



Python 기초 - 자료 구조(문자열까지)

1. 자료 구조?



1. 자료 구조?

1. 숫자형
2. Boolean
3. 주석
4. 변수
5. 문자열

1) 숫자형의 정의 및 종류

- ✓ 숫자형 : 숫자 형태로 이루어진 자료형
 - ✓ 정수형 자료 : 1, -1, 0 등 정수 형태의 자료형, integer
 - ✓ 실수형 자료 : 1.0, -1.0, 0.1, 0.0 등 실수 형태의 자료형, float

2) 사칙연산

- ✓ 숫자형은 사칙연산이 가능하다.
- ✓ 파이썬에서는 사칙연산을 그대로 지원하며 다음과 같은 종류가 있다.
 - + : 더하기
 - : 빼기
 - * : 곱하기
 - / : 나누기
 - // : 나누었을 때의 몫
 - % : 나누었을 때의 나머지
 - ** : 제곱
- ✓ 사칙 연산의 우선순위는 일반 수학과 동일하다.

3) 숫자 사이의 형변환

- ✓ 정수형은 integer로, Python에서는 int로 사용한다.
- ✓ 실수형은 float로, Python에서는 float로 사용한다.
- ✓ 형태를 변환할 때는, int(값) 혹은 float(값)의 형태로 사용하면 된다.

1) Boolean의 뜻

- ✓ Boolean은 "참" 혹은 "거짓" 을 나타내는 표현이다.
 - ✓ 참이면 True
 - ✓ 거짓이면 False
- ✓ Boolean을 나타내기 위한 논리 연산자 and, or, not이 있다.
 - ✓ And : 둘 모두 참이어야 참
 - ✓ Or : 둘 중 하나라도 참이면 참
 - ✓ Not : Boolean을 반대 표현으로 바꾸는 방법

2) Boolean의 사용

- ✓ Boolean을 나타내기 위한 논리 연산자 and, or, not이 있다.
 - ✓ And : 둘 모두 참이어야 참
 - ✓ Or : 둘 중 하나라도 참이면 참
 - ✓ Not : Boolean을 반대 표현으로 바꾸는 방법

2) Boolean의 사용

- ✓ 대소 비교가 가능하다.
 - ✓ > : 크다
 - ✓ < : 작다
 - ✓ >= : 크거나 같다
 - ✓ <= : 작거나 같다
 - ✓ == : 같다
 - ✓ != : 다르다
- ✓ Example)
 - ✓ 1 == 2 => False
 - ✓ 1 == 1 => True
 - ✓ 1 != 2 => True

2) Boolean의 사용

- ✓ 대소 비교 연산자와 논리 구조가 같이 사용된다.
 - ✓ $1 > 0 \text{ and } 1 < 2 \Rightarrow \text{True}$
 - ✓ $\text{Not } (1 > 0) \Rightarrow \text{Not } (\text{True}) \Rightarrow \text{False}$

1) 주석의 뜻

- ✓ 주석은 코드가 아니고, 말이다.
- ✓ # 으로 구분자를 나타내준다.
- ✓ 즉, # 뒤에 오는 말들은 코드가 아닌 주석이다.

2) 주석의 필요성

- ✓ # 왜 주석을 다는가?
- ✓ # 프로그래밍은 "Sequence의 연속이다."
- ✓ # 코드가 길어지다보면, 어떤 코드가 어떤 역할을 하는지 헛갈리거나,
- ✓ # 다른 사람이 내 코드를 보았을 때 헛갈릴 수 있다.
- ✓ # 따라서 주석을 통해 정보를 적어줄 수 있다.

1) 변수의 뜻

- ✓ 정의 1) 변하는 수
- ✓ 정의 2) 변수는 프로그래머가 데이터를 저장할 수 있고, 나중에 변수명을 통해 저장된 데이터를 꺼내올 수 있는 메모리의 명명된 위치
 - ✓ 즉, 어떤 값을 변수명을 통해 메모리에 저장해두고, 변수 명을 통해 부를 수 있다.
 - ✓ 직관적으로,
A라는 사람의 특징을 A의 이름으로 뇌 속에 저장하고,
A의 특징을 A의 이름으로 뇌속에서 꺼내오는 것과 같다.

2) 변수의 생성

- ✓ 일반적으로 변수명은 영어로 생성하나, 한글도 가능하다.
- ✓ 파이썬에서는 등호(=)을 통해 지정한다.
 - ✓ “지정한다”라는 표현을 사용한 이유는, 이름으로 특징을 기억에 메모리에 한 부분에 자리를 지정하기 때문
- ✓ 변수는 한 번에 여러 개를 지정할 수 있다.

3) 변수의 생성의 조건

- ✓ 변수명은 숫자가 될 수 없다.
- ✓ 변수명은 숫자로 시작할 수 없다.
- ✓ 변수명에는 하이픈(-)을 사용할 수 없다.
 - ✓ 변수명이 길어질 경우 언더바(_)를 사용하자.
- ✓ Python의 변수명은 대소문자를 구분한다.
- ✓ 변수명을 입력했는데, 색상이 바뀐다면 해당 변수명은 다른 이름으로 바꾸자.
 - ✓ Python의 내장 함수의 이름과 겹침을 의미한다.
 - ✓ 내장 함수를 새로운 변수로 덮을 경우, 나중에 해당 함수를 사용하지 못할 수 있다.

4) 변수 덮어쓰기

- ✓ 중복되는 변수명이 있다면, 나중에 정의된 변수가 선택된다.
- ✓ 이를 이용해서 " $a = a + 1$ " 과 같은 반복적인 작업도 많이 해준다.
- ✓ 반대로, 변수명을 정의하지 않고 사용하면 Name Error가 발생한다.

5) 변수 삭제

- ✓ 변수를 삭제하고 싶으면 del을 사용하면 된다.
- ✓ del 변수명



1) 문자열의 뜻

- ✓ 문자열은 “문자로 이루어진 자료 집합” 이다.

2) 문자열의 생성하기

- ✓ 문자열은 따옴표를 사용해 만들 수 있다.
 - ✓ 이 때, 작은 따옴표로 시작했다면, 문자열이 끝날 때까지는 작은 따옴표를 사용하면 안된다.
 - ✓ 만약 문자열 내부에 작은 따옴표를 넣고 싶다면, 큰 따옴표로 시작해 내부에 작은 따옴표를 넣으면 된다.
- ✓ Str() 내장 함수를 이용해 형태를 바꿀 수 있다.
- ✓ "" ""은 여러줄에 걸쳐서 문자열로 만들 수 있다.
 - ✓ 주로 여러줄의 주석을 달 때 많이 사용한다.



3) 문자열과 특수문자 표현

- ✓ 문자열에서 문장을 바꾸고 싶다면, `wn`을 사용하면 된다.
- ✓ 문자열에서 띄어 쓰고 싶다면, `wt`를 사용하면 된다.

4) 문자열 연산

- ✓ 문자열 A와 문자열 B를 더하면, AB 형태로 이어 붙어 나타난다.
- ✓ 문자열 A를 5번 곱하면, AAAAA 형태로 5번 이어 붙어 나타난다.