

# 기초 PYTHON 프로그래밍

## 12. for 반복문

1. range() 함수
2. for 반복문
3. 리스트 내에 for 반복문
4. 군집자료형의 형변환

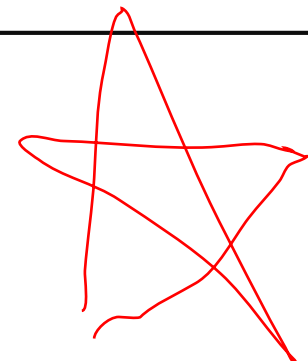
# 1. range() 함수

◆ range() 함수 : 일정 범위의 수를 반환한다.

range(b) - 0부터 b-1까지의 수를 반환한다.

range(a,b) - a부터 b-1까지의 수를 반환한다.

range(a,b,n) - a부터 b-1까지의 n 간격의 수를 반환한다



```
>>> A = range(5)
```

```
>>> type(A)
```

```
<class 'range'>
```

```
>>> print(A)
```

```
range(0, 5)
```

```
>>> B = list(range(5))
```

```
>>> print(B)
```

```
[0, 1, 2, 3, 4]
```

```
>>> list(range(5))
```

```
[0, 1, 2, 3, 4]
```

```
>>> list(range(5,10))
```

```
[5, 6, 7, 8, 9]
```

```
>>> list(range(3,10,2))
```

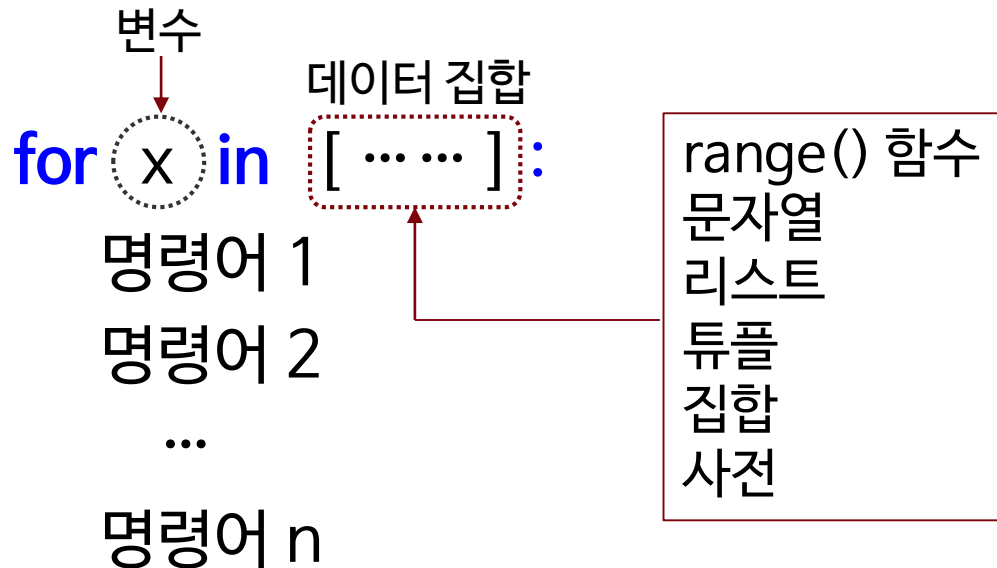
```
[3, 5, 7, 9]
```

```
>>> list(range(10,1,-2))
```

```
[10, 8, 6, 4, 2]
```

## 2. for 반복문

### ◆ for 구문을 이용한 반복문



- 데이터 집합에서 데이터를 하나씩 변수 x에 넣고 반복을 수행한다.
- for 구문에도 break와 continue를 사용할 수 있다.

### 3. range() 함수와 for 반복문

예제 1

```
for x in range(5):  
    print(x)
```

0  
1  
2  
3  
4

range(5) 자리에 [0,1,2,3,4]가 온다.  
차례대로 x값이 되어 for 블록을 수행한다.

예제 2

```
for data in range(1,10,2):  
    print(data+1)
```

2  
4  
6  
8  
10

## 4. 문자열과 for 반복문

예제 3

```
for letter in 'PYTHON':  
    print(letter)
```

P  
Y  
T  
H  
O  
N

'PYTHON'이 하나씩 letter 변수에 저장되어  
for 블록을 수행한다.

예제 4

```
for alpha in 'sogang':  
    y = alpha.upper()  
    print(y)
```

S  
O  
G  
A  
N  
G

## 5. 리스트와 for 반복문

예제 5

```
for x in [1,3,5,7,9]:  
    print(x)
```

1  
3  
5  
7  
9

[1,3,5,7,9] 원소가 하나씩 변수 x값이  
되어 for 블록을 수행한다.

예제 6

```
for x in [1,2,3,4,5]:  
    for y in [2,4,6,8,10]:  
        print(x*y, end=' ')  
    print()
```

2 4 6 8 10  
4 8 12 16 20  
6 12 18 24 30  
8 16 24 32 40  
10 20 30 40 50

## 6. 사전과 for 반복문

### ◆ keys() 메소드 사용하기

사전 score는 {학생번호:성적}으로 구성된다.

```
score = {4:90, 2:80, 1:95, 5:88, 3:92}
```

예제 7

```
for x in score :  
    print(x)
```

예제 8

```
for x in score.keys():  
    print(x)
```



1  
2  
3  
4  
5

## 6. 사전과 for 반복문

### ◆ items() 메소드 이용하기

사전 score는 {학생번호:성적}으로 구성된다.

```
score = {4:90, 2:80, 1:95, 5:88, 3:92}
```

예제 9

```
for x,y in score.items():  
    print(x,y)
```

```
1 95  
2 80  
3 92  
4 90  
5 88
```

items() 메소드를 이용하는 경우에는 키와 값을 순서대로 가져온다. 따라서 for 다음에 두 개의 변수를 사용해야 한다. x에는 키가 저장되고, y에는 값이 저장되어 이용할 수 있다.



## 6. 사전과 for 반복문

### ◆ values() 메소드 이용하기

사전 score는 {학생번호:성적}으로 구성된다.

```
score = {4:90, 2:80, 1:95, 5:88, 3:92}
```

예제 10

```
for x in score.values():  
    print(x)
```

```
95  
80  
92  
90  
88
```

## 7. 리스트 안에 for 반복문 사용하기

$$A = \{ x^2 : x \in \{0, 1, 2, \dots, 9\} \}$$

$$B = \{ 1, 2, 2^2, 2^3, \dots, 2^{10} \}$$

$$C = \{ x \mid x \in A \text{ and } x \text{ is even} \}$$

[ **표현식** **for x in 데이터집합** **if 조건** ]

```
>>> A = [ x**2 for x in range(10) ]
```

```
>>> print(A)
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

```
>>> B = [ 2**i for i in range(11) ]
```

```
>>> print(B)
```

```
[1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024]
```

```
>>> C = [ x for x in A if x%2 == 0 ]
```

```
>>> print(C)
```

```
[0, 4, 16, 36, 64]
```

## 8. 군집 자료형의 형 변환

함수명	설 명
str(A)	A를 문자열로 변환한다.
list(A)	A를 리스트로 변환한다.
tuple(A)	A를 튜플로 변환한다.
set(A)	A를 집합으로 변환한다.
dict(A)	A를 사전으로 변환한다.

A = [1,2,3,4]

B = (1,3,5)

C = {2,4,6,8,10}

D = {1:90, 2:88, 3:93}

E = tuple(A) # (1, 2, 3, 4)

F = set(B) # {1, 3, 5}

G = list(C) # [8, 10, 2, 4, 6]

H = list(D) #[1, 2, 3]

I = tuple(D) # (1, 2, 3)

J = set(D) # {1, 2, 3}