



# PYTHON\_PROGRAMMING

## 함수

# 예제

## ❖ 정의

- 정의란, 어떤 이름을 가진 코드가 구체적으로 어떻게 동작하는지를 구체적으로 기술하는 것

## ❖ 파이썬에서는 함수나 메소드를 정의할때 definition를 줄인 키워드인 **def**를 사용

## ❖ 예)

```
def 함수이름():  
    종속 문장
```

# 예제

```
result,temp = 0,0
result = int(input("수 입력 : "))
while True:
    temp = int(result%10)
    result = int(result/10)
    print(temp,end=" ")
    if not result: break;
print("\n프로그램 종료")
```

# 예제

```
def reverseCode():  
    result,temp = 0,0  
    result = int(input("수 입력 : "))  
    while True:  
        temp = int(result%10)  
        result = int(result/10)  
        print(temp,end="")  
        if not result: break;  
print("프로그램 시작")  
reverseCode()  
print("\n프로그램 종료")
```

# 예제

```
sel = 0
```

```
sel = int(input("음료 선택\n1.콜라\n2.핫6\n3.포카리\n입력 : "))
```

```
if sel == 1 : print('콜라 등장')
```

```
elif sel == 2 : print('핫6 등장')
```

```
elif sel == 3 : print('포카리 등장')
```

```
else : print('만들어 드세요^^')
```

```
if sel >=1 and sel <=3:
```

```
    print("맛있게 드세요^^")
```

# 예제

```
def sel_machine():  
    sel = 0  
    sel = int(input("음료 선택\n1.콜라\n2.하트6\n3.포카리\n입력 : "))  
    if sel == 1 : print('콜라 등장')  
    elif sel == 2 : print('하트6 등장')  
    elif sel == 3 : print('포카리 등장')  
    else : print('만들어 드세요^^')  
  
    if sel >=1 and sel <=3:  
        print("맛있게 드세요^^")  
sel_machine()
```

# 예제 1

```
def calc():  
    result=0  
    su1,op,su2 = int(input("숫자:")),input("부호 :"),int(input("숫자:"))  
    result = su1+su2  
    print(su1,'+',su2,'=',result)  
calc()
```

❖ 위의 예제를 이용하여 다음 연산들도 만드시오( − , \* , / )

## 예제 2

```
def calc():  
    result=0  
    su1,op,su2 = int(input("숫자:")),input("부호 :"),int(input("숫자:"))  
    if op == '+':result = su1+su2  
    elif op == '-':result = su1-su2  
    elif op == '*':result = su1*su2  
    elif op == '/':result = su1/su2  
    print(su1,op,su2,'=',result)  
calc()
```



# 예제

```
def cal(su1,op,su2):  
    result=0
```

```
    result = su1+su2  
    print(su1,'+',su2,'=',result)
```

```
su1,op,su2 = int(input("숫자:")),input("부호 :"),int(input("숫자:"))  
cal(su1,op,su2)
```

# 예제

```
def cal(su1,op,su2):  
    result=0  
    result = su1+su2  
    print('cal 실행')  
    return result
```

```
su1,op,su2 = int(input("숫자:")),input("부호 :"),int(input("숫자:"))  
result=cal(su1,op,su2)  
print(su1,'+',su2,'=',result)  
print("다음 문장 실행")
```

# 예제

```
def showAvrg(a,b,c):  
    print("{}와 {}의 평균".format(a,b))  
    print("값은 {}입니다".format(round(c,1)))  
def avrg(j,k):  
    total=j+k;  
    f=total/2  
    return f;  
i=2; j=3;  
f=avrg(i,j)  
showAvrg(i,j,f)  
print("다음 문장 실행")
```

# 예제

❖ 아래의 내용의 결과를 생각 후 결과 확인

```
def func2(a,b):  
    a+=5; b*=10;  
    print("func2 : a={}, b={} ".format(a,b))  
def func1():  
    a=5;b=10  
    func2(a,b)  
    print("func1 : a={}, b={} ".format(a,b))  
func1()
```

# 예제

```
num1 = input('수 입력 : ')  
num2 = int(input('수 입력 : '))  
print(type(num1))  
print(type(num2))
```

❖ 위의 내용을 토대로 사용자 만의 myType함수를 만들시오.

# 예제

```
def myType(su):  
    if type(su) == int:  
        return str(su)+":정수(int)형태입니다"  
    elif type(su) == str:  
        return su+":문자(str)형태입니다"  
    else:  
        return "어떤것인지 모르겠습니다"  
  
num1 = input("수 입력 : ")  
num2 = int(input('수 입력 : '))  
num3 = float(input('수 입력 : '))  
print(myType(num1))  
print(myType(num2))  
print(myType(num3))
```

# 예제

```
def aa(a):  
    if a==1:  
        print('1 입력')  
        print('다음 문장 실행')  
a=aa(1)  
print("리턴 값 :",a)  
print('프로그램 종료')
```

# 예제

```
def aa(a):  
    if a==1:  
        return  
        print('1 입력')  
    print('다음 문장 실행')  
a=aa(1)  
print("리턴 값 :",a)  
print('프로그램 종료')
```



# pass

```
def move():  
    print('이동')  
def attack():  
    print('공격')  
def defense():  
    print('방어')  
move()  
attack()  
defense()
```

# pass

```
def move():
```

```
    pass
```

```
def attack():
```

```
    pass
```

```
def defense():
```

```
    pass
```

```
move()
```

```
attack()
```

```
defense()
```

# pass

```
def a(a):  
    if a==1:  
        pass  
    else:  
        print('1이 아니에요')  
a(1)
```

# 문제

- ❖ 짝, 홀수를 구분하는 함수를 만드시오
- ❖ 3의 배수를 판별하는 함수를 만드시오
- ❖ 이전 예제 계산기를 입력, 출력, 연산기능으로 나눠서 실행되게 만드시오.
- ❖ 이전 예제 거꾸로 수를 반환하는 함수를 만드시오
- ❖ 이전 예제 자판기를 함수화 시키시오

# 변수 (스코핑 룰)

## ❖ 지역변수와 전역변수

- 지역변수는 한정된 지역(Local)에서만 사용되는 변수, 전역변수는 프로그램 전체(Global)에서 사용되는 변수

```
def func1() :  
    a=100  
    print("func1의 a : %d" % a)  
def func2() :  
    a=200  
    print("func2의 a : %d" % a)  
func1()  
func2()
```

# 변수 (스코핑 룰)

## ❖ 전역변수

```
def func1() :  
    print("func1의 a : %d" % a)
```

```
def func2() :  
    print("func2의 a : %d" % a)
```

```
a =200 #전역 변수
```

```
func1()
```

```
func2()
```

# 예제

```
def func1() :  
    a = 123  
    print("func1의 a : %d" % a)
```

```
def func2() :  
    print("func2의 a : %d" % a)  
a = 200 #전역 변수  
func1()  
func2()
```

# 예제

```
def func1() :  
    global a  
    a = 1222  
    print("func1의 a : %d" % a)
```

```
def func2() :  
    print("func2의 a : %d" % a)  
a = 200 #전역 변수  
func1()  
func2()
```



## 예제1

```
def display():  
    num = 10  
    print("10까지의 합 : " , sumFunc(num))  
def sumFunc(num) :  
    sum=0  
    for i in range(num+1):  
        sum+=i  
    return sum  
display()
```

❖ 위의 내용을 전역변수로 바꾸시오.

## 예제2

```
num = 10
sum=0
def display():
    sumFunc()
    print("10까지의 합 :", sum)
def sumFunc() :
    global sum
    for i in range(num+1):
        sum+=i
display()
```

# Default Parameter

```
def sum_func(x1,x2,x3 = 100) :  
    result = 0  
    result =x1 + x2 + x3  
    return result  
def display():  
    Sum = 0  
    a,b,c = 10 , 20 , 30  
    Sum = sum_func(a , b)  
    print("매개변수 2개 함수 호출 : " , Sum)  
    Sum = sum_func(a ,b , c)  
    print("매개변수 3개 함수 호출 : " , Sum)  
display()
```

# 예제

```
def alba(day=30,time=8,won=8500):  
    result = day * time * 8500  
    return result  
def display():  
    num = int(input('1.기본급₩n2.일한 날짜 입력₩n'))  
    if num == 1:  
        result = alba()  
    elif num==2:  
        day = int(input('일한 날짜 입력(몇일) : '))  
        result=alba(day)  
    print("당신의 급여 :{}원 입니다".format(result))  
display()
```

# Arbitrary Argument List

```
def sum_func(*par) :  
    result = 0  
    print("type : ",type(par))  
    print("par : ",par)  
    for num in par :  
        result = result + num  
        print("num : %d" % num)  
    return result  
  
Sum = 0  
Sum = sum_func(10 , 20)  
print("매개변수 2개 함수 : %d" % Sum)  
Sum = sum_func(10 , 20 , 30 , 40)  
print("매개변수 4개 함수 : %d" % Sum)
```

# 예제

```
def dic_func(**par) :  
    print("type(par):",type(par))  
    for k in par.keys() :  
        print("{} : {} 명입니다".format(k , par[k]))  
  
dic_func(뽕뽕뽕 = 123 , 꺾익꺾익 = 8 , test = '테스뚜')
```

# 예제

```
def change(a,b,c):  
    return a+10,b+20,c+30
```

```
a,b,c=change(10,20,30)  
d = change(10,20,30)  
print("a,b,c :",a,b,c)  
print("d:{}, type{}".format(d,type(d)))
```

# 예제

```
def swap(x,y):  
    return y,x
```

```
a=10; b=20  
print("바꾸기 전 :",a,b)  
a,b=swap(a,b)  
print("바꾼 후 :",a,b)
```



# 예제

```
swap = lambda a,b:[b,a]  
a=swap(10,20)  
print("swap 결과",a)
```

# Lambda 함수

```
lam = lambda a:a*10  
hap = lambda a,b:a+b  
noData = lambda : print("인자값 없는 람다")  
  
print(lam(10))  
print(hap(5,10))  
noData()
```

# Lambda 함수

```
def startGame():  
    print("Game Start!!!!")  
def test():  
    print("1.게임 시작")  
    print("2.게임 종료")  
    num = input("선택 : ")  
    if num=="1":  
        startGame()  
    elif num=="2":  
        end()  
end = lambda :print("게임 종료")  
test()
```

# Quiz

❖ 람다 함수를 이용하여 두 수의  $+$ ,  $-$ ,  $*$ ,  $/$  의 기능을 만들고 실행 하시오.

❖ 디폴트 매개변수

요금 구하는 프로그램 만들기.

기본요금 환승 없으면 500원 환승 할 경우 환승수에 따라 요금이 달라짐  
환승때 마다 요금의 2배씩 증가된다. 장거리는 10000원으로 처리한다.

1. 환승 안함
2. 환승 함
3. 장거리