### Mezclar argumentos posicionales y arbitrarios

Si quieres que una función acepte varios tipos diferentes de argumentos, el parámetro que acepta un número arbitrario de argumentos debe colocarse en último lugar en la definición de la función. Python combina primero los argumentos posicionales y los de palabra clave, y luego reúne los argumentos restantes en el último parámetro.

Por ejemplo, si la función necesita recoger un tamaño para la pizza, ese parámetro debe ir antes que el parámetro \*toppings:

def make\_pizza(size, \*toppings):  
 """Summarize the pizza we are about to make."""  
 print(f"\nMaking a {size}-inch pizza with the following toppings:")  
 for topping in toppings:  
 print(f"- {topping}")  
  
make\_pizza(16, 'pepperoni')  
make\_pizza(12, 'mushrooms', 'green peppers', 'extra cheese')

En la definición de la función, Python asigna el primer valor que recibe al parámetro size. Todos los demás valores que vengan después se almacenan en la tupla toppings. Las llamadas a la función incluyen primero un argumento para el tamaño, seguido de tantos ingredientes como sean necesarios.

Ahora cada pizza tiene un tamaño y un número de ingredientes, y cada información se imprime en el lugar adecuado, mostrando primero el tamaño y después los ingredientes:

Making a 16-inch pizza with the following toppings:  
- pepperoni  
  
Making a 12-inch pizza with the following toppings:  
- mushrooms  
- green peppers  
- extra cheese

[anterior](c08_26.html)[Subtema 27 de 40: (Ver todo)](c08.html)[siguiente](c08_28.html)