



# PYTHON

Day3

# RECAP

- Variables
- Operation, Operator
- String
- User Input

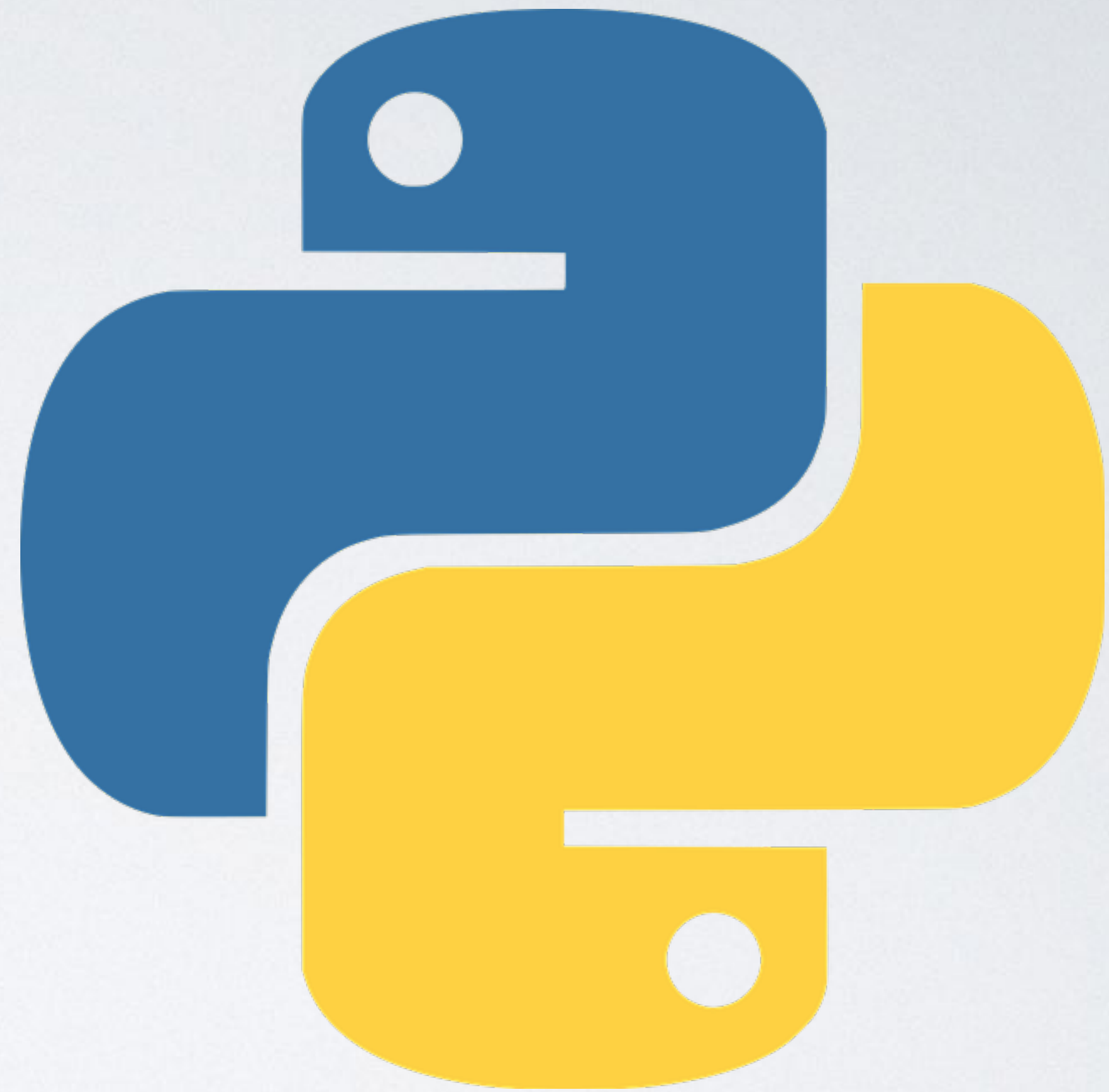


# INDEX

- Type Casting
- List
- Conditionals
  - if, else, elif
- Iterations
  - for
  - while

# PYTHON

## Type Casting





# TYPE CASTING

`_int = 1`

`_float = 3.0`

`_string = "5"`

`_another_string = "a"`

`int(_string)`

`float(_int)`

`str(_float)`

`int(_another_string)`

# TYPE CASTING

```
age = eval(input("How old are you? "))  
type(age)  
str(age)  
print(age)
```



# PYTHON

List



# LIST

- 여러 자료형을 묶어 표현하는 자료형
- Python에는 list, tuple, dictionary, set의 묶음 자료형이 존재
- 대괄호([ ]) 로 감싸며, 쉼표(,)로 요소를 구분함  
animals = ["dog", "cat", "hippo"]



# LIST - 요소 추가하기

```
>>> lang = ["python", "c", "java", "golang"]
```

```
>>> lang = []
```

```
>>> lang.append("python")
```

```
>>> lang.append("c")
```

```
>>> lang.append("golang")
```

```
>>> print(lang)
```

```
["python", "c", "golang"]
```

# LIST - 요소 끼워넣기, 삭제하기, 추출하기

```
>>> lang.insert(2, "java")
```

```
>>> lang
```

```
>>> lang.remove("golang")
```

```
>>> lang
```

```
>>> java = lang.pop(2)
```

```
>>> print(lang, java)
```



# LIST - 요소 정렬하기, 인덱스 출력하기

```
>>> numbers = [2, 10, 7, 5]
```

```
>>> numbers.sort()
```

```
>>> numbers
```

```
>>> numbers.reverse()
```

```
>>> numbers
```

```
>>> index_of_two = numbers.index(2)
```

```
>>> index_of_two
```

# LIST - 리스트 합치기

리스트에 리스트를 더한 값을 새로 선언하기

```
>>> numbers = numbers + [5, 6]
```

```
>>> numbers
```

extend 함수를 활용해 리스트를 합치기

```
>>> numbers.extend([7, 8])
```

```
>>> numbers
```



# PYTHON

Conditionals - if, else, elif



# CONDITIONAL STATEMENTS

- 돈이 100000원 이상인가?
- YES: 레스토랑으로 간다
- NO: 편의점으로 간다



# CONDITIONAL STATEMENTS

- 프로그래머가 명시한 조건이 참인지 거짓인지에 따라 다른 계산을 수행하는 것, 또는 그런 문법
- if - 만약에..

# IF - PYTHON의 조건문

```
if 현금 > 100000:
```

```
    레스토랑으로 간다
```



그런데.. 거짓인 경우는 어떡하지?

# ELSE

if 현금 > 100000:

레스토랑으로 간다

else:

편의점으로 간다



여러 조건을 두고 싶은데???

# IF IN THE ELSE

if 현금 > 100000:

레스토랑으로 간다

else:

if 현금 > 50000:

근처 밥집으로 간다

else:

편의점으로 간다



조건에 조건이 반복되면  
TAB 많이 쳐야 하네!!

# ELIF

if 현금 > 100000:

레스토랑으로 간다

elif 현금 > 50000:

근처 밥집으로 간다

else:

편의점으로 간다



# PRACTICE: IF

```
>>> money = 110000  
>>> if money > 100000:  
...     print("go to restaurant")  
...
```

go to restaurant

# PRACTICE: IF ELSE

```
>>> money = 80000  
>>> if money > 100000:  
...     print("go to restaurant")  
... else:  
...     print("go to CVS")  
...
```

go to CVS



# PRACTICE: IF IN ELSE

```
>>> if money > 100000:  
...     print("go to restaurant")  
... else:  
...     if money > 50000:  
...         print("go to bobjib")  
...     else:  
...         print("go to CVS")  
...  
go to bobjib
```

# PRACTICE: ELIF

```
>>> if money > 100000:  
...     print("go to restaurant")  
... elif money > 50000:  
...     print("go to bobjib")  
... else:  
...     print("go to CVS")  
...
```

go to bobjib



# MINI PROJECT

with pair programming

# MINI PROJECT

1. 사용자에게 이름을 입력받아 이름을 이용해 친절하게 숫자 하나를 물어 보세요(1~100)

2. 그 숫자가 미리 만든 정답과 같은지 비교한 뒤,  
같다면 축하를, 다르다면 비난을 하는 프로그램을 만들어 보세요.



# MINI PROJECT

\* Sample output

What is your name? (mirae-N)

Hi, mirae-N. guess the number(1 to 100): (12)

맞을 경우: Congratulations!

틀릴 경우: Wrong! The answer was 12.

# MINI PROJECT - BASE CODE

```
import random
```

```
answer = random.randint(1,100)
```

```
print(answer)
```

———— 이 아래로 작성해주세요 ————



# MINI PROJECT - ANSWER

```
import random
```

```
answer = random.randint(1,100)
```

```
print(answer)
```

```
username = input("What is your name? ")
```

```
guess = int(input("Hi, %s. Guess the number(1 to 100):" % username))
```

```
if guess == answer:
```

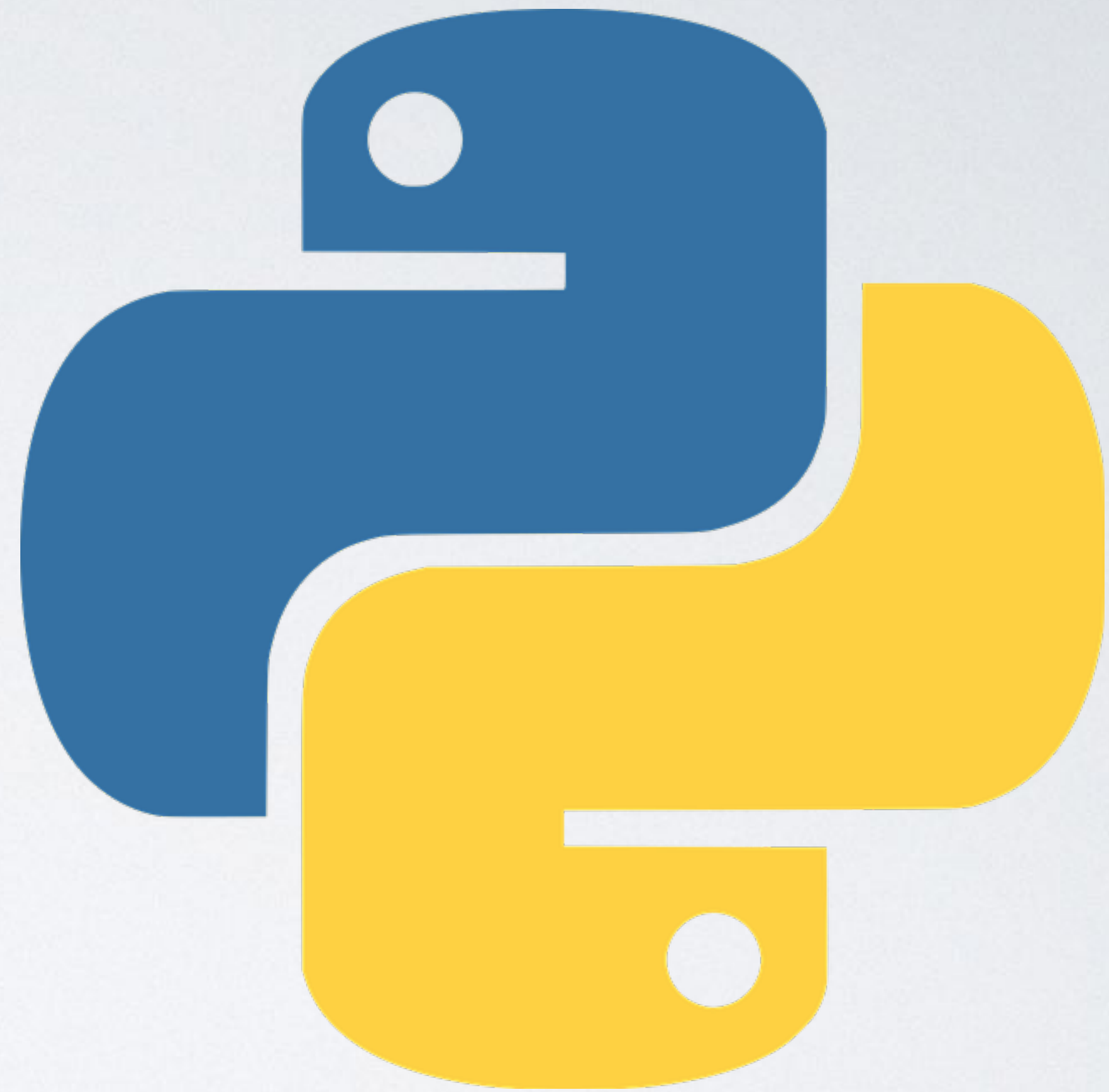
```
    print("Congratulations!")
```

```
else:
```

```
    print("Wrong! The answer was %d." % answer)
```

# PYTHON

Iterations - for





# ITERATIONS

- 어떤 문장을 반복하고 싶을 때 사용하는 구문
- for, while

# ITERATIONS - FOR

문자열 혹은 리스트 요소 하나하나를 변수에 선언하여 반복

for 변수 in (문자열 또는 리스트):

실행문



# ITERATIONS - FOR

```
>>> for i in range(1, 10+1):
```

```
...     print(i)
```

```
...
```

```
>>> for i in ["python", "c", "java"]:
```

```
...     print(i)
```

```
...
```

# ITERATIONS - MAKE AWESUM

```
>>> result = 0
>>> for j in range(1, 10+1):
...     result = result + j
...     print(result)
...
>>> result
```



# PYTHON

Iterations - while



# ITERATIONS - WHILE

조건을 만족하지 않을 때 까지 반복

while 조건:

실행문



# ITERATIONS - WHILE

```
>>> animal = ""  
>>> while animal != "dog":  
...     animal = input("What is your favorite animal? ")  
...     print("Oh, you like %s." % animal)  
...
```

What is your favorite animal? cat  
Oh, you like cat.

What is your favorite animal? lion  
Oh, you like lion.

What is your favorite animal? dog  
Oh, you like dog.

# ITERATIONS - INFINITE LOOP

```
>>> while 1:  
...     print("HAHA! Python is easy!")  
....
```



# MINI PROJECT

with pair programming

# MINI PROJECT

1. 1부터 100까지 반복을 합니다.

2. 조건문을 활용하여

3의 배수일 경우, “fizz”

5의 배수일 경우, “buzz”

15의 배수일 경우, “fizzbuzz”

나머지 경우, 그 숫자를 출력하는 프로그램을 작성하세요.



# MINI PROJECT

\* Sample output

1  
2  
fizz  
4  
buzz  
...  
fizzbuzz  
...  
100

# MINI PROJECT - ANSWER

```
for i in range(1, 100+1):  
    if i % 15 == 0:  
        print("fizzbuzz")  
    elif i % 3 == 0:  
        print("fizz")  
    elif i % 5 == 0:  
        print("buzz")  
    else:  
        print(i)
```



# NEXT CLASS

- Tuple
- Dictionary
- Set