

# Clase5\_Repso\_Archivos

April 10, 2023

## 0.0.1 Seminario de Lenguajes - Python

## 0.1 Cursada 2023

### 0.1.1 Clase 5: repaso de archivos

## 1 Repasamos archivos

```
[ ]: archivo = open('datos.txt', 'x')
```

- ¿Cómo se abre el archivo? ¿Existen otras formas? ¿Posibles errores?

## 2 Archivos de texto vs. binarios

```
[ ]: with open("ejemplos/clase5/muchachos.txt", "rb") as archivo:
    texto = archivo.read()
```

```
[ ]: print(texto)
```

```
[ ]: type(texto)
```

- El tipo **bytes** es una secuencia inmutable de bytes.
- Solo admiten caracteres ASCII.

```
[ ]: print(texto.decode('UTF-8')[:228])
```

- El tipo **str** de Python utiliza el estándar **Unicode** para representar caracteres.
- Generalmente la codificación predeterminada es **UTF-8**.
  - UTF significa **Unicode Transformation Format**, y el **8** significa que se utilizan valores de 8 bits en la codificación.

```
[ ]: with open("ejemplos/clase5/rosa_azul.jpg", "rb") as archi_rosa:
    datos = archi_rosa.read()
```

```
[ ]: datos
```

##

Luego veremos cómo procesamos este tipo de archivos utilizando librerías específicas

### 3 JSON vs. CSV

- ¿Con qué tipo de datos trabajan? (¿texto o binario?)
- ¿En qué casos usamos archivos en formato json?
- ¿En qué casos usamos archivos en formato csv?

```
[ ]: import json
datos = [
    {"nombre": "Bitcoin", "sigla": "BTC", "cotización": 28000},
    {"nombre": "Ethereum", "sigla": "ETH", "cotización": 1800},
    {"nombre": "Cardano", "sigla": "ADA", "cotización": 0.38}
]
```

```
[ ]: with open("criptomonedas.json", "w") as archivo:
    json.dump(datos, archivo)
```

```
[ ]: with open("criptomonedas.json", "r") as archivo:
    datos = json.load(archivo)
```

```
[ ]: datos_a_mostrar = json.dumps(datos, indent=4)
print(datos_a_mostrar)
```

```
[ ]: import csv

with open("criptomonedas.json", "r") as archivo_json:
    datos = json.load(archivo_json)

datos
```

```
[ ]: with open("criptomonedas.csv", "w") as archivo_csv:
    writer = csv.writer(archivo_csv)
    writer.writerow(["Criptomoneda", "Sigla", "Cotización (enU$d)"])
    for moneda in datos:
        writer.writerow([moneda["nombre"], moneda["sigla"],
↵moneda["cotización"]])
```

```
[ ]: with open("criptomonedas.csv", "r") as archivo_csv:
    csv_reader = csv.reader(archivo_csv)
    encabezado, datos = next(csv_reader), list(csv_reader)
```

```
[ ]: for moneda in datos:
    print(moneda)
```

## 4 Rutas

```
[ ]: import os
      os.getcwd()
```

```
[ ]: ruta = os.path.dirname(os.path.realpath("."))
      ruta
```

```
[ ]: ruta_completa = os.path.join(os.getcwd(), "ejemplos")
      ruta_completa
```

## 5 Exploremos el siguiente dataset de películas

```
[ ]: ruta_completa = os.path.join(os.getcwd(), "ejemplos", "clase5", "mymoviedb.csv")

      with open(ruta_completa, "r") as archivo_csv:
          csv_reader = csv.reader(archivo_csv)
          encabezado, datos = next(csv_reader), list(csv_reader)
```

```
[ ]: encabezado
```

```
[ ]: datos[:1]
```

## 6 Desafío

Queremos ver qué películas tienen un promedio de popularidad mayor a 9

Deberíamos trabajar con la columna Vote\_Average (columna 5)

```
[ ]: pelis = filter(lambda x: float(x[5])>9, datos)
```

```
[ ]: for peli in pelis:
      print(peli[1])
```

##

Recordemos que el csv contiene texto

## 7 Desafío

¿Cuáles son los idiomas de las películas del dataset?

```
[ ]: idiomas = map(lambda x: x[6], datos)
      print(list(idiomas))
```

### 7.1 ¿Nos interesa que se repita?

```
[ ]: idiomas = set(map(lambda x: x[6], datos))
```

```
[ ]: print(idiomas)
```

##

Notamos que hay datos que no son correctos.

#

Más adelante veremos cómo hay otras formas de procesar estos datasets

## 8 Accedemos al poster de las Spiderman

```
[ ]: datos[:1]
```

```
[ ]: spiderman = datos[:1][0]
poster_spiderman = spiderman[8]
poster_spiderman
```

```
[ ]: import requests

imagen = requests.get(poster_spiderman)
with open("ejemplos/clase5/poster.jpg", 'wb') as f:
    f.write(imagen.content)
```