

# 파이썬 기초 07

---

by goldmont@naver.com



스타트 파이썬  
<http://codingschool.biz>

# Chapter 07. 함수

- 함수란?
- 함수의 매개변수
- 함수의 반환 값
  - 함수의 활용
  - 파일 다루기



# 함수란?

---

- 함수 : 수학 '함수'의 개념과 영어 단어 function이  
가지는 '기능'의 의미를 포함

함수 예 : print(), input(), range(), list(), append(), remove() 등

# 함수로 '안녕하세요!' 출력

```
① def hello() :  
    print('안녕하세요!')
```

```
② hello()
```

```
③ hello()
```

```
④ hello()
```

:: 실행 결과

```
안녕하세요!  
안녕하세요!  
안녕하세요!
```



# 함수 정의와 호출

- 함수 정의

```
def 함수명() :  
    문장1  
    문장2  
    ...
```

- 함수 호출

```
...  
함수명()  
...
```

# 함수의 종류

---

- 사용자 함수 : 사용자가 함수를 직접 정의
- 내장 함수 : 파이썬 프로그램 설치 시 내장

print(), input(), range(), list(), append(), remove(), round(), int(),  
float(), str(), type()

# 함수의 매개변수

```
① def say_hello(name) :  
②     print('%s님 안녕하세요!' % name)  
  
③ say_hello('홍지수')  
   say_hello('안지영')  
④ say_hello('황예린')
```

∴ 실행 결과

홍지수님 안녕하세요!  
안지영님 안녕하세요!  
황예린님 안녕하세요!



# 매개변수 이용한 짝수/홀수 판별

```
① def even_odd(n) :  
    if n % 2 == 0 :  
        print('%d -> 짝수' % n)  
    else :  
        print('%d -> 홀수' % n)
```

```
② even_odd(15)
```

```
③ even_odd(26)
```

∴ 실행 결과

15 -> 홀수  
26 -> 짝수





# 연습문제 7-1. 매개변수 이용한 정수 합계

```
def sum((1)_____, (2)_____ ) :  
    hap = 0  
    for i in range(start, end+1) :  
        hap = hap + i  
    print('%d ~ %d의 정수의 합계 : %d' % (start, end, (3)  
_____) ))  
  
sum(1, 10)  
sum((4)_____, 200)  
sum(200, (5)_____ )
```

:: 실행 결과

1 ~ 10의 정수의 합계 : 55  
100 ~ 200의 정수의 합계 :  
15150  
200 ~ 300의 정수의 합계 :  
25250



# 함수 반환 값

```
① def inch_to_cm(inch) :  
    cm = inch * 2.54  
    return cm  
  
② num = int(input('인치를 입력하세요: '))  
③ result = inch_to_cm(num)  
④ print('%d inch => %.2f cm' % (num, result))
```

∴ 실행 결과

인치를 입력하세요: 30  
30 inch => 76.20 cm



# 반환 값 이용한 배수 판별

```
① def besu5(n) :  
    if n % 5 == 0 :  
        rel = True  
    else :  
        rel = False  
  
    return rel  
  
② num = int(input('양의 정수를 입력하세요: '))  
③ result = besu5(num)  
  
④ if result == True :  
    print('%d -> 5의 배수이다.' % num)  
else :  
    print('%d -> 5의 배수가 아니다.' % num)
```

:: 실행 결과

양의 정수를 입력하세요: 40  
40 -> 5의 배수이다.



# 연습문제 7-2. 반환 값 이용한 3의 배수 합계

```
(1)_____ sum_besu3(n) :  
    sum = 0  
    for i in range(1, (2)_____ ) :  
        if i % 3 == 0 :  
            (3)_____ = sum + i  
  
    return sum  
  
num = int(input('양의 정수를 입력하세요: '))  
result = sum_besu3((4)_____ )  
  
print('1 ~ %d까지의 정수 중 3의 배수의 합 : %d' % (num,  
(5)_____ ))
```

:: 실행 결과

양의 정수를 입력하세요: 200  
1 ~ 200까지의 정수 중 3의 배수의 합 : 6633



# 연습문제 7-3. 반환 값 이용한 원 면적과 원주

```
def cir_area((1)_____ ) :  
    area = radius * radius * 3.14  
    return (2)_____  
  
def cir_circum(radius) :  
    circum = 2 * 3.14 * (3)_____  
    return (4)_____  
  
r = float(input('반지름을 입력하세요: '))  
a = cir_area(r)  
(5)_____ = cir_circum(r)  
print('원의 면적 : %.2f, 원주의 길이 :%.2f' % ((6)_____, b))
```

:: 실행 결과

반지름을 입력하세요: 5  
원의 면적 : 78.50, 원주의 길이 :31.40



# 최대 공약수 구하기

```
① def computeMaxGong(x, y):  
②     if x > y:  
        small = y  
     else:  
        small = x  
③     for i in range(1, small+1):  
④         if((x % i == 0) and (y % i == 0)) :  
⑤             result = i  
⑥     return result
```

:: 실행 결과

첫 번째 수를 입력하세요: 20  
두 번째 수를 입력하세요: 25  
20와 25의 최대공약수 : 5



# 최대 공약수 구하기

```
⑦ num1 = int(input("첫 번째 수를 입력하세요: "))  
  num2 = int(input("두 번째 수를 입력하세요: "))  
  
⑧ max_gong = computeMaxGong(num1, num2)  
  
⑨ print('%d와 %d의 최대공약수 : %d' % (num1, num2, max_gong))
```

# 소수 구하기

```
① def isPrimeNumber(num) :  
    prime_yes = True  
    for i in range(2, a) :  
        if a % i == 0 :  
            prime_yes = False  
            break  
    return prime_yes
```

:: 실행 결과

n 값을 입력해 주세요 : 30  
2 ~ 30까지의 정수 중 소수 : 2 3 5 7 11  
13 17 19 23 29



# 소수 구하기

```
② n = int(input('n 값을 입력해 주세요 : '))  
  
③ print('2 ~ %d까지의 정수 중 소수 : ' % n, end = ' ')  
④ for a in range(2, n+1) :  
⑤     is_prime = isPrimeNumber(a)  
⑥     if is_prime :  
⑦         print(a, end=' ')
```

# 영어 단어 맞추기 퀴즈

```
① def matchWord(in_word, answer) :  
    if in_word == answer :  
        msg = '참 잘했어요~~~'  
    else :  
        msg = '단어 공부 좀 더 해야겠어요.ㅋㅋ'  
    return msg
```

:: 실행 결과

참 잘했어요~~~  
사자에 맞는 영어 단어는? tiger  
단어 공부 좀 더 해야겠어요.ㅋㅋ  
책에 맞는 영어 단어는? book  
참 잘했어요~~~  
사랑에 맞는 영어 단어는? love  
참 잘했어요~~~  
친구에 맞는 영어 단어는? mother  
단어 공부 좀 더 해야겠어요.ㅋㅋ



# 영어 단어 맞추기 퀴즈

```
② eng_dict = {'apple':'사과', 'lion':'사자', 'book':'책', 'love':'사랑', 'friend':'친구'}  
  
③ for i in eng_dict :  
④     string = input(eng_dict[i] + '에 맞는 영어 단어는? ')  
⑤     result = matchWord(string, i)  
⑥     print(result)
```

# 연습문제 7-4. 세 수중 가장 큰 수 찾기

```
def (1)_____(i, j):  
    if i > j:  
        return i  
    else :  
        return j  
  
def (2)_____(x, y, z) :  
    return maxTwo( (3)_____, maxTwo(y, z))  
  
a = int(input('첫 번째 수를 입력하세요: '))  
b = int(input('두 번째 수를 입력하세요: '))  
c = int(input('세 번째 수를 입력하세요: '))  
(4)_____ = maxThree(a, b, c)  
  
print('%d, %d, %d 중 가장 큰 수 : %d' % (a, b, c, max_num))
```

:: 실행 결과

첫 번째 수를 입력하세요: 10  
두 번째 수를 입력하세요: 50  
세 번째 수를 입력하세요: -20  
10, 50, -20 중 가장 큰 수 : 50



# 연습문제 7-5. 최소 공배수 구하기

```
def (1)_____ (x, y):  
    if x > y :  
        big = x  
    else :  
        big = (2)_____  
  
    while(True):  
        if((big % x == 0) (3)_____ (big % y == 0)):  
            result = (4)_____  
            break  
        big = big + 1  
  
    return result
```

:: 실행 결과

첫 번째 수를 입력하세요: 10  
두 번째 수를 입력하세요: 7  
10와 7의 최소공배수 : 70

# 연습문제 7-5. 최소 공배수 구하기

```
num1 = int(input("첫 번째 수를 입력하세요: "))
num2 = int(input("두 번째 수를 입력하세요: "))

min_gong = computeMinGong(num1, num2)

print('%d와 %d의 최소공배수 : %d' % (num1, num2, (5)_____ ))
```

# 파일 쓰기

```
① file = open('sample.txt', 'w')
② file.write('안녕하세요. 반갑습니다~~~')
③ file.close()
④ print('sample.txt 파일 쓰기 완료!')
```

:: 실행 결과

sample.txt 파일 쓰기 완료!!



# open() 함수의 파일 모드

파일 모드	설명
r	읽기 모드 : 파일을 읽을 때 사용
w	쓰기 모드 : 파일에 내용을 쓸 때 사용 ※ 해당 파일이 존재하지 않으면 새로운 파일을 열고, 해당 파일이 존재하면 파일을 쓸 때 기존 파일의 내용에 덮어씀
a	추가 모드 : 기존의 파일에 새로운 내용을 추가할 때 사용



# 리스트와 for문 이용한 파일 쓰기

```
① scores = ['안소영 97 80 93 97 93',  
            '정예린 86 100 93 86 90',  
            '김세린 91 88 99 79 92',  
            '연수정 86 100 93 89 92',  
            '박지아 80 100 95 89 90']  
  
② data = ""  
③ for item in scores :  
    data = data + item + '\n'
```

:: 실행 결과

```
안소영 97 80 93 97 93  
정예린 86 100 93 86 90  
김세린 91 88 99 79 92  
연수정 86 100 93 89 92  
박지아 80 100 95 89 90
```

# 리스트와 for문 이용한 파일 쓰기

④ `print(data)`

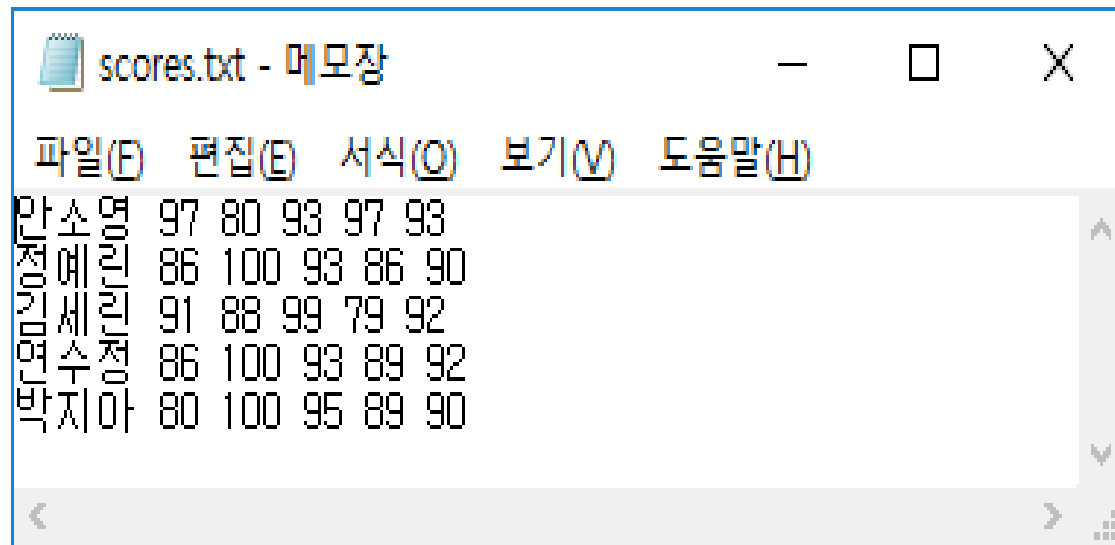
`# 파일(scores.txt)에 출력하기`

⑤ `file = open('scores.txt', 'w')`

⑥ `file.write(data)`

⑦ `file.close()`

# scores.txt 파일 내용



A screenshot of a Notepad window titled "scores.txt - 메모장". The window contains a text file with five lines of data, each representing a student's scores. The text is as follows:

안소영	97	80	93	97	93
정예린	86	100	93	86	90
김세린	91	88	99	79	92
연수정	86	100	93	89	92
박지아	80	100	95	89	90

# scores.txt 파일 읽기

```
① file = open('scores.txt', 'r')
② lines = file.readlines()

③ print(lines)

④ for line in lines :
    print(line, end='')

⑤ file.close()
```

# scores.txt 파일 읽기

:: 실행 결과

```
['안소영 97 80 93 97 93\\n', '정예린 86 100 93 86 90\\n', '김세린 91 88 99 79 92\\n',  
'연수정 86 100 93 89 92\\n', '박지아 80 100 95 89 90\\n']
```

```
안소영 97 80 93 97 93
```

```
정예린 86 100 93 86 90
```

```
김세린 91 88 99 79 92
```

```
연수정 86 100 93 89 92
```

```
박지아 80 100 95 89 90
```

# 파일의 성적 합계/평균

```
① file = open('scores.txt', 'r')
② lines = file.readlines()
③ file.close()
④ print(lines)
   print('-' * 50)
```

:: 실행 결과

```
['안소영 97 80 93 97 93\n', '정예린 86 100 93 86\n', '김세린 91 88 99 79 92\n', '연수정 86 100\n', '박지아 80 100 95 89 90\n']
```

```
-----
안소영
합계 : 460, 평균 : 92.00
-----
```

...

```
연수정
합계 : 460, 평균 : 92.00
-----
```

```
박지아
합계 : 454, 평균 : 90.80
```



# 파일의 성적 합계/평균

```
⑤ for line in lines :  
⑥     student = line.split()  
⑦     i = 0  
        sum = 0  
⑧     while i < 6 :  
⑨         if i == 0 :  
                print(student[i])  
⑩         else :  
                sum = sum + int(student[i])  
⑪         i = i + 1  
⑫ print('합계 : %d, 평균 : %.2f' % (sum, sum/5))  
    print('-' * 50)
```