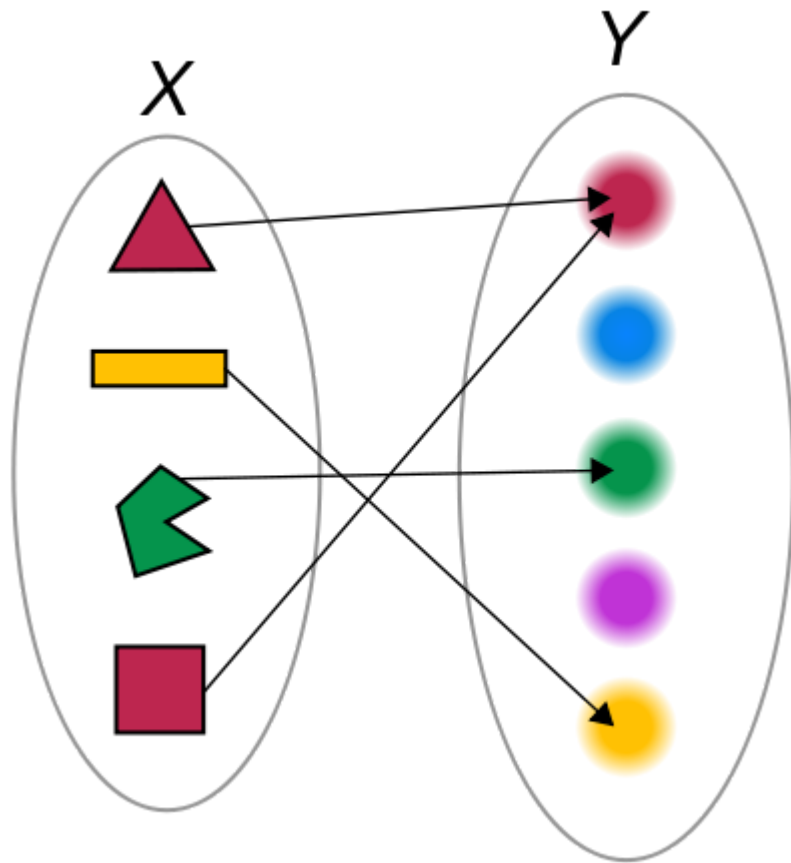


Fastcampus Data Science SCHOOL

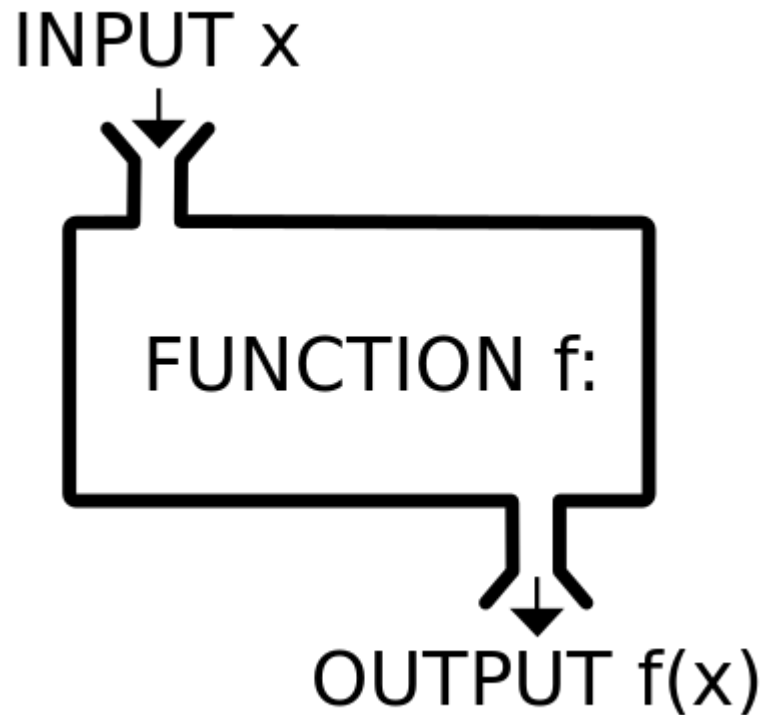
Python Basic

function



- 수학적 정의: 첫 번째 집합의 임의의 한 원소를 두 번째 집합의 오직 한 원소에 대응시키는 대응 관계
- x : 정의역 y : 공역

function



- 프로그래밍에서의 함수: 입력값을 내부에서 어떤 처리를 통해 결과값을 출력하는 것

function

```
def function(parameter):  
    실행문1  
    실행문2  
    ...
```

function

```
def awe_sum(a,b):  
    result = a + b  
    return result  
  
a = 2  
b = 3  
print(awe_sum(a,b))
```

function without input

```
def print_hello():  
    return "hello"  
  
result_hello = print_hello()  
print(result_hello)
```

function without return

```
def func_wo_return(a):  
    print("This is function without return for " + str(a) + " times")  
  
func_wo_return()
```

function with multiple return

```
def mul_return(a):  
    b = a + 1  
    return a,b
```


return skill

```
def id_check(id):  
    if id == "admin":  
        print("invalid id: admin")  
        return  
    print("valid id: ", id)
```

parameter with initialize

```
def say_hello(name="Fool", nick=True):  
    print("Hi, ", name)  
    if nick == True:  
        print("But, you are Fool")  
    else:  
        print("Oh, you are not Fool")
```

초기값을 설정할때 항상 그 인자를 마지막에 두어야 합니다.

arguments

```
def mul_sum(*args):  
    sum = 0  
    for i in args:  
        sum += i  
    return sum
```

variable outside function

```
a = "hello"
def glob_test(a):
    a += "world"
    return a

glob_test(a)
print(a)
```

```
a = "hello"
def glob_test(x):
    x += "world"
    return x

glob_test(a)
print(a)
```

Leap year

4로 나뉘어 떨어지면 윤년,
100으로 나뉘어 떨어지면 평년,
400으로 나뉘어 떨어질땐 윤년

Leap year(answer)

```
leap = False
def is_leap(y):
    if y % 4 == 0 and (y % 100 != 0 or y % 400 == 0):
        leap = True
    return leap

y = int(input("Is leap?? "))
print(is_leap(y))
```

Recursive

```
times = int(input("How many times want to curse the beast?: "))
def recurse_beast(a):
    if a == 0:
        print("curse complete!")
    else:
        print("Fusion!!!(%d times left)" % a - 1)
        recurse_beast(a-1)

recurse_beast(times)
```

File I/O

File I/O

```
f = open(filename, mode)
f.close()
```

mode

r - 읽기모드

w - 쓰기모드

a - 추가모드(파일의 마지막에 새로운 내용을 추가)

Create New File

```
f = open("Newfile.txt", 'w')  
f.close()
```

Write text

```
f = open("Newfile.txt", 'a')
for i in range(1,11):
    text = "line %d. \n" % i
    f.write(text)
f.close()
```

Read text

```
f = open("Newfile.txt", 'r')
text = f.readline()
print(text)
f.close()
```

Read All text using readlines

```
f = open("Newfile.txt", 'r')
texts = f.readlines()
for text in texts:
    print(texts)
f.close()
```

Add text

```
f = open("Newfile.txt", 'a')
for i in range(11, 20):
    text = "New line %d \n" % i
    f.write(text)
f.close()
```

Get rid of f.close()

```
with open("foo.txt", 'w') as f:  
    f.write("foo is text dummy")
```

오늘의 숙제

- [Hackerrank](#) 10 Days of Statistics Day0 풀어보기