**La vida al interior de las listas**

Ahora queremos mostrarte una característica importante y muy sorprendente de las listas, que las distingue de las variables ordinarias.

Queremos que lo memorices, ya que puede afectar tus programas futuros y causar graves problemas si se olvida o se pasa por alto.

Echa un vistazo al fragmento en el editor.

El programa:

* Crea una lista de un elemento llamada list\_1.
* La asigna a una nueva lista llamada list\_2.
* Cambia el único elemento de list\_1.
* Imprime la list\_2.

La parte sorprendente es el hecho de que el programa mostrará como resultado: [2], no [1], que parece ser la solución obvia.

Las listas (y muchas otras entidades complejas de Python) se almacenan de diferentes maneras que las variables ordinarias (escalares).

Se podría decir que:

* El nombre de una variable ordinaria es el **nombre de su contenido**.
* El nombre de una lista es el nombre de una ubicación de memoria **donde se almacena la lista**.

Lee estas dos líneas una vez más, la diferencia es esencial para comprender de que vamos a hablar a continuación.

La asignación: list\_2 = list\_1 copia el nombre del arreglo no su contenido. En efecto, los dos nombres (list\_1 y list\_2) identifican la misma ubicación en la memoria de la computadora. Modificar uno de ellos afecta al otro, y viceversa.

¿Cómo te las arreglas con eso?

list\_1 = [1]

list\_2 = list\_1

list\_1[0] = 2

print(list\_2)

[2]