

아래 프로그램을 실행하면 어떤 결과값이 나올까요?

```
x = 5
y = x
y = 3
print(x)
print(y)
```

```
5
3
```

첫 번째 줄에서는 정수 5가 변수 `x`에 지정됩니다. 그리고 두 번째 줄에서는 `x`의 값이 현재 5이기 때문에, 정수 5가 변수 `y`에 지정됩니다. 세 번째 줄에서는 `y`의 값이 3으로 바뀝니다. 따라서 이를 실행하면 5와 3이 차례로 출력됩니다.

엘리어싱 (Aliasing)

그럼 이 프로그램을 실행하면 어떠한 결과값이 나올까요?

```
x = [2, 3, 5, 7, 11]
y = x
y[2] = 4
print(x)
print(y)
```

```
[2, 3, 4, 7, 11]
[2, 3, 4, 7, 11]
```

앞서 본 변수의 예처럼 `x`는 `[2, 3, 5, 7, 11]`이 나오고, `y`는 `[2, 3, 4, 7, 11]`이 나올줄 알았는데, 결과가 예상과 다르죠? `y`의 원소만 바꾸었을 뿐인데, 왜 `x`와 `y`가 똑같이 `[2, 3, 4, 7, 11]`로 변했을까요?

엘리어싱(Aliasing)이라는 개념 때문입니다. 코드의 첫 번째 줄에서는 `[2, 3, 5, 7, 11]` 리스트에 `x`라는 이름표를 달아줍니다. 두 번째 줄에서 `y = x`라는 명령에 의해, 그 같은 리스트에 `y`라는 이름표를 달아줍니다. 세 번째 줄에서 리스트 `y`의 인덱스 2의 값을 4로 바꿔주면, 리스트는 `[2, 3, 4, 7, 11]`이 됩니다.

그런데 바뀐 `[2, 3, 4, 7, 11]`이라는 리스트가 `y`라는 이름표 뿐만 아니라, `x`라는 이름표도 갖고 있었죠? 그래서 `x`와 `y`를 출력하라는 명령이 있을 때, 모두 같은 `[2, 3, 4, 7, 11]`이 출력되는 것입니다. 두 변수가 서로 이름은 다르지만, 사실 같은 메모리 주소에 있는 값을 가리키고 있기 때문에 동일한 리스트가 출력된거죠. `y`는 `x`의 **Alias(가명)**라고 얘기할 수 있습니다.

리스트 복사

하지만 간혹 엘리머싱을 이용하지 않고 리스트의 원본을 그대로 둔 채, 리스트의 복사본만 바꾸고 싶을 때가 있습니다. 즉 **x** 라는 원본의 리스트는 그대로 둔 채, **y** 리스트의 값만 바꾸고 싶은 경우. 이럴 때는 **list** 함수를 사용하면 됩니다. **list** 함수는 원본을 복사해서 새로운 공간에 붙여넣고, 그 새로운 공간의 리스트를 리턴시켜줍니다.

```
x = [2, 3, 5, 7, 11]
y = list(x)
y[2] = 4
print(x)
print(y)
```


```
[2, 3, 5, 7, 11]
[2, 3, 4, 7, 11]
```


첫 번째 줄에서 **[2, 3, 5, 7, 11]** 을 가진 리스트에 **x** 라는 이름표가 달립니다. 두 번째 줄에서 **y = list(x)** 라는 명령에 의해, **x** 리스트의 복사본이 새로운 공간에 만들어지고 그 복사본에 **y** 라는 이름표가 달립니다. 즉 **x** 와 **y** 의 리스트는 서로 다른 메모리 주소에 저장되어 있는거죠.

따라서 세 번째 줄에서 **y** 리스트의 인덱스 **2** 의 값을 바꾸는 행위는, **x** 리스트에 아무런 영향을 미치지 않습니다. 이로써 **x** 와 **y** 를 출력하라는 명령이 있을 때, 각각 **[2, 3, 5, 7, 11]** 과 **[2, 3, 4, 7, 11]** 이 출력되는 것입니다.



수업을 완료하셨으면 체크해주세요.

 수강생 Q&A 보기

 [/questions?](/questions?assignment_id=121&sort_by=popular) 질문하기

[assignment_id=121&sort_by=popular\)](/questions/new?assignment_id=121&sort_by=popular)

[assignment_id=121&op1=%ED%94%84%EB%A1%9C%EA%B7%](#)

< 이전 강의
Aliasinq (/assignments/120)

다음 강의 > (/assignments/122)
리스트 뒤집기