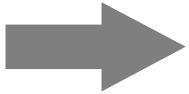
2023 N.E.T Python

Nested Loop

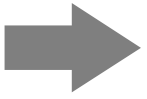
```python
for i in range(2, 10):
    for j in range(1, 10):
        print(i , '*', j, '=', i * j)
    print('---------------')
```



```
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
---------------
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
---------------
4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
```

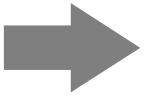
Continue & Break

```python
for i in range(20):
    if (i % 2 == 0):
        continue
    print(i)
```



```
1
3
5
7
9
11
13
15
17
19
```

```python
i = 0
while True:
    print(i)
    i += 1
    if (i == 10):
        break
```



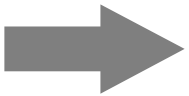
```
0
1
2
3
4
5
6
7
8
9
```

List

- **indexing & operation**

```
list = [ 'abcd', 786 , 2.23, 'john', 70.2 ]
tinylist = [123, 'john']

print(list)
print(list[0] )
print(list[1:3])
print(list[2:])
print(tinylist * 2)
print(list + tinylist)
```



```
['abcd', 786, 2.23, 'john', 70.2]
abcd
[786, 2.23]
[2.23, 'john', 70.2]
[123, 'john', 123, 'john']
['abcd', 786, 2.23, 'john', 70.2, 123, 'john']
```

List

- **input**

```
inputList = list(map(int,input.split()))
```

List

- **change & delete**

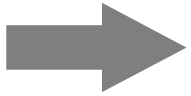
```
>>> a = [1, 2, 3]
>>> a[2] = 4
>>> a
[1, 2, 4]
```

```
>>> a = [1, 2, 3]
>>> del a[1]
>>> a
[1, 3]
```

List

- **length**

```
a = [1, 2, 3]
length = len(a)
print(length)
```



3

List

- append(요소)

```
a = [1, 2, 3]
a.append(4)
print(a)

a.append([5, 6])
print(a)
```

[1, 2, 3, 4]
[1, 2, 3, 4, [5, 6]]

- insert(인덱스, 값)

```
a = [1, 2, 3]
a.insert(0, 4)
print(a)

a.insert(3, 5)
print(a)
```

[4, 1, 2, 3]
[4, 1, 2, 5, 3]

- extend(리스트)

```
a = [1, 2, 3]
a.extend([4, 28])
print(a)

a.extend([20, 23])
print(a)
```

[1, 2, 3, 4, 28]
[1, 2, 3, 20, 23]

List

- **sort()**

```
a = [1, 5, 3, 2]
a.sort()
print(a)

a.sort(reverse=True)
print(a)
```

[1, 2, 3, 5]
[5, 3, 2, 1]

- **reverse()**

```
a = [1, 5, 3, 2]
a.reverse()
print(a)
```

[2, 3, 5, 1]

List

- **remove(값)**

```
a = ['b', 'a', 'n', 'a', 'n', 'a']
a.remove('b')
print(a)

a.remove('n')
print(a)
```

[‘a’, ‘n’, ‘a’, ‘n’, ‘a’]
[‘a’, ‘a’, ‘n’, ‘a’]

- **pop()**

```
a = [1, 2, 3]
a.pop()
print(a)
```

[1, 2]

Comprehension

1. Using loop

[표현식 for 변수 in 범위]

```
[i for i in range(10)]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

[i*3 for i in range(10)]
[0, 3, 6, 9, 12, 15, 18, 21, 24, 27]
```

Comprehension

2. Using Conditional

[표현식 for 변수 in 범위 if문]

```
[i for i in range(10)]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

[i for i in range(10) if i%2 == 0]
[0, 2, 4, 6, 8]
```

Comprehension

2. Using Conditional

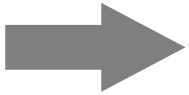
```
[i for i in range(10)]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

['even' if i%2 == 0 else 'odd' i for i in range(10)]
['even', 'odd', 'even', 'odd', 'even', 'odd', 'even', 'odd', 'even', 'odd', 'even', 'odd']
```

Tuple

```
tuple = ( 'abcd', 786 , 2.23, 'john', 70.2  )
tinytuple = (123, 'john')

print(tuple)
print(tuple[0])
print(tuple[1:3])
print(tuple[2:])
print(tinytuple * 2)
print(tuple + tinytuple)
```



```
('abcd', 786, 2.23, 'john', 70.2)
abcd
(786, 2.23)
(2.23, 'john', 70.2)
(123, 'john', 123, 'john')
('abcd', 786, 2.23, 'john', 70.2, 123, 'john')
```

Dictionary