### Analizar texto

Puedes analizar archivos de texto que contengan libros enteros. Muchas obras clásicas de la literatura están disponibles como simples archivos de texto porque son de dominio público. Los textos utilizados en esta sección proceden del Proyecto Gutenberg[(https://gutenberg.org)](https://gutenberg.org). El Proyecto Gutenberg mantiene una colección de obras literarias que están disponibles en el dominio público, y es un gran recurso si te interesa trabajar con textos literarios en tus proyectos de programación.

Extraigamos el texto de *Alice in Wonderland* e intentemos contar el número de palabras del texto. Para ello, utilizaremos el método de cadena split(), que por defecto divide una cadena siempre que encuentra algún espacio en blanco:

from pathlib import Path  
  
path = Path('alice.txt')  
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:  
❶ words = contents.split()  
❷ num\_words = len(words)  
 print(f"The file {path} has about {num\_words} words.")

He movido el archivo *alice.txt* al directorio correcto, así que el bloque try funcionará esta vez. Tomamos la cadena contents, que ahora contiene todo el texto de *Alice in Wonderland* como una cadena larga, y utilizamos split() para producir una lista de todas las palabras del libro ❶. Utilizando len() en esta lista ❷ obtenemos una buena aproximación al número de palabras del texto original. Por último, imprimimos una sentencia que informa de cuántas palabras se han encontrado en el archivo. Este código se coloca en el bloque else porque sólo funciona si el código del bloque try se ejecutó correctamente.

La salida nos dice cuántas palabras hay en *alice.txt*:

The file alice.txt has about 29594 words.

El recuento es un poco alto porque el editor proporciona información adicional en el archivo de texto utilizado aquí, pero es una buena aproximación a la longitud de *Alice in Wonderland*.

[anterior](c10_25.html)[Subtema 26 de 37: (Ver todo)](c10.html)[siguiente](c10_27.html)