

## 01장 파이썬 개요

### 1.1 파이썬 시작하기

#### 1.1.1 파이썬이란?

- 귀도 반 로섬(Guido Van Rossum)이 1991년 개발한 인터프리터(스크립트) 언어이다.
- 구글에서 만들어진 소프트웨어의 50% 이상이 파이썬
- 드롭박스(Dropbox), 인스타그램(Instagram), 장고(Django, 웹 개발을 위한 프레임워크)
- 공동 작업과 유지 보수가 매우 쉽고 편하다.



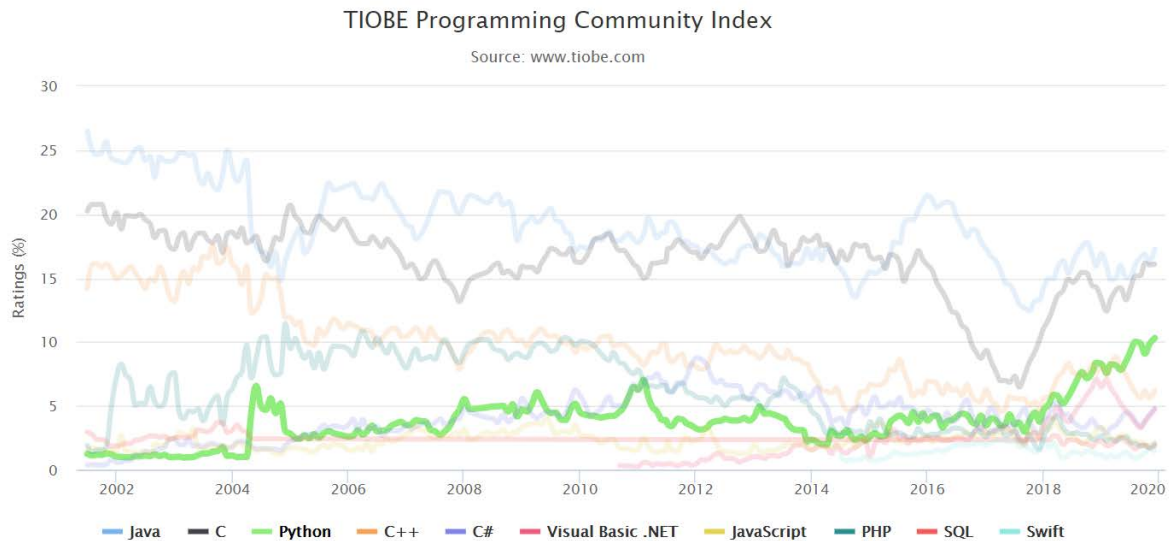
〈파이썬의 공식 로고〉

#### [꿀팁] 파이썬 이름의 유래

- 네덜란드 출신의 프로그래머인 귀도는 파이썬이라는 이름을 자신이 좋아하는 코미디 쇼인 'Monty Python's Flying Circus'에서 따왔다고 한다. 파이썬의 사전적인 의미는 고대 신화에 나오는 파르나소스 산의 동굴에 살던 큰 뱀을 뜻하여, 아폴로 신이 델파이에서 파이썬을 퇴치했다는 이야기가 전해지고 있다.

#### [꿀팁] 프로그래밍 언어 순위

- <https://www.tiobe.com/tiobe-index/>



## 1.2 파이썬의 특징

- 파이썬은 다양한 프로그래밍 패러다임을 지원하는 언어이다. 객체 지향 프로그래밍(Java, C#)과 구조적 프로그래밍(C, COBOL)을 완벽하게 지원하며 함수형 프로그래밍(R, Lisp), 관점 지향 프로그래밍(AOP) 등도 주요 기능에서 지원 된다.

### 1.2.1 파이썬은 인간다운 언어이다.

- 파이썬은 사람이 생각하는 방식을 그대로 표현할 수 있는 언어이다.

```
# 만약 4가 1,2,3,4 중에 있으면 '4가 있습니다'를 출력한다.
if 4 in [1,2,3,4]: print ("4가 있습니다")
```

### 1.2.2 파이썬은 문법이 쉬워 빠르게 배울 수 있다.

- 파이썬은 문법 자체가 아주 쉽고 간결하며 사람의 사고 체계와 매우 닮아 있다.
- 유명한 프로그래머인 에릭 레이먼드(Eric Raymond)는 파이썬을 공부한 지 단 하루 만에 자신이 원하는 프로그램을 작성할 수 있었다고 한다.

### 1.2.3 파이썬은 무료이지만 강력하다

- 사용료 걱정없이 언제 어디서든 파이썬을 다운로드하여 사용
- 파이썬과 C는 찰떡궁합
- 파이썬 라이브러리들 중에는 C로 만들어진 것도 많다.

### 1.2.4 파이썬은 간결하다

- 만약 펄(Perl)과 같은 프로그래밍 언어가 100가지 방법으로 하나의 일을 처리할 수 있다면 파이썬은 가장 좋은 방법 1가지만 이용하는 것을 선호한다.
- 실행이 되게 하려면 꼭 줄을 맞추어야 한다. 이렇듯 줄을 맞추어 코드를 작성하는 행위는 가독성에 크게 도움이 된다.

```
# 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")
```

### 1.2.5 파이썬은 프로그래밍을 즐기게 해준다

- 파이썬은 다른 것에 신경 쓸 필요 없이 내가 하고자 하는 부분에만 집중할 수 있게 해준다.
- 파이썬을 배우고 나면 다른 언어로 프로그래밍하는 것에 지루함을 느끼게 될지도 모른다.

### 1.2.6 파이썬은 개발 속도가 빠르다

- "Life is too short, You need python." (인생은 너무 짧으니 파이썬이 필요해.)

## 1.3 파이썬으로 무엇을 할 수 있을까?

### 1.3.1 파이썬으로 할 수 있는 일

- 시스템 유틸리티 제작
  - 파이썬은 운영체제의 시스템 명령어들을 이용할 수 있는 각종 도구를 갖추고 있기 때문에 이를 바탕으로 갖가지 시스템 유틸리티를 만드는 데 유리하다.
- GUI 프로그래밍
- 웹 프로그래밍
- 수치 연산 프로그래밍
- 데이터베이스 프로그래밍
- 데이터 분석, 인공지능, 사물 인터넷, 가상현실(컴퓨터 애니메이션)
  - 파이썬으로 만들어진 판다스(Pandas)라는 모듈을 이용하면 데이터 분석을 더 쉽고 효과적으로 할 수 있다.
  - 파이썬 라이브러리로 제공되는 텐서플로(TensorFlow)를 통해 머신러닝(기계학습) 프로그래밍을 할 수 있다.
  - 사물 인터넷 분야에서도 파이썬은 활용도가 높다. 한 예로서 라즈베리파이(Raspberry Pi)는 리눅스 기반의 아주 작은 컴퓨터이며 이 라즈베리파이를 제어하는 도구로 파이썬이 사용된다.
  - Maya와 같은 컴퓨터 애니메이션 소프트웨어에서 파이썬을 사용하여 애니메이션, 환경, 모션 그래픽, 가상 현실과 캐릭터를 제작한다.

### 1.3.2 파이썬으로 할 수 없는 일

- 시스템과 밀접한 프로그래밍 영역
- 모바일 프로그래밍

## 1.4 파이썬 통합개발환경(IDE) 구축

### 1.4.1 윈도우에서 파이썬 설치

- <http://www.python.org/downloads> 에서 윈도우용 파이썬 언어 패키지를 다운로드한다.
- 파이썬이 어느 곳에서든지 실행될 수 있도록 "Add Python 3.5 to PATH" 옵션을 선택하도록 하자.

#### [꿀팁] 파이썬 버전 2.x와 3.x의 차이

- 파이썬 2.0은 2000년 10월 16일 배포되었다.
- 파이썬 3.0은 2008년 12월 03일 배포되었다. 2.x 버전들과 호환되지 않는다.

```
# 1. print가 함수 형태로 변경되었다.
# 2.x style
# print "welcome to", "python"
# 3.x style
print("welcome to", "python")
# welcome to python

# 2. long 형이 없어지고 int형으로 통일되었다.
# 2.x style
# type(2**31)
# <type 'long'>
# 3.x style
type(2**31)
# <class 'int'>

# 3. "int/int"의 결과는 float으로 처리된다.
# 2.x style
# 3/2
# 1
# 3.x style
3/2
# 1.5

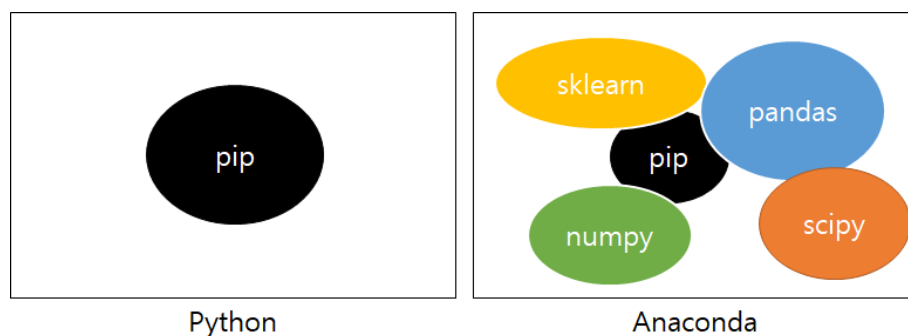
# 4. String의 Unicode 체계가 바뀌었다.
# 2.x style
# type('가')
# <type 'str'>
# type('가', decode('utf8'))
# <type 'unicode'>
# 3.x style에서는 string과 bytes로 구분된다.
type('가')
# <class 'str'>
type('가'.encode('cp949'))
# <class 'bytes'>
```

### 1.4.2 리눅스에서 파이썬 설치

```
$ python -V
$ tar xvf Python-3.6.1.tgz
$ cd Python-3.6.1
$ ./configure
$ make
$ su -
$ make install
```

### 1.4.3 Anaconda 설치 (권장)

- Python은 파이썬 공식 홈페이지에서 받을 수 있으며, pip 툴만을 포함하고 있다. 필요한 패키지나 라이브러리 등을 설치할 때 모두 수동으로 해줘야 한다.
- 아나콘다는 Python 기본 패키지에 각종 수학/과학 라이브러리들을 같이 패키징해서 배포하는 버전이다. 아나콘다에 포함된 툴들로는 panda, numpy, scipy, sklearn, matplotlib, Jupyter Notebook 등이 있습니다.
- 아나콘다 다운로드: <https://www.anaconda.com/download/>



### 1.4.4 비주얼 스튜디오 코드(vscode) 설치

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

### 1.4.5 Git 설치

- 다운로드
  - <https://www.git-scm.com/download/>
- 설정
  - <https://miaow-miaow.tistory.com/17>

```
# 명령 프롬프트에서
C:\Users\kshelp>git --version
```

```
git version 2.33.1.windows.1

C:\Users\kshelp>git config --global user.name "kshelp"

C:\Users\kshelp>git config --global user.email "kshelp@naver.com"

C:\Users\kshelp>more .gitconfig
[user]
    email = kshelp@naver.com
    name = kshelp
[core]
    editor = "\"C:\\Users\\kshelp\\AppData\\Local\\Programs\\Microsoft VS Code\\bin\\code.cmd\" --wait
```

## 1.5 첫 번째 코드 작성과 실행

### 1.5.1 파이썬 인터랙티브 셸 (Python Interactive Shell)

- 명령 프롬프트에서 파이썬 프로그램을 실행하면 파이썬 인터랙티브 셸이 실행된다.

```
C:\Users\kshelp>conda.bat activate base

(base) C:\Users\kshelp>python
Python 3.8.5 (default, Sep 3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Hello world")
Hello world
>>> 1 + 2
3
>>> 3 / 2.4
1.25
>>> 3 * 9
27
>>> a = 1
>>> b = 2
>>> a + b
3
>>> a = "Python"
>>> print(a)
Python
>>> import sys
>>> sys.exit()

(base) C:\Users\kshelp>deactivate
DeprecationWarning: 'deactivate' is deprecated. Use 'conda deactivate'.

(base) C:\Users\kshelp>conda.bat deactivate

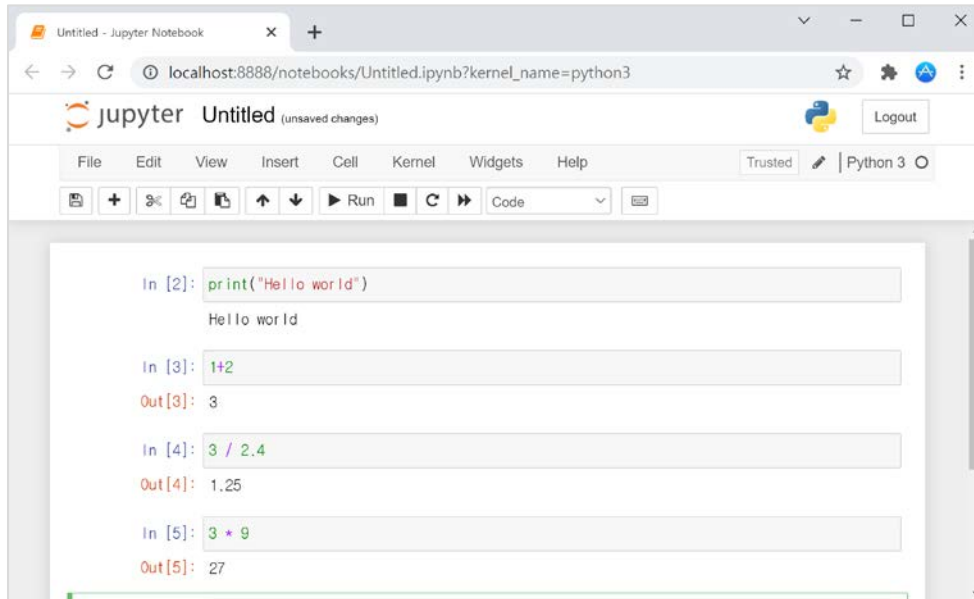
C:\Users\kshelp>
```

### 1.5.2 주피터 노트북 (Jupyter Notebook)

- Jupyter Notebook은 웹 기반의 인터랙티브 셸(명령 프롬프트) 이라고 할 수 있다.

```
// 실행 방법
```

```
C:\Users\won>jupyter notebook
[I 21:07:42.018 NotebookApp] [nb_conda_kernels] enabled, 2 kernels found
[I 21:07:43.063 NotebookApp] [nb_anacondacloud] enabled
[I 21:07:43.090 NotebookApp] [nb_conda] enabled
[I 21:07:43.285 NotebookApp] \u2713 nbpresent HTML export ENABLED
[W 21:07:43.287 NotebookApp] \u2717 nbpresent PDF export DISABLED: No module named 'nbbrowserpdf'
[I 21:07:43.654 NotebookApp] Serving notebooks from local directory: C:\Users\won
[I 21:07:43.655 NotebookApp] 0 active kernels
[I 21:07:43.657 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/
[I 21:07:43.659 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
```



## [꿀팁] Jupyter notebook 명령키

### ■ 참고

- <https://driz2le.tistory.com/200>

Command Mode (press Esc to enable)		Edit Mode (press Enter to enable)	
Enter	enter edit mode	Tab	code completion or indent
Shift-Enter	run cell, select below	Shift-Tab	tooltip
Ctrl-Enter	run cell	Ctrl-]	indent
Alt-Enter	run cell, insert below	Ctrl-[	dedent
Y	to code	Ctrl-A	select all
R	to raw	Ctrl-Z	undo
1	to heading 1	Ctrl-Shift-Z	redo
2	to heading 2	Ctrl-Y	Redo

## 1.5.4 메모장과 명령 프롬프트

### ■ 코드 작성

- 파이썬 프로그램에서는 주석을 달 때 #을 사용한다.
- 이 밖에도 #의 용도는 여러 가지가 있는데, 아래와 같이 실행 파일과 소스코드 인코딩을 명시하는 데 사용한다.
- 소스코드 인코딩의 경우, coding 지시자를 이용해 지정할 수 있는데 지정돼 있지 않은

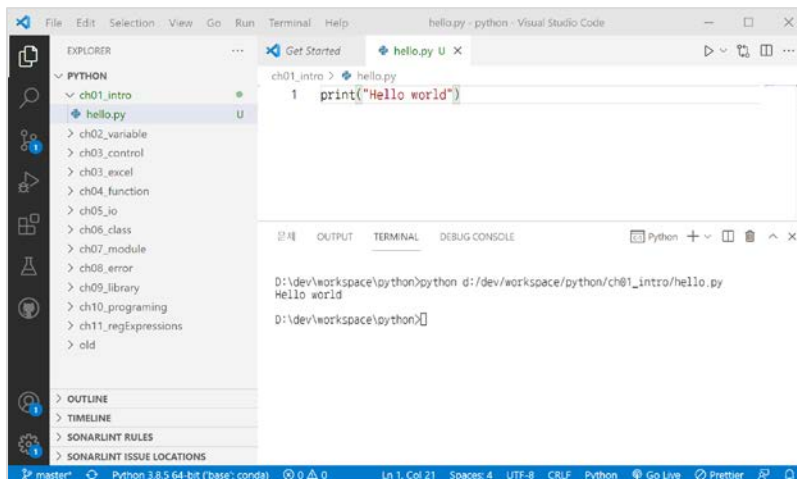
경우 아스키가 기본 소스코드 인코딩으로 설정된다.

```
# 코드 작성
C:\Users\kshelp>notepad hello.py
#!/usr/bin/python
# -*- coding: utf-8 -*-
# hello.py
print("Hello world")

# 코드 실행
C:\Users\kshelp>python hello.py
Hello world
```

### 1.5.5 비주얼 스튜디오 코드 (vscode)

- vscode에서 [파일 > 새 파일] 메뉴를 선택한다.



- Github에서 소스코드 가져오기
  - vscode에서 [Terminal > New Terminal] 메뉴를 선택하여 아래와 같이 수행한다.

```
D:\dev\workspace\pythonLab>git clone https://github.com/kshelp/python
Cloning into 'python'...
remote: Enumerating objects: 417, done.
remote: Counting objects: 100% (417/417), done.
Receiving objects: 100% (417/417), 158.73 KiB | 17.64 MiB/s, done.
remote: Compressing objects: 100% (344/344), done.
remote: Total 417 (delta 62), reused 417 (delta 62), pack-reused 0
Resolving deltas: 100% (62/62), done.
```