### Las posiciones de índice empiezan en 0, no en 1

Python considera que el primer elemento de una lista está en la posición 0, no en la 1. Esto es así en la mayoría de los lenguajes de programación, y la razón tiene que ver con cómo se implementan las operaciones de lista en un nivel inferior. Si recibes resultados inesperados, pregúntate si estás cometiendo un error simple pero común de uno en uno.

El segundo elemento de una lista tiene un índice de 1. Utilizando este sistema de conteo, puedes obtener cualquier elemento que desees de una lista restando uno a su posición en la lista. Por ejemplo, para acceder al cuarto elemento de una lista, solicitas el elemento situado en el índice 3.

A continuación se piden las bicicletas en los índices 1 y 3:

bicycles = ['trek', 'cannondale', 'redline', 'specialized']  
print(bicycles[1])  
print(bicycles[3])

Este código devuelve la segunda y la cuarta bicicleta de la lista:

cannondale  
specialized

Python tiene una sintaxis especial para acceder al último elemento de una lista. Si pides el elemento en el índice -1, Python siempre devuelve el último elemento de la lista:

bicycles = ['trek', 'cannondale', 'redline', 'specialized']  
print(bicycles[-1])

Este código devuelve el valor 'specialized'. Esta sintaxis es bastante útil, porque a menudo querrás acceder a los últimos elementos de una lista sin saber exactamente su longitud. Esta convención se extiende también a otros valores de índice negativos. El índice -2 devuelve el segundo elemento desde el final de la lista, el índice -3 devuelve el tercer elemento desde el final, y así sucesivamente.

[anterior](c03_3.html)[Subtema 4 de 23: (Ver todo)](c03.html)[siguiente](c03_5.html)