



<https://dfedorov.spb.ru>

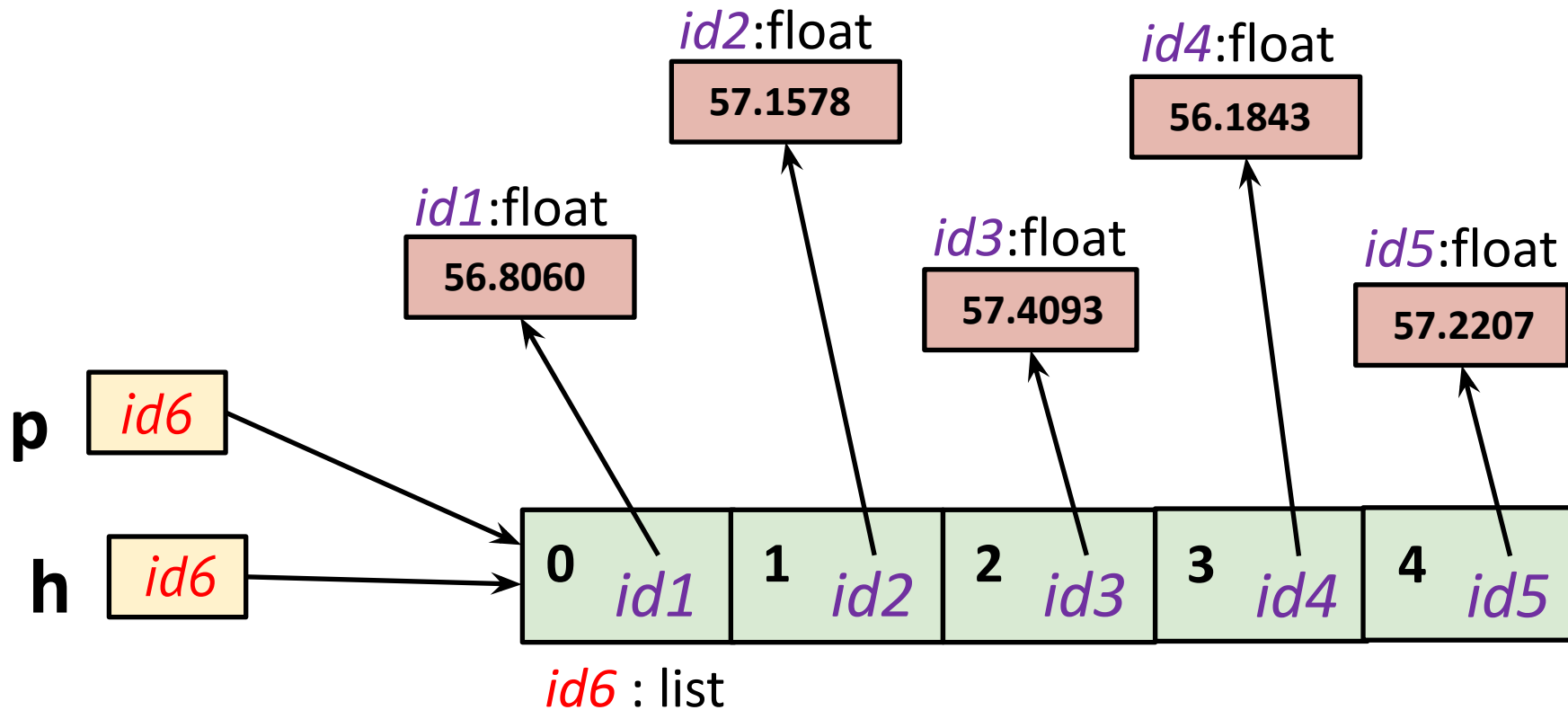
# Псевдонимы

**Псевдонимы** – альтернативные имена чего-либо.

В Python две переменные называются псевдонимами, когда они содержат **одинаковые адреса памяти**.

```
>>> h
['bonjour', 7, 'hola', -1.0, 'привіт']
>>> p = h # содержат указатель на один и тот же список
>>> p
['bonjour', 7, 'hola', -1.0, 'привіт']
>>> p[0] = 1 # модифицируем одну из переменных
>>> h      # изменилась другая переменная!
[1, 7, 'hola', -1.0, 'привіт']
>>> p
[1, 7, 'hola', -1.0, 'привіт']
>>>
```

Например, со строками такая манипуляция не пройдет, т.к. они не изменяются

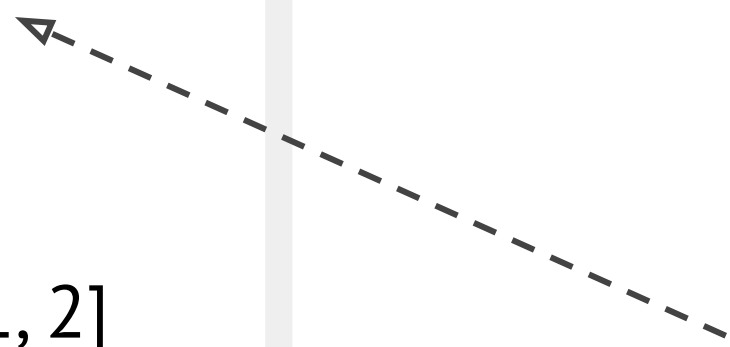


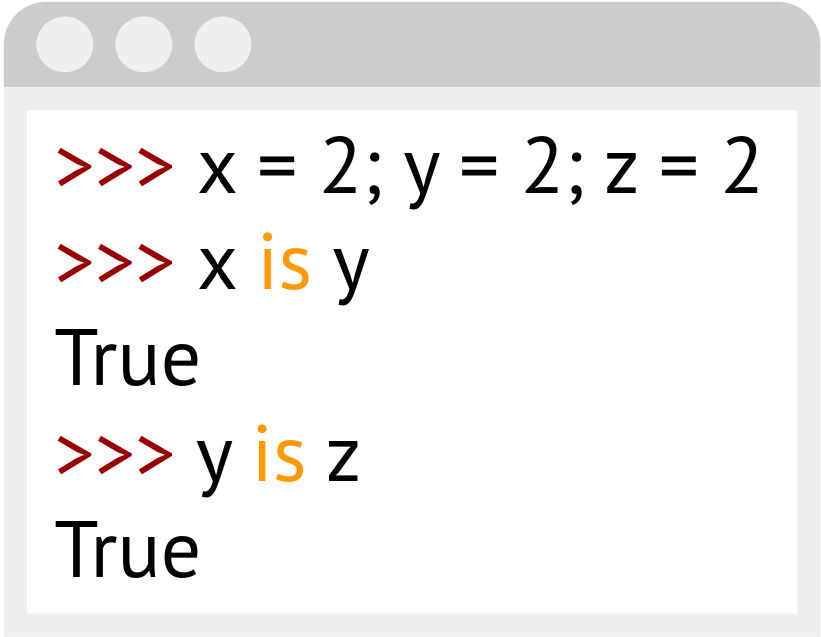
# Функция `id`

```
>>> x = y = [1, 2]
>>> x is y
True

>>> x = [1, 2]
>>> y = [1, 2]
>>> x is y
False
```

Проверяем,  
ссылаются ли  
переменные `x` и `y` на  
один и тот же объект





```
>>> x = 2; y = 2; z = 2
```

```
>>> x is y
```

```
True
```

```
>>> y is z
```

```
True
```

В целях повышения  
эффективности  
производится  
кэширование малых  
целых чисел и  
небольших строк

L = [3, 'hello', 7, 4, 'привет', 4, 3, -1]

```
>>> L[:3]
```

```
>>> L[:]
```

```
>>> L[::2]
```

```
>>> L[::-1]
```

```
>>> L[:-1]
```

```
>>> L[-1:]
```

```
>>> a = [-1, 1, 66.25, 333, 333, 1234.5]
```

```
>>> del a[0]
```

```
>>> a
```

```
[1, 66.25, 333, 333, 1234.5]
```

```
>>> del a[2:4]
```

```
>>> a
```

```
[1, 66.25, 1234.5]
```

```
>>> del a[:]
```

```
>>> a
```

```
[]
```

```
>>> lst = [4, 5, 6]
```

```
>>> lst[2:6]
```

```
[6]
```

```
>>> lst[5:6]
```

```
[]
```

```
>>>
```



```
>>> h = ['bonjour',7,'hola',-1.0,'привіт']
```

```
>>> h
```

```
['bonjour', 7, 'hola', -1.0, 'привіт']
```

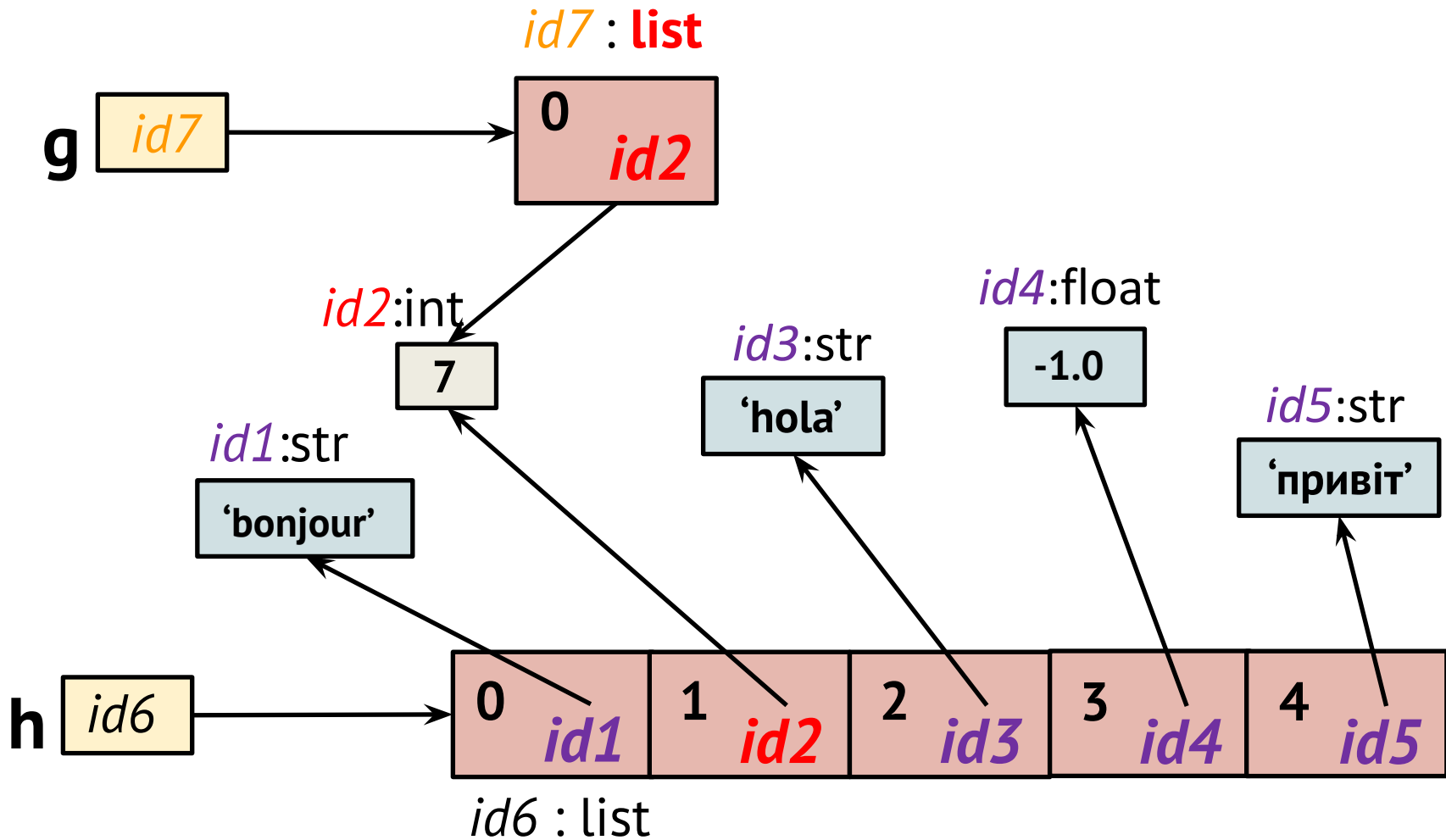
```
>>> g = h[1:2]
```

```
>>> g
```

```
[7]
```

До 2-ого элемента  
**НЕ включительно**

С 1-ого элемента  
включительно



## Поверхностное (shallow copy) копирование

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

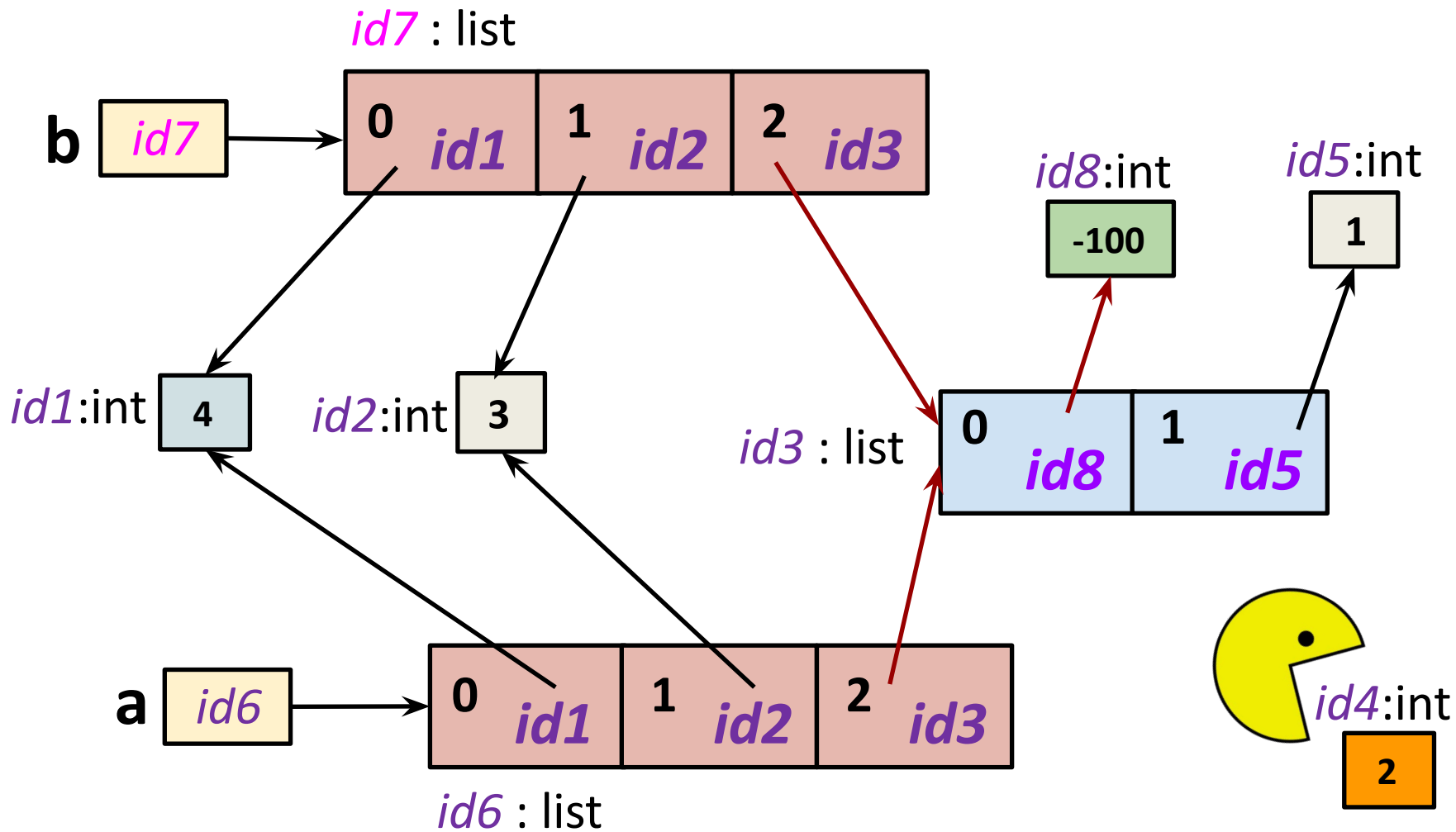
```
>>> b = a[:]
```

```
>>> b is a
```

```
False
```

```
>>> b[2][0] = -100
```

```
>>> a
```



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

```
>>> b = a[:]
```

```
>>> b is a
```

```
False
```

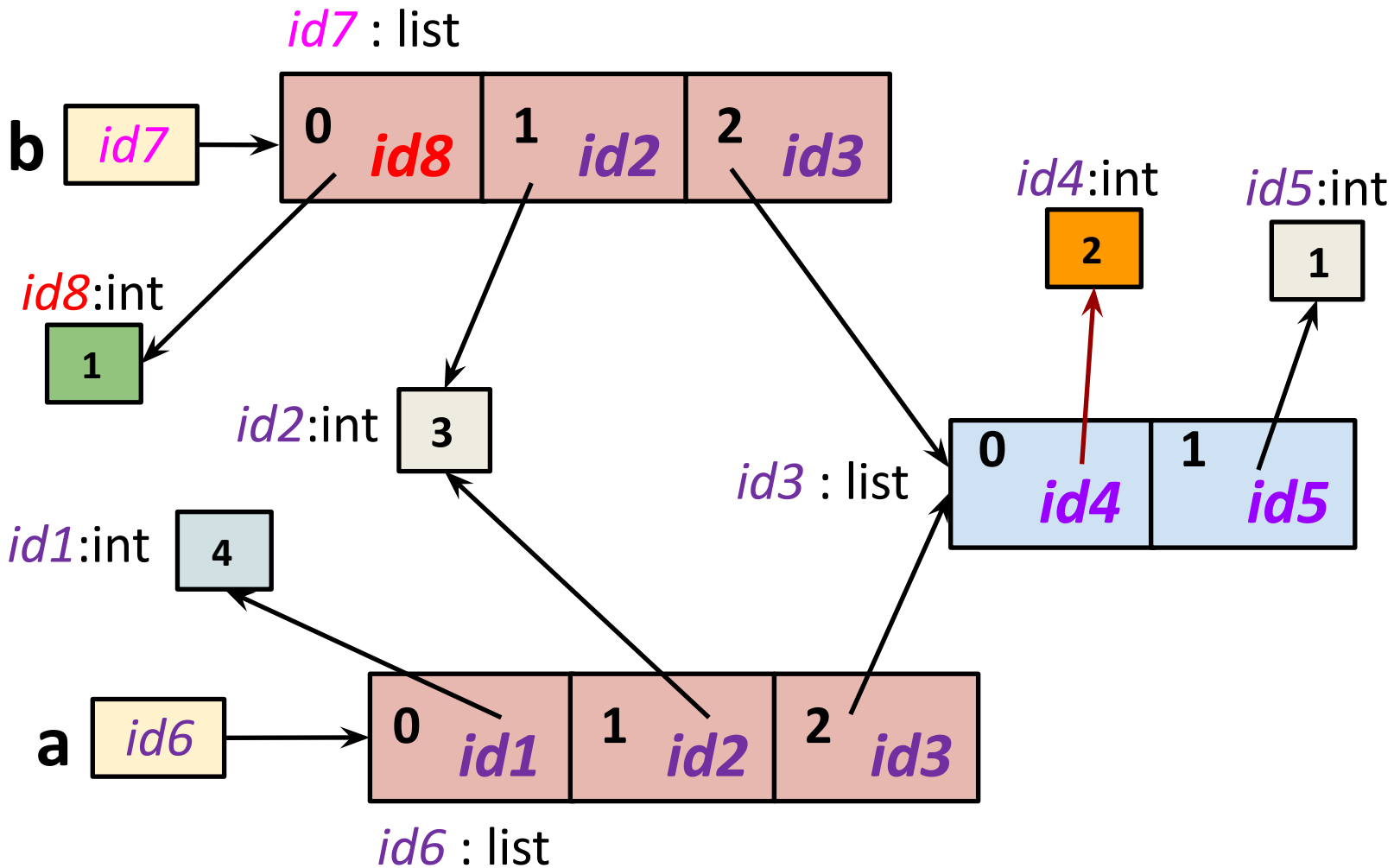
```
>>> b[0] = 1
```

```
>>> a
```

```
[4, 3, [2, 1]]
```

```
>>>
```

Почему в списке **a**  
изменения не  
произошли?



# Глубокое копирование

```
>>> import copy
>>> a = [4, 3, [2, 1]]
>>> b = copy.deepcopy(a) # .copy
>>> b[2][0] = -100
>>> a
[4, 3, [2, 1]] # список a не изменился
```

# Вложенные списки

```
>>> lst = [['A', 1], ['B', 2], ['C', 3]]
```

```
>>> lst
```

```
 [['A', 1], ['B', 2], ['C', 3]]
```

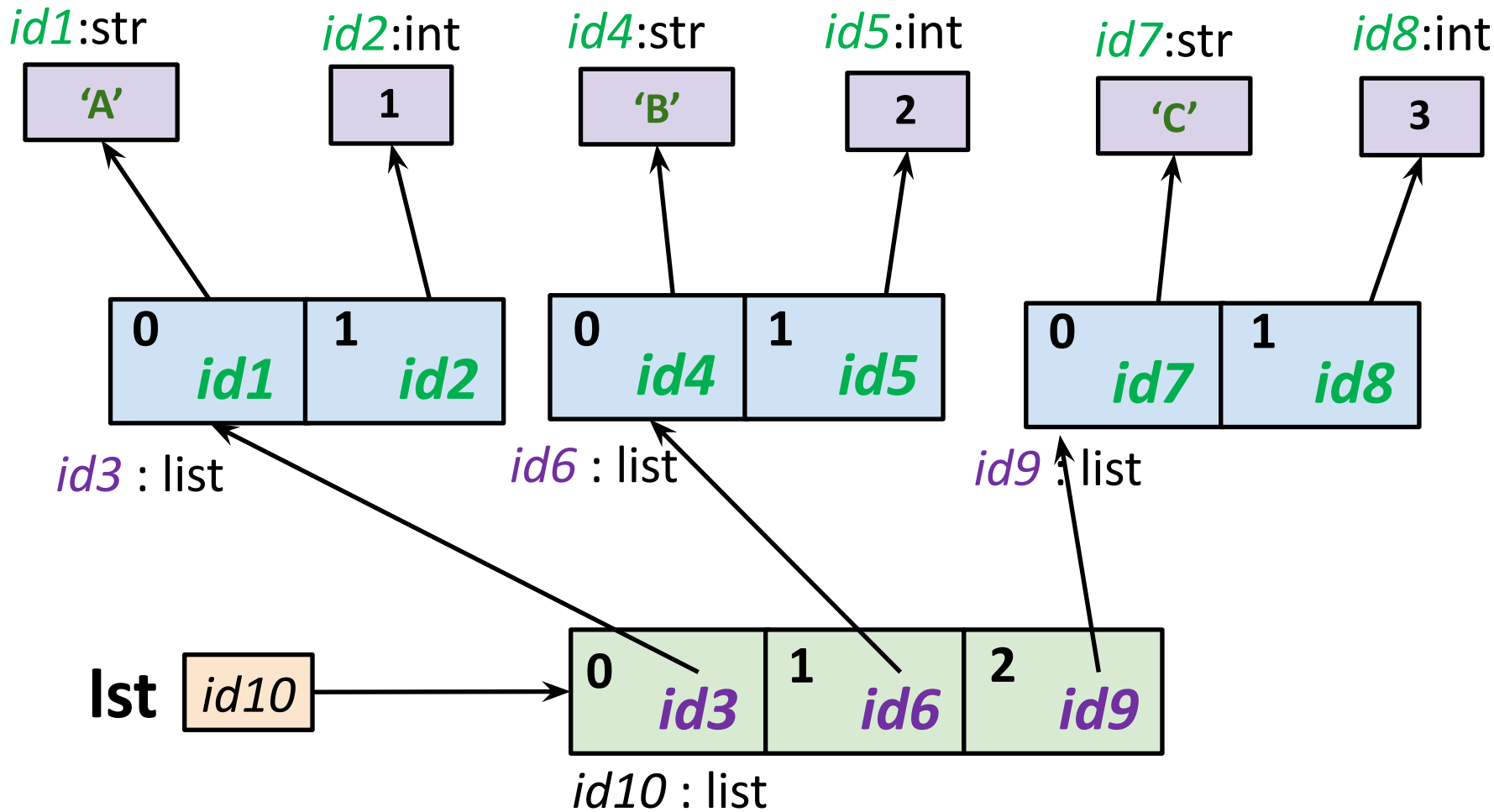
```
>>> lst[0]
```

```
 ['A', 1]
```

```
>>> lst[0][1]
```

```
 1
```





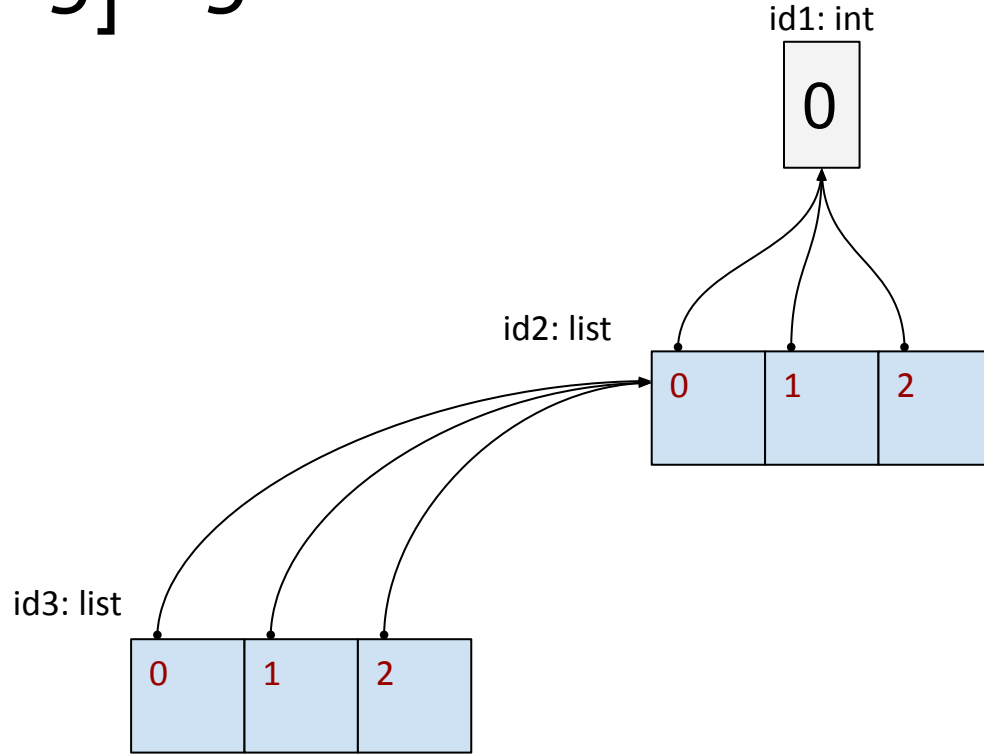
```
>>> xs = [[0] * 3] * 3
```

```
>>> xs[0][0] = 1
```

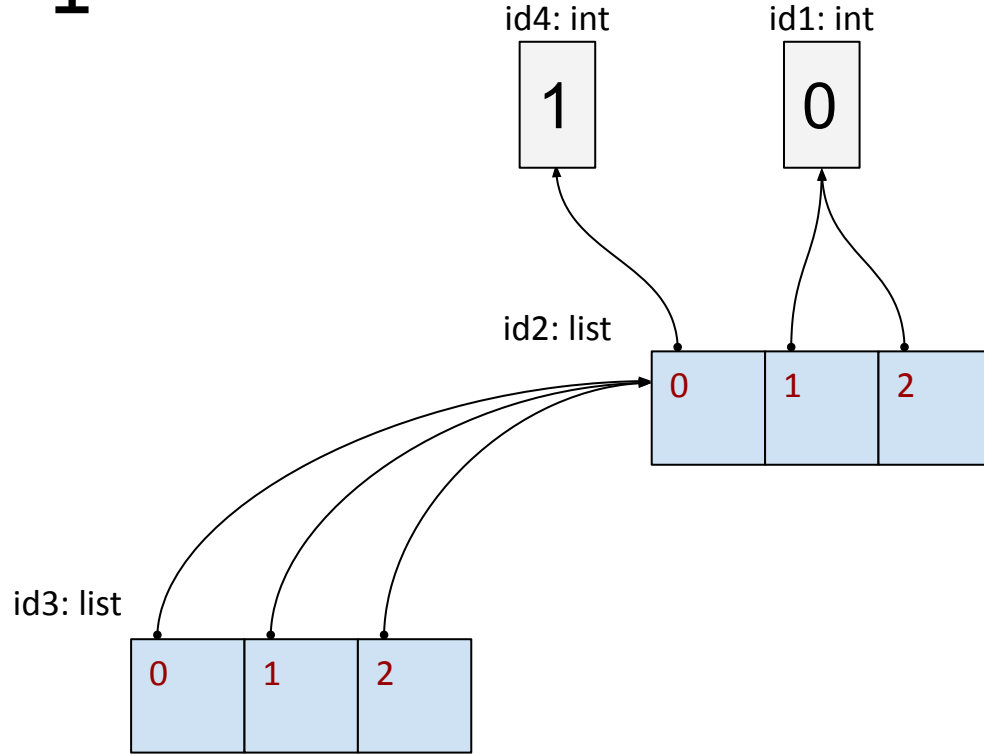
```
>>> xs
```

```
?????
```

>>> xs = [[0] \* 3] \* 3



>>> xs[0][0] = 1





<https://dfedorov.spb.ru>