

Data Analytics Based Python

SECT1. Python Introduction

IT Competency Improvement Training
Kim Jin Soo



- ◆ 컴퓨터의 기본 동작 원리
- ◆ 파이썬이 무엇인가?
- ◆ 어디에서, 누가 사용하고 있나?
- ◆ 왜 파이썬을 배워야 하는가?


컴퓨터의 동작 원리



- ❖ 컴퓨터의 두뇌, CPU
- ❖ 컴퓨터의 손, 메모리
- ❖ 컴퓨터 속의 주머니, 하드디스크
- ❖ 컴퓨터와의 대화수단, 프로그래밍 언어

중앙 처리 장치

위키백과, 우리 모두의 백과사전.

 CPU는 여기로 연결됩니다. 다른 뜻에 대해서는 [CPU \(동음이의\)](#) 문서를 참조하십시오.

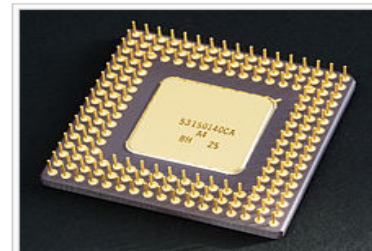
중앙 처리 장치(中央處理裝置, **문화어:** 중앙처리소자) 또는 **CPU**(**영어:** central processing unit)는 **소프트웨어** 명령의 실행이 이루어지는 **컴퓨터**의 부분, 혹은 그 기능을 내장한 **칩**을 말한다. 컴퓨터 안의 중앙 처리 장치(CPU)는 외부에서 정보를 입력 받고, 기억하고, 컴퓨터 프로그램의 명령어를 해석하여 연산하고, 외부로 출력하는 역할을 한다. 따라서 중앙 처리 장치(CPU)는 컴퓨터 부품과 정보를 교환하면서 컴퓨터 시스템 전체를 제어하는 장치로, 모든 컴퓨터의 작동과정이 중앙 처리 장치(CPU)의 제어를 받기 때문에 컴퓨터의 두뇌에 해당한다고 할 수 있다. 실제의 CPU 칩엔 실행 부분뿐만 아니라 **캐시** 등의 부가 장치가 통합된 경우가 많다.

CPU에는 MCU(Micro Control Unit)와 주변 장치(외부 확장 장체에 관한 IC)가 다 들어있는 Soc(System One Chip)가 있다. 주변 IC가 따로 달려 있을 경우, 그것은 MCU라고 할 수 있다.

CPU는 **기계어**로 쓰인 **컴퓨터 프로그램의 명령어**를 해석하여 실행한다. CPU는 프로그램에 따라 외부에서 정보를 **입력**하고, **기억**하고, **연산**하고, 외부로 **출력**한다. CPU는 컴퓨터 부품과 정보를 교환하면서 컴퓨터 전체의 동작을 제어한다.

기본 구성으로는 CPU에서 처리할 명령어를 저장하는 역할을 하는 **프로세서 레지스터**, 비교, 판단, 연산을 담당하는 **산술논리연산장치(ALU)**, 명령어의 해석과 올바른 실행을 위하여 CPU를 내부적으로 제어하는 **제어부(control unit)**와 내부 버스 등이 있다.

각종 전자 부품과 반도체 칩을 하나의 작은 칩에 내장한 전자 부품을 마이크로프로세서라고 한다. 마이크로프로세서는 전기 법통에 쓰이는 낮은 성능의 제품부터 컴퓨터에 쓰이는 높은 성능의 제품까지 매우 다양하다. 마이크로프로세서를 가운데 가장 복잡하고 성능이 높은 제품은 컴퓨터의 연산 장치로 쓰인다. 이것을 중앙 처리 장치라고 한다.

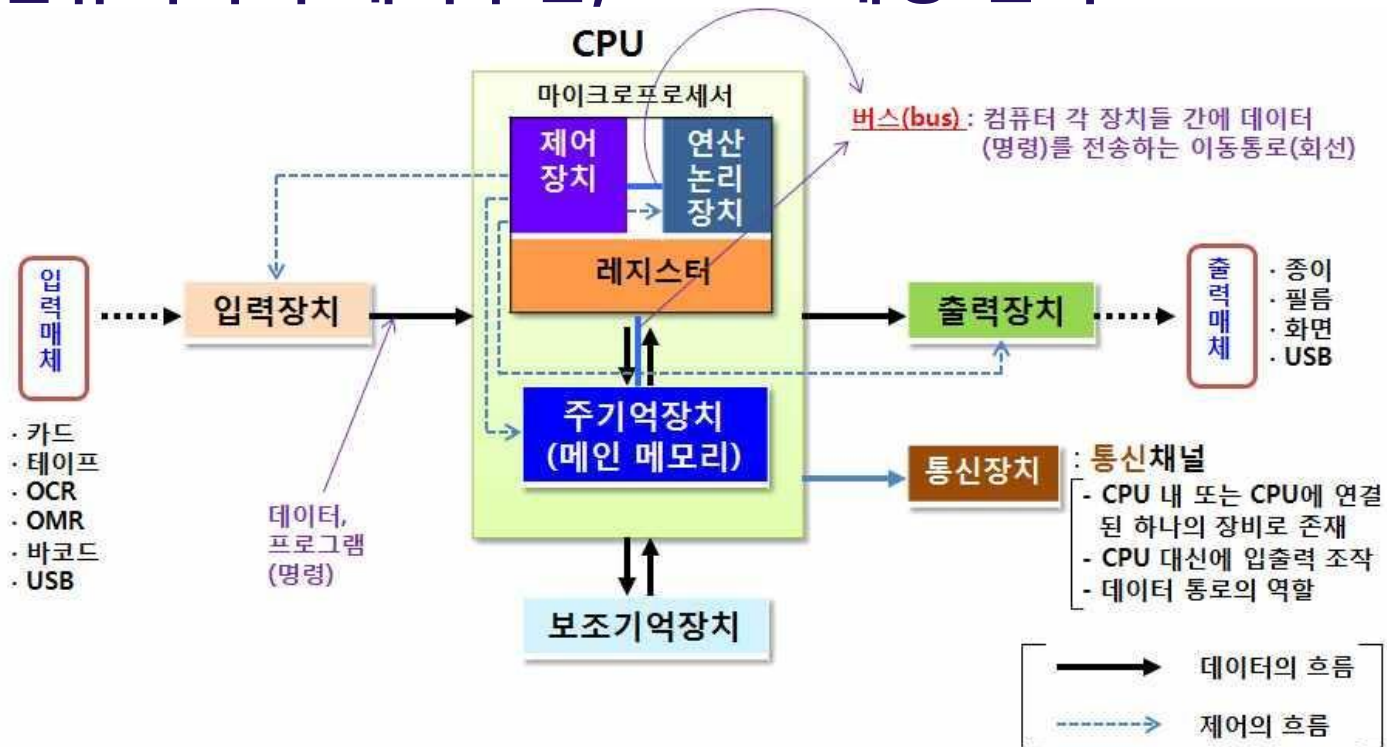


세라믹 PGA의 인텔 80486DX2 마이크로프로세서

컴퓨터의 동작 원리



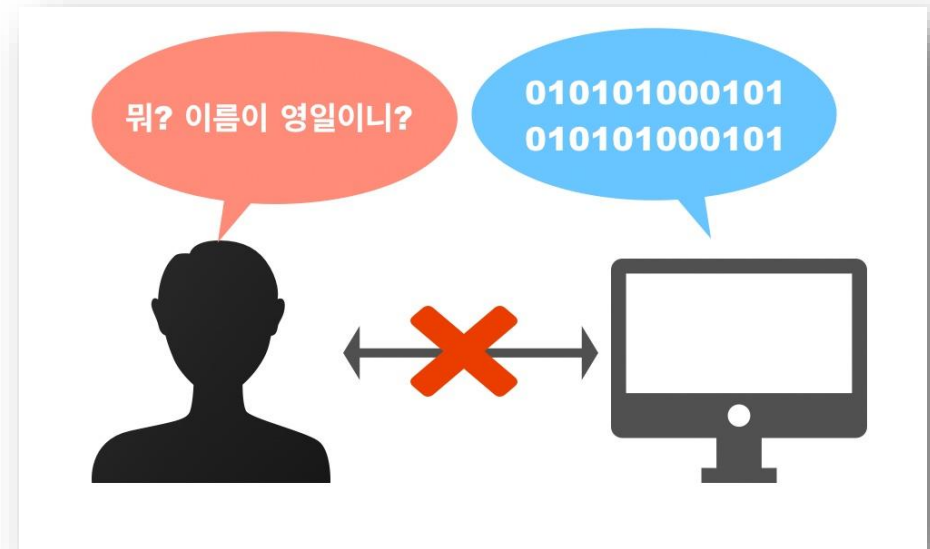
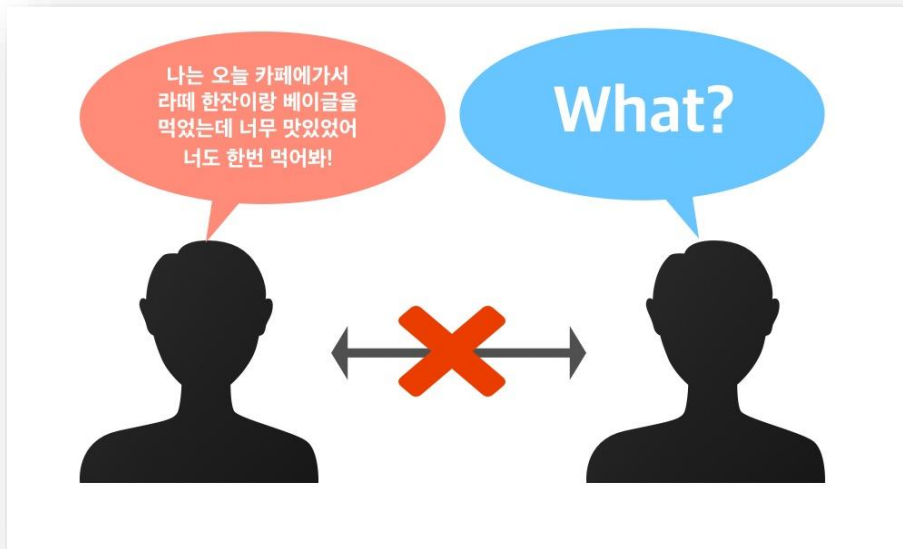
- ❖ 컴퓨터의 두뇌, CPU
- ❖ 컴퓨터의 손, 메모리
- ❖ 컴퓨터 속의 주머니, 하드디스크
- ❖ 컴퓨터와의 대화수단, 프로그래밍 언어



프로그래밍 언어의 역할



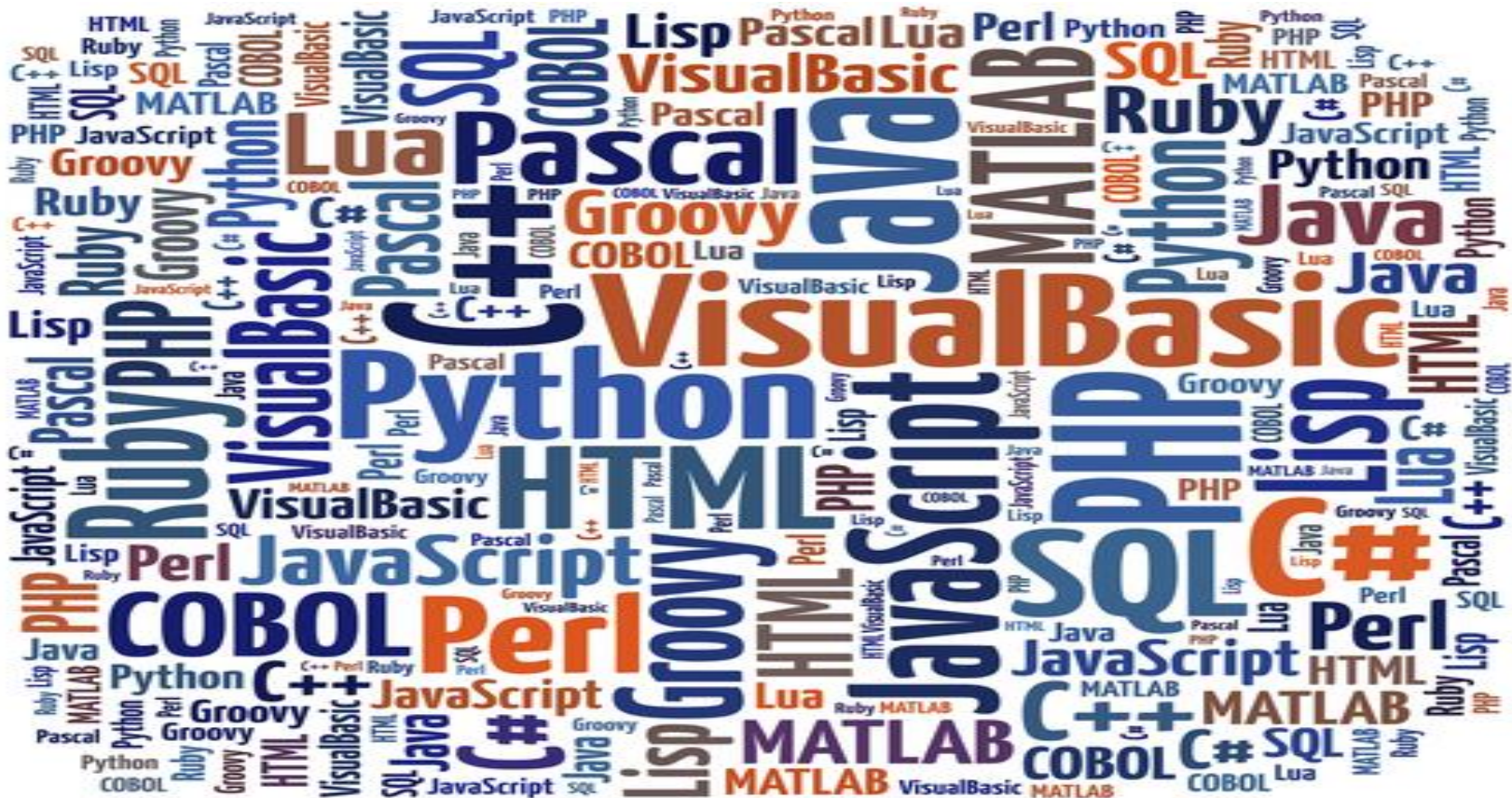
- ❖ 개발자가 되기 위해 프로그래밍 언어를 배우는 것이 아니라,
- ❖ 컴퓨터와 대화하기 위해서 컴퓨터가 쓰는 말을 배워야 한다



프로그래밍 언어의 종류



- ❖ 어떤 프로그래밍 언어를 배워야 할까?
- ❖ 언어의 한계가 곧 자기 자신의 한계



파이썬(Python)이 무엇인가?



python

noun [C]

UK  /ˈpaɪ.θən/ US  /ˈpaɪ.θɑːn/

plural **pythons** or **python**



Dave King/Dorling
Kindersley/Getty Images


a very large snake that kills animals for food by wrapping itself around them and crushing them

+ ≡

파이썬(Python)이 무엇인가?



[Python](#)[PSF](#)[Docs](#)[PyPI](#)[Jobs](#)[Community](#)

[Donate](#)[GO](#)[Socialize](#)

[About](#)[Downloads](#)[Documentation](#)[Community](#)[Success Stories](#)[News](#)[Events](#)

```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

Functions Defined

The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. [More about defining functions in Python 3](#)

[1](#)[2](#)[3](#)[4](#)[5](#)

전 세계에서 **2번째** 로 많이 사용되는 언어
(2021년 5월 TIOBE사 통계자료)

상위 20위의 프로그래밍 언어

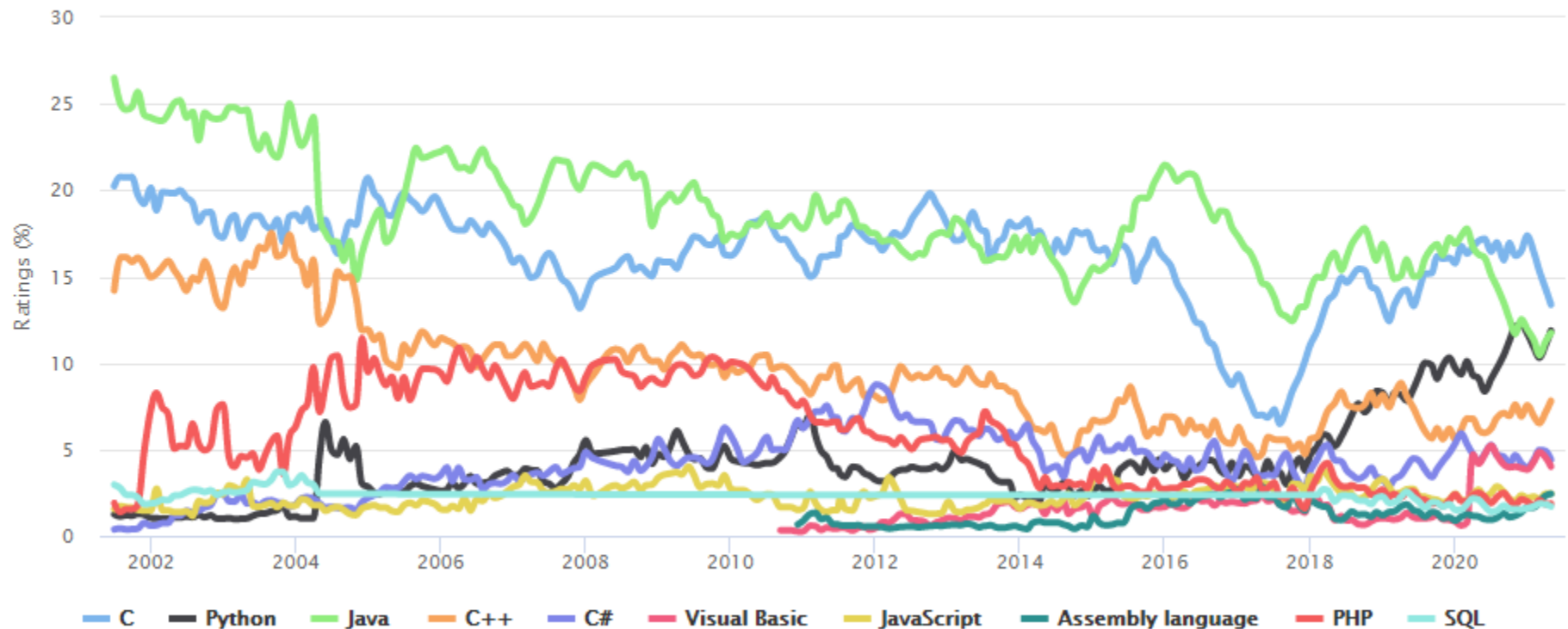


TIOBE Index for May 2021

May Headline: Python back at second position

TIOBE Programming Community Index

Source: www.tiobe.com



[참조 : <https://www.tiobe.com/tiobe-index/>]

상위 20위의 프로그래밍 언어



TIOBE Index for May 2021

May Headline: Python back at second position

May 2021	May 2020	Change	Programming Language	Ratings	Change
1	1		C	13.38%	-3.68%
2	3	▲	Python	11.87%	+2.75%
3	2	▼	Java	11.74%	-4.54%
4	4		C++	7.81%	+1.69%
5	5		C#	4.41%	+0.12%
6	6		Visual Basic	4.02%	-0.16%
7	7		JavaScript	2.45%	-0.23%
8	14	▲▲	Assembly language	2.43%	+1.31%
9	8	▼	PHP	1.86%	-0.63%
10	9	▼	SQL	1.71%	-0.38%
11	15	▲▲	Ruby	1.50%	+0.48%
12	17	▲▲	Classic Visual Basic	1.41%	+0.53%
13	10	▼	R	1.38%	-0.46%
14	38	▲▲	Groovy	1.25%	+0.96%
15	13	▼	MATLAB	1.23%	+0.06%
16	12	▼	Go	1.22%	-0.05%
17	23	▲▲	Delphi/Object Pascal	1.21%	+0.60%
18	11	▼	Swift	1.14%	-0.65%
19	18	▼	Perl	1.04%	+0.16%
20	34	▲▲	Fortran	0.83%	+0.51%

n/]

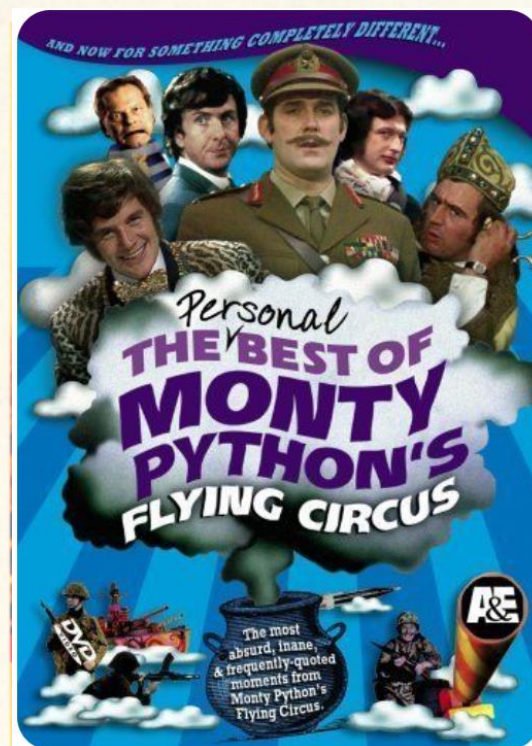
파이썬(Python)이 무엇인가?



- 귀도 반 로섬(Guido van Rossum)이 심심해서 제작
- 1991년 발표
- 객체 지향 언어



평소 좋아한 코미디에서 따옴 ▶





- ❖ 인터프리터(interpreter) 언어이다. (실행 결과를 바로 확인)
- ❖ 간결하고 쉬운 문법으로 빠르게 학습할 수 있다.
- ❖ 강력한 성능을 가진다.
- ❖ 다양한 분야에 적용할 수 있는 라이브러리가 풍부하다.
- ❖ 개발 속도가 빠르다.
- ❖ 오픈 소스(open source)이며 무료이다.

※ 현재 파이썬은 버전이 2.x 대와 3.x 대로 나뉘어
두 가지 버전이 존재 (호환되지 않음)



높은 생산성

프로그래밍 문법이
사용하기 쉽거나 (Writability)
이해하기 쉬움 (Readability)





“하나 이상의 해결법이 존재한다”

“가장 아름다운 하나의 답이 존재한다”



- Beautiful is better than ugly
- Explicit is better than implicit
- Simple is better than complex

어디에서, 누가 쓰고 있나?



- ❖ 말이 필요 없는 구글신, Google
- ❖ 구글 앱 엔진, Google App Engine
- ❖ 유튜브, Youtube : PayPal의 초창기멤버 3명이 2005년 만듦
- ❖ 오픈 소스 라이브러리, <https://developers.google.com>
- ❖ 무료 사진 공유 사이트, Instagram
- ❖ 헐리우드 영화 시각 특수 효과자의 거장, ILM
- ❖ 여행을 위한 모든 예약은 이지트립에서, 구스토닷컴
(<http://www.eztrip.com>)

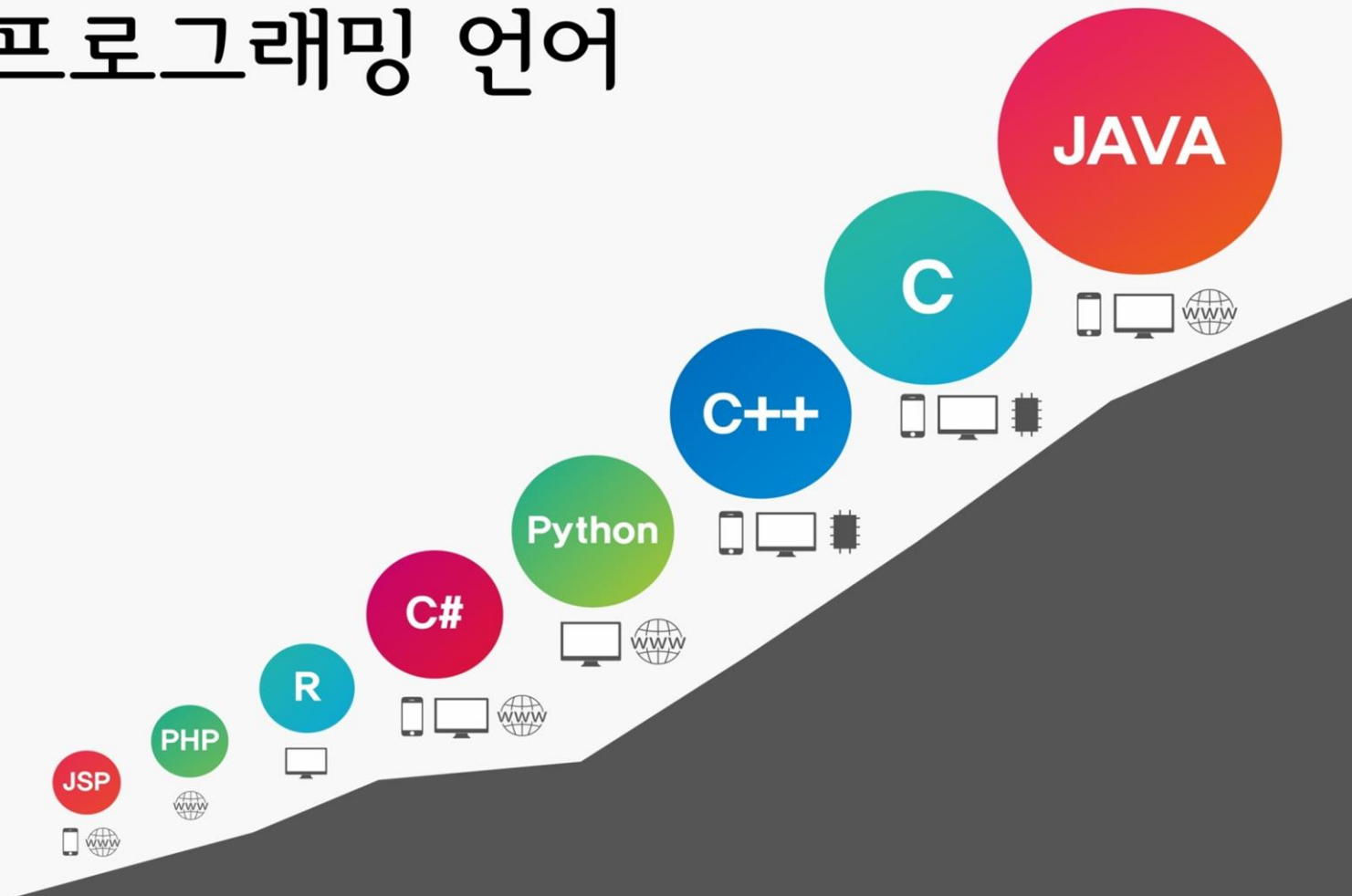
※ 파이썬으로 구현된 더 많은 성공 사례를 확인해 보자!

<http://www.python.org/about/success/>

왜 파이썬을 배워야 하나?



프로그래밍 언어





❖ 파이썬을 프로그래밍 언어로 선택한 이유

- 파이썬으로 작성한 코드는 읽기 쉽다
 - 의미 전달을 방해하는 잡음을 최대한 제거
 - 군더더기 없는 문법은 명확
 - 명시적인 코드를 작성할 수 있도록 해준다.
- 파이썬은 멀티 패러다임 언어이다.
 - 데이터와 처리 절차를 명확하게 분리해서 프로그램을 작성
 - 데이터와 처리 절차를 객체라는 틀 안에 우겨 넣은 후 객체들의 협력을 통해 프로그램을 작성
 - 상태 변경이 없는 순수한 함수들의 조합만으로 프로그램을 작성
→ 파이썬의 **절차형, 객체지향, 함수형 패러다임** 모두를 지원
- 파이썬은 유연하다.
 - 클래스 기반의 객체지향 언어인 파이썬을 자바스크립트와 유사한 형식의 프로토타입 기반의 객체지향 언어로 변모 가능하다.

❖ 정리하자면, 파이썬은 읽기 쉽고, 유연하며, 멀티 패러다임을 지원한다.