

# Clase 4\_1\_Cont\_archivos

April 6, 2022

## 0.0.1 Seminario de Lenguajes - Python

## 0.1 Cursada 2022

### 0.1.1 Archivos (cont.)

## 1 Pensemos en las siguientes situaciones:

¿Qué hacemos si queremos:

- agregar nuevos datos a los datos ya existentes?;
- modificar el puntaje de un jugador específico en un juego?;
- agregar una nueva pregunta la banco de preguntas del juego de repaso?;
- incrementar los Python Plus de un estudiante determinado?

¿Qué tienen todas estas situaciones en común?

## 2 Acceso aleatorio

- Si el valor `desde_donde` no está, se asume 0.
- Si el archivo es de texto, solamente se considera desplazar desde el comienzo del archivo, es decir el valor `desde_donde` es 0.

## 3 Veamos el siguiente ejemplo

```
[1]: archivo = open("ejemplos/clase4/imagen.txt", "rb")

archivo.seek(-18, 2)      #Esto da error si el archivo se abre en modo "r"
print(archivo.read().decode('UTF-8'))
print("*"*30)
archivo.seek(0, 0)
print(archivo.read().decode('UTF-8'))
archivo.close()
```

ivin' for today.

```
*****
Imagine there's no heaven
```

It's easy if you try  
No hell below us  
Above us, only sky  
Imagine all the people  
Livin' for today.

```
[2]: archivo = open("ejemplos/clase4/imagen.txt", "rb+")

archivo.seek(34, 0)      #Esto da error si el archivo se abre en modo "r"
archivo.write(bytes("XXXXXXXXXXXXXXXXXX", 'UTF-8'))
archivo.close()
```

- ¿Por qué convierto a bytes?

```
[4]: archivo = open("ejemplos/clase4/imagen.txt", "rb")
print(archivo.read().decode('UTF-8'))
archivo.close()
```

Imagine there's no heaven  
It's easXXXXXXXXXXXXXXXXXhell below us  
Above us, only sky  
Imagine all the people  
Livin' for today.

- Probemos esto mismo abriendo trabajando con archivos de texto.
- ¿`archivo.read(4).decode('UTF-8')`?

## 4 ¿Dónde estamos?

- `tell()`: retorna la posición actual.

```
[5]: archivo = open("ejemplos/clase4/imagen.txt", "rb")
archivo.seek(0, 2)
print(archivo.tell())
archivo.close()
```

125

## 5 Entrada y salida estándar

- Tres archivos estándares: `stdin`, `stdout`, `stderr`.
- Usados por el intérprete como entrada, salida y error estándar.
- Se los accede a través del módulo `sys`.

```
[108]: import sys

sys.stdout.write("Hola \n\n que tal")
```

Hola

que tal

```
[ ]: nombre = input()
```

```
[ ]: import sys

nombre = sys.stdin.readline()
nombre
```

## 6 Retomamos los módulos estándares

### 6.1 Módulo OS

- Incluye funciones para:
  - Procesamiento de archivos.
  - Directorios.
  - Permisos.
- Más info: <http://docs.python.org/library/os.html#module-os>

```
[110]: import os
#os.rename('ejemplos/clase5/imagene.txt', 'ejemplos/clase5/COPIA.TXT')
os.remove('bandas.dat')
```

Veamos estos otros ejemplos

```
[111]: lista = os.listdir("/home/claudia/git/")
print(lista)
print(os.sep)
```

```
['listado', 'postitulo', 'Python', 'libro-python', 'www', 'proyecto2014',
'test', 'papers', 'descarga.jpeg']
/
```

```
[112]: import os
print(os.access("/home/claudia/git/", os.W_OK))
```

True

## 7 Módulo os.path

- Provee algunas funciones útiles: exists(), isdir(), isfile()

```
[114]: import os.path

print(os.path.exists("/home/claudia/git/"))
print(os.path.isdir("/home/claudia/git/"))
```

True

True