

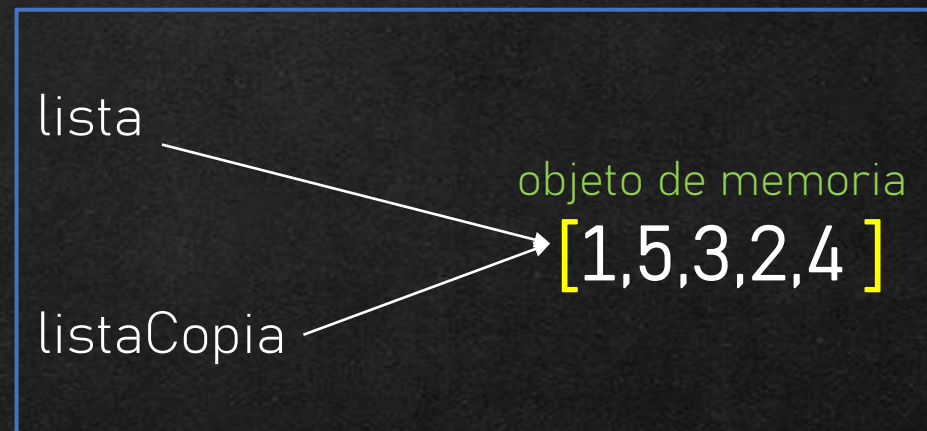
Variables de tipos estructurados

En Python las variables de tipos estructurados como **listas**, **tuplas**, **conjuntos** y **diccionarios**, así como **objetos personalizados**, tienen 2 comportamientos muy particulares que son los siguientes:

- 1.-) Dichas variables NO almacenan el dato estructurado como tal sino que almacenan una referencia al objeto de memoria que contiene el dato estructurado.
- 2.-) Cuando una variable de tipo estructurado se asigna a una nueva variable, la segunda variable recibe la referencia del objeto al cual apunta la primera.

Veámoslo a través de un gráfico:

```
lista = [1,5,3,2,4]  
listaCopia = lista
```



Variables de tipos estructurados

IMPORTANTE: En Python CASI TODO es un OBJETO.

```
lista = [1,5,3,2,4]
```

```
listaCopia = lista
```

objeto

lista

objeto

listaCopia

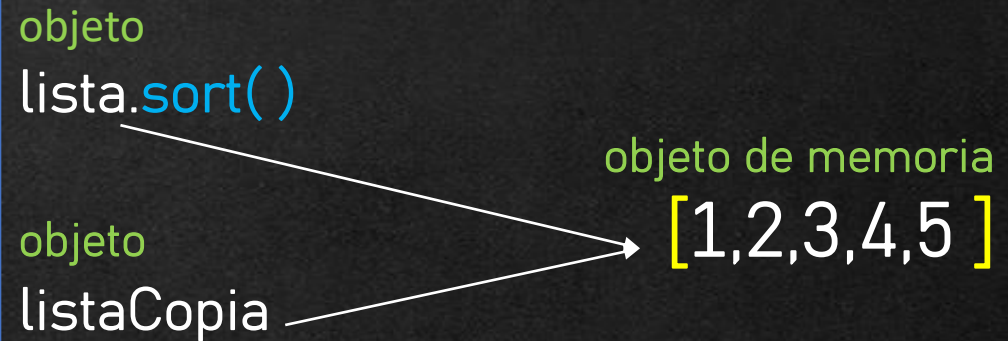
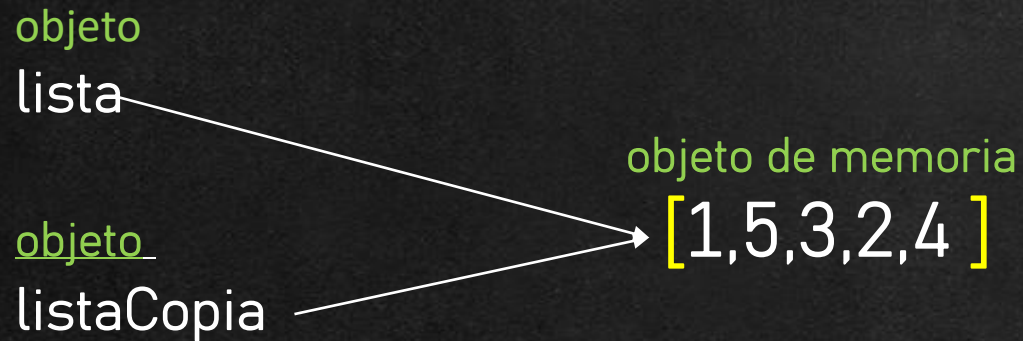
objeto de memoria

[1,5,3,2,4]



Variables de tipos estructurados

```
lista = [1,2,3,4,5]  
listaCopia = lista
```



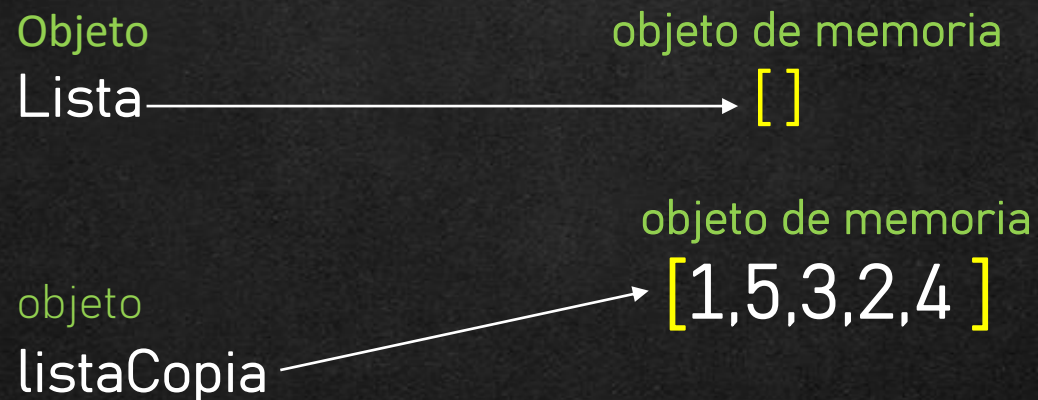
Variables de tipos estructurados

IMPORTANTE: En Python CASI TODO es un OBJETO.

```
lista = [1,5,3,2,4]
```

```
listaCopia = lista
```

```
lista = []
```





Variables de tipos estructurados

IMPORTANTE: Lo visto anteriormente no aplica a los tipos de datos primitivos.

int

float

doble

boolean

string

Las variables de estos tipos cuando son asignadas, **NO** pasan ninguna referencia sino que pasan el respectivo valor.



ProfesorenVideo.com

Muchas gracias por su atención