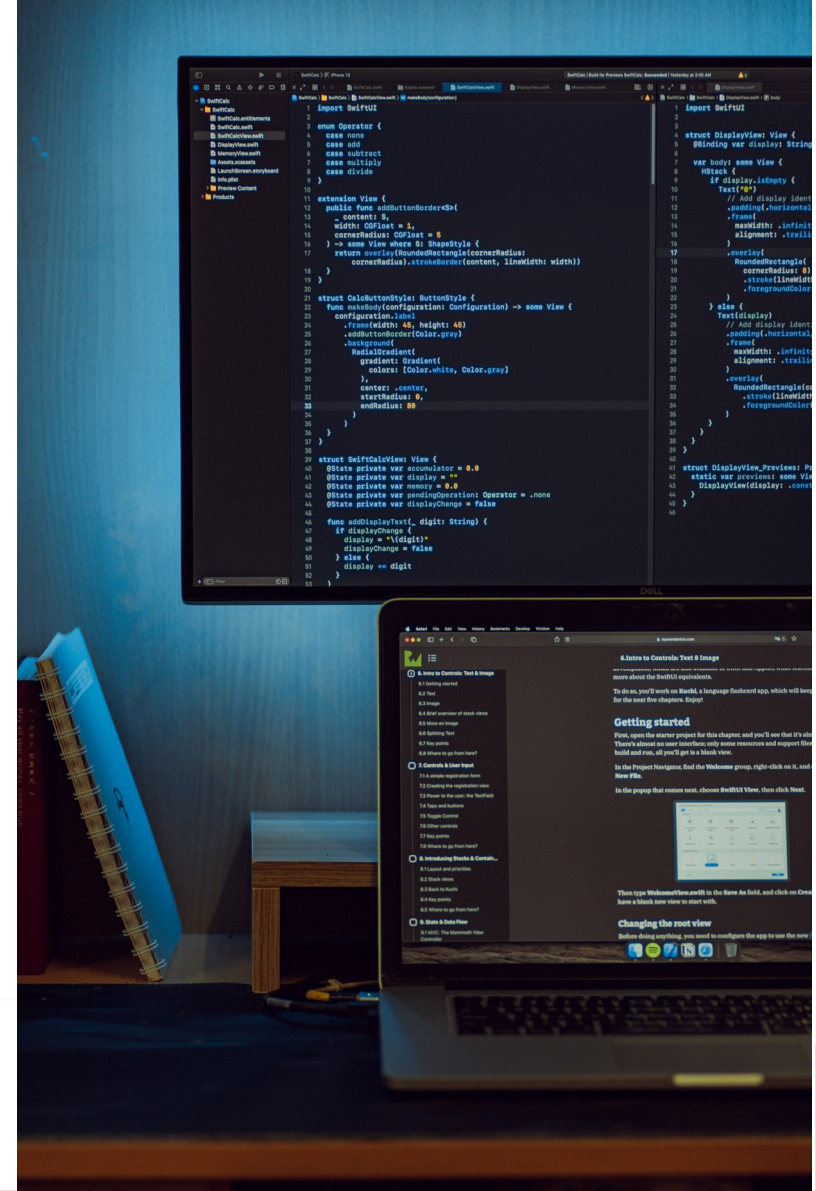


Trabajo con ficheros



Lectura/Escritura de ficheros

Vamos a intentar usar la palabra clave **with** para ahorrarnos parte del código. Usando **with** nos aseguramos que cuando el flujo de ejecución abandona el contexto creado los recursos se liberan.

```
with open("cualquier_fichero.txt", 'r') as fichero:  
    fichero.read()
```

```
with open("cualquier_fichero.txt", 'w') as fichero:  
    fichero.write("hola mundo")
```

```
fichero = open("cualquier_fichero.txt", 'r')  
try:  
    fichero.read()  
finally:  
    fichero.close()
```



Rutas de los ficheros

Si intentamos ejecutar el primer ejemplo de leer fichero:

```
File "/home/josnar/PycharmProjects/Nebrija/curso_python_basico_ficheros/ficheros/leer_fichero.py", line 2, in <module>
    with open("cualquier_fichero.txt", 'r') as fichero:
FileNotFoundError: [Errno 2] No such file or directory: 'cualquier_fichero.txt'
```

Tenemos que aprender a construir rutas correctas de ficheros.

Problema: Las rutas son diferentes según el sistema operativo

Solución: Python nos regala un módulo (os) para hacer código que funcione con cualquier sistema operativo



Módulo os

<https://docs.python.org/3/library/os.html>

Nos ofrece soluciones compatibles con cada sistema operativo de un montón de funciones que dependen del mismo.

<https://docs.python.org/3/library/os.path.html#module-os.path>

Dentro de os, os.path se encarga de gestionar las rutas (paths) de los ficheros/carpetas.



Rutas relativas/absolutas

Cuando ejecutamos un programa, nos encontramos en una ruta específica del sistema de ficheros, por lo que podemos movernos desde ahí en busca del recurso que queramos.

Nota: Para ir hacia la raíz del sistema de fichero, tendremos que usar el string “..”

Las rutas absolutas son aquellas que empiezan desde la raíz del sistema de ficheros. En linux “/” o en windows podría ser “c:”



Rutas relativas/absolutas

Cuando ejecutamos un programa, nos encontramos en una ruta específica del sistema de ficheros, por lo que podemos movernos desde ahí en busca del recurso que queramos.

Nota: Para ir hacia la raíz del sistema de fichero, tendremos que usar el string “..”

Las rutas absolutas son aquellas que empiezan desde la raíz del sistema de ficheros. En linux “/” o en windows podría ser “c:”



Construir un path

Función básica → `os.path.join()`

Se invoca con un número variable de argumentos, que se irán concatenando.

Para otras funciones útiles mirar en:

<https://docs.python.org/3/library/os.path.html#module-os.path>



Modos de apertura de un fichero

Character	Meaning
'r'	open for reading (default)
'w'	open for writing, truncating the file first
'x'	open for exclusive creation, failing if the file already exists
'a'	open for writing, appending to the end of file if it exists
'b'	binary mode
't'	text mode (default)
'+'	open for updating (reading and writing)



Lectura de un fichero

- `read(size=-1)` → Lee el número de caracteres indicado por `size`. Si `size` no se indica, lee hasta el final del fichero.
- `readline(size=-1)` → El el número de caracteres indicado por `size`, aunque si encuentra un final de línea, acaba. Si `size` no se indica, lee hasta el final de la línea.
- `readlines()` → Lee todas las líneas del fichero y las devuelve como una lista.



Escritura de un fichero

- `write(any_text)` → Añade en la posición donde se encuentra el cursor de la instancia del fichero que tenemos abierta



Caracteres especiales en un fichero

- `'\n'` → Fin de línea.
- EOF → Fin de fichero.
- `'\r'` → Retorno de carro. En ficheros generados en windows es común encontrar que una nueva línea se marca con `'\r\n'`

