



Manejo de archivos

Almacenamiento de información

01 {

[Manejo de archivos]

- Manejo de archivos
- Rutas
- Abrir un archivo
- Cerrar un archivo
- Leer un archivo de texto
- Escribir en un archivo

}

Manejo de archivos {

Hasta los momentos, nuestros programas en Python han presentado una **debilidad clave** y es que **no poseen la capacidad de manejar información de manera persistente.**

Hoy, aprenderemos a **almacenar la información localmente** en el disco duro de nuestro ordenador.

}

Manejo de archivos {

Es de suma importancia para la realización de nuestros programas el entender cómo se manejan los **archivos de texto**. Python nos permite tanto crear, como leer, actualizar y eliminar archivos.

De esta forma, podremos trabajar con la información contenida en estos archivos.

El tipo de archivo que estaremos manejando será archivo de texto plano **.txt**

}

Manejo de archivos {

Sin embargo, es bueno saber que diversidad de formatos adecuados para almacenar información con los que Python puede interactuar:

.xlsx (Excel)

.csv

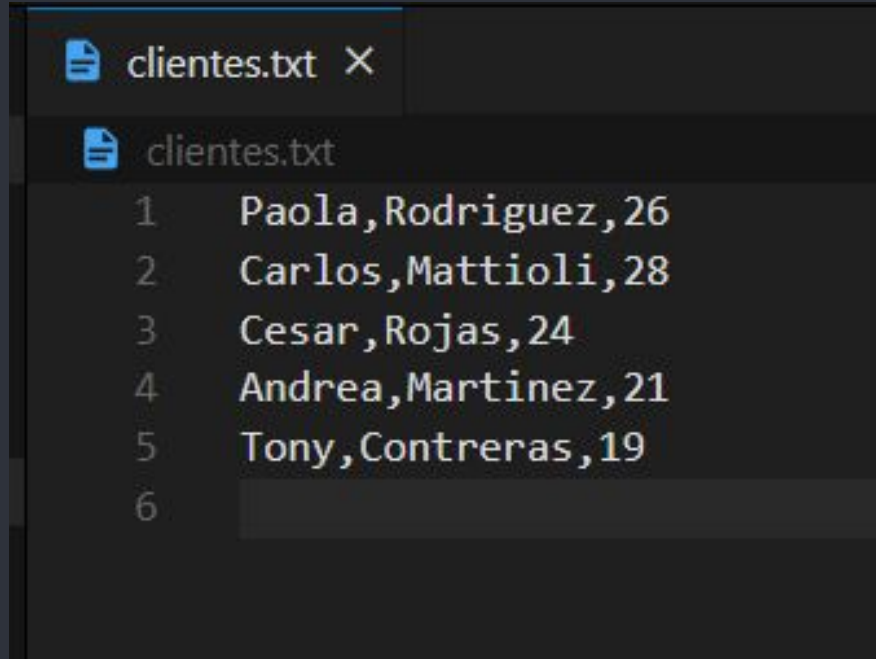
.json.

}

Archivo .txt {

1
2
3
4
5
6
7
8
9
10
11
12
13
14

}



Abrir un archivo {

Usaremos la función `open()` para abrir archivos en Python, esta función recibe dos parámetros: **nombre de archivo** y **modo**.



```
archivo = open("F:\Clase 9\clientes.txt") # Ruta (path) # r (Leer)
archivo = open("clientes.txt") # Ruta relativa(path) # r (Leer)
archivo2 = open("nuevo_archivo.txt", "x") # x (Crear)
```

}

Ejercicio {

Escribir una función que pida un **mensaje** y
guarde en un archivo con el nombre **mensaje.txt**

}

Ejercicio {

Leer **mensaje.txt** para obtener el mensaje guardado, y luego imprimirlo por la consola.

}

Rutas {

Tipo	Descripción	Ejemplo
Ruta absoluta	La ruta absoluta representa la ruta completa del archivo, partiendo del directorio raíz (depende del dispositivo).	"F:\Clase 9\clientes.txt"
Ruta relativa	La ruta relativa representa la ruta del archivo, partiendo del directorio en el que se encuentra el script .py	"clientes.txt"

}

Modos de abrir un archivo {

Modo	Descripción
r	Leer (valor por defecto). Abre un archivo para su lectura, da error si el archivo no existe.
a	Agregar. Abre un archivo para agregar información, crea el archivo si no existe. No eliminará la información anterior.
w	Escribir. Abre un archivo para sobrescribirlo, crea el archivo si no existe. Eliminará la información anterior.
x	Crear. Crea el archivo especificado, devuelve un error si el archivo existe.

Cerrar un archivo {

Debemos cerrar los archivos luego de usarlos en nuestro programa, esto se puede hacer con el **método close()**.

Siempre se deben cerrar los archivos porque, en algunos casos, los cambios realizados en el archivo no se mostrarán hasta que se cierre el archivo.



```
archivo = open("clientes.txt", "r")
archivo.close()
```

Leer un archivo con read() {

La función **open()** nos retorna un objeto **Archivo** que tiene un método para leer su contenido llamado **read()**.




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

}

Leer un archivo con read() {

También podemos indicar **cuantos caracteres queremos leer**
del archivo:



```
archivo = open("clientes.txt", "r")
data = archivo.read(15)
print(data)
archivo.close()
```

}

Leer un archivo con readLine() {

Además, podemos también **leer líneas del archivo** con el método **readLine()**:



```
archivo = open("clientes.txt", "r")
linea1 = archivo.readline() # Leemos la linea 1
linea2 = archivo.readline() # Leemos la linea 2
print(linea1)
print(linea2)
archivo.close()
```

Leer un archivo con for {

Entonces, **si queremos leer todo el archivo** podemos hacerlo con un **ciclo for** de la siguiente forma:




```
archivo = open("clientes.txt", "r")
for linea in archivo:
    print(linea)
archivo.close()
```

}

Leer un archivo con for {

Tambien podemos usar **readLines()** para leer todo el **archivo**. Este método nos retorna una **lista de strings**, donde cada string es una línea del archivo.



```
archivo = open("clientes.txt", "r")
lista = archivo.readlines()
print(lista)
archivo.close()
```

Escribir en un archivo {

Tenemos dos **modos** para escribir en un archivo:

a (Agregar al final del archivo)

w (Elimina la información anterior y agrega lo indicado).

}

Escribir en un archivo con a {




```
archivo = open("clientes2.txt", "a") # Abrimos el archivo
archivo.write("Paola,Sisiruca,35\n") # Agregamos al final del archivo
archivo.close() # Cerramos el archivo

archivo = open("clientes2.txt", "r") # Abrimos el archivo
for linea in archivo:
    print(linea)
archivo.close() # Cerramos el archivo
```

}

Escribir en un archivo con w {



```
archivo = open("clientes2.txt", "w") # Abrimos el archivo
archivo.write("Paola,Sisiruca,35\n") # Agregamos al final del archivo
archivo.close() # Cerramos el archivo

archivo = open("clientes2.txt", "r") # Abrimos el archivo para su lectura
for linea in archivo:
    print(linea)
archivo.close() # Cerramos el archivo
```

}