[Session 1] Python Basics Exercise

Note

You don't need to solve everything, because it is not mandatory and graded, just for practice! We will provide a solution after the real-time class. But, if you have some problem and ask us, then we will give you some help or hints. **Good luck!**

Exercise 1

It is a non-programming exercise. Just predict the output of the following code, try this **WITHOUT RUNNING CODE**.

(1.1) Arithmetic operations

```
print(15 + 5 - 3 * 2)
print(15 ** 2)
print(8 / 4)
print(15 // 4)
print(5 % 4)
print(4 % 8)
print(4 // -3)
```

(1.2) Basic use of `print()` function

```
print("a", "b", 3, sep="")
print("55", end=" ")
print("abc")
```

(1.3) String operations

```
print("One"[2])
a = "Two"
print(a[1])
print(a + "four")
print("1" + "2")
print("1" * 10)
print("A\b")
print("O\\O")
print("\nn\n\nn")
```

(1.4) Conditionals

```
result = ""
if 1:
    result += "1"
if []:
    result += "2"
if 0:
    result += "3"
if ():
    result += "4"
if 0.0000000000001:
    result += "5"
if None:
```

```
    result += "6"
if True or False and not False:
    result += "7"
if 1 is 1:
    result += "8"
if "":
    result += "9"

print(result)
```

(1.5) Loop

```
num1 = 0
for i in range(100):
    num1 += i
print(num1)

num2 = 0
for i in [1, 3, 5]:
    for j in [1, 3, 5]:
        num2 += i * j
print(num2)

num3 = 100
num4 = 0
while num3:
    num3 -= 1
    if num3 % 2 == 0:
        continue
    num4 += num3
print(num4)
```

Exercise 2

You will implement something, according to the description.

(2.1) Function 1

Define a function named "is_even" that uses a parameter, `n`. It returns True when `n` is even, and returns False when `n` is odd number. Modify the following code and write your solution.

```
def is_even(n):
    pass
```

Your function should meet the followings...

- `is_even(0)` should be True
- `is_even(4)` should be True
- `is_even(-12412491)` should be False
- `is_even(1)` should be False

(2.2) Function 2

Implement a function named "greeting", that uses parameter name. It returns nothing, but it prints "hello, (some name)!". If no argument is given, then the name should be "Anonymous". Modify the following code and write your solution.

```
def greeting(name):
    pass
```

Your function should meet the followings...

- `greeting("Kevin")` should print, `"hello, Kevin!"`
- `greeting("")` should print, `"hello, !"`
- `greeting()` should print, `"hello, Anonymous!"`

(Additional Question) How about the result of `greeting("\b" * 7)`?

(2.3) Class 1

A class, `Student`, has contains some data: name, ID, and grade. You only need to implement just one method:

- **`get_info()`**: It has no return value, but it prints student's name with the following format. For example, if the name is "Kevin", ID is 205224, and he is 3rd grade:

```
Name: Kevin
ID: 205224
Grade: Junior
```

Instead of representing grade as number, **you should use Freshman, Sophomore, Junior, and Senior**.

You can use the following skeleton.

```
class Student:
    def __init__(self, name, id, grade):
        self.name = name
        self.id = id
        self.grade = grade

    def get_info(self):
        pass
```

Your method should print the followings:

```
a = Student("Bob", 20181055, 4)
a.get_info()
```

```
Name: Bob
ID: 20181055
Grade: Senior
```

```
b = Student("Sandy", 202020412, 1)
b.get_info()
```

```
Name: Sandy
ID: 202020412
Grade: Freshman
```