

# 항공우주공학과 202004142 이승신 수치해석 HW01

## 1. Python의 역사

Python은 1991년 02월 20일 네덜란드계 프로그래머 귀도 반 로섬에 의해 만들어졌다.

"Life is short, You need Python." -Moto of Python-

## 2. Python의 특징

Python은 인터프리터 방식의 High-Level 프로그래밍언어 이다. 프로그래밍 언어는 인간친화적인 관점으로 분류가능한데 사람이 쓰는 문법과 유사하면 High-Level, 컴퓨터 구조에 가까우면 Low-Level 프로그래밍 언어로 분류한다.

또한 프로그래밍 언어를 기계어로 전환하는 방법에 따라 컴파일 방식, 인터프리터 방식으로 구분 가능한데, 컴파일 방식은 컴파일러가 작성된 스크립트를 모두 읽어 들여 이를 기계어로 변환하고 기계어로 변환된 오브젝트 파일을 실행시키는 것이고 인터프리터 방식은 인터프리터가 스크립트를 한 줄씩 읽으면서 즉각적으로 기계어로 변환 후 실행하는 방식이다. 보편적으로 컴파일 방식이 실행속도가 뛰어나나 소스코드가 수정되면 처음부터 다시 컴파일을 해야 하는 불편함이 있다. 반대로 인터프리터 방식은 컴파일 방식에 비해 실행속도가 느리지만 소스코드를 수정해도 수정된 부분을 즉각적으로 반영할 수 있다는 장점이 있다, 즉 디버깅과 개발속도 측면에서 유리하다.

마지막으로 다른 프로그래밍 언어와 달리 중괄호를 사용하지 않고 들여쓰기를 이용해 가독성이 매우 뛰어나다.

## 3. Python의 활용분야

Python은 다양한 분야에서 활용되고 있다. 특히나 Python의 무궁무진한 라이브러리가 이를 가능하게 하는데, 대표적인 예로 OpenCV 라이브러리는 이미지 프로세싱 분야에서 쓰이고, Numpy, Matplotlib 는 수치해석 및 데이터분석 분야에서 쓰인다. 또한 Flask, Django 라이브러리로 웹서버를 개발 가능하고, Pymysql 라이브러리로 SQL데이터베이스 작업 또한 가능하다.

하지만 수많은 라이브러리를 다양한 사람들이 개발하기 때문에 라이브러리간 충돌이 발생할 가능성이 있다. 이를 방지하기 위해 Python은 Venv, Conda 같은 가상환경을 지원해 각각 독립된 환경에서 Python을 실행 가능하게 하여 불필요한 라이브러리에 의한 프로그램

램 충돌을 방지한다.

#### 4. Python 메모리 관리

Python은 High-Level 프로그래밍 언어로, 가비지컬렉터라 불리는 메모리 관리자가 자동으로 변수에 메모리를 할당하고 해제하는 형태이다. 덕분에 개발자는 메모리 관리를 직접 해야하는 C언어 계열과 달리, 아주 쉽게 프로그램을 개발 할 수 있다. 하지만 자동으로 메모리가 관리되는 만큼 메모리 누수가 발생 할 수가 있고, 비효율적으로 메모리가 운영 될 수 있다. 따라 시스템 리소스가 적은 임베디드 시스템과 같은 곳에서 사용하는 것은 부적합 하다.

따라 퍼포먼스, 고도의 최적화가 필요한 곳은 비록 개발하기 어렵고 느리지만, C,C++를 활용하는 것이 유리하다. 왜냐하면 C언어 계열은 포인터, 즉 사용자가 직접 메모리를 할당하고 해제 할 수 있어 메모리를 개발자의 의도대로 효율적으로 사용 할 수 있기 때문이다. 하지만 개발자의 역량에 따라 프로그램의 성능이 매우 달라 질 수 있다는 것도 유의해야 한다.