

close()		
Descripción	Ejemplo	
Cierra los archivos	<pre>f = open("demofile.txt", "r") print(f.read()) f.close()</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método close() cierra un archivo abierto. Siempre se debe cerrar los archivos, en algunos casos, debido al almacenamiento del búfer, los cambios realizados no se pueden ver hasta cerrar el archivo abierto.	file.close()	Sin parámetros

detach()	
Descripción	Ejemplo
Retorna o Devuelve el flujo de datos original desde el búfer.	N/A

fileno()		
Descripción	Ejemplo	
Retorna o Devuelve un número que representa el flujo, desde la perspectiva del SO. Generalmente se refiere a un descriptor de flujo o 'file descriptor', que es un número que el SO usa para identificar un flujo de datos (archivo, input, output)	<pre>f = open("demofile.txt", "r") print(f.fileno())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método fileno() devuelve el descriptor de archivo de la secuencia, como un número. Se producirá un error si el SO no utiliza un descriptor de archivo.	file.fileno()	Sin parámetros

flush()		
Descripción	Ejemplo	
Vaciar el búfer interno. Puede borrar el búfer al escribir en un archivo.	<pre>f = open("myfile.txt", "a") f.write("Now the file has one more line!") f.flush() f.write("...and another one!")</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método flush() limpia el búfer interno.	file.flush()	Sin parámetros

isatty()		
Descripción	Ejemplo	
Devuelve si el flujo de archivo es interactivo o no. Comprueba si el archivo está conectado a un dispositivo terminal.	<pre>f = open("demofile.txt", "r") print(f.isatty())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método isatty() retorna True si el flujo es interactivo, ejemplo: conectado a un dispositivo terminal. Sirve para que los programas sepan si están interactuando con un usuario real o trabajando con datos redirigidos.	file.isatty()	Sin parámetros

read()		
Descripción	Ejemplo	
Lee el contenido del archivo. Retorna el contenido del archivo.	<pre>f = open("demofile.txt", "r") print(f.read())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método read() devuelve el número especificado de bytes del archivo. El valor predeterminado es -1, lo que significa el archivo completo.	file.read()	size Es opcional.

readable()		
Descripción	Ejemplo	
Retorna si el flujo del archivo se puede leer o no. Comprueba si el archivo es legible.	<pre>f = open("demofile.txt", "r") print(f.readable())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método readable() retorna True si el archivo es legible, False si no.	file.readable()	Sin parámetros

readline()		
Descripción	Ejemplo	
Retorna una línea del archivo. Retorna la primer línea del archivo.	<pre>f = open("demofile.txt", "r") print(f.readline())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método readline() devuelve una línea del archivo. También puede especificar cuántos bytes de la línea desea devolver, utilizando el parámetro de tamaño.	file.readline(size)	size Es opcional. size, es la cantidad de bytes que quiero devolver.

readlines()		
Descripción	Ejemplo	
Retorna una lista de líneas del archivo. Devuelve todas las líneas del archivo, como una lista donde cada línea es un elemento en el objeto lista.	<pre>f = open("demofile.txt", "r") print(f.readlines())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método readlines() devuelve una lista que contiene cada línea del archivo como un elemento de lista. Se utiliza el parámetro de pista para limitar el número de línea devueltas. Si el número total de bytes supera el número especificado, no se devolverán más líneas.	file.readlines(hint)	hint Es opcional. Si el número de bytes devueltos excede el valor indicado en hint, no se devolverán más líneas.

seek()		
Descripción	Ejemplo	
Cambia la posición del archivo. Cambia la posición actual del archivo a 'X' y devuelve el resto de la línea.	<pre>f = open("demofile.txt", "r") f.seek(4) print(f.readline())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método seek() establece la posición actual del archivo en una secuencia de archivos. También devuelve la nueva posición.	file.seek(offset)	offset Es requerido, un número que representa la posición donde se establecerá la posición actual del flujo de archivo.

seekable()		
Descripción	Ejemplo	
Retorna si el archivo permite cambiar la posición del flujo. Comprueba si el archivo se puede buscar.	<pre>f = open("demofile.txt", "r") print(f.seekable())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método seekable() devuelve True si el archivo se puede buscar, False si no. Se puede buscar un archivo si permite el acceso al flujo del archivo, como el método seek().	file.seekable()	Sin parámetros

tell()		
Descripción	Ejemplo	
Retorna la posición actual del archivo. Encuentra la posición actual del archivo.	<pre>f = open("demofile.txt", "r") print(f.tell())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método tell() devuelve la posición actual del archivo en una secuencia de archivos. <b>Consejo:</b> puede cambiar la posición actual del archivo con este método.	file.tell()	Sin parámetros

truncate()		
Descripción	Ejemplo	
Redimensiona el archivo a un tamaño especificado. Abra el archivo con “a” para agregarlo y luego trunque el archivo a 20 bytes.	<pre>f = open("demofile2.txt", "a") f.truncate(20) f.close()  #open and read the file after the truncate: f = open("demofile2.txt", "r") print(f.read())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método truncate() redimensiona el archivo al número de bytes. Si no se especifica el tamaño, se utiliza la posición actual.	file.truncate(size)	size Opcional. El tamaño del archivo (en bytes) después del truncado.

writable()		
Descripción	Ejemplo	
Retorna si el archivo se puede escribir o no. Comprueba si el archivo se puede escribir.	<pre>f = open("demofile.txt", "a") print(f.writable())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método writable() devuelve True si el archivo se puede escribir, False si no. Un archivo se puede escribir si se abre utilizando “a” para agregar o “w” para sobrescribir.	file.writable()	Sin parámetros

write()		
Descripción	Ejemplo	
Escribe la cadena especificada en el archivo. Abra el archivo con “a” para agregarlo y luego agregue algo de texto al archivo.	<pre>f = open("demofile2.txt", "a") f.write("See you soon!") f.close()  #open and read the file after the appending: f = open("demofile2.txt", "r") print(f.read())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
El método write() escribe texto específico en el archivo. El lugar donde se insertará el texto dependerá del modo de archivo y de la posición de la transmisión. “a” el texto se inserta al final del archivo “w” borra el archivo y luego se inserta el texto	file.write(byte)	byte El objeto de texto o byte que se insertará.

writelines()		
Descripción	Ejemplo	
<p>Escribe una lista de strings en el archivo.</p> <p>Abra el archivo con “a” para agregar, luego agregue una lista de textos en el archivo.</p>	<pre>f = open("demofile3.txt", "a") f.writelines(["See you soon!", "Over and out."]) f.close()  #open and read the file after the appending: f = open("demofile3.txt", "r") print(f.read())</pre>	
Definición y Uso	Sintaxis	Valor de Parámetros
<p>El método writelines() escribe los elementos de una lista en el archivo.</p> <p>El lugar donde se insertará el texto dependerá del modo de archivo y de la posición de la transmisión.</p> <p>“a” el texto se inserta al final del archivo</p> <p>“w” borra el archivo y luego se inserta el texto</p>	<pre>file.writelines(list)</pre>	<p>list</p> <p>La lista de textos que se le pasan al método.</p>