + N)



- [파일] [저장(S)] 메뉴 클릭(단축키: Ctrl + S)
- 폴더 지정 후 hello.py 이름으로 저장

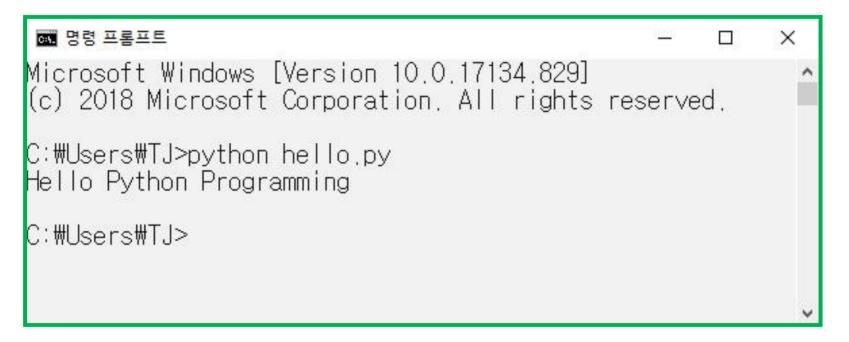


- ❖ 실행하기
 - 코드 파일 저장한 폴더로 이동
 - [Shift] + 빈 공간 마우스 오른쪽 버튼 클릭
 - [여기서 명령 창 열기(W)] 메뉴 클릭

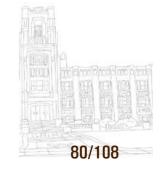




■ 명령 프롬프트에 python hello.py 입력 → [Enter] 키

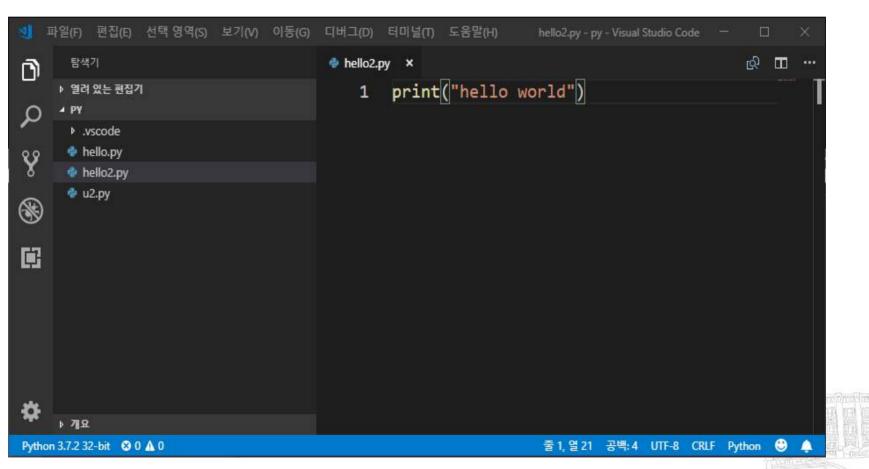


- ❖ 파이썬 명령어로 파일 실행하기
 - python <파일 경로>



05. 폴더 열기

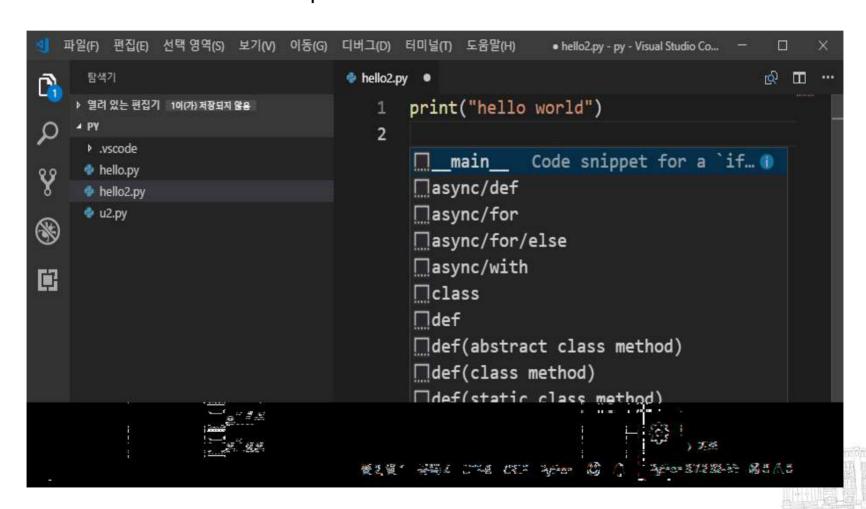
- ❖ 폴더 열기
 - [파일] [폴더 열기] 코드 파일 저장한 폴더 선택
 - 저장된 모든 파일 목록 확인



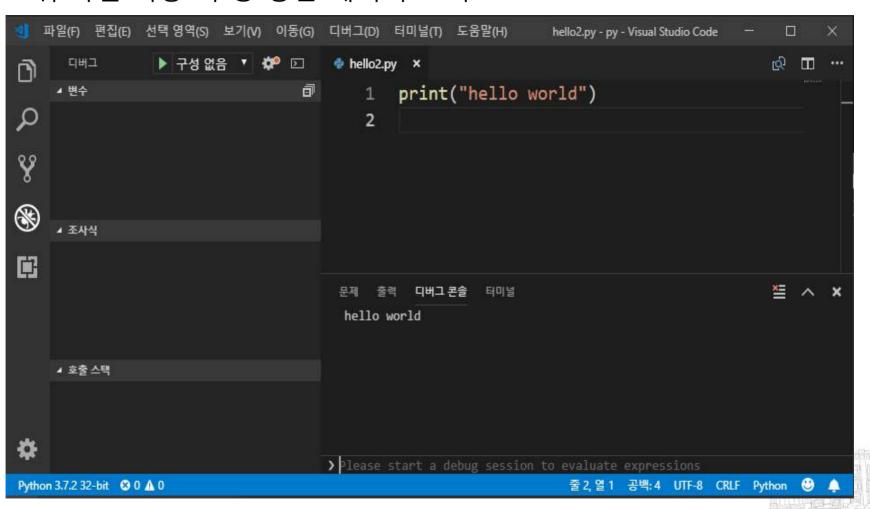
- ❖ Visual Studio Code 확장 프로그램 설치
 - 위 파일 저장 시 창 상단 메시지 표시
 - [정보] 이 파일 형식에 대해 Python 확장이 권장됩니다.
 - [권장 사항 표시] [확장] [Python] [설치] 클릭



- ❖ Visual Studio Code 확장 프로그램 설치
 - 코드 입력시 CTRL + Spacebar 클릭



- ❖ Visual Studio Code 확장 프로그램 설치
 - 위 파일 저장 시 창 상단 메시지 표시



07. 코딩 전용 폰트

- ❖ 글자의 구분
 - oO0 : 소문자 영문, 대문자 영문, 숫자 0
 - | I | : 소문자 영문 엘, 대문자 영문 아이, 특수기호 파이프
- ❖ 글자의 너비 통일
 - 알파벳을 입력할 경우





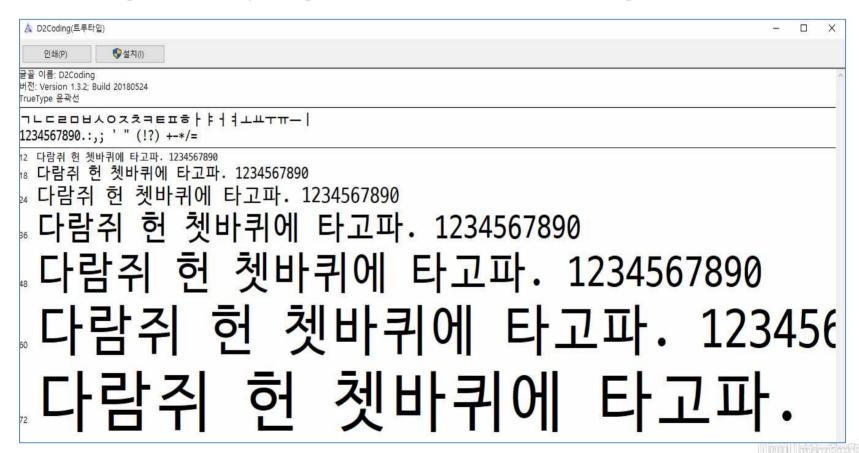


- ❖ Visual Studio Code의 경우 코딩전용 폰트 사용
 - D2codingfont : https://github.com/naver/d2codingfont



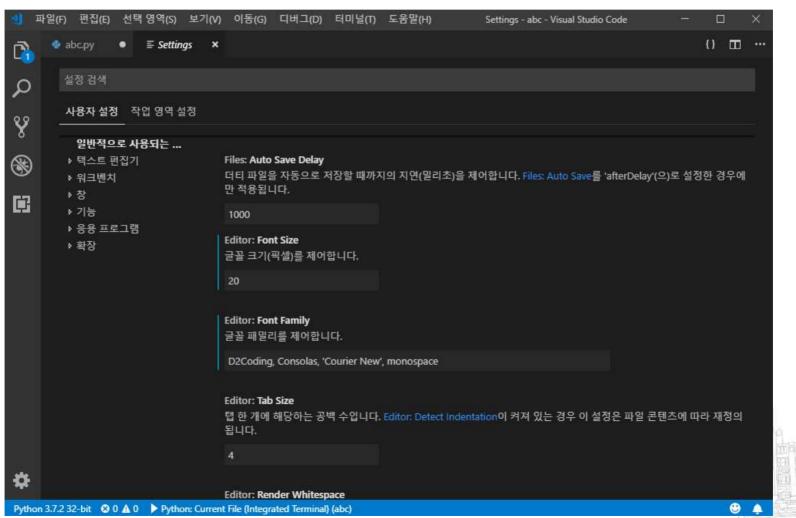
07. 코딩 전용 폰트

- ❖ Visual Studio Code의 경우 코딩전용 폰트 다운로드
 - D2codingfont : https://github.com/naver/d2codingfont



07. 코딩 전용 폰트

- ❖ Visual Studio Code의 경우 코딩전용 폰트 사용
 - [파일] [기본 설정] [설정]



■ 파이썬 개발 환경 설정

1. 운영체제 선정

운영체제	장점	단점
윈도(Windows)	국내 사용자가 쓰기에 가장 쉬운 운영체제로, 프로 그래밍을 시작하는 사람에게 상당히 좋은 선택임	다른 사용자가 개발한 다양한 모듈을 설치하기 어렵고, 참고 문서가 부족함
리눅스(Linux)	모듈 설치가 매우 쉽고, 다양한 레퍼런스를 인터 넷에서 검색하기 쉬움	초기 사용자가 사용하기에는 운영체제 자체가 매우 어려움
맥 OS(Mac OS) 기본적으로 리눅스의 장점이 있으면서, 윈도처럼 안정적임 다른 운영체제보다 상대		다른 운영체제보다 상대적으로 가격이 비쌈

[운영체제별 특징]



- 파이썬 개발 환경 설정
- 2. 파이썬 인터프리터 선정

종류	<u>d</u> g	
Python	일반적인 파이썬, 기본적인 모듈 포함	
Anaconda	다양한 과학 계산용 모듈을 묶어 패키지로 제공	
Canopy	다양한 과학 계산용 모듈을 묶어 패키지로 제공	

[주요 파이썬 인터프리터]



■ 파이썬 개발 환경 설정

3. 코드 편집기 선정

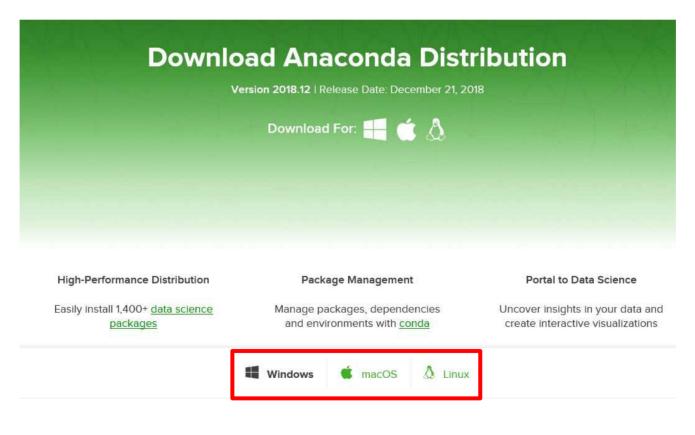
종류	설명
메모장	윈도의 기본 문서 편집 도구
VI editor	리눅스의 기본 문서 편집 도구
Sublime Text	프로그래밍에 특화된 문서 편집 도구
Atom	프로그래밍에 특화된 문서 편집 도구(Github에서 개발)
PyCharm	다양한 기능을 갖춘 파이썬 전용 개발 도구(IDE)
Visual Studio Code	MS에서 제공하는 코드 전용 에디터, 다양한 기능으로 최근 Atom과 함께 가장 많은 관심을 받고 있음

[주요 코드 편집기와 IDE]



■ 파이썬 인터프리터 설치: Anaconda

1. 우선 Anaconda 다운로드 페이지(https://www.anaconda.com/download)에 접속하여 해당 운영체제를 클릭해준다.





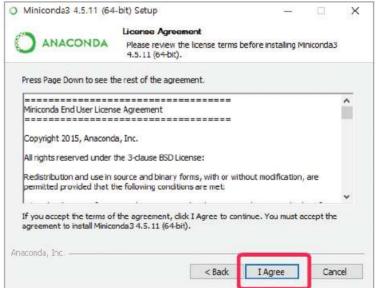
- 파이썬 인터프리터 설치: Anaconda
- 2. 버전에 맞추어 [Download] 버튼을 클릭해준다.



■ 파이썬 인터프리터 설치: Anaconda

③ 다운로드한 인스톨러를 실행하고 [Next]를 클릭한 후, 다음 화면에서 [I Agree]를 클릭한다.

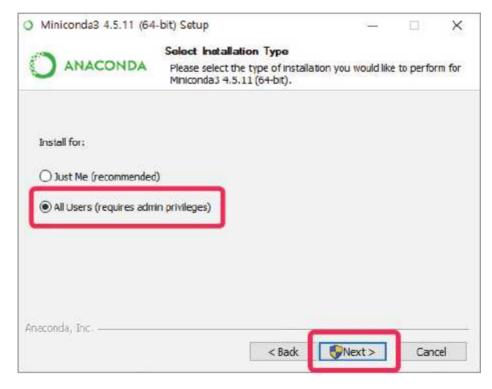




[Miniconda 설치 진행]



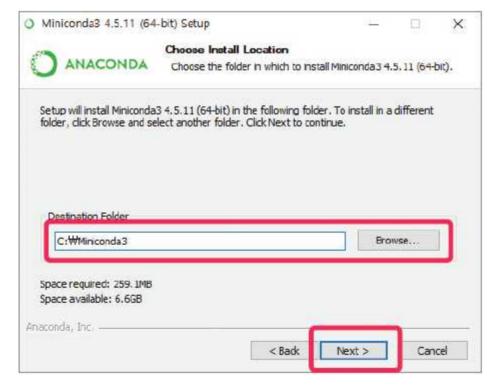
- 파이썬 인터프리터 설치: Anaconda
- 4. 인스톨 타입 설정 창에서 'All Users'를 선택하고, [Next]를 클릭한다.



[Miniconda 인스톨 타입 설정]



- 파이썬 인터프리터 설치: Anaconda
- 5. 인스톨 경로 설정 창에서 [Browse]를 클릭하여 적절한 경로를 지정하고, [Next]를 클릭한다.

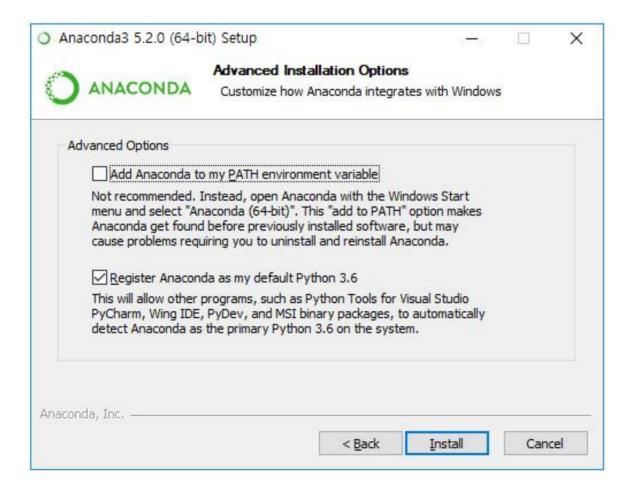


[Miniconda 인스톨 경로 설정]



■ 파이썬 인터프리터 설치: Anaconda

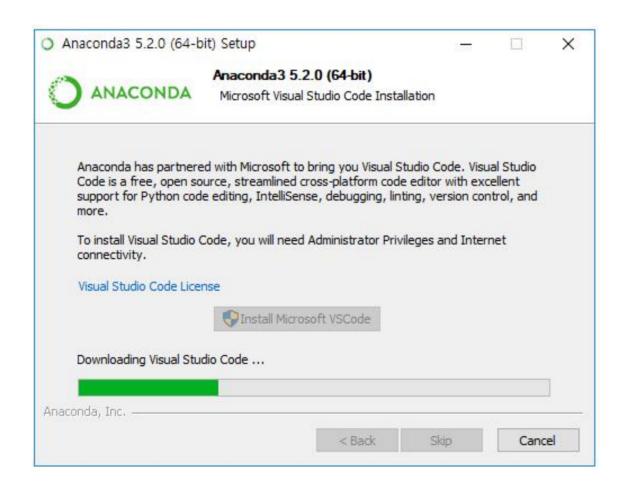
6. 인스톱 옵션 설정 창에서 기본값으로, [Install]을 클릭한다.





0

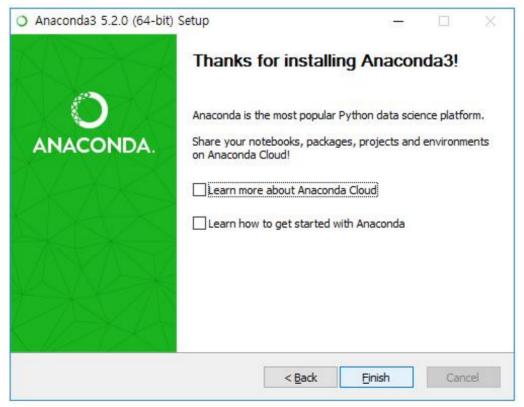
- 파이썬 인터프리터 설치: Anaconda
- 7. 아나콘다 프로그램 설치 진행 중





■ 파이썬 인터프리터 설치: Anaconda

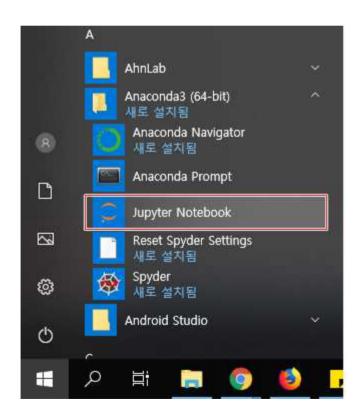
8. 설치가 완료되면 <윈도> + <R> 키를 누른 후, 실행 창에서 'cmd'를 입력하고 <Enter> 키를 누른다. cmd 창이 열리면 'python'을 입력한 후 <Enter> 키를 눌러 파이썬이 작동하는지 확인한다.





•

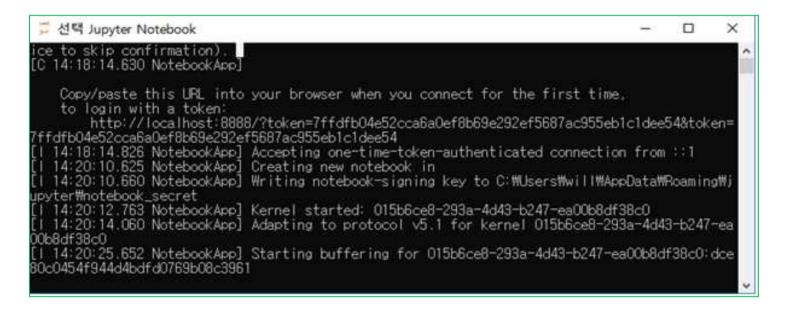
- 파이썬 인터프리터 설치: Jupyter Notebook
- 1. 주피터 노트북은 프로그램을 작성하면서 주석도 기입하고 code의 line별로 실행도 할 수 있는 유용한 툴이며, 아나콘다를 설치하면 주피터 노트북도 함께 설치가 된다.

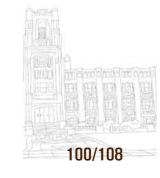




•

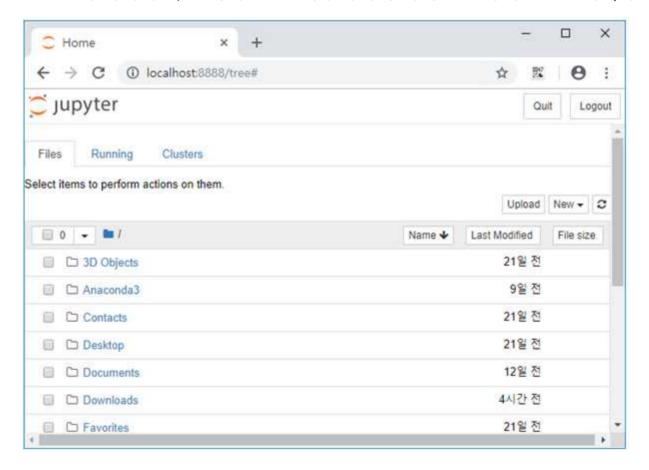
- 파이썬 인터프리터 설치: Jupyter Notebook
- 2. [시작] 메뉴에서 주피터 노트북을 찾아서 선택해준다.





•

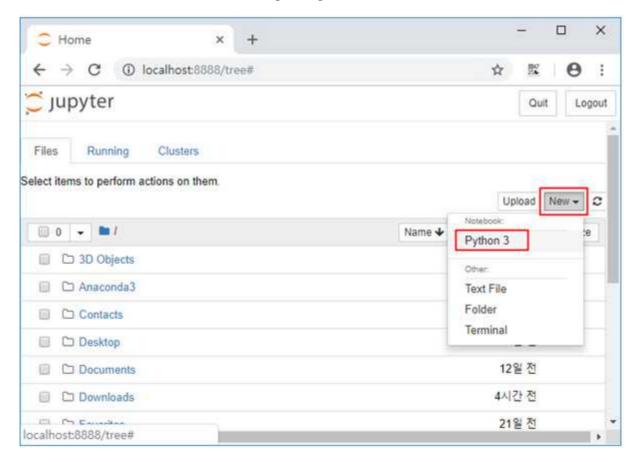
- 파이썬 인터프리터 설치: Jupyter Notebook
- 3. 콘솔창이 나타나고, 잠시 후 웹 브라우저에 주피터 노트북이 실행된다. (기본 웹브라우저 지정)





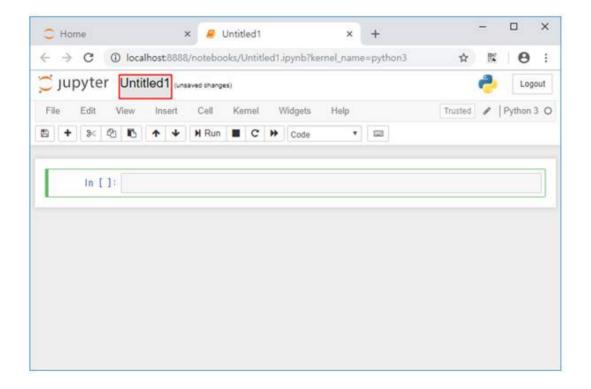
•

- 파이썬 인터프리터 설치: Jupyter Notebook
- 4. 주피터 노트북은 우측 상단의 [New] 버튼을 클릭하면 메뉴창이 보인다.



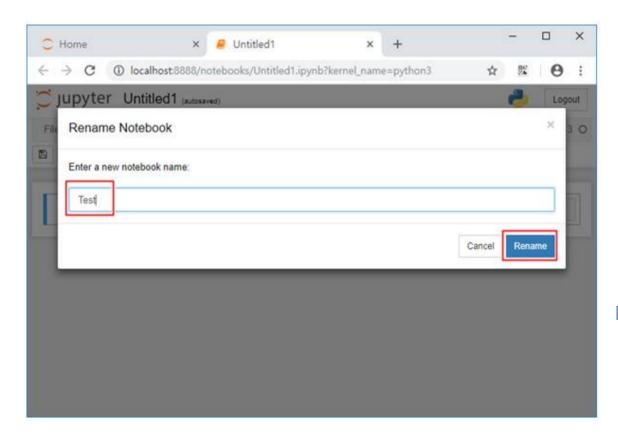


- 파이썬 인터프리터 설치: Jupyter Notebook
- 프로젝트 인 경우 폴더를 생성한 후 파일을 작성한다.
 여기서는 [Python 3] 항목을 선택한다.





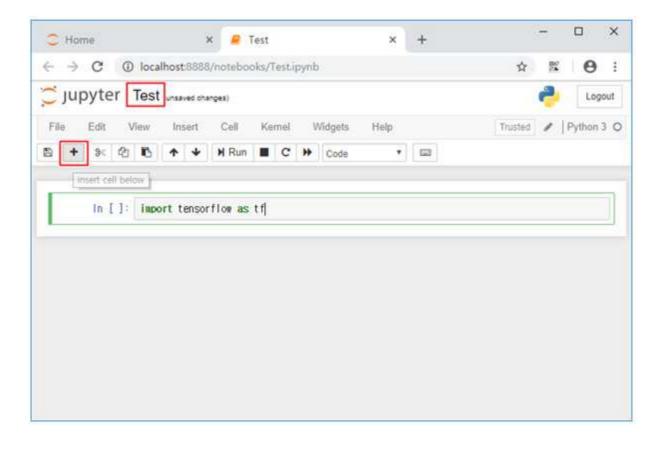
- 파이썬 인터프리터 설치: Jupyter Notebook
- 6. 주피터 노트북 화면 상단에 있는 [Untitled1] 이름을 클릭하여 파일명을 변경할 수 있다.





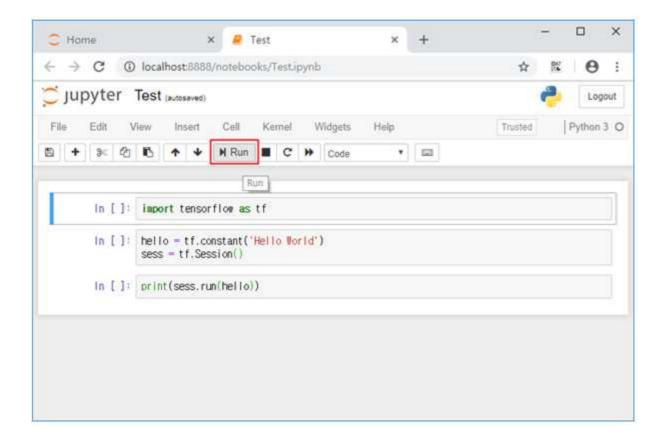
•

- 파이썬 인터프리터 설치: Jupyter Notebook
- 7. 주피터 노트북에서 코딩을 시작할 때 라인이 하나밖에 없다. [+] 표시를 클릭해서 라인을 하나 씩 추가해주며 코딩을 진행한다.





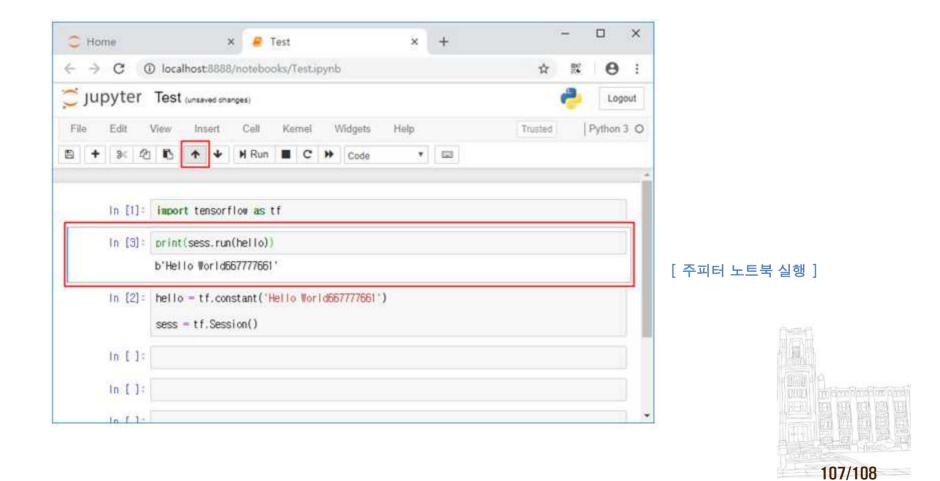
- 파이썬 인터프리터 설치: Jupyter Notebook
- 첫번째 블록을 선택하고 [Run] 버튼을 클릭하면 실행이 되면서 진행된다.
 [Run] 버튼을 클릭할 때마다 한 블럭씩 실행을 한다.

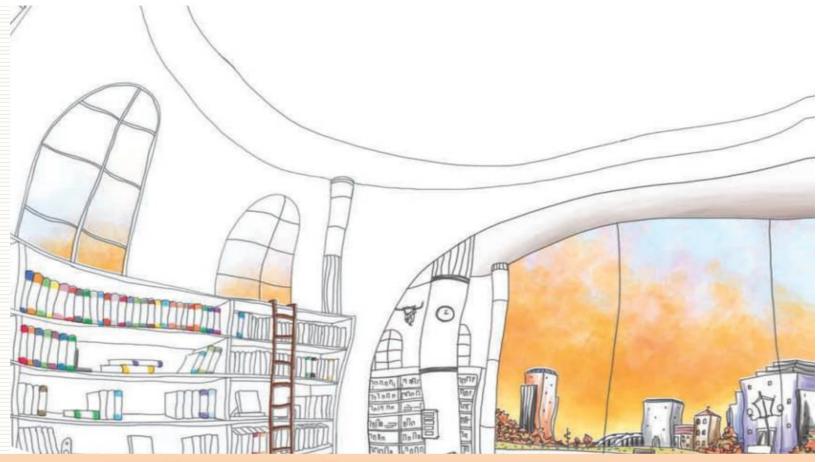


「주피터 노트북 실행]



- 파이썬 인터프리터 설치: Jupyter Notebook
- 9. 이동할 라인을 선택하고 화살표를 이용해서 위 / 아래 로 코드의 위치를 변경할 수 있다.





Thank You!

파이썬

