

# 파이썬

## 1강. 파이썬 개요

# 1. 프로그래밍 언어의 개념

- 프로그래밍 언어 : 컴퓨터가 이해하는 말로 컴퓨터에서 작동하는 소프트웨어(엑셀, 한글, 인터넷 익스플로러 등)를 만드는 도구
- 프로그래머 : 프로그래밍 언어를 사용해 소프트웨어나 앱을 만드는 사람



# 1. 프로그래밍 언어의 개념

- **프로그래밍 (programming)**  
-> 프로그램을 만드는 것
- **프로그램 (program)**  
-> 미리 작성된 것
- **컴퓨터 프로그램 (computer program)**  
-> 컴퓨터가 무엇을 해야 할지 미리 작성한 진행 계획

Pro + Gram = ProGram  
미리 + 작성된 것 = 미리 작성된 것

## 2. 프로그래밍 언어의 종류

- 수백 가지가 넘는 종류 중 많이 사용되는 프로그래밍 언어는 C/C++, 자바(Java), HTML, PHP, 파이썬 등













다양한 스포츠



다양한 프로그래밍 언어

### 3. 프로그래밍 랭킹

Feb 2022	Feb 2021	Change	Programming Language		Ratings	Change
1	3	▲		Python	15.33%	+4.47%
2	1	▼		C	14.08%	-2.26%
3	2	▼		Java	12.13%	+0.84%
4	4			C++	8.01%	+1.13%
5	5			C#	5.37%	+0.93%
6	6			Visual Basic	5.23%	+0.90%
7	7			JavaScript	1.83%	-0.45%
8	8			PHP	1.79%	+0.04%
9	10	▲		Assembly language	1.60%	-0.06%
10	9	▼		SQL	1.55%	-0.18%

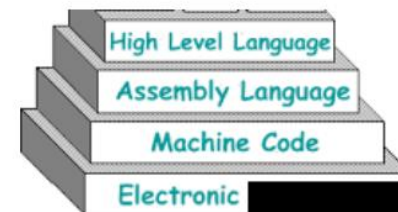
[출처 : <https://www.tiobe.com/tiobe-index/>]

## 4. 프로그래밍 언어의 구조

- **이진 숫자 (binary digit)**
  - 0과 1로 이루어진 컴퓨터의 언어
  - 이진 코드 (binary code) : 이진 숫자로 이루어진 코드
- **프로그래밍 언어 (programming language)**
  - 사람이 이해하기 쉬운 언어로 프로그램을 만들기 위해 만들어짐
- **소스 코드 (source code)**
  - 프로그래밍 언어로 작성한 프로그램
  - 코드 실행기
- **프로그래밍 언어는 컴퓨터가 이해할 수 없으므로**
  - 이를 이진 숫자로 변환해 주는 역할

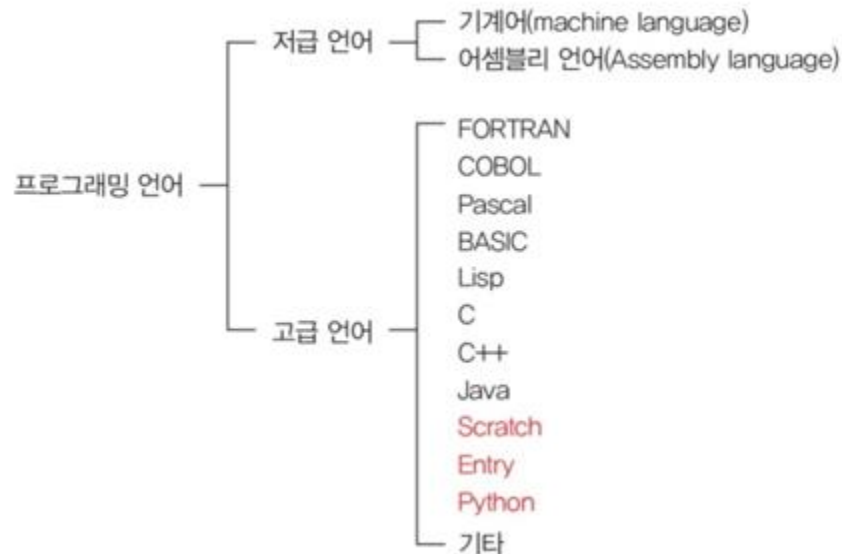
## 5. 프로그래밍 언어의 개발

- 컴퓨터는 1950년대에 들어와서 본격적으로 개발되기 시작
- 소프트웨어보다 하드웨어의 개발에 중점을 둔 시대
- 기계어(machine language)를 사용하여 프로그래밍이 어려움
- 그 후 어셈블리 언어(Assembly language)가 개발되어 사용됨
- 1957년부터 FORTRAN과 같은 고급 언어가 개발되기 시작
- 그 후 현재 상당히 인기를 끌고 있는 C, Java 등이 개발됨
- 최근 비전공자도 가능한 Scratch, Entry, Python 등이 개발됨



## 6. 저급 언어와 고급 언어

- 프로그래밍 언어는 인간의 의사를 컴퓨터에 전달하는 소프트웨어
- 저급 언어(low-level language)와 고급언어(high-level language)
- 저급 언어는 기계어와 어셈블리 언어
- 고급 언어는 C, C++, Java, Scratch, Python 등
- 고급 언어는 하드웨어와 멀고, 저급 언어는 하드웨어에 가까움





## 7. 저급 언어

- 저급 언어는 기계어와 어셈블리 언어 2가지
- 저급 언어를 통하여 하드웨어에 대한 직접적인 제어가 가능
- 저급 언어 사용은 하드웨어와 관련된 풍부한 지식이 필요
- 초보자는 저급 언어로 프로그램을 작성하기는 어려움

기계어는 0과 1의 조합으로 이루어진다.

'더하라'는 명령어는 00101010이고,

'빼라'는 명령어는 00101110이고,

'곱하라'는 명령어는 00101100,

'나누라'는 명령어는 00101011과 같이 나타낸다.

## 8. 어셈블리 언어

- 기계어의 명령을 알기 쉬운 기호로 표시해서 사용
- 덧셈의 경우 'ADD X'와 같은 형태로 표현
- 뺄셈의 경우 SUBTRACT X로 나타냄
- 기계어를 사용한 작업보다 훨씬 더 간편해 졌음
- 여전히 기계어에 가까운 저급 언어로 분류됨

기계어는 0과 1의 조합으로 이루어진다.

'더하라'는 명령어는 00101010이고,

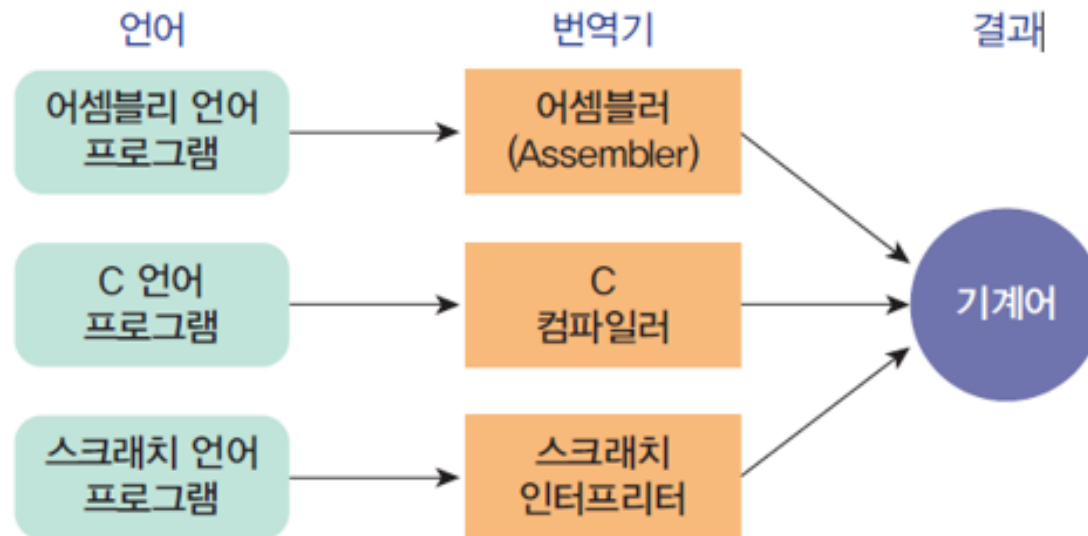
'빼라'는 명령어는 00101110이고,

'곱하라'는 명령어는 00101100,

'나누라'는 명령어는 00101011과 같이 나타낸다.

## 9. 컴파일러 방식과 인터프리터 방식

- 컴퓨터는 0과 1의 조합으로 된 숫자만 다룰 수 있음
- 중간 역할을 해 줄 프로그래밍 언어가 필요함
- 작성된 프로그램을 기계어로 바꾸어주는 소프트웨어가 필요
- 어셈블러, 컴파일러, 인터프리터



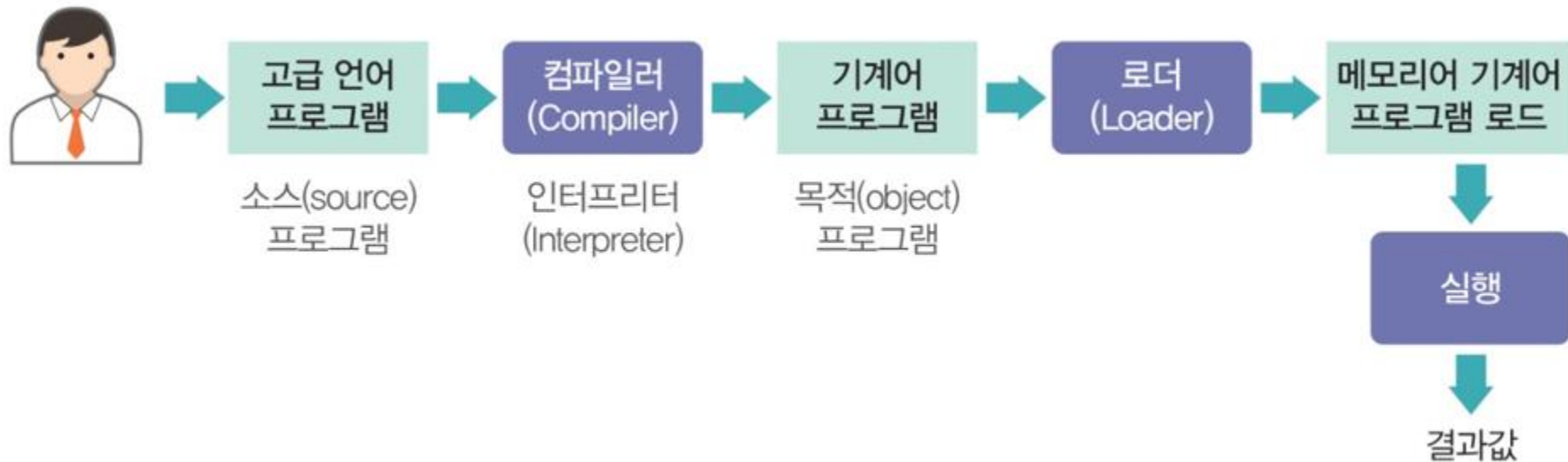
## 10. 컴파일러(compiler)와 인터프리터(interpreter) 특징

- 고급 언어는 반드시 기계어로 번역된 후 실행됨
- 번역과 실행 방법에 따라 컴파일러 방식과 인터프리터 방식
- 컴파일러의 경우 한 프로그램을 한꺼번에 번역한 뒤 실행
- 인터프리터의 경우 명령어 단위로 한 행씩 번역해가며 실행

특징 \ 방식	컴파일러 방식	인터프리터 방식
번역방법	프로그램 전체 번역	실행되는 행 단위로 번역
장점	한번 번역하면 빠른 시간 내에 전체 실행이 가능함	큰 기억 장치가 필요하지 않으며 번역 과정이 비교적 간단함
단점	프로그램의 일부를 수정하는 경우에도 전체를 다시 컴파일해야 함	반복문이 많은 경우 매 반복 때마다 번역해야함
결과물	목적 언어로 된 프로그램	실행의 결과
적용언어	FORTTRAN, Pascal, COBOL, Ada, C 등	Lisp, Scratch, Python 등

## 11. 고급 언어 프로그램의 전체 실행 순서

- ① 소스 프로그램이 번역기에 의해 목적 프로그램으로 변환됨
- ② 목적 프로그램이 로더(loader)란 기억장치에 장착
- ③ 실행하여 결과값이 만들어짐



## 12. 파이썬 역사

- 배우기도 쉽고 결과도 바로 확인할 수 있어 초보자에게 적합한 프로그래밍 언어
- 귀도 반 로섬(1956년~)이라는 프로그래머가 C 언어로 제작해 1991년에 공식으로 발표
- 사전적인 의미는 비단뱀으로 로고도 파란색과 노란색 비단뱀 두 마리가 서로 얹혀 있는 형태



## 13. 파이썬 장점

- **강력한 기능을 무료로 사용할 수 있다**
  - 파이썬은 오픈 소스이며, 비용을 지불하지 않고 무료로 사용 가능.  
다양한 추가 라이브러리도 무료
- **읽기 쉽고 사용하기 쉽다**
  - 직관적인 코드를 사용해 C나 자바 같은 언어보다 읽기 쉬워  
프로그램을 빨리 제작할 수 있어 비용 절감 효과 제공
- **사물인터넷과 잘 연동된다**
  - 라즈베리파이 기반의 사물인터넷이 파이썬을 잘 지원하므로  
사물인터넷 개발 및 운영에 적극 활용

## 13. 파이썬 장점

- **다양하고 강력한 외부 라이브러리들이 풍부하다**
  - 파이썬에서 제공하는 라이브러리뿐 아니라, 외부에서 제공하는 다양한 서드 파티(Third Party) 라이브러리까지 사용 가능
- **강력한 웹 프레임워크를 사용할 수 있다**
  - 파이썬의 웹 프레임워크를 사용해 강력하고 빠른 웹 환경을 구축 가능



## 14. 파이썬 단점

- 느린 속도
  - 파이썬은 컴파일러 언어가 아닌 스크립트 언어이기 때문에 컴파일러 언어보다 느림
  - 이를 보완하려고 많은 파이썬 패키지를 최적화시키고 있음
  - C언어에 비해 일반적으로 10~350배 느림
- 모바일 컴퓨팅 분야에 지원이 약하고 하드웨어 제어 등과 관련된 부분 사용이 어려움

## 15. 파이썬 특징

- 실행 시간에 자료형을 검사
- 객체의 멤버에 무제한으로 접근할 수 있음
  - 속성, 전용의 메서드 혹은 모듈을 만들어 제한할 수 있음
- 모듈, 클래스, 객체와 같은 언어의 요소가 내부에서 접근할 수 있고, 리플렉션을 이용한 기술을 쓸 수 있음

## 16. 파이썬 특징점

하이브리드 언어	java의 객체지향과 c언어의 함수지향의 특징을 동시에 갖는 대화형 인터프리터 언어이다.
동적인 자료형 결정	동적으로 자료형을 결정한다. 자료형 선언이 없다.
플랫폼 독립적 언어	대부분의 운영체제(윈도우, 리눅스, 유닉스 등)에서 동작한다.
간결하고 쉬운 문법	비교적 간결하고, 쉬운 문법으로 프로그래밍 구현이 가능하다.
풍부한 라이브러리와 오픈 소스	지속적으로 버전이 업그레이드 되면서 수많은 게임, 데이터 분석, 인터넷 프로그래밍 등 다양한 분야에서 사용할 수 있는 수 많은 라이브러리를 제공하며, 대부분 오픈 소스로 제공된다.
메모리 자동 관리	쓰레기 수집(Garbage Collection) 기능을 이용하여 사용이 종료된 데이터를 자동으로 제거하여 메모리의 효율성을 높여준다.

## 17. 파이썬 종류

Cython	C로 작성된 인터프리터
스택리스 파이썬	C 스택을 사용하지 않는 인터프리터
자이썬	자바 가상머신 용 인터프리터 (과거엔 제이파이썬이라 함)
IronPython	.NET 플랫폼 용 인터프리터
PyPy	파이썬으로 작성된 파이썬 인터프리터

## 18. 파이썬 활용

- 실제 파이썬은 많은 분야에서 상용 스크립트 언어로 채택되어 사용되고 있다.
- 도움말 문서도 잘 정리되어 있으며, 유니코드 문자열을 지원해서 다양한 언어의 문자 처리도 가능하다.
- 또한 배우기 쉽고, 짧은 시간에 사람들이 원하는 프로젝트 구현이 가능하다.
- 그리고 효율적인 고수준의 자료구조를 제공하여 데이터 처리가 매우 용이하다.



**THANK YOU**