

# 1주차

## 1-1. 파이썬 프로그래밍의 기초, 자료형

### print()

```
print('Hello World!')
```

*Hello World!* 를 출력하는 명령어(함수)

### 자료형과 기본 자료형

자료형(datatype): 자료(data)의 형식

- 숫자(number): 정수 및 실수
- 문자열(string): 문자의 나열로 이루어진 자료형
- 불(boolean): True, False

### 주석

- 메모와도 같은 기능
- 프로그램에 영향을 끼치지 않음
- `#` 을 맨 앞 줄에 삽입

```
# Hello World!를 출력  
print('Hello World!')
```

## 1-2. 숫자형 변수

### 정수형

정수형(Integer): 정수를 가리키는 자료형

```
a = 123  
b = -7  
c = 0
```

- `a`, `b`, `c` 모두 정수형 변수

### 실수형

실수형(Floating-point): 소수점이 포함된 숫자

```
a = 1.2  
b = -3.45
```

- `a`, `b` 모두 실수형 변수

### 숫자형을 활용하기 위한 연산자

사칙연산

```
a = 3  
b = 4  
print(a + b)  
print(a * b)  
print(a / b)  
print(a // b)
```

```
print(a % b)
print(a ** b)
```

## 1-3. 문자열 자료형 변수

### 문자열

문자열(string)

- "Life is too short, You need Python"
- "a"
- "123"

### 문자열 만들기

1. 큰따옴표(")로 양쪽 둘러싸기
  - "Hello World"
2. 작은따옴표(')로 양쪽 둘러싸기
  - 'Python is fun'
3. 큰따옴표 3개를 연속(""")으로 써서 양쪽 둘러싸기
  - """Life is too short, You need Python"""
4. 작은따옴표 3개를 연속('')으로 써서 양쪽 둘러싸기
  - '''Life is too short, You need Python'''

### 문자열에 따옴표 포함시키기

1. 문자열에 작은따옴표(') 포함시키기
  - 문자열을 큰 따옴표로 감싸기
  - "Python's favorite food is perl"
2. 문자열에 큰따옴표(") 포함시키기
  - 문자열을 작은따옴표로 둘러싸기
  - "Python is very easy." he says.'
3. 백슬래시(\)를 사용해서 작은따옴표(')와 큰따옴표(")를 문자열에 포함시키기
  - 'Python\'s favorite food is perl'
  - "\"Python is very easy.\" he says."

### 여러 줄인 문자열을 변수에 대입하고 싶을 때

1. 줄을 바꾸는 이스케이프 문자 \n 삽입하기
  - "Life is too short\nYou need python"

### 이스케이프 문자

문자	설명
\n	개행
\t	탭 간격
\\	문자 \를 그대로 표현
\'	작은따옴표(')를 그대로 표현
\"	큰따옴표(")를 그대로 표현

### 문자열 연산하기

1. 문자열 더해서 연결하기(Concatenation)

```
head = "Python"
tail = " is fun!"
print(head + tail)
```

## 2. 문자열 곱하기

```
a = "python"
print(a * 2)
```

## 3. 문자열 곱하기 응용

```
print("=" * 50)
print("My Program")
print("=" * 50)
```

## 4. 문자열 길이 구하기

```
a = "Life is too short"
print(len(a))
```

## 문자열 인덱싱과 슬라이싱

"I love Hello, World"

0	1	2	3	4	5	6	7
I		l	o	v	e		H

```
# 문자열 인덱싱
a = "I love Hello, World"
print(a[4]) # v
print(a[12])# ,

print(a[-1]) # d
print(a[-2]) # l
print(a[-5]) # W

# 문자열 슬라이싱
print(a[0:4]) # I lo
print(a[0:3]) # I l
print(a[0:5]) # I lov
print(a[5:7]) # e
print(a[7:]) # Hello, World
print(a[:12]) # I love Hello
print(a[:]) # I love Hello, World
print(a[7:-7]) # Hello
```

```
a = "20010331Rainy"
date = a[:8]
weather = a[8:]
print(date)
print(weather)
```

## 문자열 관련 함수

- 문자 개수 세기(count)

```
a = "hobby"
print(a.count('b'))
```

- 위치 알려주기(find)

```
a = "Python is the best choice"
print(a.find('b')) # 문자열에서 b가 처음 나온 위치
print(a.find('k')) # 없는 문자를 찾으면?
```

- 위치 알려주기2(index)

```
a = "Life is too short"
print(a.index('t')) # 문자열에서 t가 처음 나온 위치
print(a.index('k')) # 없는 문자를 찾으면? -> 오류 발생
```

- 문자열 삽입(join)

```
a = ",".join('abcd')
print(a)
```

- 소문자를 대문자로 바꾸기

```
a = "hi"
print(a.upper())
```

- 대문자를 소문자로 바꾸기

```
a = "HI"
print(a.lower())
```

- 왼쪽 공백 지우기(lstrip), 오른쪽 공백 지우기(rstrip), 양쪽 공백 지우기(strip)

```
a = " hi "
print(a.lstrip())
print(a.rstrip())
print(a.strip())
```

- 문자열 바꾸기(replace)

```
a = "Life is too short"
print(a.replace("Life", "Your leg"))
```

- 문자열 나누기(split)

```
a = "Life is too short"
b = a.split() # 공백을 기준으로 문자열 나눔
print(b)
c = "a:b:c:d"
d = c.split(':') # : 기호를 기준으로 문자열 나눔
print(d)
```

## 1-4. 변수와 입력

### 변수 만들기/사용하기

```
# 변수 선언과 할당
pi = 3.14159265
r = 10

# 변수 참조
print("원주율 =", pi)
print("반지름 =", r)
print("원의 둘레 =", 2*pi*r)
print("원의 넓이 =", pi*r*r)
```

## 복합 대입 연산자

```
a = 30

# a = a + 10
a += 10
# a = a - 10
a -= 10
# a = a * 10
a *= 10
# a = a / 10
a /= 10
# a %= 10
a = a % 10
# a = a ** 2
a **= 2
```

```
string = "안녕하세요"
string += "!!"
print(string)
```

## 사용자 입력: `input()`

```
# input() 괄호 안의 입력한 내용을 "프롬프트 문자열" 이라고 하며,
# 사용자로부터 입력을 요구하는 안내 내용을 포함
# 사용자가 입력한 내용을 "문자열"로 받음

string = input('인사말을 입력하세요 : ')
print(string)
```

## 입력 내용을 숫자로 바꾸기: `int()`, `float()`

```
# 정수
a = input('정수 a 입력 : ')
int_a = int(a)

b = input('정수 b 입력 : ')
int_b = int(b)

print("a + b =", a + b)
print("int_a + int_b =", int_a + int_b)
```

```
# 실수
a = input('실수 a 입력 : ')
```

```
float_a = float(a)

b = input('실수 b 입력 : ')
float_b = float(b)

print("a + b =", a + b)
print("float_a + float_b =", float_a + float_b)
```

- `ValueError` : 자료형을 변환할 때 '변환할 수 없는 것'을 변환하려고 하면 발생하는 예외

숫자를 문자열로 바꾸기: `str()`

```
a = 10
b = 20
str_a = str(a)
str_b = str(b)

print("a + b =", a + b)
print("str_a + str_b =", str_a + str_b)
```