

코딩(Coding)

→ 자동화 / 버튼 클릭했는데 바로 작동 => 반복되는 것을 지속적으로 시킬 때
내부적으로 계산이 되어서 작동이 되는 상태

모든 컴퓨터 프로그램은 코딩으로 구성되어 있다. (모든 명령 -> 자동화)

e.g. c-언어, 파이썬, 자바 등등 (컴퓨터 용어)

그렇다면, Language의 공통점?

모든 language의 기본 요소 1. 문법, 2. 단어 => Combination

단어? 의미(정보)를 포괄 / 정보의 그릇

문법? 어떤 순서, combination으로 도출되는가? / 완전한 대화

컴퓨터 용어에서 단어의 역할

=> 변수 (variable) [숫자, 문자]

중요부분!

1. 변수라는 그릇에다가 정보를 넣는 것 (assign) / variable assignment
2. Conditioning 이 필요하다 [if~] / if conditioning
3. 여러 번 반복 / loop
4. 함수를 익혀야 됨 [입력 -> 출력 (결과가 존재)] / packaging

<Python 용어 정리>

"="의 의미

➔ 오른쪽의 있는 정보(의미)를 왼쪽에 있는 변수(variable/단어)로 assign 한다

Print (-> 함수의 일종)

변수 xx 를 screen에다가 뿌리는 함수 [이미 프로그램 내에 만들어진 함수]

한칸 -> **cell**

다른 cell을 만드는 방법

➔ 파란 테두리 상태 / **a(above), b(below)**

<-> 지우는 방법: **x (delete)**

문자를 의미로 담고 싶을 때: (변수 형태x, 의미 형태)

➔ **""**, **"** 로 따옴표 해놓기

Run 키 (실행키)

1. Run 키를 직접 누르거나

2. **Shift + enter** 단축키 누르기

Type (-> 함수의 일종)

어떠한 변수의 형태인가? 찾는 함수

List ([] / **대괄호**로 묶는다) / str, int, float 든 어떤 요소가 들어있더라도 대괄호 안이면 무조건 list!

e.g. a = [1, 2, 3, 'love']

tuple[=list랑 같은 말 / () **소괄호**로 묶는다] => 보안이 훨씬 더 강함

e.g. a = (1, 'love', [1, 'bye'])

Int (정수)

E.g. a = 1

Float (실수)

e.g. a = 1.5

str (문자)

e.g. a = 'love'

dict

a = {'a': 'apple', 'b': 'banana'}

'표제어': '설명' 형태

Dictionary에 2개가 들어있음

밖에 {} **중괄호**를 씌움 -> dictionary

몇 개 구분은 , (comma)로

표제어: 설명 쌍은 : (colon)로