

# 프로그래밍특론

국민대학교  
이정미 교수

kook906@kookmin.ac.kr

# 반복문 (FOR, WHILE)

# 학습내용

반복문 – for문

반복문 – while문

반복문예제

실습예제

반복문 - for문

# 반복문 - for문

- 기본 형식

for 변수 in range(시작값, 종료값, 증가값):  
    명령문

for 변수 in range(반복값):  
    명령문

## for문 실습

```
for 변수 in range(1, 5, 1):  
    print("반복문의 기본  
    형식입니다")
```

5는  
종료값이므로  
4번까지만  
명령이  
실행된다.

## 실행 결과

반복문의 기본 형식입니다  
반복문의 기본 형식입니다  
반복문의 기본 형식입니다  
반복문의 기본 형식입니다

# 반복문 - for문

## for문 실습

```
for n in range (1, 5, 1) :  
    print("안녕하세요")
```

## 실행 결과

안녕하세요  
안녕하세요  
안녕하세요  
안녕하세요

## for문 실습

```
for n in range (5) :  
    print("*안녕하세요")
```

## 실행 결과

\*안녕하세요  
\*안녕하세요  
\*안녕하세요  
\*안녕하세요  
\*안녕하세요

# 반복문 - for문

## for문 실습

```
for n in range (1, 5, 1) :  
    print("Count %d" % n )  
print("종료합니다.")
```

## 실행 결과

```
Count 1  
Count 2  
Count 3  
Count 4  
종료합니다.
```

## for문 실습

```
for n in range (5) :  
    print("Count %d" % n )  
print("종료합니다.")
```

## 실행 결과

```
Count 0  
Count 1  
Count 2  
Count 3  
Count 4  
종료합니다.
```

# 반복문 - for문

for 변수 in (시퀀스):  
 반복 몸체인\_문장들

## for문 실습

```
for i in 1,2,3:  
    print('로그인')  
    print('캐릭터 및 아이템 등 여러 옵션 선택')  
    print('게임 실행')  
    print('게임 종료')
```



## 반복문 - for문

- 여러 개의 값을 갖는 시퀀스에서 변수에 하나의 값을 순서대로 할당한다.
- 할당된 변수값을 갖고 블록의 문장들(문장1,문장2)를 순차적으로 실행한다.
- 반복 몸체인 문장1, 문장2에서 변수를 사용할 수 있다.
- 시퀀스의 그 다음 값을 변수에 할당해 다시 반복 블록을 실행한다.
- 이러한 과정을 시퀀스의 마지막 항목까지 수행한다.

## 반복문 - for문

for 변수 in (시퀀스):  
 반복 몸체인\_문장들

### for문 실습

#% 04-06numseq.py 수의 나열에서 합과 평균 구하기

```
sum = 0
for i in 1.1, 2.5, 3.6, 4.2, 5.4:
    sum += i
    print(i, sum)
else:
    print('합: %.2f, 평균: %.2f' % (sum, sum / 5))
```

## 반복문 - for문

- 내장 함수 range()를 사용한 for문
- range(5)
  - 정수 0에서 4까지 5개의 항목인 정수로 구성되는 시퀀스  
✓0 1 2 3 4

# 반복문 - for문

## for문 실습

```
# %% 04-08coffeeorder.py 3회에 걸쳐 커피 주문 받기
for i in range(3):
    coffee = input("주문하세요! [아메리카노] [카페라떼] [카푸치노] >> ")
    if coffee == '아메리카노':
        print('%s 주문' % coffee)
    elif coffee == '카페라떼':
        print('%s 주문' % coffee)
    elif coffee == '카푸치노':
        print('%s 주문' % coffee)
    else:
        print('모르겠어요.')
else:
    print('주문을 마치겠습니다.')
```

# 반복문 - for문

## for문 실습

# % 04-08coffeeorder2.py      3회에 걸쳐 커피 주문 받기

```
for i in range(3):
    coffee = input("주문하세요! [아메리카노] [카페라떼] [카푸치노] >> ")

    if coffee == '아메리카노' or coffee == '카페라떼' or coffee == '카푸치노':
        print('%s 주문' % coffee)
    else:
        print('모르겠어요.')

else:
    print('주문을 마치겠습니다.')
```

# 반복문 - for문

## for문 실습

#% 04-09findnum.py 지정된 최소 한 자릿수가 포함된 두 자리 정수 찾기

```
n = input("10진수의 한 자릿수 입력 >> ")
print('두 자릿수 정수에서 최소 한 자릿수가 %s인 정수 찾기: ' % n)
print(' 결과 '.center(50, '='))
```

```
for i in range(10, 100):
    snum = str(i)
    if n in snum:
        print(i, end= ' ')
```

# 반복문 - for문

## for문 실습

```
#% 04-10checkrides.py 어린이를 위한 놀이기구 탑승 검사
MAXNUM = 4
MAXHEIGHT = 130
more = True
cnt = 0
while more:
    height = float(input("키는 ? "))
    if height < MAXHEIGHT:
        cnt += 1
        print('들어가요.', '%d명' % cnt)
    else:
        print('커서 못 들어갑니다.')
    if cnt == MAXNUM:
        more = False
    else:
        print('%d명 모두 찾습니다. 다음 번에 이용하세요.' % cnt)
```

## 반복문 - while문



# 반복문 - while문

## 기본 형식

while (조건) :

수행할 문장1

수행할 문장 :


## while문 실습

```
n = 1
```

```
while n <= 5 :
```

```
    print('안녕하세요')
```

```
    print('수고하셨습니다.')
```



무한 루프에  
빠집니다.

당황하지 말고!!!!

control + c 를 눌러서 중지 시킨다.

# 반복문 - while문

## 무한 루프

True 는 무한 루프를 발생시킨다.

### while문 실습

```
import time
x=1
while True :
    print( x )
    time.sleep(1)
print("end")
```



당황하지 말고!!!!  
control + c 를 눌러서 중지 시킨다.

# 반복문 - while문과 for문 비교

## 기본 형식

while (조건) :  
    수행할 문장1  
    수행할 문장2 :  
    ...

for 변수 in range(반복값):  
    명령문

### while문 실습

5번 반복

```
n = 1
while n <= 5 :
    print( " 안녕하세요 " )
    n=n+1
print("수고하셨습니다.")
```

### for문 실습

5번 반복

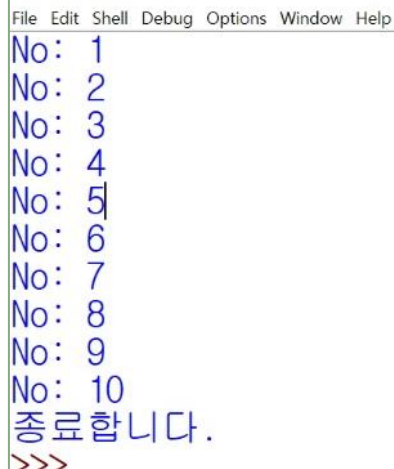
```
for 변수 in range(5):
    print("반복문의 기본형식입니다")
```

# 반복문 - while문과 for문 비교

## while문 실습

```
n = 1
while n <= 10 :
    print("No : ", n)
    n += 1
print("종료합니다.")
```

## while문 실행 결과

A screenshot of a Python interpreter window. The window has a menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The output area shows the execution of a while loop. The text is as follows:

```
No: 1
No: 2
No: 3
No: 4
No: 5
No: 6
No: 7
No: 8
No: 9
No: 10
종료합니다.
>>>
```

# 반복문 - while문과 for문 비교

## while문 실습 응용

```
n = 1
sum=0
while n<=10 :

    sum=sum+n
    print("No:",n,"합계:",sum)
    n=n+1
print("종료합니다.")
```

## while문 실습 결과

File Edit Shell Debug Options Window Help

```
No: 1 합계: 1
No: 2 합계: 3
No: 3 합계: 6
No: 4 합계: 10
No: 5 합계: 15
No: 6 합계: 21
No: 7 합계: 28
No: 8 합계: 36
No: 9 합계: 45
No: 10 합계: 55
종료합니다.
```

# 반복문 - while문과 for문 비교

## while문 실습 응용

```
print("구구단 7단")
```

```
n = 1
```

```
while n <= 9 :
```

```
    print("7 x %d = %d" % (n, 7*n) )
```

```
    n = n + 1
```

## while문 실습 결과

File Edit Shell Debug Options Window Help

=====

구구단 7단

7 x 1 = 7

7 x 2 = 14

7 x 3 = 21

7 x 4 = 28

7 x 5 = 35

7 x 6 = 42

7 x 7 = 49

7 x 8 = 56

7 x 9 = 63

## 반복문 예제

# If문 예제

다음은 참고로 1에서 99까지 정수인 난수를 2개 생성해 곱하기의 결과를 출력하는 프로그램을 작성하시오.

- 반복 while True: 로 계속 난수 발생하여 출력
- 표준 입력으로 y 또는 n을 입력 받아 n이면 반복을 종료하고 프로그램을 마침

```
import random
while True:
    num1 = random.randint(1, 99);
    num2 = random.randint(1, 99);
    result = num1 * num2
    print('%d * %d = %d' % (num1, num2, result))
    print()
    more = input('계속 y / n ? ')
    if more in 'nN':
        break
```



# If문 예제

정수 1개를 표준 입력으로 받아 소수인지를 판별하는 프로그램을 작성하시오.

- 2 이상의 정수를 입력하여 입력한 정수의 판별

소수(prime number)인지를 판별한 2 이상의 정수 입력 >> 11

정수 11는 소수입니다.

```
n = int(input('소수(prime number)인지를 판별한 2 이상의 정수 입력 >> '))
cnt = 2
for i in range(2, n // 2 + 1):
    if n % i == 0:
        cnt += 1
        break
if cnt == 2:
    print('정수 %d는 소수입니다.' % n)
else:
    print('정수 %d는 소수가 아닙니다.' % n)
```

수고 하였습니다.