
Lecture 2. 입력과 문자열

BAE SEKANG(배세강)



Contents

1. Data Types

- 기본 자료형
- type()
- 자료형 변환

2. Input

- 입력
- input()의 활용

3. String

- 문자열 표현
- 문자열 연산
- Escape 문자
- f-string
- len()



Data Types

♣ 기본 자료형

자료형	표현	예시
정수	int	3, 0, -10, 9999
실수	float	3.14, -0.01, 2.0
문자	str	"Hello World!", 'c', "123", '@#\$\$%'
논리	bool	True, False
복소수	complex	(5 + 3j), (2 + 0j)
...	...	...



Data Types

♣ type()

```
print(type(10))
```

```
print(type(3.14))
```

```
print(type("Hello World!"))
```

```
print(type(True))
```

```
>> <class 'int'>
```

```
>> <class 'float'>
```

```
>> <class 'str'>
```

```
>> <class 'bool'>
```

Q. class는 뭐지?



Data Types

♣ 자료형 변환

- int(), str(), float()

```
a = 3  
print(type(a))
```

```
>> <class 'int'>
```

```
b = float(a)  
print(b)
```

```
>> 3.0
```

```
c = str(b)  
print(c)  
print(type(c))
```

```
>> 3.0
```

```
>> <class 'str'>
```



Data Types

♣ 자료형 변환

- int와 float의 변환

```
# float -> int
```

```
a = 123.456  
print(int(a))
```

```
>> 123
```

```
# int -> float
```

```
b = 5  
print(float(b))
```

```
>> 5.0
```



Data Types

♣ 자료형 변환

- `int(3.14)`와 `int("3.14")`의 차이는?

```
a = 3.14
```

```
b = "3.14"
```

```
print(int(a))
```

```
print(int(b))
```

```
>> 3
```

```
>> ValueError
```



Input

♣ 입력

- input()

```
# input()
```

```
word = input()  
print(word)
```

```
print(type(word))
```



Input

♣ 입력

- input 함수 내부에 설명을 적을 수 있다.

```
# input("설명")  
  
food = input("먹고 싶은 음식은? ")  
print(food)  
  
print(input("마시고 싶은 음료는? "))
```



Input

♣ input()의 활용

- 두 숫자를 입력 받아 더한 결과를 출력하는 프로그램

이 프로그램은 무엇이 문제일까?

```
a = input("첫 번째 숫자: ")  
b = input("두 번째 숫자: ")  
print(a + b)
```



Input

♣ input()의 활용

- 두 숫자를 입력 받아 더한 결과를 출력하는 프로그램

int형끼리 덧셈을 해야한다!

```
a = int(input("첫 번째 숫자: "))  
b = int(input("두 번째 숫자: "))  
print(a + b)
```



String

♣ 문자열 표현

- '문자', "문자", """문자"""

```
str1 = 'c'
```

```
str2 = "apple"
```

```
# multiline string
```

```
str3 = '''
```

```
이것은 아주 긴 문장을 나타내며, 띄어쓰기와  
줄바꿈도 포함된 그런 문장입니다. Python 재밌다!  
'''
```



String

♣ 문자열 연산

- 문자열 합치기(String Concatenation): +를 사용

```
a = "가"  
b = "방"  
print(a+b)
```

>> 가방

```
print("3" + "5")
```

>> 35



String

♣ 문자열 연산

- 문자열 곱셈: '문자' * N의 형태로, '문자'가 N개가 된다.

```
print("E" * 10)
```

```
>> EEEEEEEEEEEE
```

```
print("안녕" * 3)
```

```
>> 안녕안녕안녕
```

```
print(type("This is str" * 5))
```

```
>> <class 'str'>
```



String

♣ Escape 문자

예약 문자	내용
\n	줄 바꿈(New line)
\t	Tab
\\	\ 문자 자체
'\''	' 문자 자체
'\"'	" 문자 자체
\b	Backspace
...	...



String

♣ String Formatting

- %를 활용한다.

```
# Example
name = "홍길동"
age = 26
pi = 3.141592

print("%s의 나이는 %d살입니다." % (name, age))
print("PI: %.2f" % pi)
```



String

♣ f-string

- f"{연산}"의 형태로 사용

```
# Example
money = 10000

print(f"내가 가진 돈은 {money}원입니다.")

a = 1
b = 2
print(f"{a} + {b} = {a+b}")
```



String

♣ len()

- 문자열의 길이를 반환하는 함수

```
my_string1 = "abcde"  
print(len(my_string1))
```

>> 5

```
my_string2 = "Good Night!"  
k = len(my_string2)  
print(k)
```

>> 11

- 함수에 대해서는 이후에 더 자세히 다룬다.



String

♣ 문자열 메소드

- .upper(), .lower(): 대소문자 변환
- .replace(old, new): old를 new로 변환
- .find(target): target의 index 반환
- ...

```
a = "ABCDE"
print(a.lower())
a = a.replace("BC", "HI")
print(a)
```



String

♣ print() 심화

- end: 출력할 문자 끝에 붙여 출력할 문자를 정함(기본은 "\n")
- sep: print의 인자 사이에 넣어 출력할 문자를 정함(기본은 " ")

```
print("HELLO")  
print("GOOD MORNING!", end="^^")
```

```
print()  
print(1, 2, 3, 4, 5)  
print(10, 20, 30, 40, 50, sep="★")
```



QUIZ

Problem 1.

`print(type(-21.3))`을 실행한 결과는 무엇일까?

Problem 2.

두 정수를 a , b 를 입력 받을 때, $a \times b$ 의 값을 출력하는 프로그램을 작성한다.

Problem 3.

문자열 A 와 B 를 합친 문자열의 길이를 출력하는 프로그램을 작성한다.



THANK YOU!

Any Question?