



El Objeto file en Python

Módulo Files: manejo de archivos

Abrir archivos: open

Python incluye la función `open()` la cual permite abrir archivos. Esta regresa un objeto de archivo que permite leer o modificar el archivo.



Al abrir un archivo en Python debes de indicarle el modo en que deseas abrirlo. Por defecto, Python usa el modo 'rt' (lectura para texto).

Modo	Descripción
't'	Modo texto
'b'	Modo binario
'r'	Lectura
'w'	Escritura (si el archivo existe, lo trunca, y si no, lo crea)
'x'	Crea un archivo si no existe, si ya existe marca error
'a'	Escribe al final de un archivo (si no existe, lo crea)
'+'	Lectura y escritura

Algo que debemos de tomar en cuenta con los archivos es que después de utilizarlos es necesario cerrarlos usando el método `close()` para que Python libere los recursos que se ocuparon para abrirlo. Ejemplo:

```
#Abrir archivo
objeto_archivo = open('ejemplo.txt')

#Operaciones con el archivo
contenido=objeto_archivo.read()

#Cerrar archivo
objeto_archivo.close()

print (contenido)
```

With

Usando la declaración *with* podrás asegurarte de que el archivo se cierre después de ejecutar el bloque de código correspondiente sin la necesidad de usar el método `close()`. Ejemplo:

```
with open ('ejemplo.txt') as objeto_
archivo: contenido = objeto_archivo.read()

print (contenido)
```

Los objetos de archivo cuentan con distintos métodos que podrás usar para leer o escribir en un archivo:

<code>read()</code>	Sirve para leer el contenido de un archivo. Requiere abrir el archivo en modo lectura.
<code>readline()</code>	Funciona de manera similar, pero solo lee una línea del archivo. También requiere modo lectura.
<code>readlines()</code>	Regresa una lista con el contenido de cada línea de un archivo como elemento. También requiere modo lectura.
<code>write()</code>	Permite escribir en el archivo. Requiere modo escritura.