

# 얕은 복사와 깊은 복사

🕒 작성일시	@2022년 7월 25일 오후 11:10
☰ 내용	문법 파이썬
▼ 주차	2주차
📎 자료	

## 할당(Assignment)

- 대입 연산자(=)
  - 리스트 복사 확인하기
  - 대입 연산자(=)를 통한 복사는 해당 객체에 대한 **객체 참조를 복사**

```
original_list = [1, 2, 3]
copy_list = original_list
print(original_list, copy_list) # [1, 2, 3] [1, 2, 3]

copy_list[0] = 'hello'
print(original_list, copy_list) # ['hello', 2, 3] ['hello', 2, 3]
```

## 얕은 복사(Shallow copy)

- Slice 연산자를 활용하여 같은 원소를 가진 리스트지만 연산된 결과를 복사 (다른 주소)
  - 단, **1차원에서만** 가능!

```
a = [1, 2, 3]
b = a[:]

print(a, b) #[1, 2, 3] [1, 2, 3]

b[0] = 5
print(a, b) #[1, 2, 3] [5, 2, 3]
```

- 2차원에서는 적용x

```
a = [1, 2, ['a', 'b']]
b = a[:]
```

```
print(a, b) # [1, 2, ['a', 'b']], [1, 2, ['a', 'b']]

b[2][0] = 0

print(a, b) # [1, 2, [0, 'b']], [1, 2, [0, 'b']]
```

## 깊은 복사(Deep copy)

```
import copy

a = [1, 2, ['a', 'b']]
b = copy.deepcopy(a)

print(a, b) # [1, 2, ['a', 'b']], [1, 2, ['a', 'b']]

b[2][0] = 0

print(a, b) # [1, 2, ['a', 'b']], [1, 2, [0, 'b']]
```