

Python Programming

Módulo 2

Trabajar con archivos txt

Escribir y leer archivos

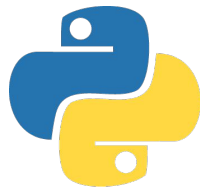
Al igual que en otros lenguajes de programación en Python es posible crear, abrir y leer archivos.

La función incorporada `open()` toma como argumento la ruta de un archivo y retorna una instancia del tipo `file`.

```
f = open("archivo.txt", modo)
```

El *modo* es la forma de cómo se va a trabajar el archivo. Para leer, `modo="r"`. Este es el único modo que cuando se usa `open()` y no se pasa de qué manera interactuar con el archivo se toma automáticamente `"r"`.

Para escribir y crear se usa `modo="w"`, además en este modo si el archivo existe se sobrescribe. Y por último el `modo="a"` de append para añadir contenido al final del archivo si existe el mismo.



En cualquiera de los casos anteriores, si no se especifica una ruta, el fichero se busca en el directorio actual (ruta relativa). Y si vas a especificar la ruta (ruta absoluta), la misma se escribe con doble barra para separar los directorios o barra invertida.

```
"C:\\Users\\Anonimo\\Desktop\\Python Programming\\mi_archivo.txt"
```

```
"C:/Users/Anonimo/Desktop/Python Programming/mi_archivo.txt"
```



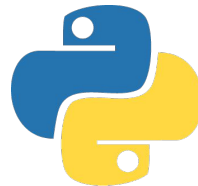
Métodos para trabajar con archivos

Cuando abrimos un archivo para leer, el método `.read()` retorna el contenido del archivo abierto en formato string.

```
f = open("archivo.txt", "r")  
print(f.read())  
f.close()
```

Con `f.readline()` leemos una sola línea del archivo; el carácter de fin de línea (`\n`). Entonces se retorna una cadena.

Si quieres leer todas las líneas de un archivo, puedes usar `f.readlines()` que crea una lista, y cada elemento de esta es un renglón del archivo.



Por otro lado, cuando abrimos un archivo en modo escritura, usamos el método `.write()`

```
f = open("archivo.txt", "w")  
f.write("Hola mundo")  
f.close()
```

La función `write()` graba una cadena en el archivo. Si trabajamos en modo "w" reemplazará todo el contenido del archivo, si el archivo existe, si no existe, lo crea.

Recordemos que para añadir datos al final del archivo sin borrar información previa, el fichero debe abrirse en la modalidad `append` ("a") y también se usa `.write()`.

```
f = open("archivo.txt", "a")  
f.write("Hola ")  
f.write("mundo")  
f.close()
```

También podemos usar el método `writelines()` y pasarle una lista como argumento para que guarde el contenido.

Cuando trabajamos con archivos es importante cerrarlo cuando acabamos de trabajar con ellos, vía el método `close()`. Aunque es verdad que los ficheros normalmente son cerrados y guardados automáticamente, es importante especificarlo para evitar tener comportamientos inesperados.

¡Sigamos trabajando!