### Trabajar con varios archivos

Añadamos más libros para analizar, pero antes traslademos el grueso de este programa a una función llamada count\_words(). Esto facilitará la ejecución del análisis para varios libros:

**word\_count.py**

from pathlib import Path  
  
def count\_words(path):  
❶ """Count the approximate number of words in a file."""  
 try:  
 contents = path.read\_text(encoding='utf-8')  
 except FileNotFoundError:  
 print(f"Sorry, the file {path} does not exist.")  
 else:  
 # Count the approximate number of words in the file:  
 words = contents.split()  
 num\_words = len(words)  
 print(f"The file {path} has about {num\_words} words.")  
  
path = Path('alice.txt')  
count\_words(path)

La mayor parte de este código no ha cambiado. Sólo ha sido sangrado y trasladado al cuerpo de count\_words(). Es una buena costumbre mantener los comentarios actualizados cuando modificas un programa, así que el comentario también se ha cambiado a un docstring y se ha reformulado ligeramente ❶.

Ahora podemos escribir un bucle corto para contar las palabras de cualquier texto que queramos analizar. Para ello, almacenamos los nombres de los archivos que queremos analizar en una lista y, a continuación, llamamos a count\_words() para cada archivo de la lista. Intentaremos contar las palabras de *Alice in Wonderland*, *Siddhartha*, *Moby Dick*, y *Little Women*, todos ellos de dominio público. He dejado intencionadamente *siddhartha.txt* fuera del directorio que contiene *word\_count.py*, para que podamos ver lo bien que nuestro programa gestiona un archivo que falta:

from pathlib import Path  
  
def count\_words(filename):  
 --snip--  
  
filenames = ['alice.txt', 'siddhartha.txt', 'moby\_dick.txt',   
 'little\_women.txt']  
for filename in filenames:  
❶ path = Path(filename)  
 count\_words(path)

Los nombres de los archivos se almacenan como simples cadenas. A continuación, cada cadena se convierte en un objeto Path ❶, antes de la llamada a count\_words(). El archivo *siddhartha.txt* que falta no tiene ningún efecto sobre el resto de la ejecución del programa:

The file alice.txt has about 29594 words.  
Sorry, the file siddhartha.txt does not exist.  
The file moby\_dick.txt has about 215864 words.  
The file little\_women.txt has about 189142 words.

Utilizar el bloque try-except en este ejemplo proporciona dos ventajas significativas. Evitamos que nuestros usuarios vean un rastreo, y dejamos que el programa siga analizando los textos que es capaz de encontrar. Si no atrapamos el FileNotFoundError que plantea *siddhartha.txt*, el usuario vería un rastreo completo, y el programa dejaría de ejecutarse tras intentar analizar *Siddhartha*. Nunca analizaría *Moby Dick* ni *Little Women*.

[anterior](c10_26.html)[Subtema 27 de 37: (Ver todo)](c10.html)[siguiente](c10_28.html)