



파이선으로 배우는 데이터구조

Data Structures Learning with Python

김영훈

한양대학교 ERICA 인공지능학과

If 문

- ▶ 조건문을 사용하면 조건이 참인지 거짓인지에 따라 다양한 코드 블록을 실행할 수 있습니다.

```
if condition:  
    # 조건이 참이면 실행할 코드
```

If 문

▶ If....else...

```
if condition:  
    # 조건이 참이면 실행할 코드  
else:  
    # 조건이 거짓일 때 실행할 코드
```

If 문

▶ If....elif...

```
if condition1:  
    # 조건1이 참이면 실행할 코드  
elif condition2:  
    # 조건2가 참이면 실행할 코드
```

For 문

- ▶ 루프를 사용하면 코드 블록을 여러 번 실행할 수 있습니다.

```
for i in range(10):  
    print(i)
```

For 문

- ▶ 루프를 사용하면 코드 블록을 여러 번 실행할 수 있습니다.

```
fruits = ["apple", "banana", "cherry"]  
for fruit in fruits:  
    print(fruit)
```

루프 제어 문

- ▶ 루프제어 문은 루프의 실행을 변경할 수 있습니다.
 - ▶ break: 루프를 조기에 종료합니다.
 - ▶ continue: 현재 반복의 나머지 부분을 건너뛰고 다음 반복으로 이동합니다.
 - ▶ pass: 널 연산으로, 실행 시 아무 일도 일어나지 않습니다.

```
for num in range(5):  
    if num == 3:  
        break  
    print(num)
```

루프 제어 문

- ▶ 루프제어 문은 루프의 실행을 변경할 수 있습니다.
 - ▶ break: 루프를 조기에 종료합니다.
 - ▶ continue: 현재 반복의 나머지 부분을 건너뛰고 다음 반복으로 이동합니다.
 - ▶ pass: 널 연산으로, 실행 시 아무 일도 일어나지 않습니다.

```
for i in range(1, 11): # Numbers from 1 to 10
    if i % 3 == 0:
        continue
    print(i)
```


루프 제어 문

- ▶ 루프제어 문은 루프의 실행을 변경할 수 있습니다.
 - ▶ break: 루프를 조기에 종료합니다.
 - ▶ continue: 현재 반복의 나머지 부분을 건너뛰고 다음 반복으로 이동합니다.
 - ▶ pass: 널 연산으로, 실행 시 아무 일도 일어나지 않습니다.

```
tasks = ["task1", "task2", "task3"]

for task in tasks:
    if task == "task1":
        print(task)
        pass
    elif task == "task2":
        print(task)
        pass
    else:
        pass
```

중첩 루프

- ▶ 루프와 조건을 서로 중첩하여 보다 복잡한 제어 구조를 만들 수 있습니다.

```
for i in range(3):  
    for j in range(3):  
        print(f"i: {i}, j: {j}")
```



코딩 시작!

연습 문제 풀이

For/While 1.

- ▶ 1부터 100 사이의 숫자 중에서 7 또는 5의 배수를 출력하는 코드를 작성합니다.

Expected Output

5
7
10
14
...
90
91
95
98

```
for i in range(1, 101):  
    if i % 7 == 0 or i % 5 == 0:  
        print(i)
```

For/While 2.

- ▶ (별찍기) 다음과 같은 패턴을 출력하는 Python 프로그램을 작성하십시오.

Expected Output

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

```
n = 5
for i in range(n):
    for j in range(i):
        print('*', end='')
    print("")
```

```
for i in range(n,0,-1):
    for j in range(i):
        print('*', end='')
    print("")
```

For/While 3.

- ▶ 일련의 숫자 중에서 짝수의 개수와 홀수의 개수를 출력하는 Python 프로그램을 작성하십시오.

Expected Output

Number of even :
4

Number of odd : 5

```
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9]
count_odd = 0
count_even = 0
for x in numbers:
    if not x % 2:
        count_even += 1
    else:
        count_odd += 1
print("Number of even :", count_even)
print("Number of odd :",count_odd)
```

For/While 4.

- ▶ 0부터 6까지의 숫자를 출력하되 3과 6을 제외하고 출력하는 Python 프로그램을 작성하십시오.

Expected Output

0 1 2 4 5

```
for x in range(6):  
    if (x == 3 or x == 6):  
        continue  
    print(x, end=' '  
print("\n")
```

For/While 5.

- ▶ 0부터 50 사이의 피보나치 수열을 출력하는 Python 프로그램을 작성하십시오.

Expected Output

1
1
2
3
5
8
13
21
34

```
x, y = 0, 1
```

```
while y < 50:  
    print(y)  
    x, y = y, x + y
```

```
a, b = 1, 1
```

```
print(a)  
while True:  
    if b > 50:  
        break
```

```
print(b)
```

```
t = a + b
```

```
a = b
```

```
b = t
```


For/While 6.

- ▶ 1부터 20까지의 숫자를 출력하되, 숫자에 3, 6, 9가 들어있는 경우 대신에 '짝'을 출력하는 Python 프로그램을 작성하십시오.

Expected Output

1
2
짝
4
5
...
17
18
짝

```
for i in range(1, 20):  
    n = i % 10  
  
    if n == 3 or n == 6 or n == 9:  
        print('짝')  
    else:  
        print(i)
```

For/While 7.

- ▶ 숫자의 각 자리가 짝수인 10에서 80(둘 다 포함) 사이의 숫자를 찾는 Python 프로그램을 작성하세요.

Expected Output

20
22
24
...
62
64
66
68
80

```
for i in range(10, 80):  
    c1 = i % 10 % 2 == 0  
    c2 = i // 10 % 2 == 0  
    if c1 and c2:  
        print(i)  
  
for n in range (10, 81):  
    a = n % 10  
    b = n // 10  
  
    if a % 2 == 0 and b % 2 == 0:  
        print(n)
```

For/While 8.

- ▶ 입력받은 숫자의 구구단(1부터 9까지)를 만드는 Python 프로그램을 작성하세요.

Input

Input a number: 5

Expected Output

5 x 1 = 5

5 x 2 = 10

...

5 x 8 = 40

5 x 9 = 45

```
n = int(input("Input a number: "))
```

?

```
print(n, 'x', i, '=', n * i)
```

For/While 9.

- ▶ for 루프를 사용하여 다음 패턴을 구성하는 Python 프로그램을 작성하세요.

Expected Output

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```

```
for i in range(10):
    print(str(i) * i)
```

For/While 10.

- ▶ 1부터 50까지의 숫자들의 평균과 분산을 구하는 프로그램을 작성하시오.

Expected Output

Mean: 25.5

Variance:208.25

```
mean = 0  
var = 0
```

?

?

```
print('Mean:', mean)  
print('Variance:', var)
```

그리고 mean과 var
계산을 각각 한 줄에
계산하는 방법도 있음

For/While 11.

- ▶ 입력받은 수가 짝수일 때까지 사용자의 입력을 받는 프로그램을 작성하십시오.

Expected Output

number: 1 ↵
number: 3 ↵
number: 2 ↵

```
n = int(input('number:'))
```

```
while ?:  
    n = int(input('number:'))
```

For/While 12.

- ▶ (별찍기) 다음과 같은 패턴을 출력하는 Python 프로그램을 작성하십시오. :
L

Expected Output

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

```
for i in range(6):  
    print('*')  
for i in range(5):  
    print('*', end='')
```

For/While 13.

- ▶ while 루프를 사용하여 1부터 10까지 자연수를 출력하는 Python 프로그램을 작성하십시오

Expected Output

1
2
3
4
5
6
7
8
9
10

```
i = 1  
while i <= 10:  
    print(i)  
    i += 1
```


For/While 14.

- ▶ 중첩 루프를 사용하여 다음 숫자 패턴을 인쇄하는 Python 프로그램을 작성하세요.

Expected Output

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

```
row = 5
for i in range(1, row + 1, 1):
```

?

```
print("""
```

For/While 15.

- ▶ 사용자로부터 숫자를 받아 1부터 주어진 숫자까지의 모든 숫자의 합을 계산하는 프로그램을 작성하세요. 예를 들어, 사용자가 10을 입력한 경우 출력은 55가 되어야 합니다.

Input

Enter number : 10

Expected Output

Sum is: 55

```
s = 0
```

```
n = int(input("Enter number :"))
```

```
?
```

```
print("\n")
```

```
print("Sum is: ", s)
```

For/While 16.

- ▶ 다음 조건을 만족하는 숫자만 목록에서 표시하는 프로그램을 작성하세요. 1) 숫자는 5로 나누어야 합니다. 2) 숫자가 150보다 크면 건너뛰고 다음 숫자로 이동합니다. 3) 숫자가 500보다 크면 루프를 중지합니다.

Input

[12, 75, 150, 180,
145, 525]

Expected Output

75
150
145

```
numbers = [12, 75, 150, 180, 145, 525]
```

```
for item in numbers:
```

?

For/While 17.

- ▶ while 루프를 사용하여 숫자의 총 자릿수를 계산하는 프로그램을 작성하세요.

Input
76542

```
n = int(input())  
count = 0
```

Expected Output
5

```
while n > 0:
```

?

```
    print(count)
```

For/While 18.

- ▶ 중첩 for 루프를 사용하여 다음과 같은 패턴을 출력하는 Python 프로그램을 작성하십시오.

Expected Output

```
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

```
for i in range(5):
    for j in range(5 - i):
        print(5 - j - i, end=' ')
    print()
```

For/While 19.

- ▶ 초항(a)과 공차(d)가 주어졌을 때, 5개의 항을 가지는 등차수열을 나타내는 프로그램을 작성하시오.

Expected Output

a: 1 ↵

d: 3 ↵

1

4

7

10

13

```
a = int(input('a:'))
```

```
d = int(input('d:'))
```

```
for i in range(5):
```

```
    print(a + d * i)
```

For/While 20.

- ▶ for 루프를 사용하여 -10에서 -1까지의 숫자를 출력하는 프로그램을 작성하세요.

Expected Output

-10
-9
-8
-7
-6
-5
-4
-3
-2
-1

```
for num in range(-10, 0, 1):  
    print(num)
```

For/While 21.

- ▶ 1부터 30까지의 숫자 중에서 모든 소수를 표시하는 프로그램을 작성하세요.

Expected Output

2
3
5
7
11
13
17
19
23
29

```
for i in range(2, 31):  
    is_prime = True  
    for j in range(2, i):  
        if i % j == 0:  
            is_prime = False  
  
    if is_prime:  
        print(i)
```


For/While 22.

- ▶ 입력받은 이진수 숫자를 십진수로 나타내는 프로그램을 작성하시오.

Input
11010

```
n = 11010  
out = 0  
val = 1
```

Expected Output
26

```
while n > 0:  
    out += (n % 10) * val  
    val *= 2  
    n = n // 10  
  
print(out)
```

For/While 23.

- ▶ 주어진 정수를 뒤집는 프로그램을 작성하세요.

Input
76542

Expected Output
24567

```
num = int(input())
output = 0

while num > 0:
    reminder = num % 10
    output = (output * 10) + reminder
    num = num // 10
print(output)
```

For/While 24.

- ▶ 루프를 사용하여 주어진 리스트의 홀수 인덱스 위치에 있는 원소를 출력하는 프로그램을 작성하세요.

Input

[10, 20, 30, 40, 50,
60, 70, 80]

Expected Output

20 40 60 80

```
my_list = [10, 20, 30, 40, 50, 60, 70, 80]
for i in range(len(my_list)):
    if i % 2 == 1:
        print(i, end=" ")
```

For/While 25.

- ▶ 1부터 주어진 숫자까지의 모든 숫자의 세제곱을 출력하는 프로그램을 작성하세요.

Input

6

Expected Output

1

8

27

64

125

216

```
n = int(input())
```

```
for i in range(1, n + 1):  
    print(i * i * i)
```

For/While 26.

- ▶ n항까지의 계열의 합을 계산하는 프로그램을 작성하세요. 예를 들어, n=5이면 계열은 $2 + 22 + 222 + 2222 + 22222 = 24690$ 이 됩니다.

Input

n = 5

start = 2

Expected Output

Sum of above series
is: 24690

```
n = 5
```

```
start = 2
```

```
sum_seq = 0
```

```
for i in range(0, n):
```

```
    sum_seq += start
```

```
    start = start * 10 + start
```

```
print("Sum of above series is:", sum_seq)
```

For/While 27.

- ▶ 다음과 같은 패턴을 출력하는 Python 프로그램을 작성하십시오. : Z

Expected Output

```
*****
```

```
    *
```

```
  *
```

```
* 
```

```
* 
```

```
* 
```

```
*****
```

```
for i in range(7):  
    print('*', end='')  
print()
```

```
for i in range(2, 7):  
    for j in range(7 - i):  
        print(' ', end='')  
    print('*')
```

```
for i in range(7):  
    print('*', end='')  
print()
```

For/While 28.

- ▶ 입력받은 숫자의 약수를 출력하는 Python 프로그램을 작성하십시오.

Input

12

Expected Output

1

2

3

4

6

12

```
n = int(input())
```

```
for i in range(1, n + 1):
```

```
    if n % i == 0:
```

```
        print(i)
```

For/While 29.

- ▶ 입력받은 정수에 대하여 다음 두 동작을 수행하는 프로그램을 작성하시오.
 - ▶ 짝수이면 2로 나누고, 홀수이면 3배해서 1을 더함
 - ▶ 1이 될 때 까지 반복

Input

5

```
n = int(input())  
print(n)
```

Expected Output

5

16

8

4

2

1

```
while n != 1:  
    if n % 2 == 0:  
        n = n // 2  
    else:  
        n = 3 * n + 1  
print(n)
```


For/While 30.

- ▶ 입력받은 숫자의 팩토리얼을 계산하는 Python 프로그램을 작성하십시오.

Input

5

Expected Output

120

```
n = input()
out = 1
for i in range(1, int(n) + 1):
    out *= i
print(out)
```