



# Python 핵심문법

김효관 |

# 1. 파이썬 기본문법

## 주요 문법

1-1 반복하기

1-2 조건 판단하기

1-3 자주사용하는내용 함수화 하기

1-4 사용자 입력변수 받기

# 1. 파이썬 핵심문법

## 1. 반복하기 (while, for)



```
tvList = [ UN40EN001, UN40EN002, UN40EN003, UN40EN004]
```

헉! tv목록앞에 제품목록을 전부 붙여야하는데.... 어떻게하지?  
하나씩 하면 되겠네..

```
preFix = "LEDTV_"  
tvList[0] = prefix + tvList[0]  
tvList[1] = prefix + tvList[1]
```

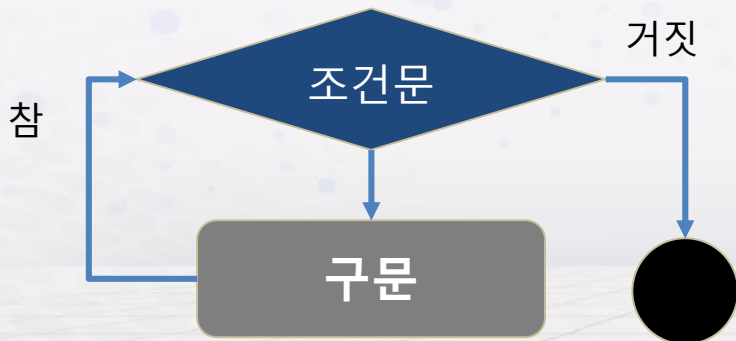
만약.. 목록이 십만개가 넘는다면?? .....

# 1. 파이썬 핵심문법

## 1. 반복하기 (while, for)

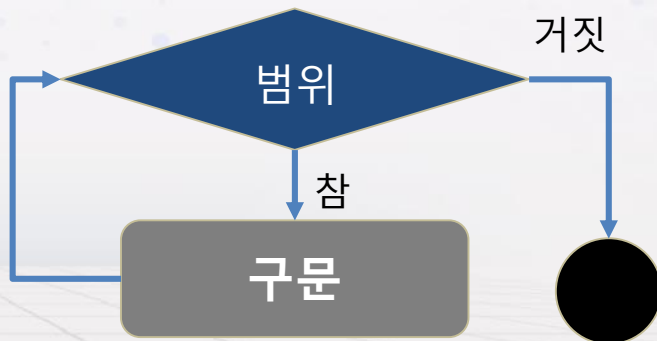
조건문이 참인 경우 조건문 이하 구문을 수행  
만약 거짓이면 구문 수행하지 않음

while



주어진 범위 안에서  
구문 수행

for



# 1. 파이썬 핵심문법

## 1. 반복하기 (while, for)

```
while (조건) loop : 조건 실패 시 조건 문 탈출  
for (조건) : 조건 실패 시 조건 문 탈출
```

### Example

```
tvList = [ 'UN40EN001', 'UN40EN002', 'UN40EN003', 'UN40EN004' ]
```

#### for 문

```
preFix = "LEDTV_"  
### range: last index is not included  
### list(range(0,4))  
for i in range(0,4):  
    tvList[i] = preFix + tvList[i]
```

for문은 변수가 특정 범위안에서  
구문연산 (구문은 탭으로 한칸땀)

#### while 문

```
preFix = "LEDTV_"  
### range: last index is not included  
### list(range(0,4))  
i=0  
listLength = len(tvList)  
while(i < listLength ):  
    tvList[i] = preFix + tvList[i]  
    i = i+1
```

while문은 조건식을  
만족하는 동안 구문연산

# 1. 파이썬 핵심문법

## 1. 반복하기 (while, for)

### 2중 for문

### Example

```
for n in [1,2]:  
    print("{0}단 시작".format(n))  
    for i in [1,2,3,4,5,6,7,8,9]:  
        print("{0} * {1} = {2} ".format(n, i, n*i))
```

#### 2중 for 문

```
for n in [1,2]:  
    print("{0}단 시작".format(n))  
    for i in [1,2,3,4,5,6,7,8,9]:  
        print("{0} * {1} = {2} ".format(n, i, n*i))
```

1단 시작

```
1 * 1 = 1  
1 * 2 = 2  
1 * 3 = 3  
1 * 4 = 4  
1 * 5 = 5  
1 * 6 = 6  
1 * 7 = 7  
1 * 8 = 8  
1 * 9 = 9
```

2단 시작

```
2 * 1 = 2  
2 * 2 = 4  
2 * 3 = 6  
2 * 4 = 8  
2 * 5 = 10  
2 * 6 = 12  
2 * 7 = 14  
2 * 8 = 16  
2 * 9 = 18
```

# 1. 파이썬 핵심문법

## 1. 반복하기 (while, for)

### 2중 for문 (break & continue)

#### Example

```
for n in [1,2]:  
    print("{0}단 시작".format(n))  
    for i in [1,2,3,4,5,6,7,8,9]:  
        if i > 5:  
            break  
        print("{0} * {1} = {2} ".format(n, i, n*i))
```

break 를 만나면 상위 반복문  
한 개를 탈출함

```
for n in [1,2]:  
    print("{0}단 시작".format(n))  
    for i in [1,2,3,4,5,6,7,8,9]:  
        if i%2 == 0:  
            continue  
        print("{0} * {1} = {2} ".format(n, i, n*i))
```

continue 이후 구문만  
수행하지 않음

\* else 반복문 수행도중 break 로 인하여 종료되지 않았을때 수행됨

```
for n in [1,2]:  
    print("{0}단 시작".format(n))  
    for i in [1,2,3,4,5,6,7,8,9]:  
        if i > 5:  
            break  
        print("{0} * {1} = {2} ".format(n, i, n*i))
```

1단 시작  
1 \* 1 = 1  
1 \* 2 = 2  
1 \* 3 = 3  
1 \* 4 = 4  
1 \* 5 = 5  
2단 시작  
2 \* 1 = 2  
2 \* 2 = 4  
2 \* 3 = 6  
2 \* 4 = 8  
2 \* 5 = 10

#### 2중 for 문 + continue

```
for n in [1,2]:  
    if n%2 == 0:  
        continue  
    print("{0}단 시작".format(n))  
    for i in [1,2,3,4,5,6,7,8,9]:  
        print("{0} * {1} = {2} ".format(n, i, n*i))
```

1단 시작  
1 \* 1 = 1  
1 \* 2 = 2  
1 \* 3 = 3  
1 \* 4 = 4  
1 \* 5 = 5  
1 \* 6 = 6  
1 \* 7 = 7  
1 \* 8 = 8  
1 \* 9 = 9

```
tvList2 = [ 'UN40EN001', 'UN40EN002',  
            'UN40EN003', 'UN40EN004']  
리스트가 출력되도록 하세요
```

힌트: `print( tvList[0] )`



# 1. 파이썬 핵심문법

## 2. 조건 판단하기



```
tvList = [ UN40EN001, LEDTV_UN40EN002,  
            LEDTV_UN40EN003, UN40EN004]
```

헉! tv목록앞에 제품목록을 전부 붙여야하는데.... 붙어있는게 있고 없는게 있네.. 어떻게 하지?  
붙어 있는지 조건을 판단해야겠는데...

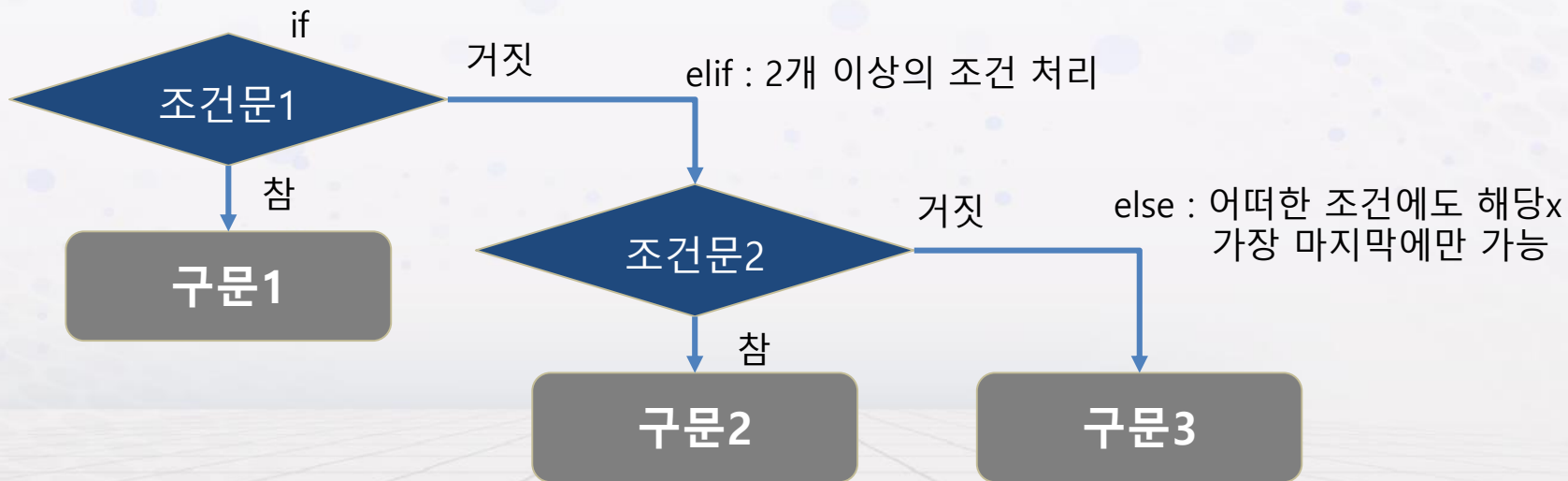
(만약, LEDTV\_가 있으면 넘기고... LEDTV\_가 없으면 넣고..)

조건문은 어떻게 사용할 수 있을까요?

# 1. 파이썬 핵심문법

## 2. 조건 판단하기

조건문을 평가하고 참인 경우만 구문이 수행  
2개 이상의 구문은 들여쓰기로 블록을 지정



# 1. 파이썬 핵심문법

## 2. 조건 판단하기

```
if [조건문]: # 조건이 일치하는 경우 실행
    [조건문 참인경우 실행]
else:
    [조건문 불일치 시 실행]
```

### Example

```
tvList2 = [ 'UN40EN001', 'LEDTV_UN40EN002', 'LEDTV_UN40EN003', 'UN40EN004']
```

```
preFix = "LEDTV_"
### range: last index is not included
### list(range(0,4))
for i in range(0,4):
    preFixCount = tvList2[i].count(preFix)
    if(preFixCount == 0):
        tvList2[i] = preFix + tvList2[i]
    else:
        tvList2[i] = tvList2[i]
```

미션: 모든 목록에 LEDTV를 붙여야한다. (단, 이미 붙어있으면 제외한다)

```
tvList2 = [ 'UN40EN001', 'LEDTV_UN40EN002', 'LEDTV_UN40EN003', 'UN40EN004']
```

```
preFix = "LEDTV_"
### range: last index is not included
### list(range(0,4))
for i in range(0,4):
    preFixCount = tvList2[i].count(preFix)
    if(preFixCount == 0):
        tvList2[i] = preFix + tvList2[i]
    else:
        tvList2[i] = tvList2[i]
```

```
tvList2
```

```
tvList2 = [ 'UN40EN001', 'LEDTV_UN40EN002',  
            'LEDTV_LEDTV_UN40EN003', 'UN40EN004']  
리스트에서 LEDTV가 한 개만 작동하도록 하세요
```

힌트:

A = "haiteam\_"

B = A\*3

B <- haiteam\_haiteam\_haiteam\_

A01, 한국  
A02, 미국  
A03, 인도  
A01  
A02  
A03

한국  
미국  
인도

List ['A01', '한국', 'A02', '미국', 'A03', '프랑스']  
국가코드만 출력하세요

SCORE 변수를 80으로 선언 후  
90점 이상 A, 80~90점 B, 이외 C 점수로  
GRADE변수에 할당하는 프로그램을 구현하세요

# 1. 파이썬 핵심문법

## 3. 자주 사용하는 내용 함수화 하기



화면에는 소수점 2자리까지만 표현해야하는데  
결과 값이 계속 소수점이 많은데 반복수식을  
따로 관리할 수 있을까

```
a = 10  
b = 5.3  
display2Digit = a/b  
(1.8867924528301887)
```

### 소수점 반올림 프로세스

```
target = 1.8867924528301887  
target_step1 = target * 100           → 188.67924528301887  
target_step2 = round(target_step1)    → 189  
target_step3 = target_step2/100       → 1.89
```

# 1. 파이썬 핵심문법

## 3. 자주 사용하는 내용 함수화 하기

```
def 함수명(입력변수#1, 입력변수#2,...):  
    return 결과값
```

### Example

*# This function adds two numbers*

```
def roundTwo(value):  
    step1 = value * 100  
    step2 = round(step1)  
    step3 = step2/100  
    return step3
```

```
a = 10  
b = 5.3  
display2Digit = a/b  
answer = roundTwo(display2Digit)
```

*# This function adds two numbers*

```
def roundTwo(value):  
    step1 = value * 100  
    step2 = round(step1)  
    step3 = step2/100  
    return step3  
  
a = 10  
b = 5.3  
display2Digit = a/b  
print(display2Digit)  
answer = roundTwo(display2Digit)  
answer
```

1.8867924528301887

1.89



리스트를 인수로 받아서 최대, 최소값을 뺀  
평균을 구하는 함수를 생성하세요.

# 1. 파이썬 핵심문법

## 4. 사용자 입력변수 받기

```
입력변수 = input("입력을 위한 문구")
```

### Example

```
name = input("성함을 입력하세요:")  
print(name)
```

```
In [68]: name = input("성함을 입력하세요:")  
         print(name)
```

```
성함을 입력하세요:김효관  
김효관
```

사칙연산(+,-,\*,/)을 할수 있는 함수를 생성한 후  
사용자의 변수를 2개 inputX, inputY를 입력받아  
print문을 활용하여 더하기 예제를 출력하세요

# 참조. 이동평균법

추세를 알기 쉽도록 구간을 옮겨가면서 내는 평균

추세구간 : 5구간

주차 정보	1	2	3	4	5	6	7	8	9	10	11
판매량	100	200	300	200	100	150	100	120	300	500	300
추세선	200	200	180	190	170	134	154	234	264	305	367



3주기준 자기포함  
양옆 날개 2개씩 펼친 평균

4주기준 자기포함  
양옆 날개 2개씩 펼친 평균

