

## 자료형과 연산자

1. 변수 파이썬은 변수를 선언하기 위한 타입이 없으므로 변수 선언 과정이 없다.

### 변수 이름 규칙

- 문자, 숫자, 밑줄(\_) 포함
- 숫자로 시작할 수 없다.
- 대소문자를 구분한다.
- 공백, 문장부호, 특수문자 불가
- 예약어는 사용할 수 없다.

## 2. 화면 입출력

- input() : 사용자로부터 값을 입력 받기 위한 함수. 입력한 값은 항상 문자열이다.
- print() : 화면 출력 sep, end 사용

## 3. 기본 자료형

### 숫자형

- 정수 (int)
- 실수(float) 정수형과 정수형의 나눗셈 연산의 결과는 실수형
- 복소수(complex)
- 실수와 허수의 합으로 나타내는 수 체계
- 허수는 j문자를 붙여서 표현한다.

### 문자형

- 겹따옴표('')와 홑따옴표('')로 묶어서 사용한다.
- 여러줄 문자열 : 겹따옴표나 홑따옴표를 3개 사용한다.

## 탈출 문자

\n	줄 바꿈
\t	탭
\r	리턴(행의 첫 번째 열로 돌아옴)
\0	널(null)
\\	\ 문자 표시
\'	'(홀따옴표) 문자 표시
\"	"(겹따옴표) 문자 표시

## 문자열 연결

- 문자열과 문자열을 + 하면 문자열을 연결한다
- 문자열을 공백으로 연결해도 문자열을 연결한다 ex) "Hello" "World"=> "HelloWorld"
- 문자열과 숫자를 \* 연산 하면 문자열을 곱셈한 숫자만큼 반복한다.

## 문자열 인덱싱

- 첫 문자의 인덱스 : 0 음수는 맨 뒤의 문자부터 의미 맨 마지막 : -1

## 문자열 슬라이싱

- [start:stop] 부분 문자열을 뺀다. stop 위치의 문자는 포함하지 않는다
- [start:stop:step] 매 step번째 아이템을 추출해준다.

## 논리형(Bool) True / False

- False로 판단되는 값

- None
- False
- 숫자 타입 0에 해당하는 것(예: 0, 0L, 0.0, 0j)
- 빈 문자 (예: "", "")
- 빈 튜플 또는 리스트 (예: (), [])
- 빈 딕셔너리 (예: {})

#### 4. 포매팅 문자, 숫자, 날짜 데이터에 형식을 지정

'{[인덱스]:[공백대체문자][정렬방법][자릿수][타입]}' . format(변수명)

- 인덱스 : format()함수의 인수 중에서 해당 자리에 출력할 인수의 인덱스
- 공백 대체 문자 : 공백 대체 문자를 작성할 시 정렬문자를 꼭 적는다.
- 정렬 방법 : < 왼쪽 정렬, > 오른쪽 정렬, ^ 가운데 정렬
- 자릿수 : 변수의 값을 출력할 최대 자릿수를 지정
- 타입 : 출력 형식을 지정.

'd'는 10진 정수, 'f'는 실수(부동소수점), 's'는 문자열을 의미.

숫자 형식 'b'는 2진수, 'o'는 8진수, 'x'는 16진수로 출력.

- 날짜 출력 %Y 연도 %m 월 %d 일 %H 시간 %M 분 %S 초

#### 5. 산술 연산자

// 몫 % 나머지 \*\* 제곱

#### 6. 문자열 다루기

"", ''	문자열을 만든다
+	문자열을 연결한다
len("문자열")	문자열의 길이를 반환한다
[start:stop]	문자열을 start 위치부터 stop위치까지 자른다. (stop은 포함 안 됨)
[start:stop:step]	문자열을 start 위치부터 stop위치까지 매 step마다 반환한다.
split('delimiter')	문자열을 delimiter로 잘라 리스트로 반환한다
'delimiter'.join(["str1",...])	문자열 리스트를 delimiter로 연결한다
capitalize()	첫 문자를 대문자로, 나머지 문자를 소문자로 바꾼다.
upper / lower()	문자열을 모두 대문자/소문자로 바꾼다
startswith() / endswith()	특정 문자로 시작하는지/끝나는지를 식별해서 논리값을 반환한다
find() / index()	특정 문자의 인덱스를 반환한다
isalnum() / isalpha() / isnumeric() / isdecimal()	문자열이 숫자인지 문자인지 판별한다
replace(old, new)	old문자를 new문자로 치환한다