### Un diccionario dentro de otro diccionario

Puedes anidar un diccionario dentro de otro diccionario, pero tu código puede complicarse rápidamente cuando lo hagas. Por ejemplo, si tienes varios usuarios para un sitio web, cada uno con un nombre de usuario único, puedes utilizar los nombres de usuario como claves de un diccionario. A continuación, puedes almacenar información sobre cada usuario utilizando un diccionario como valor asociado a su nombre de usuario. En el siguiente listado, almacenamos tres datos sobre cada usuario: su nombre, apellidos y ubicación. Accederemos a esta información recorriendo los nombres de usuario y el diccionario de información asociado a cada nombre de usuario:

**many\_users.py**

users = {  
 'aeinstein': {  
 'first': 'albert',  
 'last': 'einstein',  
 'location': 'princeton',  
 },  
  
 'mcurie': {  
 'first': 'marie',  
 'last': 'curie',  
 'location': 'paris',  
 },  
  
 }  
  
❶ for username, user\_info in users.items():  
❷ print(f"\nUsername: {username}")  
❸ full\_name = f"{user\_info['first']} {user\_info['last']}"  
 location = user\_info['location']  
  
❹ print(f"\tFull name: {full\_name.title()}")  
 print(f"\tLocation: {location.title()}")

Primero definimos un diccionario llamado users con dos claves: una para cada uno de los nombres de usuario 'aeinstein' y 'mcurie'. El valor asociado a cada clave es un diccionario que incluye el nombre, apellidos y ubicación de cada usuario. A continuación, recorremos en bucle el diccionario users ❶. Python asigna cada clave a la variable username, y el diccionario asociado a cada nombre de usuario se asigna a la variable user\_info. Una vez dentro del bucle principal del diccionario, imprimimos el nombre de usuario ❷.

A continuación, empezamos a acceder al diccionario interno ❸. La variable user\_info, que contiene el diccionario de información del usuario, tiene tres claves: 'first', 'last', y 'location'. Utilizamos cada clave para generar un nombre completo y una ubicación de cada persona con un formato ordenado, y luego imprimimos un resumen de lo que sabemos de cada usuario ❹:

Username: aeinstein  
 Full name: Albert Einstein  
 Location: Princeton  
  
Username: mcurie  
 Full name: Marie Curie  
 Location: Paris

Observa que la estructura del diccionario de cada usuario es idéntica. Aunque Python no lo requiere, esta estructura facilita el trabajo con diccionarios anidados. Si el diccionario de cada usuario tuviera claves diferentes, el código dentro del bucle for sería más complicado.

[anterior](c06_25.html)[Subtema 26 de 28: (Ver todo)](c06.html)[siguiente](c06_27.html)