

## 8. 변수

### 8.1. 변수를 만드는 방법

#### 변수?

- a=2, b=3 에서 a, b 를 변수라고 한다
- 파이썬에서 사용하는 변수 = 객체

#### 객체란?

- 우리가 지금까지 보아온 자료형의 데이터 (값) 을 의미한다
- a = [1, 2, 3] 선언 시
  - [1, 2, 3] 값을 가지는 리스트 데이터 (객체) 가 자동으로 메모리에 생성되고 변수 a 는 [1, 2, 3] 리스트가 저장된 메모리의 주소를 가리키게 된다.

### 8.2. 메모리 주소 확인 (id)

id 함수로 확인 가능

```
a = 123 print(id(a))
```

### 8.3. 리스트를 복사하고자 할 때

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

b 변수에 a 변수를 대입하면 b는 a와 동일하다 다만 [1, 2, 3] 이라는 리스트 객체를 참조하는 변수가 a 변수 1개에서 b 변수가 추가되어 2개로 늘어났을 뿐

id 함수를 사용하면 사실을 알 수 있다

```
print(id(a))
```

결과 : 4303029896

```
print(id(b))
```

결과 : 4303029896

그렇다면 b변수를 생성할 때 a변수의 값을 가져오면서 a와는 다른 주소를 가리키도록 만들 수는 없을까?

## 1. [:]이용하기 (리스트 전체를 가리킴)

```
a = [1, 2, 3] #a리스트 선언
b = a[:] #b라는 변수에 a리스트 값을 다 넣는다
a[1] = 4 #a 리스트 인덱스 1에 4추가
print(a) #결과 [1, 4, 3]
print(b) #결과 [1, 2, 3]
```

a리스트 값을 바꾸더라도 b리스트 에는 아무런 영향이 없다

## 2. copy 모듈 이용하기

```
from copy import copy
a = [1, 2, 3]
b = copy(a)
```

서로 다른 객체를 가르키고 있는지 확인 할 수 있다

b is a 결과는 False

---

## 8.4. 변수를 만드는 여러가지 방법

### 1. 튜플

```
a, b = ('python', 'life')
(a, b) = ['python', 'life']
```

### 2. 리스트

```
[a, b] = ['python', 'life']
```

### 3. 여러 개의 변수에 같은 값 대입하기

```
a = b = 'python'
```

\

#### 4. 위 방법을 사용하여 두 변수의 값 바꾸기

```
a = 3
b = 5
a, b = b, a
print(a) #5
print(b) #3
```