

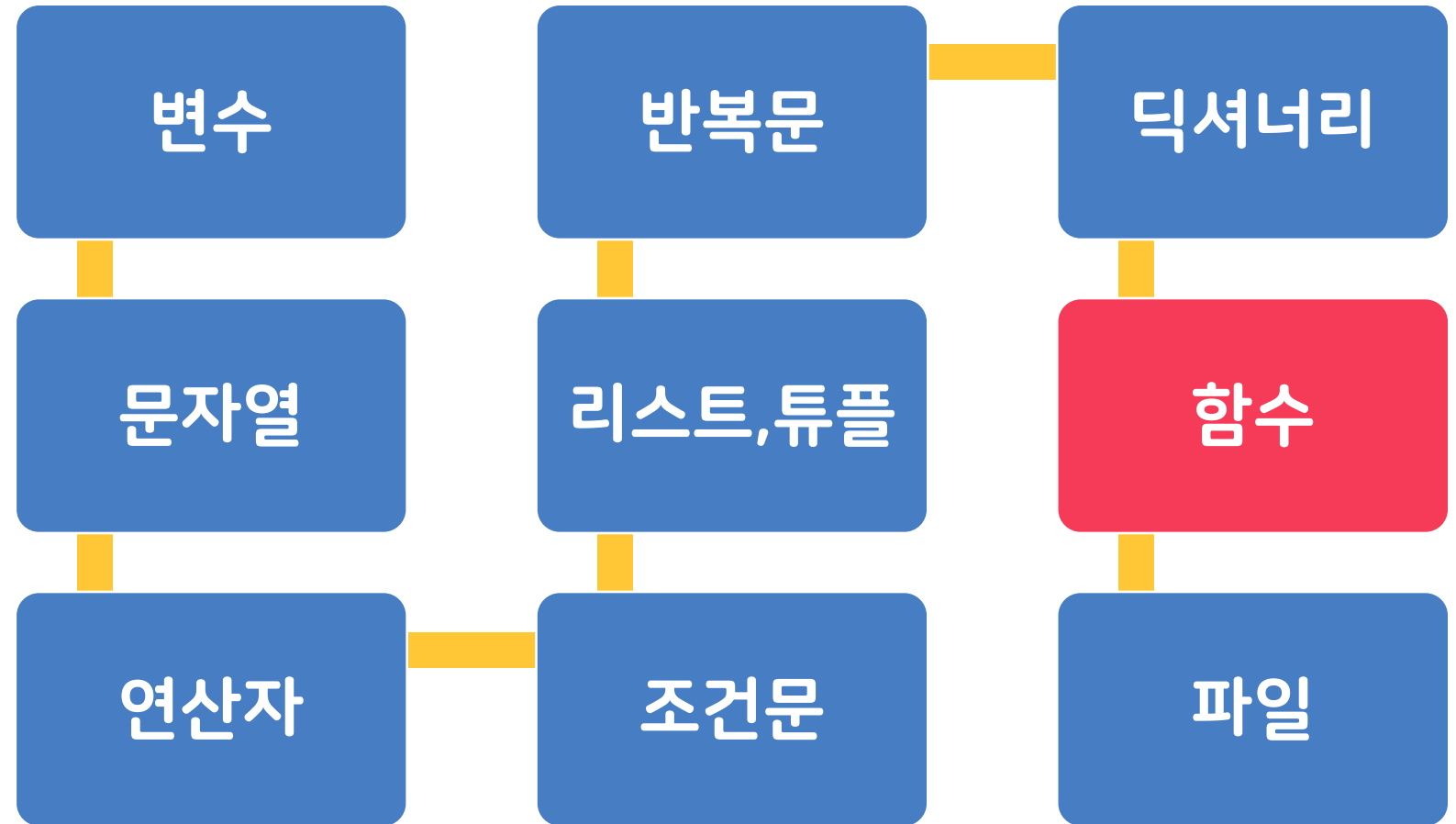


Smart Media
스마트미디어인재개발원

임보미 강사



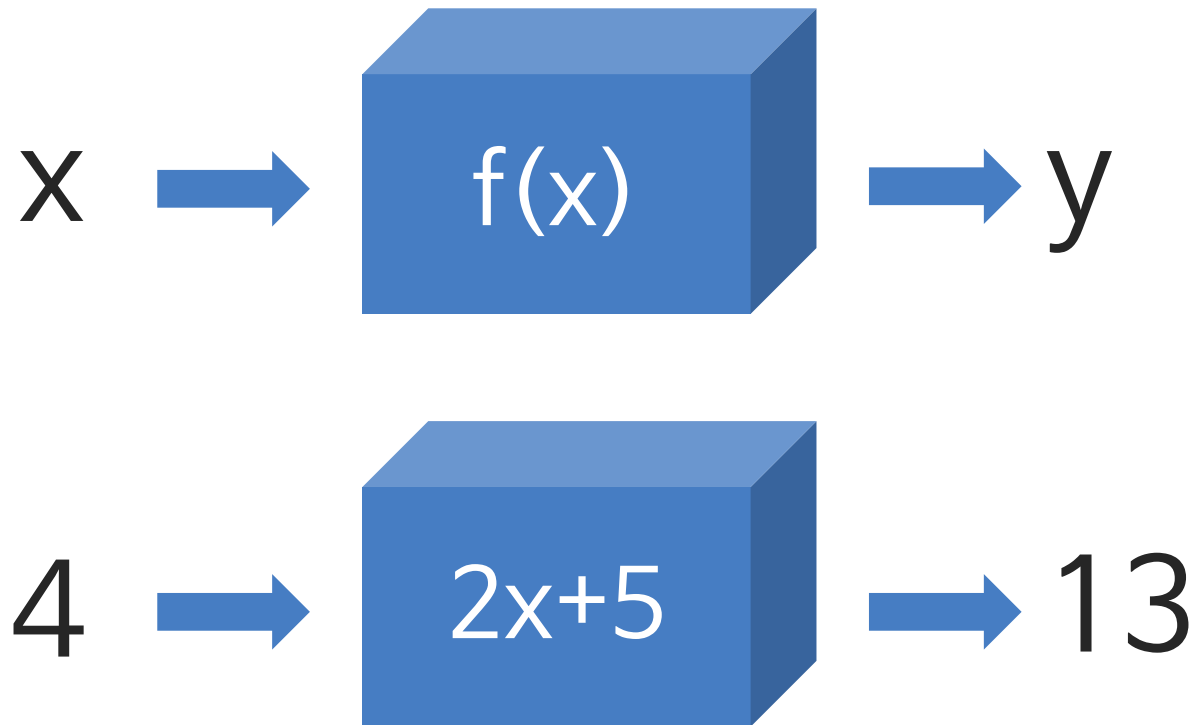
수업 진행방향





학습목표

- 함수에 대해 알 수 있다.
- 함수를 정의하고 활용 할 수 있다.



함수란?

- 하나의 특별한 목적의 작업을 수행하기 위해 독립적으로 설계된 코드의 집합

함수를 사용하는 이유

- 반복적인 프로그래밍을 피할 수 있다.
- 모듈화로 인해 전체적인 코드의 가독성이 좋아진다.
- 프로그램에 문제가 발생하거나 기능의 변경이 필요할 때에도 손쉽게 유지보수가 가능하다.

데이터 전처리로 텍스트의 "ㅋ"을 모두 제거하고 싶을 때

- 데이터는 review_list1, review_list2에 대입되어 있음

```
review_list1 = ["월 스미스가 하드캐리ㅋㅋㅋㅋㅋㅋ",  
                "자스민 너무 멋지고 팬 엄청나게 생길듯"]  
review_list2 = ["색감도 노래도 너무 화려하고 재밌었어요ㅋㅋ",  
                "오늘부터 디즈니 팬입니다ㅋㅋ",  
                "디즈니 의 새로운해석도 놀랍고. 월스미스도 신의한수!"]
```

```
s = "월 스미스가 하드캐리ㅋㅋㅋㅋㅋㅋ"
s.replace("ㅋ", "")
```

'월 스미스가 하드캐리'

define : 정의를 내리다

def

함수명(매개 변수):

(colon, 콜론)

실행문장

return 반환 변수

들여쓰기 (Tab, Space*4)

함수호출

- 함수명(인수1, 인수2)

```
def 함수명(입력인수):  
    실행문장  
    return 반환변수
```

```
def number_sum(num1, num2):  
    result = num1 + num2  
    return result
```

```
number_sum(3, 10)
```

13

```
number = number_sum(3, 10)  
number
```

13

두 수를 입력 받아서 뺀 결과를 return하는 함수를 정의하시오.

```
1 num1 = int(input('첫 번째 정수 입력>> '))
2 num2 = int(input('두 번째 정수 입력>> '))
3 # 빼기 기능을 하는 함수 두 인자값 입력하여 호출
4 result = number_sub(num1,num2)
5 print(result)
```

첫 번째 정수 입력>> 10

두 번째 정수 입력>> 3

7

두 수를 입력 받아서 뺀 결과를 return하는 함수를 정의하시오.

```
1 num1 = int(input('첫 번째 정수 입력>> '))
2 num2 = int(input('두 번째 정수 입력>> '))
3 # 빼기 기능을 하는 함수 두 인자값 입력하여 호출
4 result = number_sub(num1,num2)
5 print(result)
```

첫 번째 정수 입력>> 10

두 번째 정수 입력>> 3

7

두개의 정수를 받아 두개의 숫자 중 10에 더 가까운 수를
반환하는 함수 close10을 만들어보세요
(만약 두 숫자 모두 10과의 차이가 같다면 첫번째 수를 반환)

```
num1 = int(input("첫번째 정수 입력 : "))
num2 = int(input("두번째 정수 입력 : "))
result = # close10 함수 호출
print("10에 가까운 수 : {}".format(result))
```

첫번째 정수 입력 : 12

두번째 정수 입력 : 13

10에 가까운 수 : 12

첫번째 정수 입력 : 2

두번째 정수 입력 : 3

10에 가까운 수 : 3

첫번째 정수 입력 : 5

두번째 정수 입력 : 15

10에 가까운 수 : 5

두 수를 입력 받아서 원하는 연산을 수행하여 결과를 return하는 함수를 정의하시오.

```
num1 = int(input("첫 번째 정수 입력 >> "))  
num2 = int(input("두 번째 정수 입력 >> "))  
op = input("연산자 입력(+, -) >> ")  
result = cal(num1, num2, op)  
print("결과 : {}".format(result))
```

```
첫 번째 정수 입력 >> 5  
두 번째 정수 입력 >> 3  
연산자 입력(+, -) >> +  
결과 : 8
```

```
첫 번째 정수 입력 >> 5  
두 번째 정수 입력 >> 3  
연산자 입력(+, -) >> -  
결과 : 2
```

독스트링(docstring)

- 함수의 설명을 작성 (Shift + <Tab>)

```
def cal(num1, num2, op):  
    """덧셈과 뺄셈을 계산하는 함수"""  
    if op == '+':  
        return num1+num2  
    else:  
        return num1-num2
```

cal()

Signature: cal(num1, num2, op)

Docstring: 덧셈과 뺄셈을 계산하는 함수

가변 매개변수(variable parameters)

- 함수 호출 시 몇 개의 인수가 전달될지 알 수 없다면, 사용자가 직접 매개변수의 개수를 정할 수 있도록 선언

```
def 함수명(*매개변수):  
    실행문장  
    return 반환변수
```

가변 매개변수(variable parameters)

- 전달된 모든 인수는 튜플(tuple)의 형태로 저장

```
def add(*args):  
    print(args)
```

args → arguments

```
add(1,2,3)
```

```
(1, 2, 3)
```

```
add(1,2,3,4,5,6,7,8,9,10)
```

```
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
```

가변 매개변수를 활용해 모든 숫자를 더해서 반환하는 함수를 작성하시오.

```
def add(*args):  
    ?
```

```
add(1,2,3)
```

6

```
add(1,2,3,4,5,6,7,8,9,10)
```

55

가변 매개변수(variable parameters)

- 딕셔너리 형태로 함수 내부에서 처리하고 싶을 때

```
def 함수명(**매개변수):  
    실행문장  
    return 반환변수
```

가변 매개변수(variable parameters)

- 딕셔너리 형태로 함수 내부에서 처리하고 싶을 때

```
def print_map(**kwargs):  
    print(kwargs)  
    for key,value in kwargs.items():  
        print(key,"/",value)
```

kwargs → keyword arguments

```
print_map(하나=1)
```

```
{'하나': 1}  
하나 / 1
```

```
print_map(one=1, two=2)
```

```
{'one': 1, 'two': 2}  
one / 1  
two / 2
```

Return 함수의 반환값은 언제나 하나이다.

```
def add_sub(num1, num2):  
    return num1+num2, num1-num2
```

```
add_sub(10, 7)
```

Return 함수의 반환값은 언제나 하나이다.

```
def add_sub(num1, num2):  
    return num1+num2, num1-num2
```

```
result_add, result_sub = add_sub(10,7)  
print(result_add)  
print(result_sub)
```

17

3

기본값 설정(default parameters)

```
def power_of_N(num, power=2):  
    return num**power
```

```
power_of_N(|)
```

Signature: power_of_N(num, power=2)

기본값 설정(default parameters)

```
def power_of_N(num, power=2):  
    return num**power
```

```
power_of_N(3)
```

9

```
power_of_N(3,3)
```

27

```
power_of_N(3,power=5)
```

243



수고하셨습니다!



다음시간에는?

파일(file)