### Comprensión de listas

El método descrito anteriormente para generar la lista squares consistía en utilizar tres o cuatro líneas de código. Un *list comprehension* te permite generar esta misma lista en una sola línea de código. Una comprensión de lista combina el bucle for y la creación de nuevos elementos en una sola línea, y añade automáticamente cada nuevo elemento. Las comprensiones de lista no siempre se presentan a los principiantes, pero las he incluido aquí porque lo más probable es que las veas en cuanto empieces a mirar el código de otras personas.

El siguiente ejemplo construye la misma lista de números cuadrados que has visto antes, pero utiliza una comprensión de lista:

**squares.py**

squares = [value\*\*2 for value in range(1, 11)]  
print(squares)

Para utilizar esta sintaxis, empieza con un nombre descriptivo para la lista, como squares. A continuación, abre un conjunto de corchetes y define la expresión para los valores que quieras almacenar en la nueva lista. En este ejemplo, la expresión es value\*\*2, que eleva el valor a la segunda potencia. A continuación, escribe un bucle for para generar los números que quieras introducir en la expresión, y cierra los corchetes. El bucle for de este ejemplo es for value in range(1, 11), que introduce los valores del 1 al 10 en la expresión value\*\*2. Observa que no se utilizan dos puntos al final de la expresión for.

El resultado es la misma lista de números cuadrados que has visto antes:

[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

Escribir tus propias comprensiones de listas requiere práctica, pero te merecerá la pena una vez que te sientas cómodo creando listas normales. Cuando estés escribiendo tres o cuatro líneas de código para generar listas y empiece a parecerte repetitivo, plantéate escribir tus propias comprensiones de listas.

[anterior](c04_17.html)[Subtema 18 de 41: (Ver todo)](c04.html)[siguiente](c04_19.html)