

Python Programming

Módulo 03 - Desafío

¿Qué son los desafíos?

- Son **ejercicios complementarios** a los realizados en clase.
- **Permiten practicar y consolidar los conocimientos adquiridos.**
- Es recomendable que los realices antes de rendir el examen de la clase.



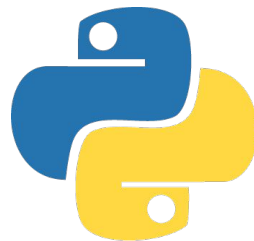
Ejercicio 1: Cliente

Crear una clase cliente que pida por argumento los atributos: `nombre_apellido(str)` y el `cuit(str)`, usar guiones del medio).

Únicamente tendrá un método que sea `.mostrar_info()` que muestra por pantalla el nombre del cliente y su cuit.

Ejemplo al crear una instancia:

```
cliente1 = Cliente(nombre_y_apellido, cuit)
```



Ejercicio 2: Quedó registrado

Tomar de la sección llamada *Descarga*, el **Programa Buscar Archivos**. El mismo recibe como argumento una ruta y una extensión. Transformar el mismo creando un menú, donde tengamos las siguientes opciones.

Menú

```
1 - Buscar archivos con determinada
extensión
2 - Salir
>>>
```

Si se selecciona la opción 1, se procede a pedir la ruta y la extensión por teclado.

Tratar de usar alguna manera para borrar pantalla y que siempre volvamos al menú recurrentemente, pero que la pantalla quede limpia.

Además, podríamos implementar el uso del módulo ["tabulate"](#) (investigar) que se instala con "pip", y sirve para armar una tabla, entonces se podría listar los archivos con un número de costado indicando el orden que se fueron encontrando en esa tabla.

El resultado en pantalla sería algo como esto:

```
1 - Buscar archivos con determinada extensión
2 - Salir

>>> 1
Ingrese ruta: C:\carpeta_ejemplo
Ingrese extensión de archivo a buscar: .py
+---+-----+
| ID |          ARCHIVO          |
+---+-----+
| 1  |          boot.py          |
| 2  |          dolar.py         |
| 3  |          envio.py         |
+---+-----+
Toque ENTER para continuar...
```

La opción salir finaliza el programa.



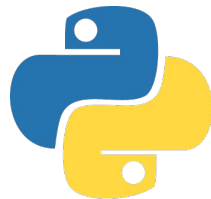
Ejercicio 3: Backup

Como bien sabemos "pip" nos permite instalar módulos de terceros. Pero además de pasarle el argumento "install", podemos pasar otros.

Por ejemplo, "pip list" nos muestra una lista de los módulos instalados. Existen más comandos, para más información, puedes ejecutar desde la consola del sistema operativo: "pip -h".

Este ejercicio trata de crear un script que use el módulo subprocess para ejecutar "pip list" y de esa manera con `capture_output` tomar la lista y logremos guardar en un "txt", sólo los nombres de los módulos.

Este programa será una automatización útil para tener a mano y hacer un "backup".



En la sección de **Descargas** encontrarás los recursos necesarios para realizar los ejercicios y su resolución para que verifiques cómo te fue.



¡Terminaste el módulo!
Estás listo para rendir el examen