

프로그래밍 개요

프로그램(Program)이란

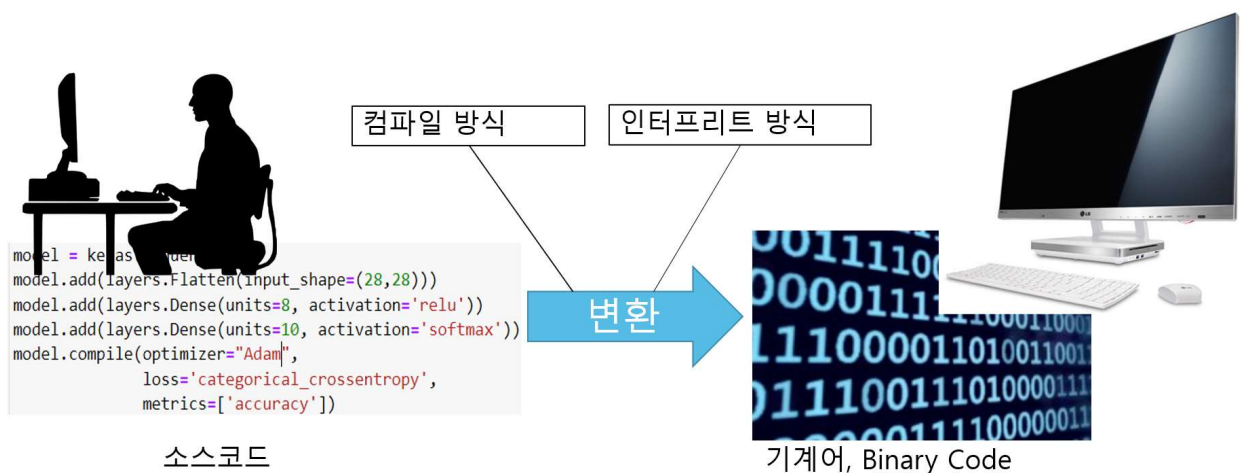
- 컴퓨터에 특정 작업을 실행시키기 위한 처리 방법과 순서를 논리적으로 작성한 명령문들의 집합

관련 용어

- 로직(Logic)
 - 프로그램이 시작해서 목적인 결과를 낼 때까지 일의 순서, 논리적인 흐름을 **프로그램 로직**이라 한다.
- 프로그래밍(Programming)
 - 로직을 작성하는 작업
 - 코드를 작성하는 일이므로 코딩이라고도 한다.
- 프로그래밍 언어(Programming Language)
 - 프로그램을 작성할 때 사용하는 언어
 - 범용적인 언어: 파이썬, 자바, C언어
 - 특수 목적의 언어: R, SQL
 - <https://www.tiobe.com/tiobe-index/> (<https://www.tiobe.com/tiobe-index/>)
- Library, API
 - 프로그램을 작성하는데 자주 반복적으로 사용되는 코드들을 미리 작성해 제공하는 것을 말한다.
 - 파이썬에서는 패키지라고도 한다.

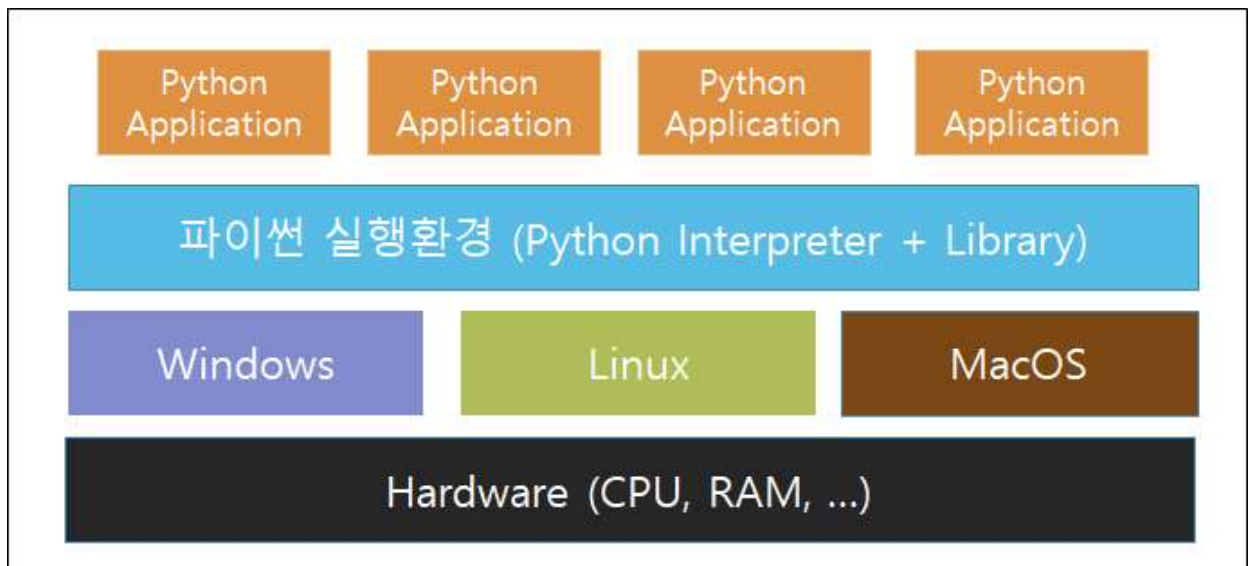
프로그래밍 언어

- 프로그램 작성시 언어
 - 프로그램은 사람이 이해하는 언어로 작성한다.
 - High Level Language 이라고 한다.
 - High Level Language로 코드를 작성한 것을 **Source code** 라고 한다.
- 프로그램 실행시 언어
 - 컴퓨터는 사람의 언어를 이해하거나 저장하지 못한다. 0과 1(on/off)의 **2진 데이터(binary code)의 기계어**를 사용한다.
 - Low Level Language
 - 기계어로 변환된 코드를 **Binary Code**라고 한다.



Compile과 Interpret

- 사람이 이해하는 High Level Language 로 작성한 프로그램 코드를 실행할 때는 기계가 이해하는 언어인 기계어(Binary code)로 변환해야 한다.
언어들 마다 변환하는 방식은 다르지만 다음 두가지 방식이 있다.
- **Compiled 방식**
 - ****Compiler(컴파일러)****라는 변환프로그램을 이용해 소스 코드를 한번에 변환해 기계어로된 파일을 생성한다.
 - 소스코드가 아닌 변경된 기계어 파일(바이너리 파일)을 실행한다.
 - **장점**
 - 프로그램 실행속도가 빠르다.
 - **단점**
 - **OS(Platform) 종속적이다.** 그래서 OS별로 프로그램을 작성해야 한다.
- **Interpreted 방식**
 - 소스코드를 바로 실행한다.
 - ****Interpreter(인터프리터)****라는 실행환경(Runtime Environment)이 실행시 명령문단위로 기계어로 변환하여 실행한다.
 - **장점**
 - **OS(Platform) 독립적이다.** 그래서 프로그램을 하나만 구현하면 OS상관없이 실행된다.
 - 단 각 OS에는 그 환경에 맞는 Interpreter가 설치되어 있어야 한다.
 - **단점**
 - 실행하는 도중 기계어로 변환하기 때문에 **속도가 느리다**
 - **파이썬은 Interpreted 방식의 언어이다.**



Python(파이썬)



- 1991년 네덜란드 프로그래머 귀도 판 로섬(Guido van Rossum) 이 만든 프로그래밍 언어.
- 평이한 영어로 이해할 수 있는 코드와, 일상적인 업무에 적용할 수 있고 프로그램 개발시 긴 시간을 필요로 하지 않는 언어를 목표로 개발함.
- 파이썬 3
 - 2008년 12월 3일에 기존 2.X 버전과 분리하여 하위호환이 안되는 버전으로 발표됨.
 - 2.x와 3.x는 각각 따로 버전업을 하다 파이썬 2 버전은 2020년 1월 이후 지원이 중단되었다.

파이썬의 특징

- 구문의 가독성이 좋다.
 - 코드는 작성하는 것 보다 읽는 경우가 많다. 는 명제 아래 코딩 스타일의 일관성을 유지하여 가독성을 높이는 것을 중시한다.
- 유지보수성이 좋다.
 - 파이썬은 배우기 쉽고, 코드를 읽기 쉽기 때문에 유지보수성이 좋다.
- Interpreted 언어이므로 OS 독립적인 프로그램 작성이 가능
- 다른 언어와의 유연한 확장 구조
 - C나 Java 언어로 작성된 함수들을 호출할 수 있다.
- 동적타입(Dynamically typed)
 - 변수의 타입을 고정하지 않기 때문에 하나의 변수에 다른타입의 값을 대입할 수 있다.

- 방대한 라이브러리
 - 강력한 표준라이브러리 제공
 - 잘 갖춰진 생태계
 - 풍부한 라이브러리들을 다양한 집단에서 제공하여(3rd party Library) 개발 생산성이 좋다.

파이썬에서는 라이브러리를 패키지라고도 표현한다.

파이썬으로 할 수 있는 것

- 업무 자동화
 - 스크립트
 - 다른 프로그램이나 system을 제어하는 프로그램을 스크립트라고 한다.
 - 파이썬으로 스크립트를 작성하여 컴퓨터로 하는 다양한 작업을 자동화 시킬 수 있다.
- 범용 어플리케이션 프로그램 개발
 - GUI 기반의 독립형 어플리케이션 개발
- 데이터 과학과 머신러닝
 - 최근에 파이썬이 각광 받는 이유
 - numpy, pandas, scikit-learn등 데이터분석과 머신러닝, 딥러닝관련 강력한 파이썬 라이브러리들이 있다.
- 웹 어플리케이션, Open API
 - Django, Flask, Fast API 와 같은 파이썬 기반 웹 프레임워크를 이용하면 웹 어플리케이션을 빠르고 쉽게 개발 할 수 있다.

파이썬으로 할 수 없는 것

- 시스템 프로그래밍(하드웨어 전용 드라이버, 펌웨어등). (C/C++등을 이용)
- 모바일 앱 개발
 - kivy 라이브러리를 이용하면 가능은 하나 아직 활성화 되지 않았다.

파이썬 실행환경에 라이브러리 설치

PyPI (Python Package Index) 이용

- 파이썬 3rd party Library들의 저장소
 - 파이썬의 Library들이 관리되고 있는 라이브러리 중앙 저장소
 - 파이썬은 필요한 라이브러리들을 pip 명령어를 이용해 PyPI에서 다운 받아 설치 하여 사용한다.
 - <https://pypi.org/> (<https://pypi.org/>)
 - 패키지 검색 사이트

명령어	설명
pip install 패키지명	패키지 install
pip install --upgrade 패키지명	패키지 업그레이드
pip uninstall 패키지명	패키지 제거
pip list	설치된 패키지 조회
pip show 패키지명	특정 패키지의 정보 조회
pip install --upgrade pip	pip 툴 업그레이드

Conda package repository 이용

- 아나콘다에서 관리하는 패키지 저장소
- <https://anaconda.org/anaconda/repo> (<https://anaconda.org/anaconda/repo>)
 - 패키지 검색 사이트

명령어	설명
conda install 패키지명	패키지 install
conda update 패키지명	패키지 업그레이드
conda remove 패키지명	패키지 제거
conda list	설치된 모든 패키지 조회
conda list 패키지명	특정 패키지의 정보 조회