

Loop 2

By 윤명근 / 박수현

수업목표

- What is a nested loop
- Variable loops
- Variable nested loops
- Examples

What is a nested loop

- Nested for loop
 - 문법

```
for iterating_var in sequence:  
    for iterating_var in sequence:  
        statements(s)  
    statements(s)
```

http://www.tutorialspoint.com/python/python_nested_loops.htm

What is a nested loop

- Nested for loop

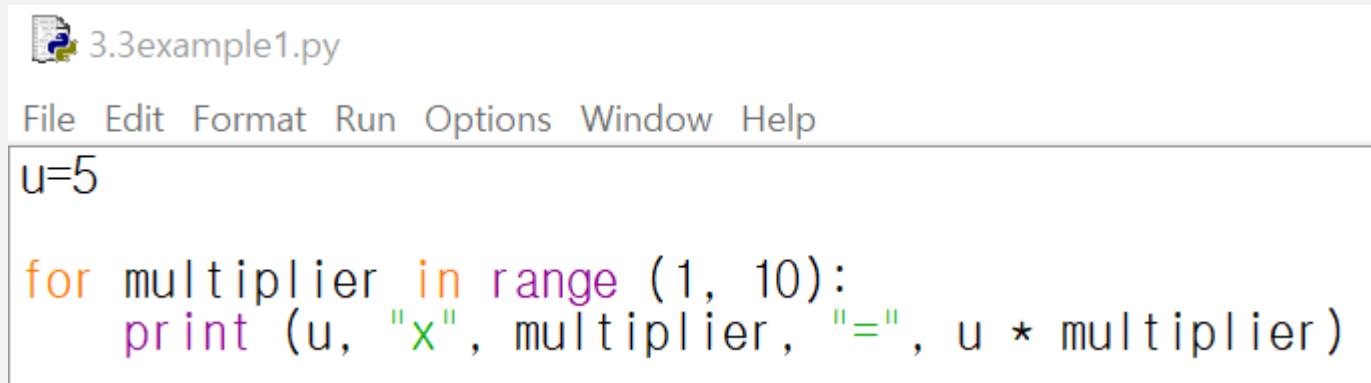
```
3.3example11.py
File Edit Format Run Options Window
for i in range (10):
    for j in range (10):
        print('*', end='')
    print ()
```

```
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
>>>
```

- 줄 바꿈 문자를 제거하기 위해서 위와 같이 끝 문자를 지정할 수 있는 **end 파라미터**를 설정하면 됨. 지정하지 않으면 디폴트로 \n 문자가 세팅

What is a nested loop

- 곱셈표 출력
 - 5단 출력: for loop 하나면 해결

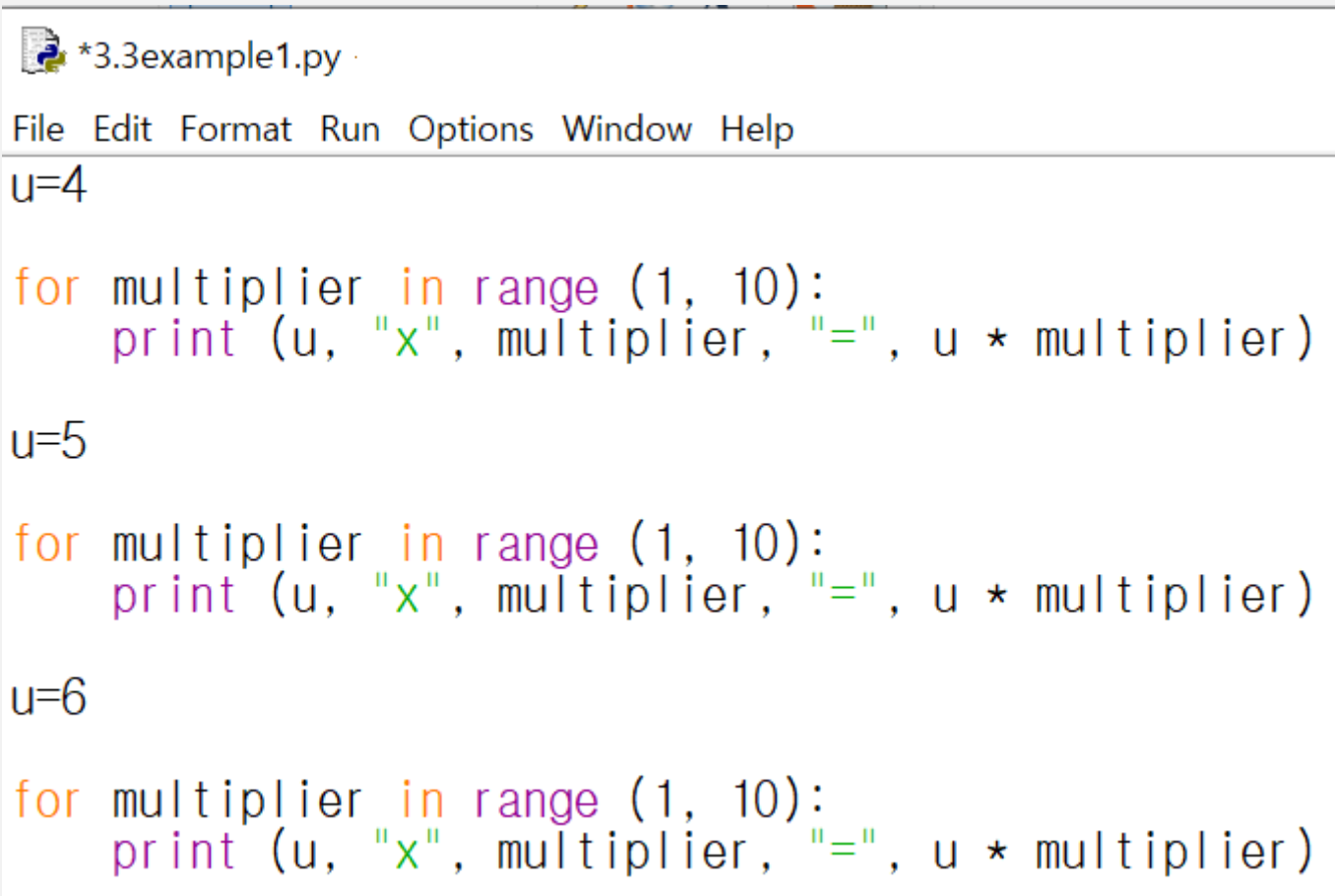


```
3.3example1.py
File Edit Format Run Options Window Help
u=5
for multiplier in range(1, 10):
    print(u, "x", multiplier, "=", u * multiplier)
```

```
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
>>>
```

What is a nested loop


- 곱셈표 출력
 - 4단부터 6단까지 출력: 세 개의 for loop으로 해결 → **Bad idea ! Why?**



```
*3.3example1.py
File Edit Format Run Options Window Help
u=4
for multiplier in range (1, 10):
    print (u, "x", multiplier, "=", u * multiplier)
u=5
for multiplier in range (1, 10):
    print (u, "x", multiplier, "=", u * multiplier)
u=6
for multiplier in range (1, 10):
    print (u, "x", multiplier, "=", u * multiplier)
```

What is a nested loop

- 곱셈표 출력 - 1
 - 4단부터 6단까지 출력: 이중 for loop으로 해결!

 3.3example2.py

File Edit Format Run Options Window Help

```
for i in range (4, 7):  
    print("--", i, "단")  
  
    for multiplier in range (1, 10):  
        print (i, "x", multiplier, "=", i * multiplier)  
  
    print()
```


```
-- 4 단  
4 x 1 = 4  
4 x 2 = 8  
4 x 3 = 12  
4 x 4 = 16  
4 x 5 = 20  
4 x 6 = 24  
4 x 7 = 28  
4 x 8 = 32  
4 x 9 = 36
```

```
-- 5 단  
5 x 1 = 5  
5 x 2 = 10  
5 x 3 = 15  
5 x 4 = 20  
5 x 5 = 25  
5 x 6 = 30  
5 x 7 = 35  
5 x 8 = 40  
5 x 9 = 45
```

```
-- 6 단  
6 x 1 = 6  
6 x 2 = 12  
6 x 3 = 18  
6 x 4 = 24  
6 x 5 = 30  
6 x 6 = 36  
6 x 7 = 42  
6 x 8 = 48  
6 x 9 = 54
```

What is a nested loop

- 곱셈표 출력 - 2
 - 4단부터 6단까지 출력: 이중 for loop으로 해결!

 3.3example2-1.py

File Edit Format Run Options Window Help

```
for i in range(4, 7) :  
  
    print("-- {}단".format(i))  
  
    for j in range(1, 10) :  
        print("{} x {} = {}".format(i, j, i*j))  
    print()
```

-- 4단
4 x 1 = 4
4 x 2 = 8
4 x 3 = 12
4 x 4 = 16
4 x 5 = 20
4 x 6 = 24
4 x 7 = 28
4 x 8 = 32
4 x 9 = 36

-- 5단
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45

-- 6단
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54

Variable loops

- 반복횟수가 가변적인 상황
 - Ex) 입력된 횟수만큼 별표 출력

 star.py

File Edit Format Run Options Window Help

```
print("\n1) for 사용")

num = int(input("2) -- Type the number of stars : "))
for multiplier in range(1, num+1):
    print("*", end='')

print("\n\n3) print 사용")

num = int(input("4) -- Type the number of stars : "))
print("*" * num, end='')
```

```
1) for 사용
2) -- Type the number of stars : 5
*****

3) print 사용
4) -- Type the number of stars : 10
*****
^^^
```

Variable nested loops

- 반복횟수가 가변적인 상황
 - Ex) 입력된 횟수만큼 별표 출력

```
variable_nested_loop.py
File Edit Format Run Options Window Help
row = int(input("\n-- Type the number of rows : "))
col = int(input("-- Type the number of columns : "))

print("\n1) row = {}, col = {}".format(row, col))

print("\n2) for 사용")
for i in range(1, row + 1) :
    for j in range(1, col + 1) :
        print("*", end='')

    print()

print("\n3) print 사용")
for i in range(1, row + 1) :
    print("*" * col)
```

```
-- Type the number of rows : 5
-- Type the number of columns : 10
```

1) row = 5, col = 10

2) for 사용

```
*****
*****
*****
*****
*****
```

3) print 사용

```
*****
*****
*****
*****
*****
```


Nested loop

- Nested while loop

```
while expression:  
    while expression:  
        statement(s)  
    statement(s)
```

Nested loop

- Nested while loop

 3.2.example12.py

File Edit Format Run Options Window Help

```
row = int(input("Wn-- Type the number of rows : "))
col = int(input("-- Type the number of columns : "))
```

```
print("Wn1) row = {}, col = {}".format(row, col))
```

```
i = 0
```

```
while (i < row) :
    j = 0

    while(j < col) :
        print('*', end='')
        j = j + 1
```

```
print()
i = i + 1
```

```
-- Type the number of rows : 5
-- Type the number of columns : 20
```

```
1) row = 5, col = 20
```

```
*****
*****
*****
*****
*****
```

```
///
```

실습

- 구구단 출력
- For loop을 while loop으로 전환

실습

- '*' 삼각형 출력
- 주어진 n 에 따라 높이가 n 인 직각 삼각형 출력

– $n=3$

*

**

– $n=5$

*

**

실습

- '*' 삼각형 출력
- 주어진 n에 따라 높이가 n인 직각 삼각형 출력

```
n = int(input("n > "))  
  
for i in range(n):  
    for j in range(i + 1):  
        print("*", end="")  
    print()
```

실습

- 자연수 n 을 입력 받고, 다음과 같이 첫 번째 줄에는 '*'을 1개, 둘째 줄에는 2개 n 번째 줄에는 n 개를 출력하는 파이썬 코드를 작성하세요.

```
n > 5
*
**
***
****
*****
```


실습

- 자연수 n 을 입력 받고, 다음과 같이 첫 번째 줄에는 '*'을 1개, 둘째 줄에는 2개 n 번째 줄에는 n 개를 출력하는 파이썬 코드를 작성하세요.

```
n = int(input("n > "))

for i in range(n):
    for j in range(n - i - 1):
        print(end = ' ')
    for j in range(i + 1):
        print("*", end="")
    print()
```

숙제 1 – 피라미드 출력

- 자연수 n 을 입력 받고, 다음과 같은 모양을 출력하시오. 첫 번째 줄에는 '*'을 1개, n 번째 줄에는 $(2*n-1)$ 개를 출력하는 파이썬 코드를 작성하세요.

```
n > 5
*
***
*****
*****
*****
*****
```

숙제 - 2 조화급수와 로그

- 조화급수와 로그

- 조화급수는 다음과 같이 정의된다.

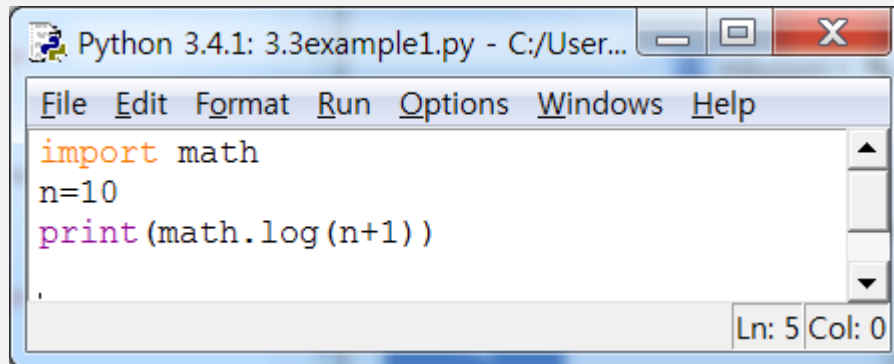
- $1/1 + 1/2 + 1/3 + 1/4 + \dots$

- n 번째 항까지의 조화급수는 $(1/1 + 1/2 + 1/3 + 1/4 + \dots + 1/n)$ 이다.

- 아래 그림($n=5$)과 같이, n 번째 항까지의 조화급수는 $\int_1^{n+1} \frac{1}{x} dx$ 와 거의 같아지므로, $\ln(n+1)$ 로 계산될 수 있다.

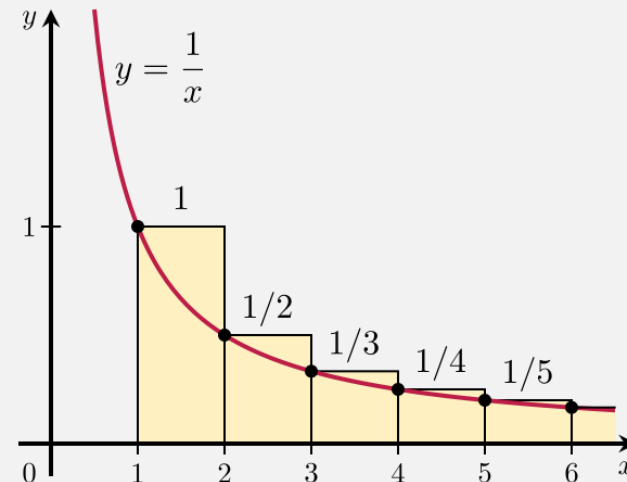
- $k=[1..100]$ 범위에서 실제 조화급수의 값과 $\ln(n+1)$ 로 측정한 값을 출력하고, 오차율((측정값-실제값)/실제값)도 출력하시오.

- $\ln(n+1)$ 을 계산하려면 `import math`가 필요



```
Python 3.4.1: 3.3example1.py - C:/User...
File Edit Format Run Options Windows Help
import math
n=10
print(math.log(n+1))
Ln: 5 Col: 0
```

국민대학교 소프트웨어학부



Homework

- 제출방법

파일명 :

이름-학번-피라미드 출력.py

이름-학번-harmonic-series.py

- 파일이 여러 개일 경우 zip으로 묶어서 ecampus 숙제제출 link에 upload
- 제출마감
 - 2023.10.17(화) 13:00
 - 제출 마감 일시까지만 제출 가능. 마감일시 이후 ecampus 숙제제출 링크 자동 close