

Fastcampus

Computer Science Extension School

Python Basic_Day2

Review

- Computer Science
- Computation vs Calculation
- Computational Thinking
- Computer Architecture
- Python Introduce
- Install & run Jupyter
- Arithmetic

Jupyter Notebook

```
$ pip install jupyter  
$ pip list
```

```
$ jupyter notebook
```

Hello python!

So, let's try!!

```
print("hello python!")
```

Numbers & Math

<object> <operator> <object>

```
print(3 + 7)
print(10 - 3)
print(15 / 7)
print(34 * 100)
```

Numbers & Math

```
print(15 / 7)
print(15 / 5)
type(15 / 5)

print(15 // 5)
type(15 // 5)

print(7 % 3)

print(15 ** 3)

print(34 * 100)
print(3 * 2.5)
type(3 * 2.5)
```

Boolean

```
print(3 < 7)
print(10 < 3)
print(15 > 7)
print(3 >= 3)
print(3 <= 10)
print(34 == 100)
print(34 != 100)
```

Variable

```
print("hello python!")  
hello = "hello"  
python = "python!"  
print(hello, python)
```

```
num1 = 14  
num2 = 5  
  
print(num1+num2)  
print(num1-num2)  
print(num1*num2)  
print(num1/num2)
```


Small Project

반지름(`r=10`)을 선언한 뒤, 이를 이용하여 원의 지름, 둘레, 넓이, 구의 겹넓이, 부피를 각각 출력하는 파이썬 파일을 만들어보세요.(`pi=3.1415`)

sample output

```
r = 10 ==> print("r =", r)
d = 20
c = 62.830
a = 314.15
gnb = 1256.0000
v = 4188.666666666667
```

Let's Code PYTHONIC

Important Python Enhance Proposal

Layout

- 들여쓰기: 공백 4칸 or 탭(섞어쓰면 안됨)
- 한 줄은 79자(120자도 상관없음)
- 클래스정의와 최상위 함수는 두 줄을 띄움
- 클래스 내 메소드는 한 줄을 띄움

Important Python Enhance Proposal

Variables

- `_variable` : 내부적으로 사용되는 변수
- `print_` : 파이썬 키워드와 충돌 방지

Naming Convention

- 클래스 이름은 `CamelCase`
- 함수, 변수, 메소드 이름은 `snake_case`

파이썬에서 쓰이지 않는 네이밍 규칙

- `chHungarianNotation`
- `javaScriptStyleCamelCase`

Syntax

문법, 구조, 또는 언어 문장 내에 있는 구성요소의 순서

"나는 입니다 학생" (Syntax Error)

"나는 학생 입니다" (Syntactically Valid)

"Python"5 (Syntax Error)

3.6 * 12 (Syntactically Valid)

type casting

`float(3) --> int to float`

`int(3.6) --> float to int`

`str(1) --> int to string`

`int("12") --> string to int`

input

```
name = input("What is your name? ")  
print("Hi, ", name)
```

input with evaluation

```
input("How old are you? ")  
eval(input("How old are you? "))
```


type casting with input

```
int(input("How old are you? "))
```

Small Project Again!

사용자의 입력을 받아 반지름(r)을 선언한 뒤, 이를 이용하여 원의 지름, 둘레, 넓이, 구의 겉넓이, 부피를 각각 출력하는 파이썬 파일을 만들어보세요.
($\pi=3.1415$)

Strings

Strings

```
some_string = "python"  
len(some_string)
```

- index

p	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

```
some_string[3:5] = "ho"  
some_string[1:5:2] = "yh"  
some_string[::] = some_string[0:len(some_string):1]  
some_string[::-1] = some_string[-1:-len(some_string):-1]  
some_string[::-1] = "nohtyp"
```

but, strings are immutable

```
>>> some_string = "python"

>>> some_string[0] = "c"
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'str' object does not support item assignment

>>> some_string = "c" + some_string[1:]
```

String Functions

```
func = "python is easy programming language"  
func.count('p')
```

```
func.find('p')
```

```
comma = ","  
func = comma.join('python')
```

```
func.split(',')
```

```
python_is_easy = "python is easy"  
python_is_easy.split()
```

```
python_is_easy.replace("python", "golang")
```

String Functions

```
some_string = "    computer    "  
some_string.strip()
```

```
some_string = ",,,Fastcampus..."  
some_string.strip(",")  
some_string.strip(".")
```

String Formatting - old way

```
print("I have a %s, I have an %s." % ("pen", "apple"))
```

```
%s - string  
%c - character  
%d - Integer(decimal)  
%f - floating-point  
%o - 8진수(Octal)  
%x - 16진수(hexadecimal)  
%% - %
```


String Formatting - New way

```
print("I have a {}, I have an {}".format("pen", "apple"))
```

```
print("I have a {0}, I have an {1}".format("pen", "apple"))
```

```
print("I have a {0}, I have an {0}".format("pen", "apple"))
```