

내장함수

내장 함수 1

- import 시키지 않고 사용할 수 있는 함수
- `abs(x)`
 - 숫자의 절대값을 돌려주는 함수

```
abs(3)
abs(-3)
abs(1+2j)
*  $\text{abs}(a+bj) = \sqrt{a^2 + b^2}$ 
```

- `all(x)`
 - 반복가능한(iterable) x가 모두 참인경우 true, 거짓이 하나라도 있을 경우 false를 반환

```
all([1,2,3])
all([1,2,3,0])
all([1,2,3, ""])
```

```
def all(iterable):
    for element in iterable:
        if not element:
            return False
    return True
```

내장 함수 2

- `any(x)`

- 반복가능한 x 중 하나라도 참이 있을 경우 True

```
any([1,2,3])  
any([0, ""])
```

- `chr(x)` / `ord(c)`

- 정수 형태의 아스키 코드 값을 입력 받아 그에 해당하는 문자를 출력 하는 함수

```
chr(97)  
chr(48)
```

```
ord('a')  
ord('0')
```

- `dir(x)`

- 객체가 가지고 있는 변수나 함수를 리스트 형태로 보여주는 함수

```
num = [1,2,3,4]  
dir(num)
```

내장 함수 3

- `divmod(x,y)`

- 두 개의 숫자를 입력값으로 받아, 몫과 나머지를 튜플 형태로 반환

```
divmod(7,3)
```

- `enumerate(x)`

- 시퀀스 자료형(리스트, 튜플, 문자열)을 입력으로 받아 `enumerate`객체를 반환함
- `enumerate`객체는 순서와 실제값을 갖는 객체

```
data = list(enumerate("greenjoa"))  
print( data)
```

- `eval(x)`

- 실행가능한 문자열을 입력으로 받아 실행한 결과를 반환하는 함수

```
eval('1+2')  
eval('divmod(4,3)')
```

내장 함수 4

- filter(function, iterable)
 - 연속열의 각 원소에 대해 함수를 적용한 값으로 새로운 리스트를 만들

```
def positive(l):  
    result = []  
    for i in l:  
        if i > 0:  
            result.append(i)  
    return result  
  
print(positive([1,-3,2,0,-5,6]))
```

```
def positive(l):  
    return l > 0  
  
print(list(filter(positive, [1,-3,2,0,-5,6])))
```

- hex(x)
 - 정수값을 받아서 16진수로 변환하는 함수

```
hex(234)
```

내장 함수 5

- `id(object)`
 - 객체의 고유값(레퍼런스)를 반환하는 함수

```
a=3  
id(3)  
id(a)
```

- `int(x)`
 - 스트링 형태의 숫자나 실수 등을 정수로 변환하는 함수

```
int('3')  
int(3.5)  
int('11', 2) 이진수 값  
int('1A', 16) 16진수 값
```

- `isinstance(object, class)`
 - 객체가 클래스의 인스턴스인지 판단하는 함수

```
isinstance(10, int)
```

내장 함수 6

- `lambda`
 - 익명의 함수를 만들 때 사용
 - Return 구문을 사용하지 못함

`lambda 인수1, 인수2, ... : 인수를 이용한 표현식`

```
sum = lambda a, b : a+b  
sum(3,4)
```

```
def sum(a, b):  
    return a+b
```

```
l = [lambda a,b:a+b, lambda a,b:a*b]  
l[0](3,4)  
l[1](3,4)
```

```
print(list(filter(lambda x: x > 0, [1,-3,2,0,-5,6])))
```

내장 함수 7

- `len(s)`
 - Iterable 자료형(문자열, 리스트, 튜플, 딕셔너리, 집합)을 입력으로 받아 그 길이를 돌려주는 함수

```
len([1,2,3,4])  
len("greenjoa")
```

- `list(s)`
 - Iterable 자료형을 입력받아 리스트를 새로 만들어 주는 함수

```
list("greenjoa")  
list((1,2,3))  
a = [1,2,3]  
b = list(a)  
* a와 b의 id가 다름
```


내장 함수 8

- `map(f, iterable)`
 - 함수(f)와 iterable 자료형의 입력으로 각 자료형의 각 요소가 함수 f에의 해 수행된 결과를 반환하는 함수

```
def two_times(l):  
    result = []  
    for i in l:  
        result.append(i*2)  
    return result
```

```
result = two_times([1,2,3,4])  
print(result)
```

```
def two_times(x): return x*2  
  
list(map(two_times, [1,2,3,4]))
```

```
list(map(lambda a: a*2, [1,2,3,4]))
```

- `max(iterable)` / `min(iterable)`
 - 최대 / 최소 값을 반환하는 함수

```
max([1,2,3])  
min("python")
```

내장 함수 9

- `repr(object)`
 - 객체를 출력할 수 있는 문자열 형태로 돌려주는 함수
 - `eval` 함수의 입력으로 주로 쓰임

```
eval(repr("hi".upper()))  
eval(str("hi".upper())) 에러 발생
```

- `sorted(iterable)`
 - 정렬한 리스트를 반환하는 함수

```
sorted([3,1,2])  
sorted("zero")  
sorted(['a','c','b'])
```

내장 함수 10

- `zip(iterable, iterable, . . .)`
 - 동일한 개수의 요소 값을 갖는 반복 가능한 자료형을 묶어주는 역할

```
list(zip([1,2,3], [4,5,6], [7,8,9]))  
list(zip("abc", "def"))
```