

Chapter 02\_033. for문과 while문 비교

—

for문이 적합한 경우

while문이 적합한 경우

for문이 적합한 경우

while문이 적합한 경우

## • for문이 적합한 경우

- 횟수에 의한 반복이라면 for문이 while문 보다 적합하다.

1부터 10까지의 합을 구하는 경우 for문이 while문 보다 코드가 간결한다.

### for문

```
sum = 0
for i in range(1, 11):
    sum += i
print('sum : {}'.format(sum))
```

### while문

```
sum = 0
n = 1
while n < 11:
    sum += n
    n += 1
print('sum : {}'.format(sum))
```

for문이 적합한 경우

while문이 적합한 경우

## • while문이 적합한 경우

- 조건에 의한 반복이라면 while문이 for문 보다 적합하다.

1부터 시작해서 7의 배수의 합이 50이상인 최초의 정수 출력

for문

```
sum = 0
maxInt = 0
for i in range(1, 101):

    if i % 7 == 0 and sum <= 50:
        sum += i
        maxInt = i

    print('i : {}'.format(i))

print('7의 배수의 합이 50이상인 최초의 정수 : {}'.format(maxInt))
```

for문이 적합한 경우

while문이 적합한 경우

## • while문이 적합한 경우

- 조건에 의한 반복이라면 while문이 for문 보다 적합하다.

1부터 시작해서 7의 배수의 합이 50이상인 최초의 정수 출력

### while문

```
sum = 0
maxInt = 0
n = 1
while n <= 100 and sum <= 50:
    n += 1

    if n % 7 == 0:
        sum += n
        maxInt = n

    print('n : {}'.format(n))

print('7의 배수의 합이 50이상인 최초의 정수 : {}'.format(maxInt))
```

## • 실습

조건에 의한 반복 이란?

while문 사용 방법

실습

- 다음 반복 실행을 위해서 for문과 while문 중 가장 적합한 구문을 이용해서 프로그램을 만들어 보자.

자동차 바퀴가 한번 구를 때마다 0.15mm씩 마모된다고 한다.

현재의 바퀴 두께가 30mm이고 최소 운행 가능 바퀴 두께가 20mm라고 할 때 앞으로 구를 수 있는 횟수를 구해보자.

```
currentThickness = 30
rotationCount = 0
removeThickness = 0.15

while currentThickness >= 20:
    rotationCount += 1
    currentThickness -= removeThickness

safeRotationCount = rotationCount - 1
print('운행 가능 횟수 : {}'.format(safeRotationCount))
```



운행 가능 횟수 : 66