

Python Programming

Módulo 04 - 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: Contactos I

Necesitamos almacenar: Nombre, teléfono, email y dirección de proveedores de materias primas para una pyme.

Vamos a desarrollar un programa con un pequeño menú que nos permita almacenar esos datos.

El menú puede ser como el siguiente:

Menú:

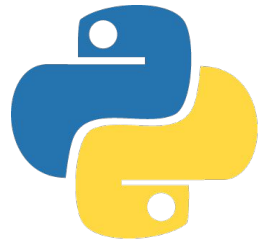
- 1. Ingreso de nuevo contacto
 - 2. Salir
- >>>

Si eligen la opción 1, se carga un nuevo contacto. Los datos se cargan por teclado.

Si, en cambio, selecciona la opción 2, salimos.

Tener en cuenta que es un menú recurrente, tener detalles de prolijidad, como borrar la pantalla y otras cuestiones del estilo, puede resultar en una mejor presentación.

- La base de datos será motor Sqlite3 con nombre *datos.sqlite* y la tabla debería de llamarse *contactos*.
- Todos los campos serán del tipo "text" .
- Si deseamos podemos agregar un campo delante de todo que sea el ID. Ese campo ID (opcional) puede ser auto incremental.



Ejercicio 2: Contactos II

Al ejercicio anterior, agregar al menú de las opciones una opción de **mostrar todos los contactos** que, haciendo uso del módulo `"tabulate"`, podríamos mostrar una tabla en la consola con todos los contactos.

También agregar una opción de **buscar por nombre**, con alguna coincidencia en minúsculas o en mayúsculas. Al buscar `"To"`, buscaría los contactos que tengan en el campo nombre el texto `"to"` en alguna parte, y mostraría todos sus datos.

El nuevo menú:

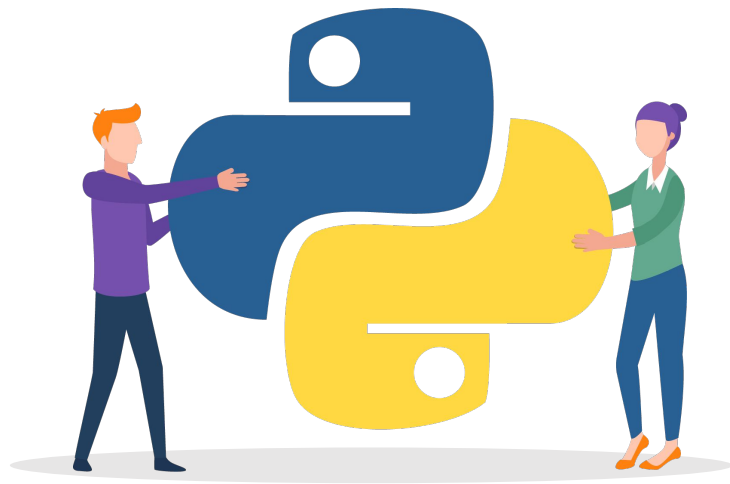
Menú:

1. Ingreso de nuevo contacto
 2. Mostrar todos los contactos
 3. Encontrar contacto por aproximación en el nombre
 4. Salir
- >>>

Ejercicio 3: Comma Separated Values

Antes de entrar al ejercicio, te vamos a contar qué son los **archivos csv**.

Los **archivos CSV** son un tipo de documento en formato abierto sencillo para representar datos en forma de tabla, pero en texto plano. Las columnas se separan por comas y las filas por saltos de línea. Se emplean mucho en análisis de datos y otras áreas.



Por ejemplo, esta tabla:

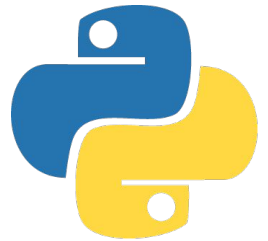
Año	Marca	Modelo	Descripción	Precio
1997	Ford	E350	ABS and Air conditioner	3000.00
1998	Chevy	Venture	Extended Edition	4900.00
1999	Chevy	Venture	Extended Edition	5000.00
1996	Jeep	Grand Cherokee	Moon roof	4799.00

Como archivo csv se vería así:

```
Año,Marca,Modelo,Descripción,Precio
1997,Ford,E350,ABS and Air conditioner, 3000.00
1998,Chevyr,Venture,Extended Edition,4900.00
1999,Chevy,Venture,Extended Edition,5000.00
1996,Jeep,Grand Cherokee,Moon roof,4799.00
```

En Python podemos crear o abrir un csv, con la función `open()` y adentro de la función pasando como argumento el nombre del archivo con extensión csv y no como txt. Los modos son iguales a los que manejamos con los archivos de texto ("`w`", "`r`" y "`a`"). Además, dentro del archivo debemos respetar el delimitador (la coma) y el salto de línea para indicar las filas.

Esta estructura se puede armar concatenando textos con "comas", y con "`\n`" el salto de línea, etc. También se puede usar un módulo de la librería estándar de Python que se llama "`csv`" y hace las cosas un poco más automáticas.



Ahora si... el enunciado:

Se desea hacer un script que cuando se ejecute, descargue la tabla de contactos (incluida en la sección **Descargas**) a un archivo csv.

El script debe de estar dentro del mismo directorio que la base de datos, así no tenemos que poner ninguna ruta absoluta, y de esta manera se toma por defecto la ruta relativa.

Recordar que el procedimiento es similar al de guardar un txt pero cambiando la extensión a csv.

Ejemplo:

```
f = open("contactos.csv", "w")
```

Al finalizar, abrir el archivo csv con algún editor de código (puede ser con el mismo VSCode) y verificar que la estructura sea la correcta.

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