



El futuro digital
es de todos

MinTIC

«Misión
TIC2022»

Fund. de Programación

Grupos 79,80,81



UNIVERSIDAD
DE ANTIOQUIA

Facultad de Ingeniería

RESUMEN SESIÓN ANTERIOR





ABRIR ARCHIVO

La forma más sencilla para leer un archivo es usando el método `open()`

`open("archivo", "modo")`

modos:

- **'r'** modo lectura (read)
- **'w'** modo escritura (write)
- **'a'** para agregar (append)

El operador `+` agrega funcionalidades al modo eg. **"r+"**



LEER ARCHIVOS

El archivo debe ser abierto (**open**) en un modo que permita la lectura

- **read():** Retorna todo el archivo como un string
- **readline():** Retorna linea por linea
- **readlines():** Retorna el archivo como una lista, donde cada elemento de la lista es una línea

ESCRIBIR Y CERRAR ARCHIVO

- Para escribir se utiliza el método **write()** El cual recibe como argumento un string

file.write("string de datos a guardar")

**Si el modo es w sobrescribe

- EL método **close()** cierra un archivo abierto, lo ideal es que después de realizar una tarea con el archivo se cierre su conexión



SENTENCIA WITH PARA ABRIR ARCHIVOS

La sentencia with se introdujo en python para, entre otras cosas, trabajar con archivos, con esta no es necesario llamar la sentencia close ya que with permite agrupar los procesos de trabajo con el archivo y se encarga de controlar el flujo.

with open('file', 'modo') as f:
 f.metodoA()
 f.metodoB()



ESCRIBIR UN JSON

Se debe abrir un flujo al archivo en modo escritura y usar el método **dump()**

```
dict={1:"dato"}
```

```
with open('file.json','w') as f:  
    json.dump(dict,f)
```




CARGAR UN JSON

Se debe abrir un flujo al archivo en modo lectura y usar el método **load()**

```
with open('file.json','r') as f:  
    dict=json.load(f)
```


Manejo de archivos CSV

JUAN FERNANDO GONZÁLEZ
GRUPOS 79,80,81
Semana 7 Sesión 1





ARCHIVOS CSV

Los archivos CSV (comma-separated values) es un tipo de archivo sencillo que permite almacenar los datos en forma de tabla, normalmente se separan las columnas por coma (,) y las filas por la línea del archivo. Este tipo de archivo es ampliamente usados en aplicaciones de comercio electrónico ya que son muy fáciles de procesar.



MÓDULO CSV

Python cuenta con distintos módulos para trabajar con archivos CSV tales como pandas y csv.

El módulo csv implementa distintas clases y métodos para ejecutar distintas tareas con este tipo de archivos



LEER UN ARCHIVO CSV

Se debe abrir un flujo hacia el archivo con la función **open()**, y para mejorar la compatibilidad lo mejor es agregar el parámetro **newline=""** a open. Posteriormente se usa el método **reader()** el cual retorna un objeto iterable que permite recorrer la tabla línea por línea

```
with open("tabla.csv", newline="") as fcsv:  
    tabla=csv.reader(fcsv)
```



ESCRIBIR UN ARCHIVO CSV

Se utiliza el método **write** y se debe tener abierto un flujo de datos en modo escritura al archivo

```
with open(tabla.csv', 'w', newline='') as csvfile:  
    spamwriter = csv.writer(csvfile, delimiter=' ')  
    spamwriter.writerow([lista de datos])
```

- **writerow**
- **writerows**



LEER TABLA COMO DICCIONARIO

Con el método **DictReader** se puede leer un archivo csv y poner su contenido en un diccionario, donde las llaves del diccionario están dadas por el parametro **fieldnames**, si no se define, se toman como llaves los valores de la primera fila del archivo

```
with open('tabla.csv') as csvfile:  
    reader = csv.DictReader(csvfile)
```



ESCRIBIR TABLA DESDE UN DICCIONARIO

Con el método **DictWriter** se pueden escribir datos desde un diccionario a un archivo csv, en este caso el parámetro **fieldnames** define la primera fila de la tabla

```
with open('tabla.csv', w', newline='') as csvfile:  
    writer = csv.DictWriter(csvfile, fieldnames=[lista])  
    writer.writeheader()  
    writer.writerow({datos})
```




ACTIVIDAD

Se te ha asignado la tarea de crear una clase que permita manejar los datos de clientes de una empresa por medio de una tabla csv, la clase debe permitir leer y agregar los siguientes datos:

id de producto, cantidad, precio

Esta clase será consumida por los desarrolladores encargados del desarrollo de la GUI