

1장. 파이썬이란 무엇인가?

- I. 파이썬이란?
- II. 파이썬의 특징
- III. 파이썬으로 무엇을 할 수 있을까?
- IV. 파이썬 설치하기
- V. 파이썬 둘러보기
- VI. 파이썬과 에디터

- ❖ 1990년 귀도 반 로섬(Guido Van Rossum)이 개발한 인터프리터 언어
 - 자신이 좋아하는 코미디쇼인 '몬티 파이썬의 날아다니는 서커스'에서 이름을 따왔음
 - 고대 신화에 나오는 파르나소스 산의 동굴에 살던 큰 뱀
- ❖ 컴퓨터 프로그래밍 교육을 위해 많이 사용하지만, 기업의 실무를 위해서도 많이 사용하는 언어임
 - 구글에서 만들어진 소프트웨어의 50% 이상이 파이썬임
 - 온라인 사진 공유 서비스 인스타그램(Instagram)
 - 파일 동기화 서비스 드롭박스(Dropbox)
- ❖ 공동 작업과 유지 보수가 매우 쉽고 편함
- ❖ 국내에서도 사용자 층이 더욱 넓어지고 있음

❖ 파이썬은 인간다운 언어임

- 사람이 생각하는 방식을 그대로 표현할 수 있는 언어
- `if 4 in [1,2,3,4]: print (" 4가 있습니다 ")`

❖ 파이썬은 문법이 쉬워 빠르게 배울 수 있음

- 문법 자체가 아주 쉽고 간결하며, 사람의 사고 체계와 매우 닮아 있음
- 파이썬을 공부한 지 단 하루 만에 자신이 원하는 프로그램을 작성 (Eric Raymond)
- 1주일이면 충분 (프로그래밍 유 경험자)

❖ 파이썬은 무료이지만 강력함

- 사용료 걱정 없이 언제 어디서든 파이썬을 다운로드하여 사용 (Open Source)
- 파이썬과 C는 찰떡궁합
 - ✓ 프로그램 전반적인 뼈대는 파이썬으로 만들고 빠른 실행속도를 필요로 하는 부분은 C로 만들어서 파이썬 프로그램 안에 포함
 - ✓ 파이썬 라이브러리들 중에는 C로 만들어진 것도 많음

❖ 파이썬은 간결함

- 펄(Perl) 프로그램 언어가 100가지 방법으로 하나의 일을 처리할 수 있다면, 파이썬은 가장 좋은 방법 1가지만 이용하는 것을 선호
- 실행이 되게 하려면 꼭 줄을 맞추어야 됨

```
# simple.py
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")
```

❖ 파이썬은 프로그래밍을 즐기게 해 줌

- 다른 것에 신경 쓸 필요 없이 내가 하고자 하는 부분에만 집중할 수 있게 해줌

❖ 파이썬은 개발 속도가 빠름

- 파이썬의 엄청난 개발 속도를 두고 유행처럼 퍼진 말
 - ✓ "Life is too short, You need python." (인생은 너무 짧으니 파이썬이 필요해.)

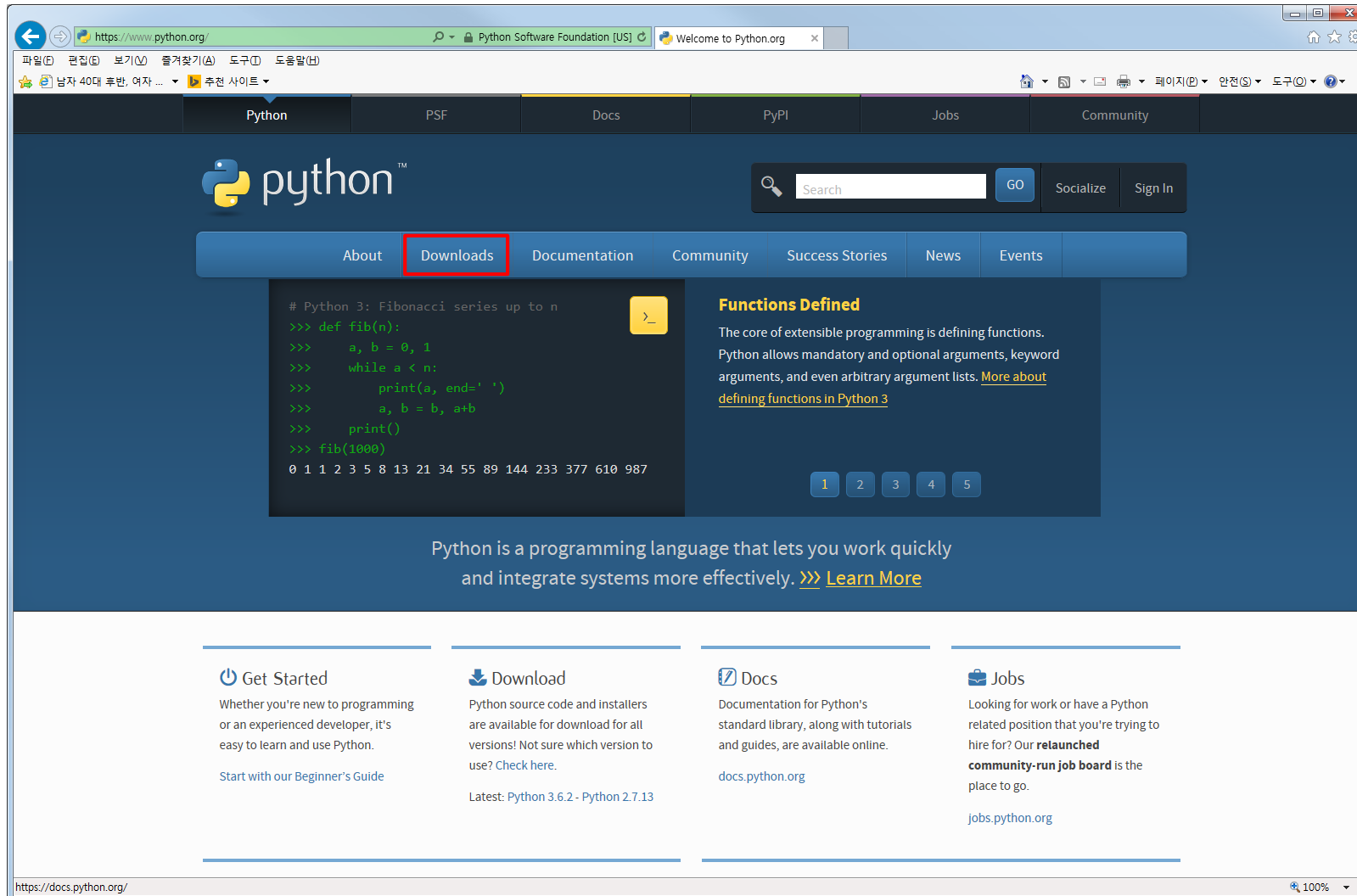
❖ 파이썬으로 할 수 있는 일

- 시스템 유틸리티 제작
- GUI 프로그래밍
- C/C++와의 결합
- 웹 프로그래밍
- 수치 연산 프로그래밍
- 데이터베이스 프로그래밍
- 데이터 분석, 사물 인터넷

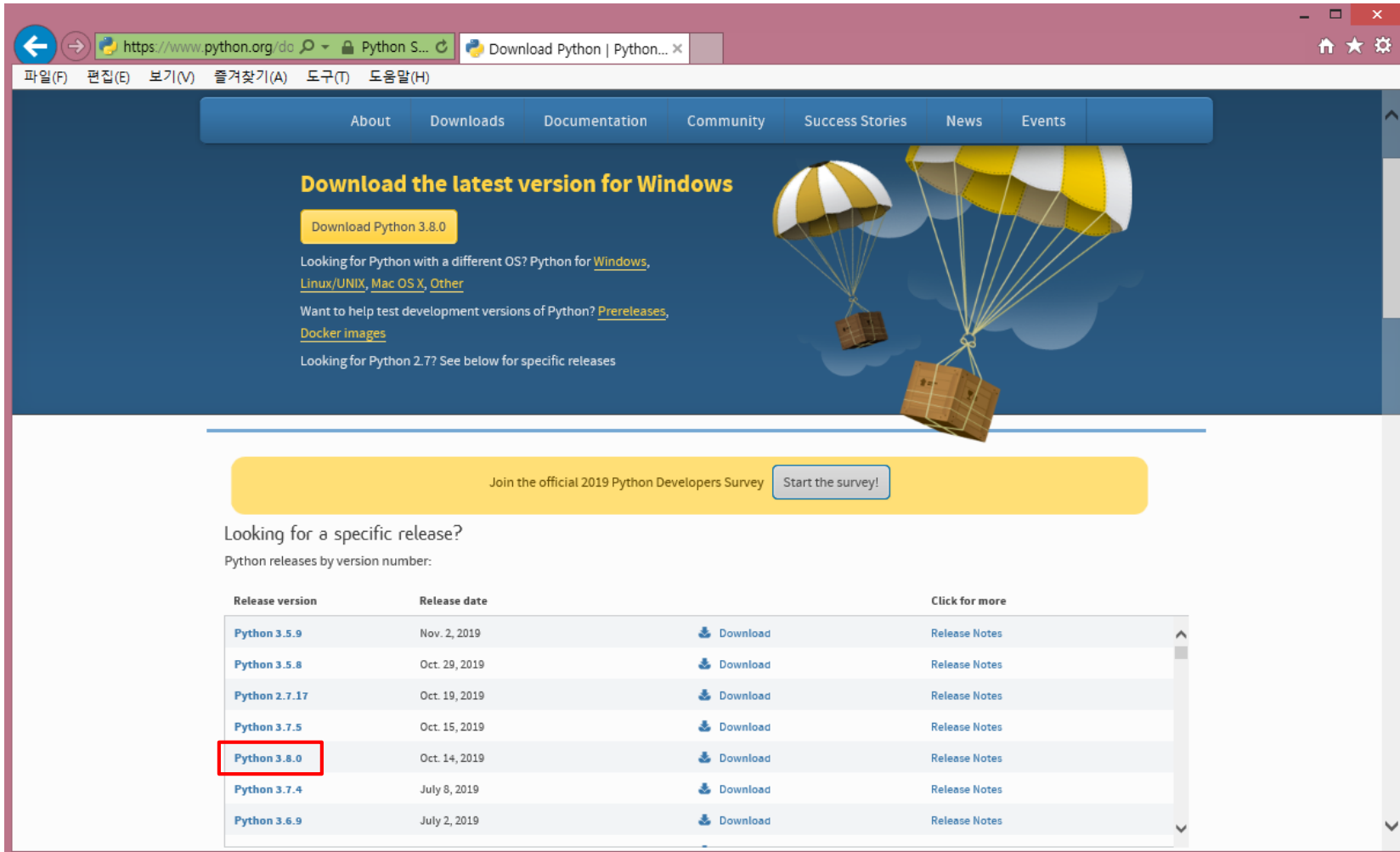
❖ 파이썬으로 할 수 없는 일

- 시스템과 밀접한 프로그래밍 영역
- 모바일 프로그래밍

- ❖ 파이썬 홈페이지(www.python.org)에서 Downloads 버튼을 클릭함



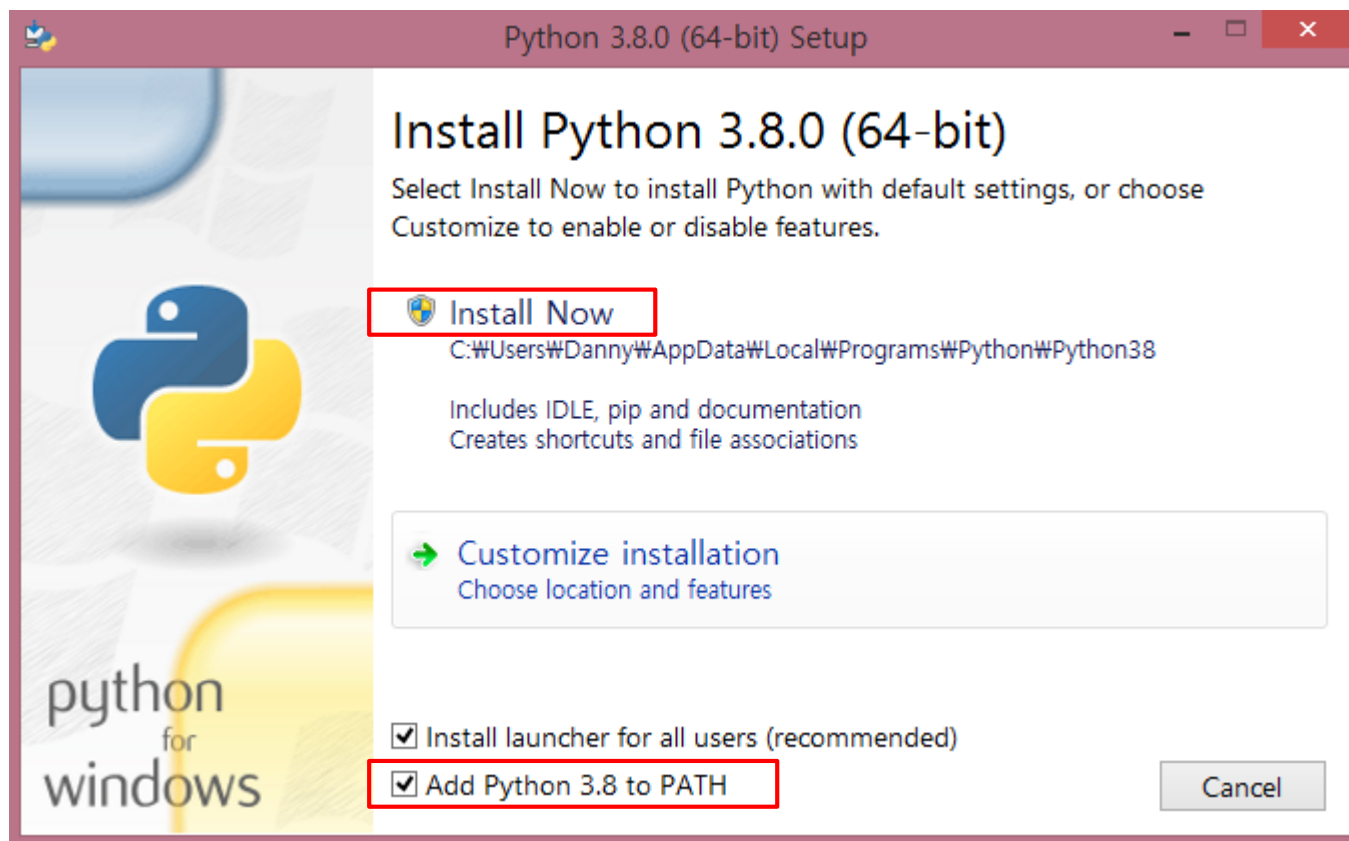
- ❖ Python 3.8.x로 시작하는 가장 최신버전을 선택함
 - Window x86-64 executable installer을 다운로드 함



The screenshot shows the Python.org website's download page for Windows. The main heading is "Download the latest version for Windows" with a prominent "Download Python 3.8.0" button. Below this, there are links for other operating systems and development resources. A survey banner is also visible. At the bottom, a table lists various Python releases, with "Python 3.8.0" highlighted by a red rectangular box.

Release version	Release date	Click for more
Python 3.5.9	Nov. 2, 2019	Download Release Notes
Python 3.5.8	Oct. 29, 2019	Download Release Notes
Python 2.7.17	Oct. 19, 2019	Download Release Notes
Python 3.7.5	Oct. 15, 2019	Download Release Notes
Python 3.8.0	Oct. 14, 2019	Download Release Notes
Python 3.7.4	July 8, 2019	Download Release Notes
Python 3.6.9	July 2, 2019	Download Release Notes

- ❖ 파이썬이 어느 곳에서든지 실행될 수 있도록 'Add Python 3.8 to PATH' 선택 후, 'Install Now'를 선택하면 설치가 진행됨

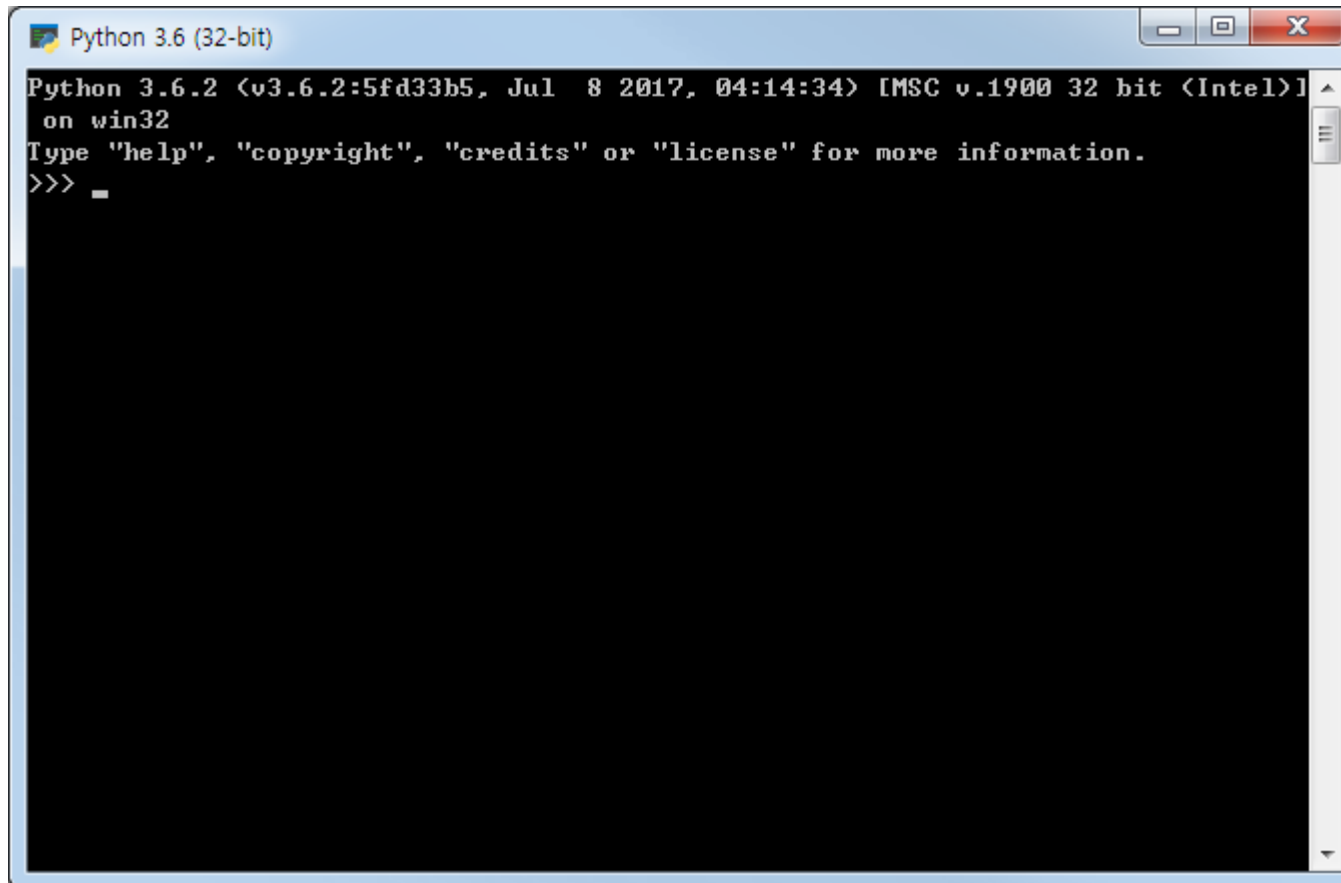


- ❖ 설치가 종료되면 Close 버튼을 클릭함



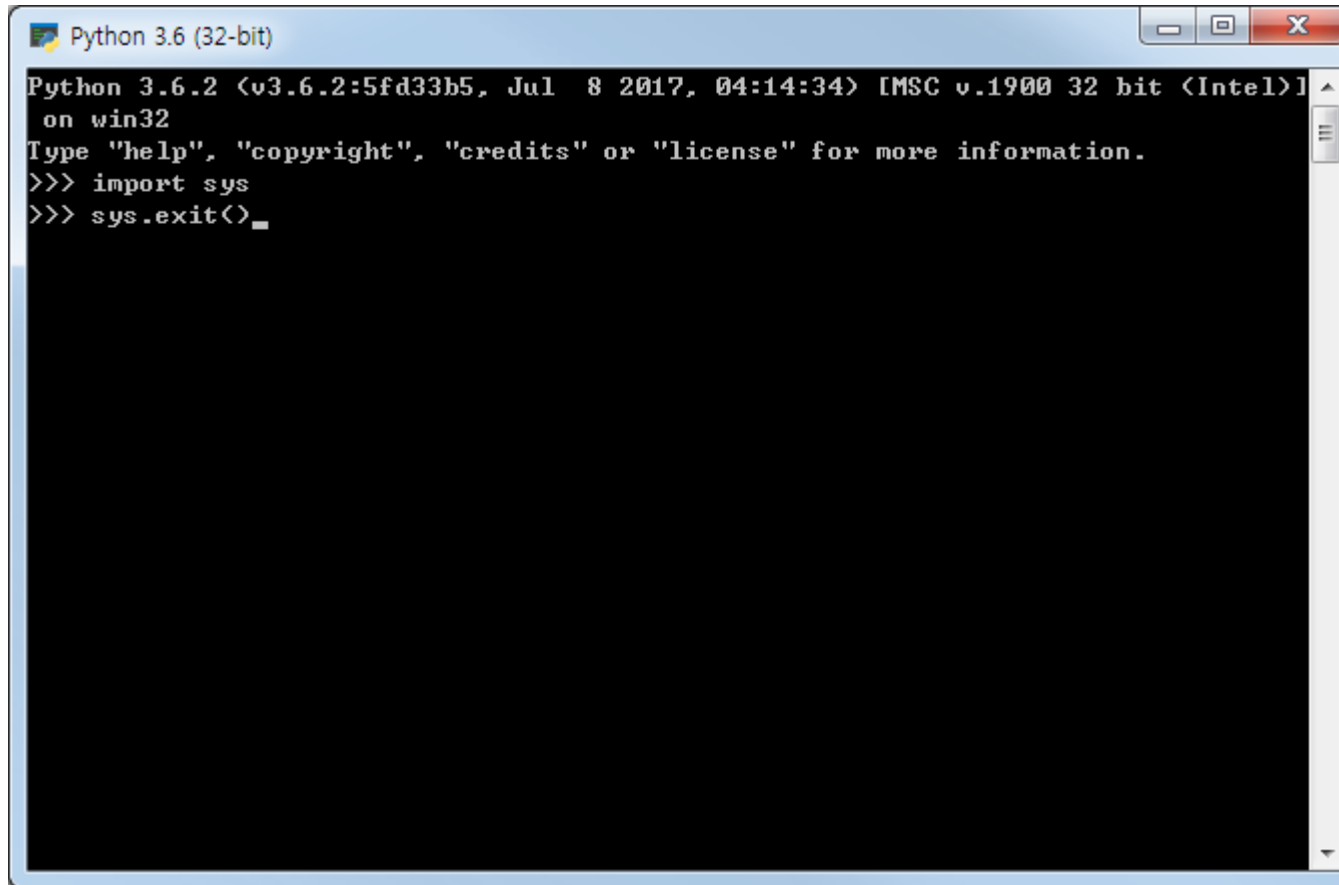
❖ 파이썬 기초 실습 준비하기

- [시작] 메뉴에서 [모든 프로그램 → Python 3.8 → Python 3.8(64-bit)] 선택
- 아래의 화면을 대화형 인터프리터라고 함(Python shell)

A screenshot of a Windows application window titled "Python 3.6 (32-bit)". The window has a standard Windows title bar with minimize, maximize, and close buttons. The main content area has a black background with white text. The text displayed is: "Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)] on win32", followed by "Type 'help', 'copyright', 'credits' or 'license' for more information.", and then the prompt ">>> " with a cursor. A vertical scrollbar is visible on the right side of the text area.

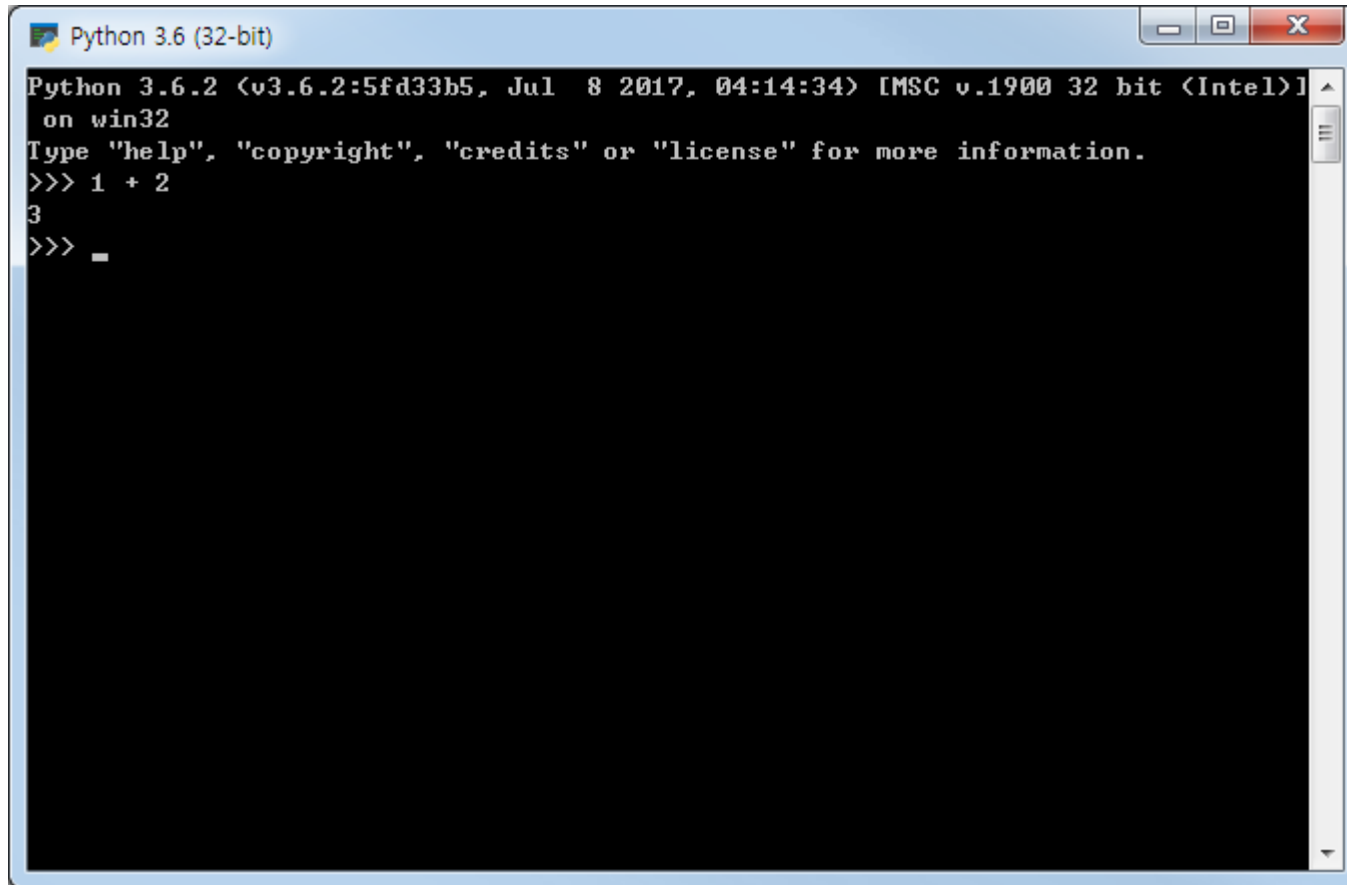
```
Python 3.6 (32-bit)
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

- 파이썬을 종료하려면 [Ctrl + Z → Enter] 또는 아래의 화면과 같이 sys 모듈을 사용하여 종료할 수 있음



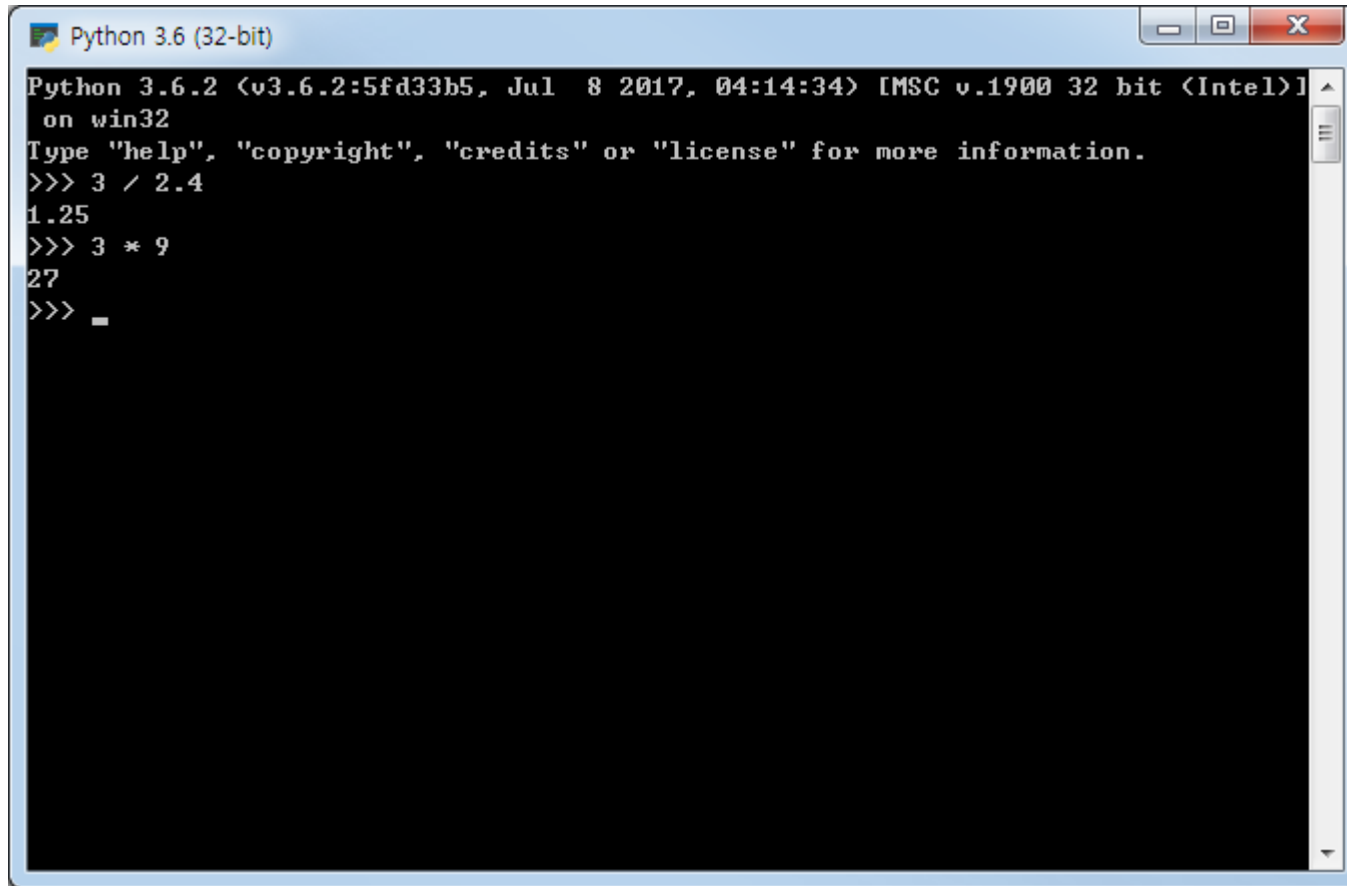
```
Python 3.6 (32-bit)
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import sys
>>> sys.exit()
```

- ❖ 파이썬 기초 문법 따라 해보기
 - 사칙연산 - 더하기



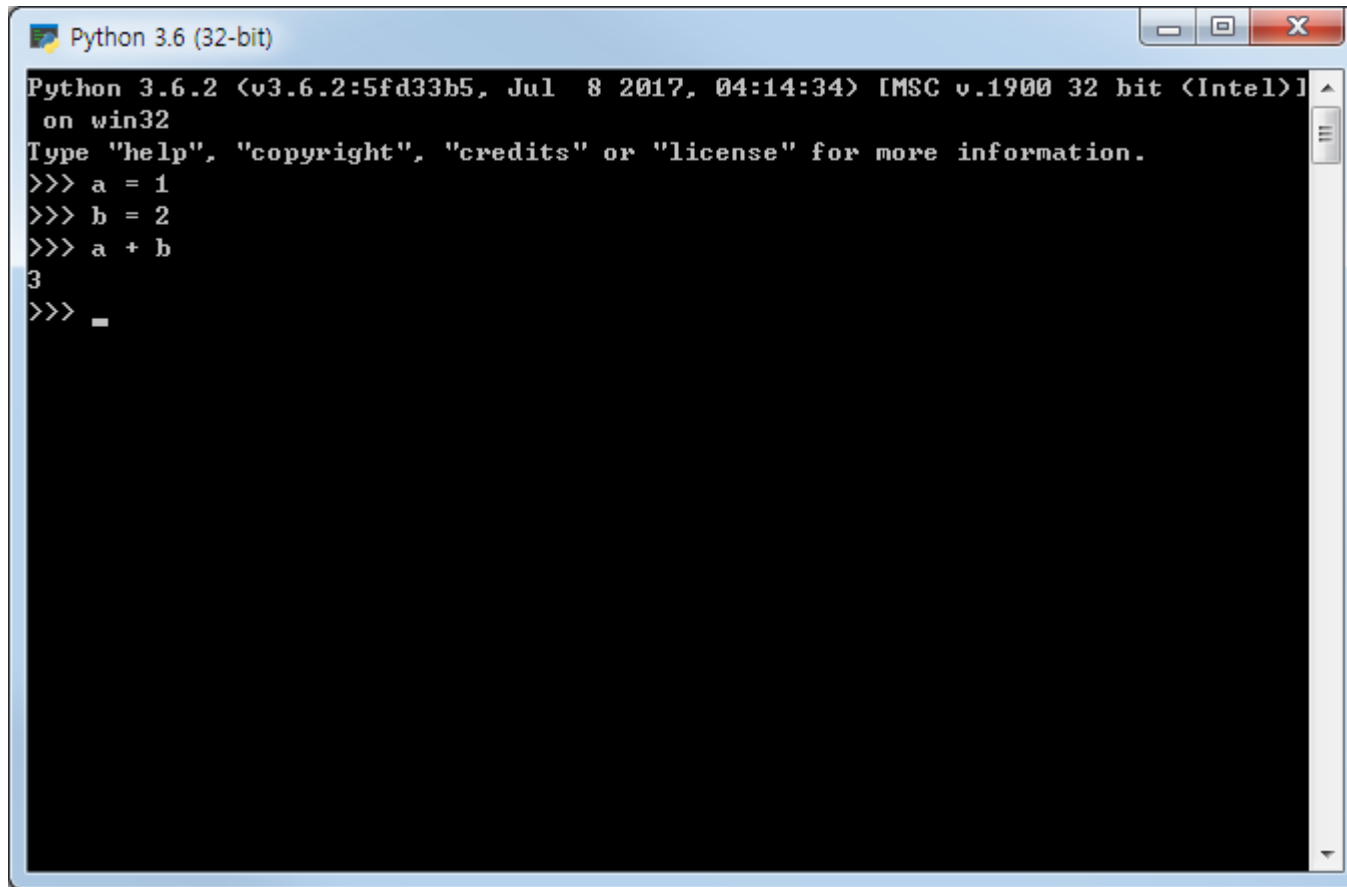
```
Python 3.6 (32-bit)
Python 3.6.2 <v3.6.2:5fd33b5, Jul  8 2017, 04:14:34> [MSC v.1900 32 bit <Intel>]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> 1 + 2
3
>>> _
```

- 사칙연산 - 나눗셈과 곱셈



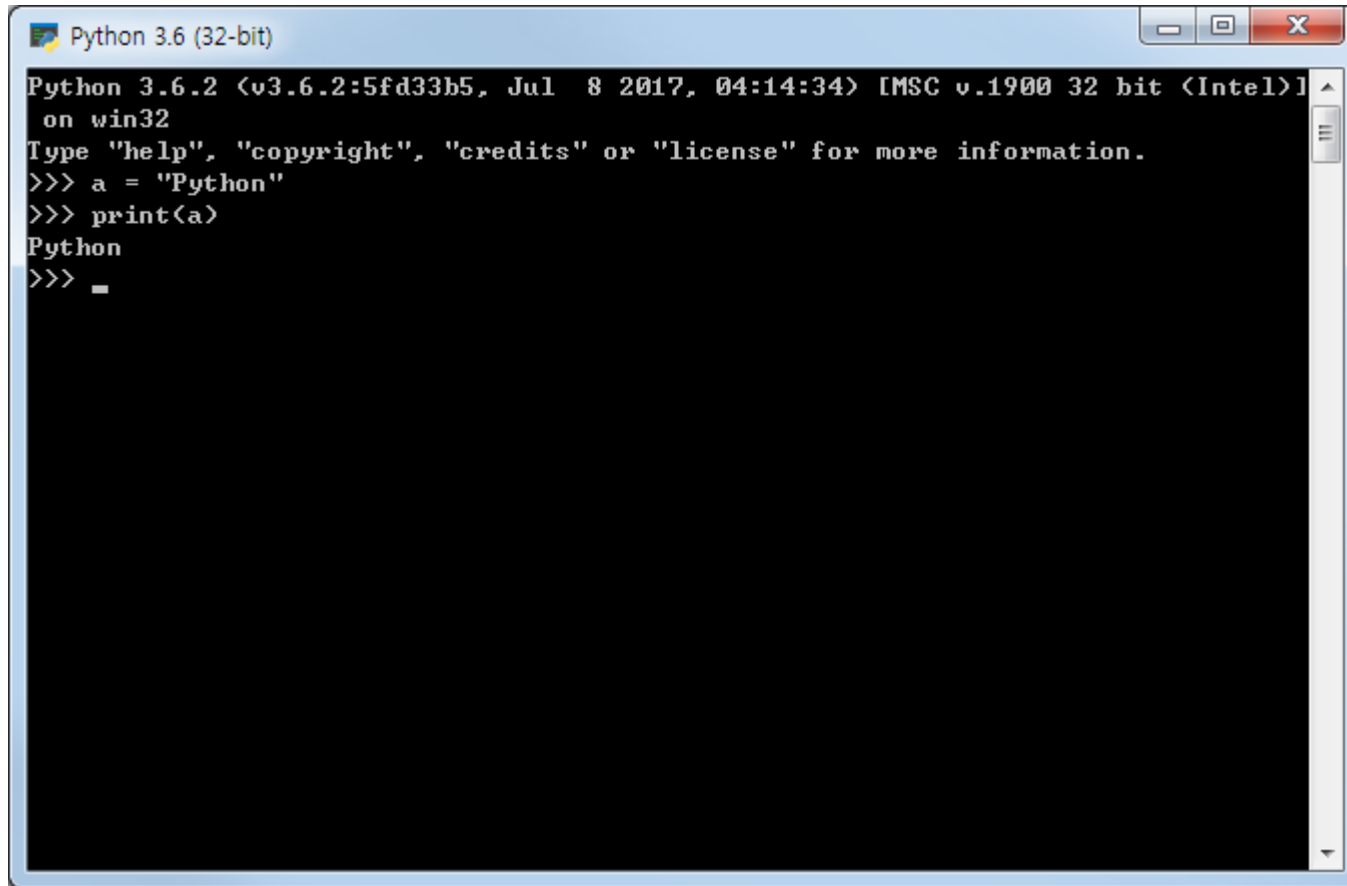
```
Python 3.6 (32-bit)
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> 3 / 2.4
1.25
>>> 3 * 9
27
>>> _
```


❖ 변수에 숫자 대입하고 계산하기



```
Python 3.6 (32-bit)
Python 3.6.2 <v3.6.2:5fd33b5, Jul 8 2017, 04:14:34> [MSC v.1900 32 bit <Intel>]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> a = 1
>>> b = 2
>>> a + b
3
>>> _
```

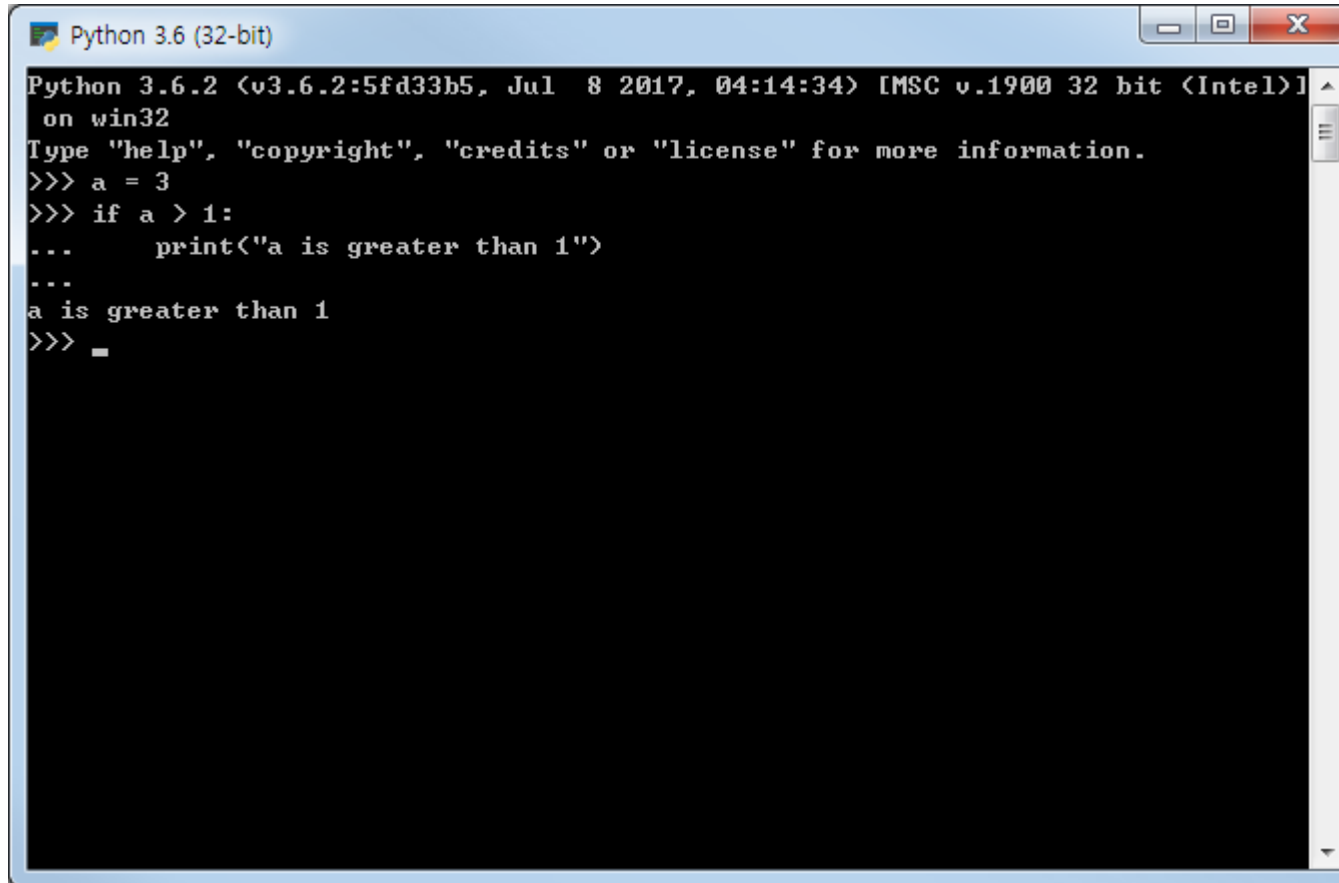
❖ 변수에 문자 대입하고 출력하기



```
Python 3.6 (32-bit)
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> a = "Python"
>>> print(a)
Python
>>> _
```

❖ 조건문 if

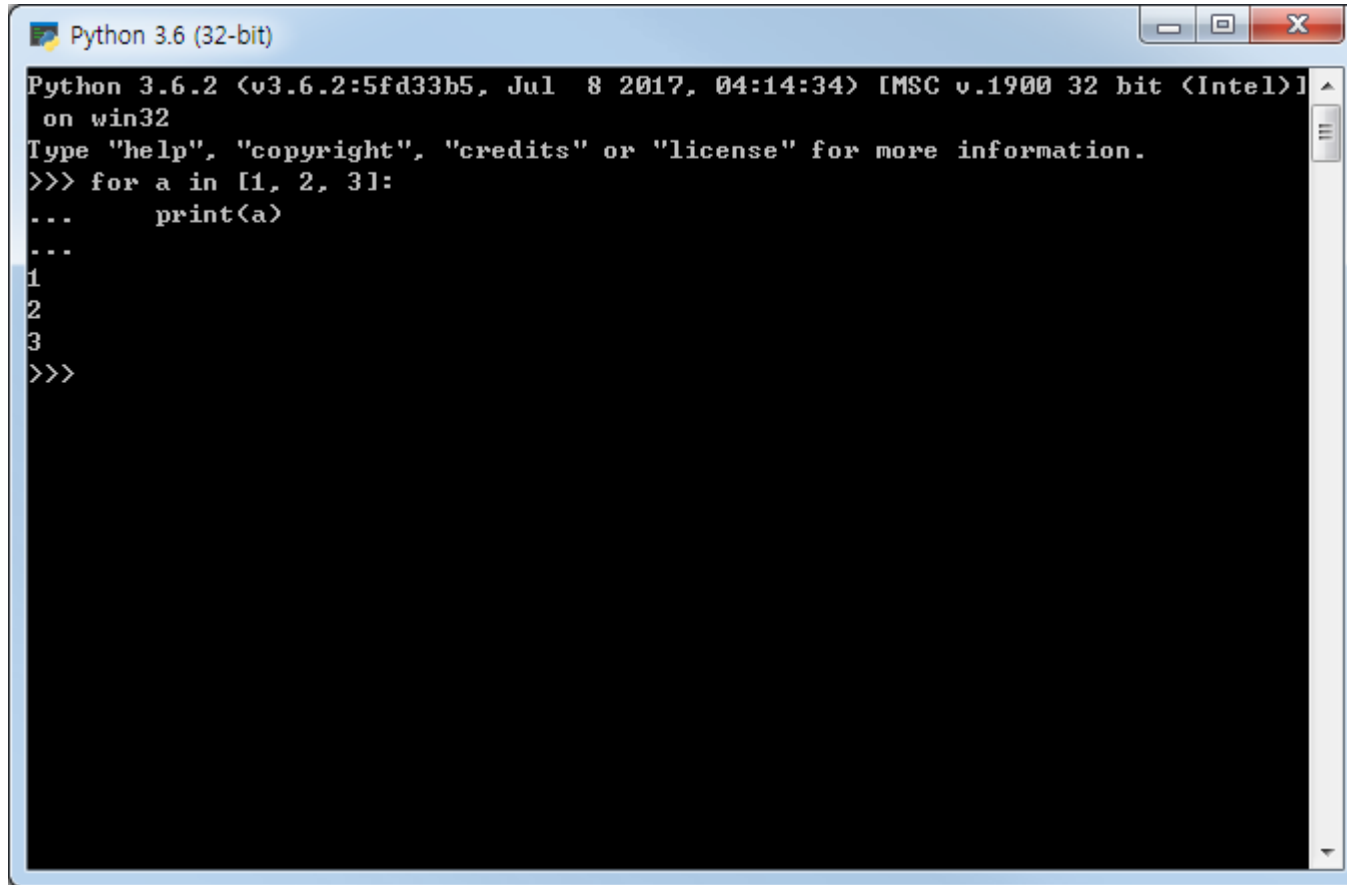
- 아래의 예제는 a가 1보다 크면 'a is greater than 1'이라는 문장 출력
✓ '...'는 아직 문장이 끝나지 않았음을 의미



```
Python 3.6 (32-bit)
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> a = 3
>>> if a > 1:
...     print("a is greater than 1")
...
a is greater than 1
>>> _
```

❖ 반복문 for

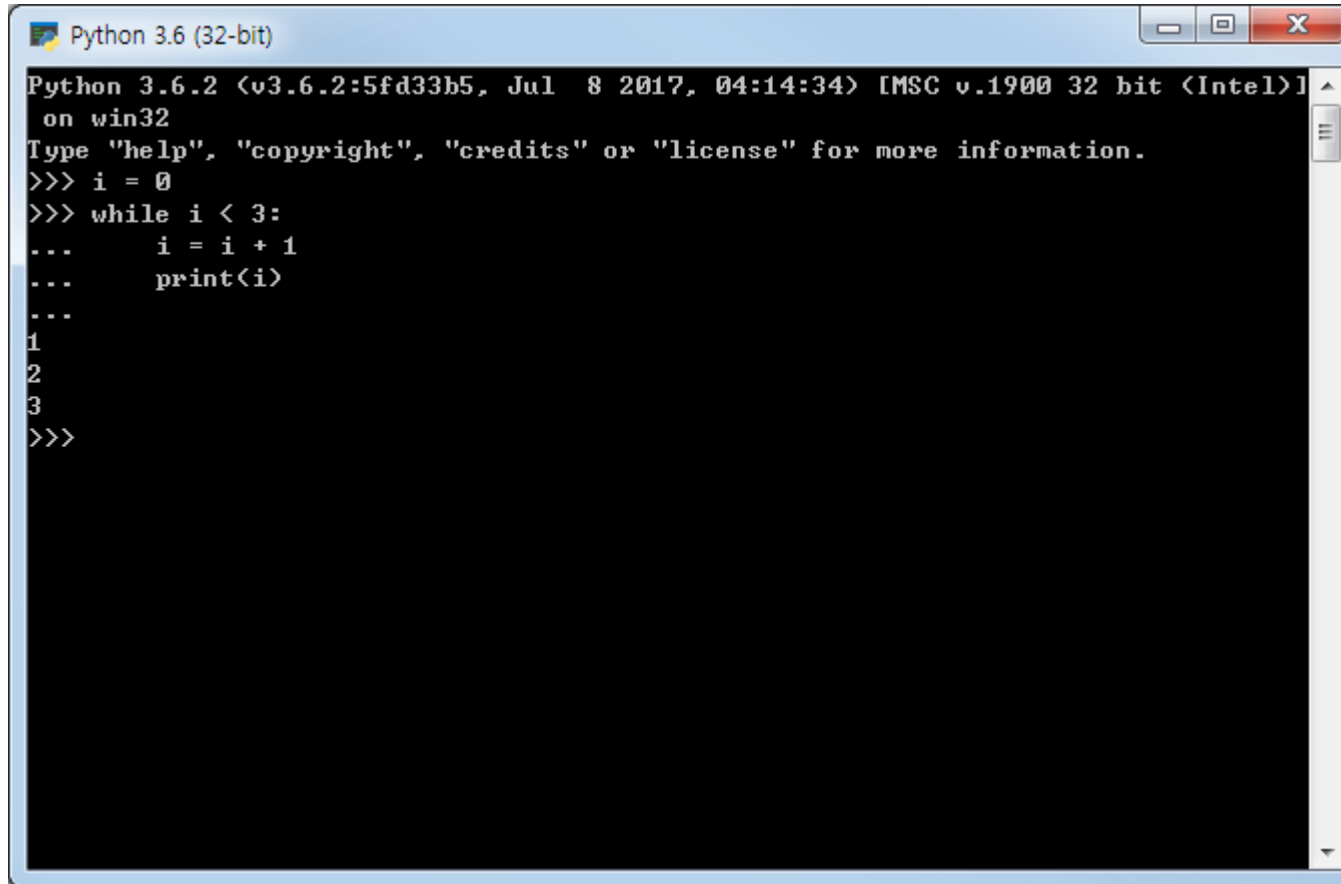
- 아래의 예제는 for문을 이용하여 [1, 2, 3] 안의 값들을 하나씩 출력



```
Python 3.6 (32-bit)
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> for a in [1, 2, 3]:
...     print(a)
...
1
2
3
>>>
```

❖ 반복문 while

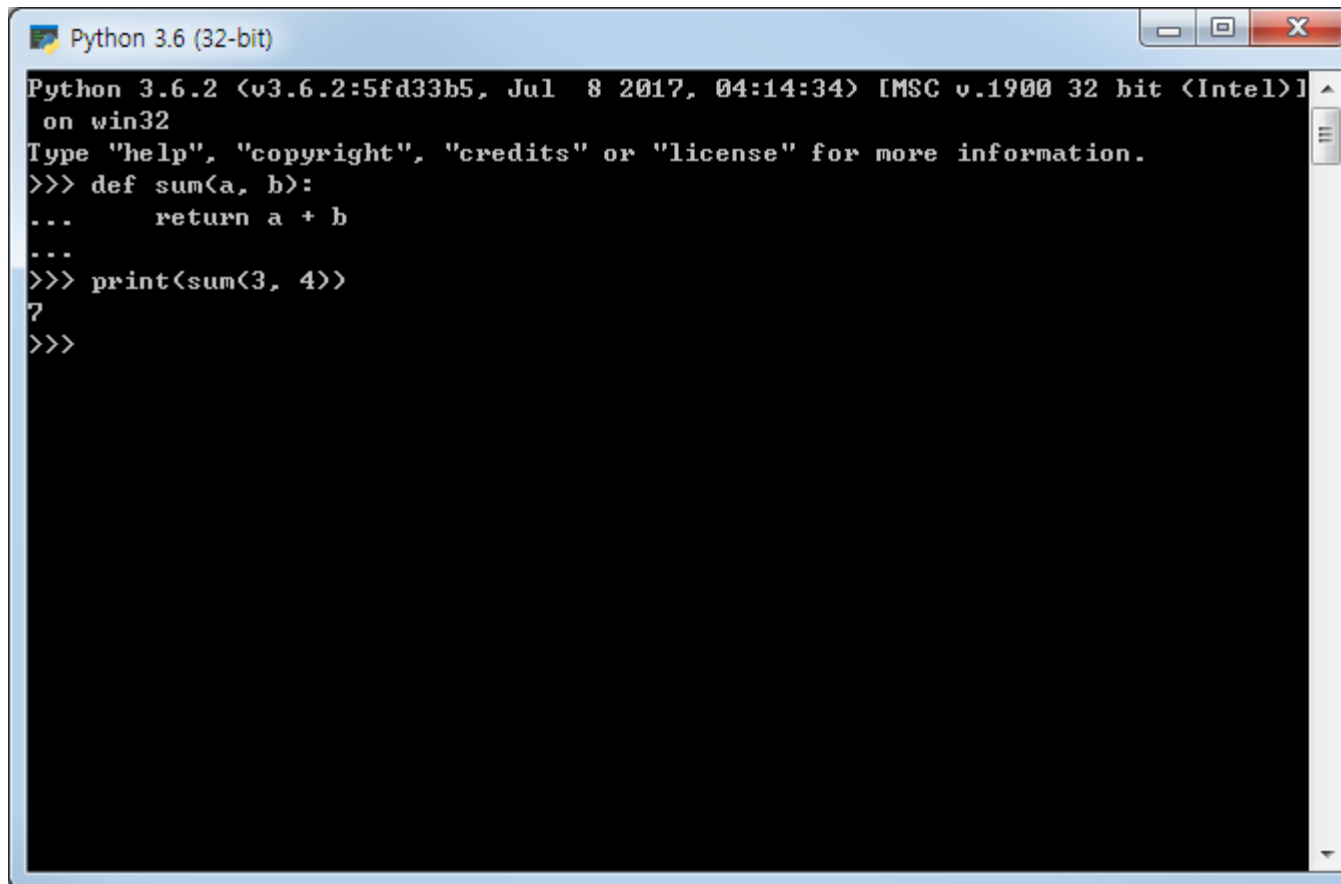
- 아래의 예제는 i 값이 3보다 작은 동안 $i = i + 1$ 과 $\text{print}(i)$ 를 수행



```
Python 3.6 (32-bit)
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> i = 0
>>> while i < 3:
...     i = i + 1
...     print(i)
...
1
2
3
>>>
```

❖ 함수

- def는 함수를 만들 때 사용하는 예약어임
- 아래의 예제는 `sum(a, b)`에서 `a, b`는 입력값이고, `a + b`는 결과 값임
- 즉 3, 4가 입력으로 들어오면 `3+4`를 수행하고 결과값 7을 돌려 줌

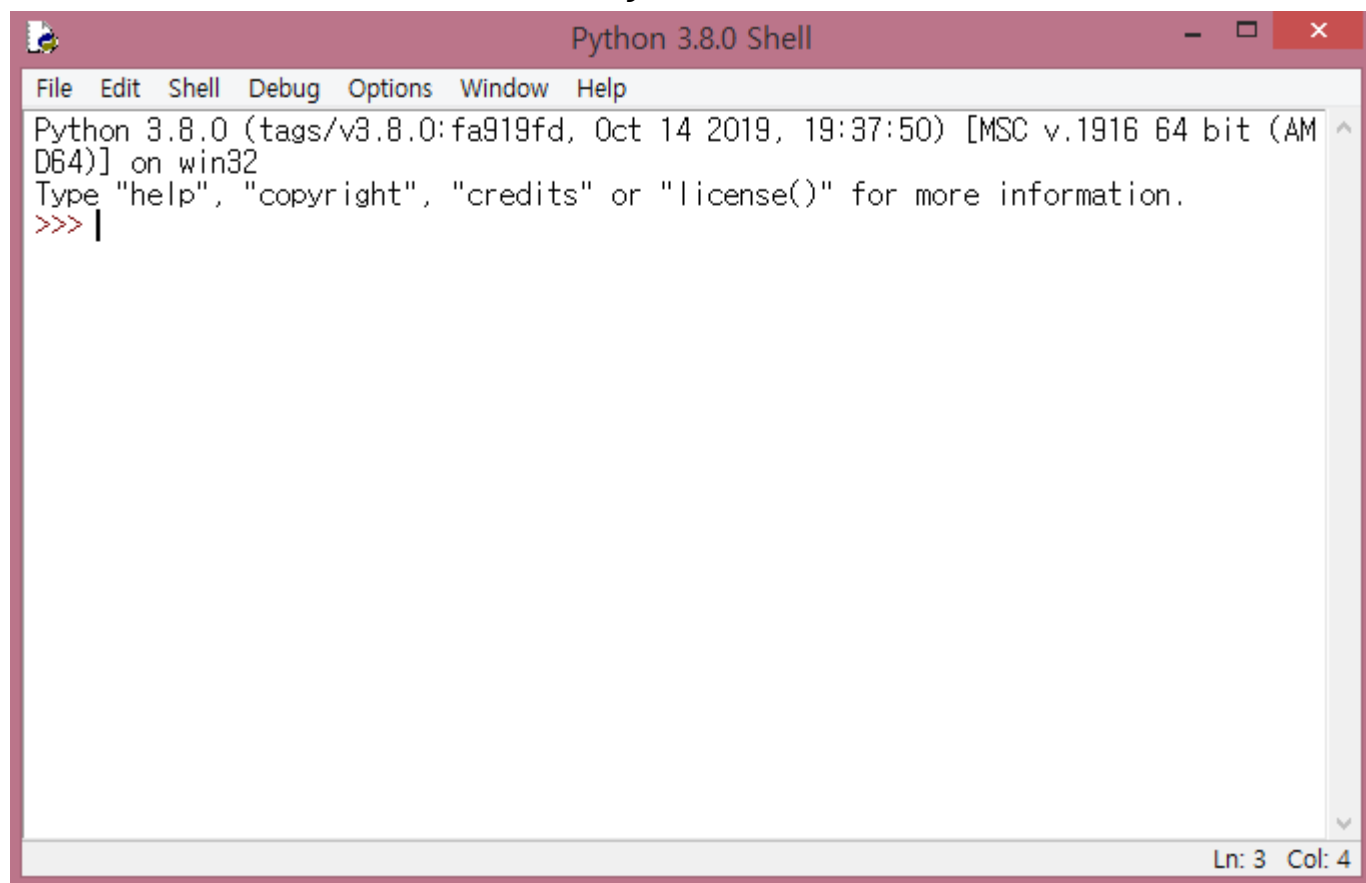


```
Python 3.6 (32-bit)
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> def sum(a, b):
...     return a + b
...
>>> print(sum(3, 4))
7
>>>
```

- ❖ 에디터란 소스 코드를 편집할 수 있는 프로그래밍 툴을 말함
- ❖ 파이썬 프로그램을 작성할 수 있는 여러 가지 에디터
 - 파이썬 IDLE(Integrated Development and Learning Environment)
 - ✓ 파이썬 IDLE(아이들)은 파이썬을 설치할 때 기본으로 설치되는 에디터
 - 파이참(PyCharm)
 - ✓ 파이썬 전용 에디터로 소스 코드를 매끄럽게 작성할 수 있게 자동 완성, 문법 체크 등 편리한 기능을 제공함
 - ✓ 파이참 홈페이지(www.jetbrains.com/pycharm)에서 다운로드 함
 - 비주얼 스튜디오 코드
 - ✓ 파이썬 전용 에디터는 아니지만, 여러 가지 언어를 지원하는 에디터임

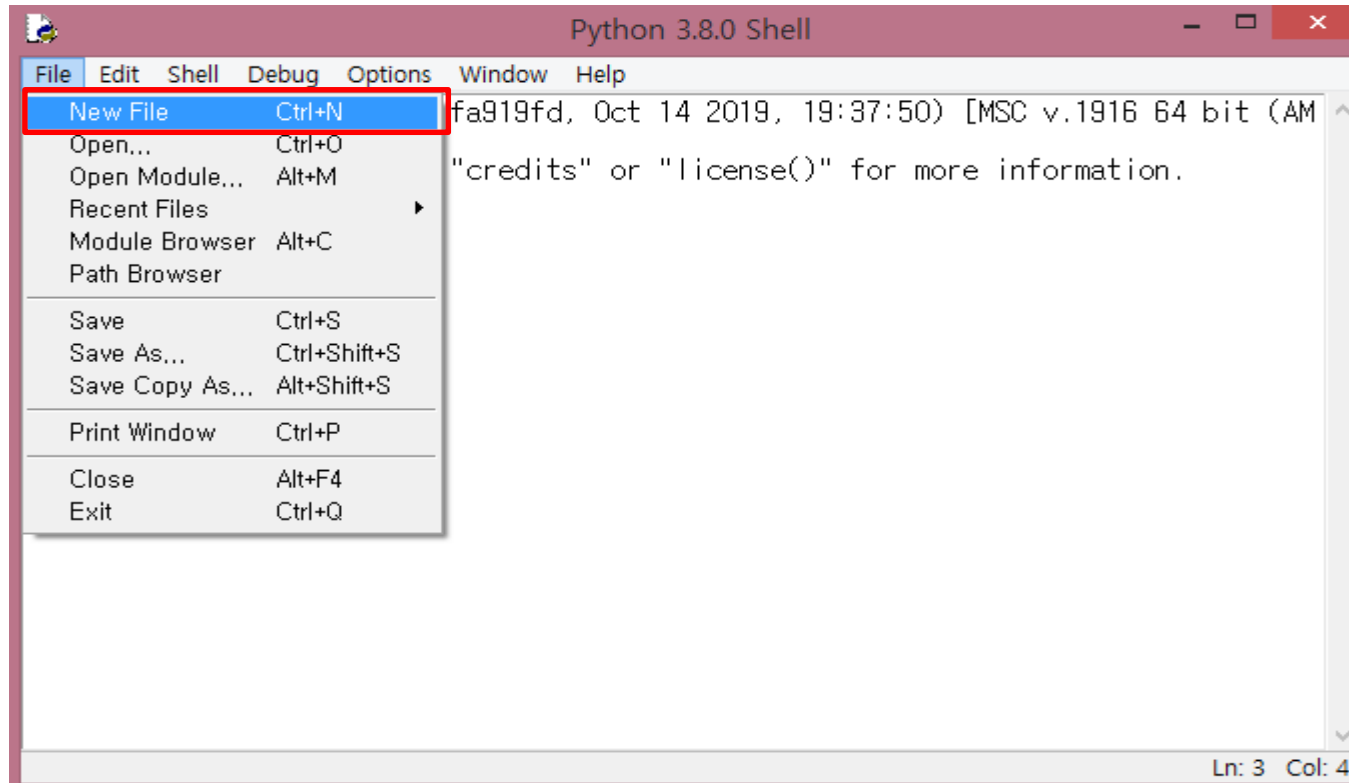
❖ IDLE로 파이썬 프로그램 작성하기

- 파이썬 IDLE(Integrated Development and Learning Environment)은 파이썬 프로그램 작성을 도와주는 통합 개발 환경임
- [시작 → 모든 프로그램 → Python 3.8 → IDLE]을 선택해 파이썬 IDLE을 실행함



```
Python 3.8.0 Shell
File Edit Shell Debug Options Window Help
Python 3.8.0 (tags/v3.8.0:fa919fd, Oct 14 2019, 19:37:50) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```


- IDLE은 두 가지 창으로 구성됨
 - ✓ **IDLE 셸 창(Shell Window):** IDLE 에디터에서 실행한 프로그램의 결과가 표시되는 창으로 파이썬 셸과 동일한 기능을 수행함.
 - ✓ **IDLE 에디터 창(Editor Window):** IDLE 에디터가 실행되는 창
- IDLE 에디터 실행하기
 - ✓ IDLE 셸 창 메뉴에서 [FILE → New File]을 선택해 실행함



- [IDLE 에디터] 파이썬 프로그램 작성하기

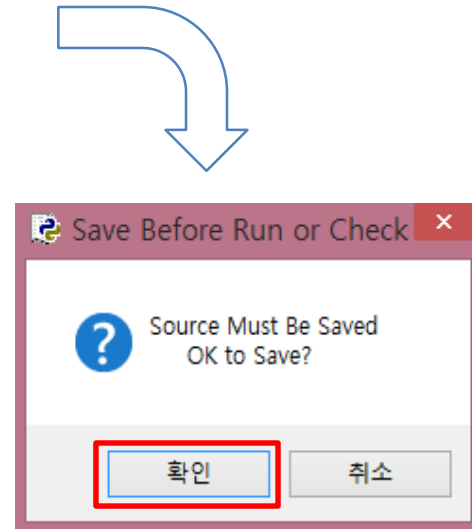
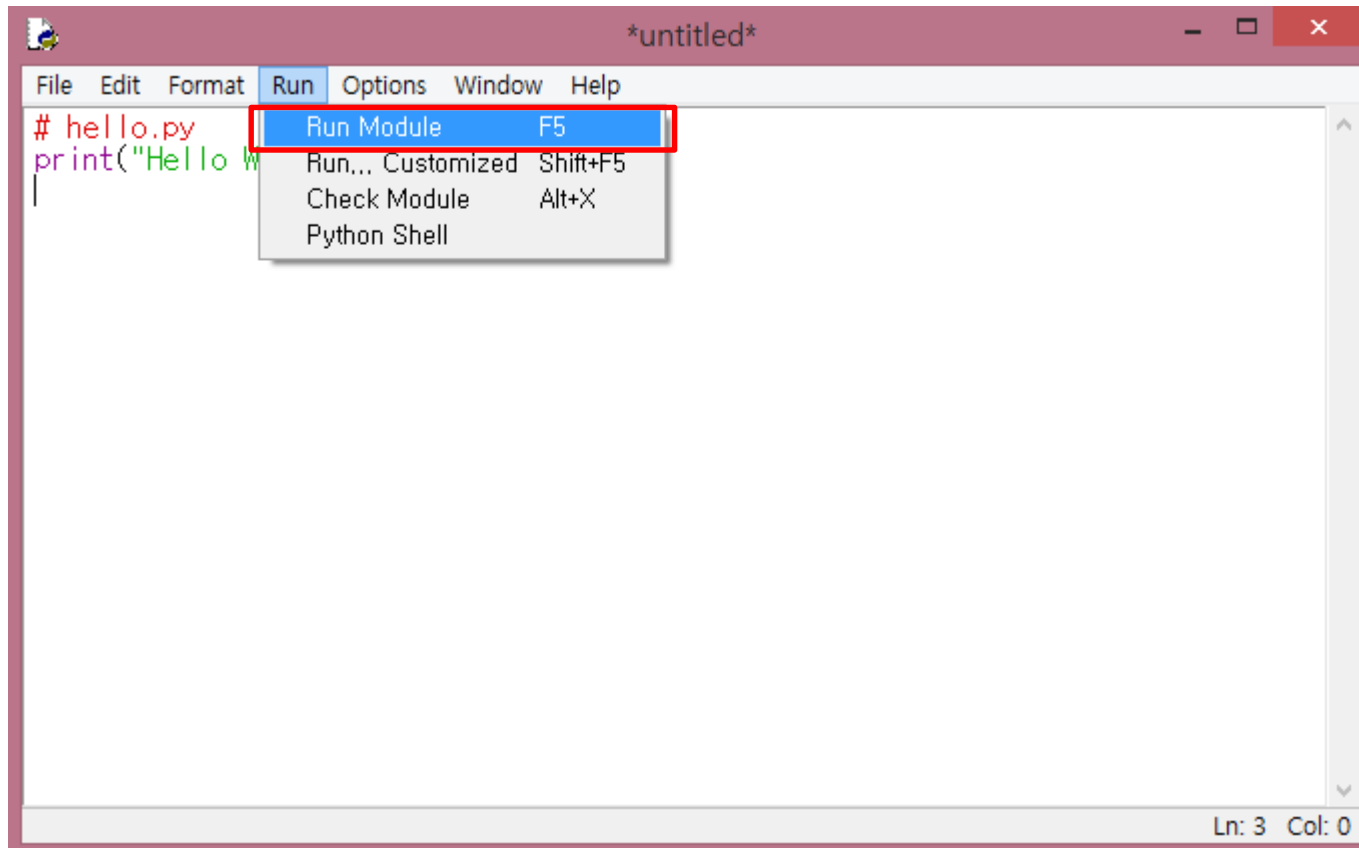


The screenshot shows the IDLE Python editor window. The title bar reads '*untitled*'. The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code editor contains two lines of Python code: a comment line '# hello.py' and a print statement 'print("Hello World")'. The status bar at the bottom right indicates 'Ln: 3 Col: 0'.

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

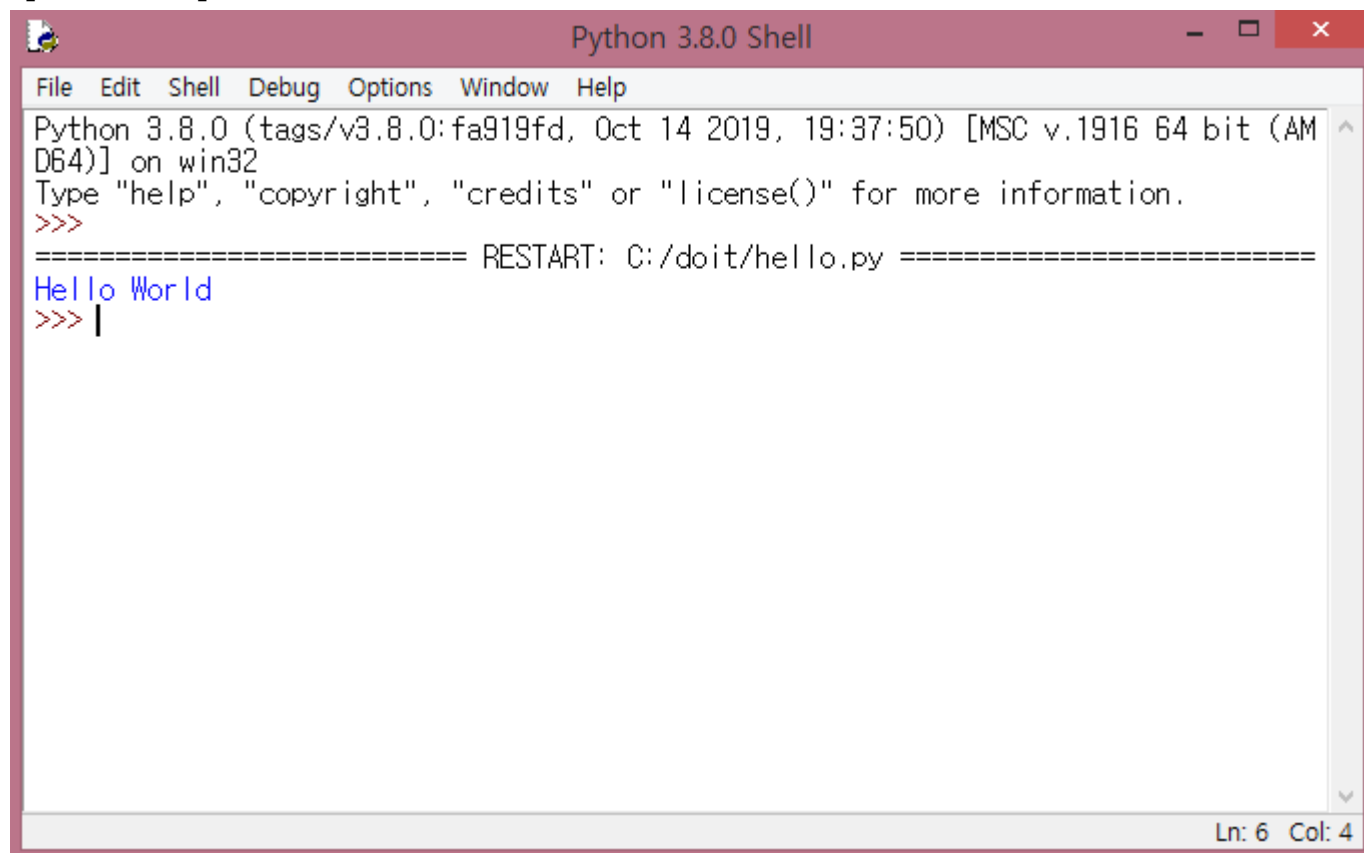
- ✓ #은 주석으로 그 줄 끝까지 프로그램 수행에 전혀 영향을 주지 않음
- ✓ 주석은 프로그래머를 위한 코멘트로 프로그램 소스에 설명문을 달 때 사용함
- ✓ 여러 줄로 이루어진 주석은 큰따옴표 세 개(""") 사이에 주석문을 작성하면 됨

- [IDLE 에디터] 파이썬 프로그램 실행하기



- ✓ IDLE 에디터 창 메뉴에서 [Run → Run Module]을 선택함(단축키:F5)
- ✓ 프로그램을 실행하면 파일을 먼저 저장하라는 대화상자가 나옴
- ✓ [확인]을 선택하고, C:\wdoit 디렉터리에 hello.py라는 이름으로 저장함

- [IDLE 셸] 파이썬 프로그램 실행결과 보기



The screenshot shows a window titled "Python 3.8.0 Shell". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main text area displays the following content:

```
Python 3.8.0 (tags/v3.8.0:fa919fd, Oct 14 2019, 19:37:50) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/doit/hello.py =====
Hello World
>>> |
```

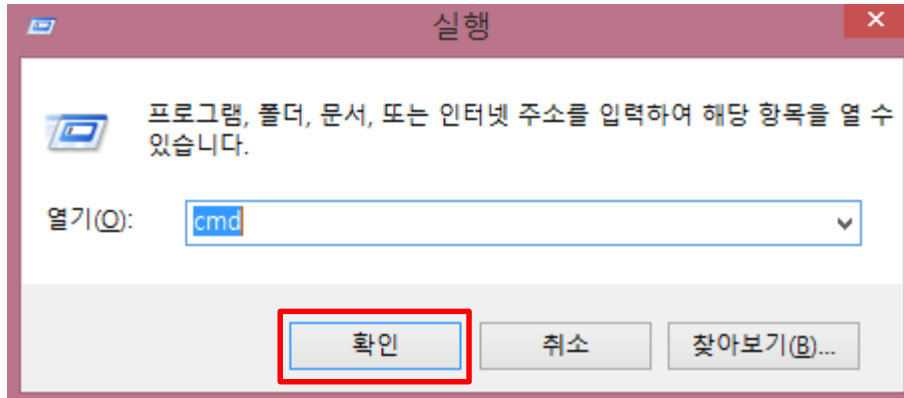
The status bar at the bottom right indicates "Ln: 6 Col: 4".

✓ 파일을 저장하면 자동으로 파이썬 프로그램이 실행됨

- 여러 줄을 작성하고 저장해야 하는 프로그램은 "IDLE 에디터"에서 작성하고, "IDLE 셸"은 실행 결과를 표시하는 용도로 사용함
- 간단한 예제는 "파이썬 셸"에서 입력하고 결과를 바로 확인함

❖ 명령 프롬프트 창에서 파이썬 프로그램 실행하기

■ 명령 프롬프트 창 열기



✓ [윈도우 키 + R]을 누르고, 빈 칸에 'cmd'를 입력하고 [확인]을 선택함

■ 파이썬 프로그램 실행하기

