



**SEAS**  
CAMPUS SEAS

**Oriol Chiva Hidalgo**

**Trabajo Obligatorio. Python**

**10/2025**

**GRUPO SAN VALERO**

**SEAS, Estudios Superiores Abiertos**

**ZARAGOZA**



### Propuesta de trabajo

El siguiente ejercicio consiste en escribir un programa para gestionar una agenda telefónica con los nombres y los teléfonos de los conocidos de una persona. El programa incorporará las funciones para:

- Consultar el teléfono de una persona.
- Añadir el teléfono de una persona nueva.
- Eliminar el teléfono de una persona.

La agenda debe estar guardada en el fichero de texto agenda.txt donde el nombre de las personas y su teléfono deben aparecer separados por comas y cada persona debe aparecer en una línea distinta.

Se deberán implementar las siguientes funciones:

- `obten_telefono(fichero, persona)`  
# esta función devuelve el teléfono de una persona.
- `inserta_telefono(fichero, persona, telefono)`  
# esta función inserta el teléfono de una persona. Dará un mensaje de error si el fichero no existe
- `elimina_telefono(fichero, persona)`  
# esta función elimina el teléfono de una persona.
- `menu()`  
# presenta un menú con las opciones disponibles para gestionar la agenda.

Como se acaba de ver, el programa estará conducido por un menú, que será el siguiente:

\_\_\_\_\_ MENÚ \_\_\_\_\_

1. Obtener un teléfono.
2. Insertar un teléfono.
3. Eliminar un teléfono.
4. Salir

NOTA: Todas las funciones devolverán un mensaje de error si no se han podido llevar a cabo correctamente.

### Objetivos del trabajo

Con el presente trabajo se pretende:

- Que el alumno desarrolle un programa gestionado por menús en el que se trabajen muchos de los conceptos estudiados en el curso.

### Bibliografía

- Manual de la asignatura.
- Ejercicios realizados a lo largo del curso.

### Criterios de evaluación

La evaluación, es una componente fundamental de la formación. Este trabajo obligatorio formará parte de tu calificación final. En esta tabla, se resumen los aspectos a valorar y el porcentaje que representa cada uno de los mismos.

CONTENIDOS GENERALES	10%
Explicación general del programa, claridad de la programación, limpieza del código y presentación.  Claridad en los conceptos. Desarrollo modular de forma que sea aprovechable para otras aplicaciones.	10%
TEMAS DE ESPECIALIDAD	90%
Explicación correcta del desarrollo de las funciones que intervienen, así como de la programación del menú.	20%
Desarrollos de las funciones que intervienen en el programa.	50%
Prueba de funcionamiento en todos los casos posibles que se puedan presentar, incluyendo aquellos que implican fallos en la introducción de los datos.	20%

### Fecha límite de recepción de trabajos

La fecha de entrega deberá ser anterior a la fecha fin de matrícula.

### Ficha de Corrección del Trabajo

(Espacio reservado para anotaciones del profesor y doble corrector).

Profesor:	
Alumno (Código / Nombre):	
Fecha de Entrega:	Fecha de Calificación:

### Observaciones sobre el trabajo

Este espacio está reservado para que el profesor titular describa anotaciones que considera importantes sobre la realización del trabajo

También está destinado para que el profesor que efectúa la doble corrección pueda realizar sus anotaciones, asimismo se podrán describir las conclusiones a las que se ha llegado tras realizar la doble corrección.

### Formato de presentación

1. Se presentará el código en el formato .py de los programas de Python. Las explicaciones se presentarán en formato informático.
2. Se valorará el realizar otras versiones del programa para trabajar con otros tipos de archivos, además del archivo .txt del enunciado.



### Desarrollo del trabajo

Mi trabajo se ha realizado, con la idea de separar todo en funciones, de esta manera es mucho más fácil donde encontrar los errores y además se puede ir reutilizando el código durante el programa.

Nos piden que usemos un fichero llamado "agenda.txt" así que solo empezar el programa los hemos creado. En él se han ido añadiendo o eliminando los usuarios que deben tener un número de teléfono.

El enunciado nos pide que en cada función si hay un error que lo devuelva, la forma más correcta es usando el try catch, pero también nos avisan que se valorará correctamente otro tipo de versiones, por eso también en vez de usar un try catch se utiliza un if-else.

Adjunto imágenes demostrando los posibles errores que pueden tener el programa y también su funcionamiento:

```
C:\Users\Chiva\AppData\Local\Microsoft\WindowsApps\python3.12.exe "C:\Users\Chiva\Dropbox\PC\Desktop\SEAS\Python\Unidad 1\trabajo_obligatorio.py"

----- MENÚ -----

1. Obtener un teléfono
2. Insertar un teléfono
3. Eliminar un teléfono
4. Salir

Seleccione una opción: 2
Nombre del cliente: oriol

Teléfono del cliente: 738273
Número de teléfono incorrecto.

Teléfono del cliente: jwdwue
Número de teléfono incorrecto.

Teléfono del cliente: 662452103
Número de teléfono correcto.

Cliente Oriol con el número 662452103 añadido correctamente

----- MENÚ -----

1. Obtener un teléfono
2. Insertar un teléfono
3. Eliminar un teléfono
```



----- MENÚ -----

1. Obtener un teléfono
2. Insertar un teléfono
3. Eliminar un teléfono
4. Salir

Seleccione una opción: **1**

Nombre del cliente: **newne**

No existe ningún cliente con el nombre 'Newne'.

----- MENÚ -----

1. Obtener un teléfono
2. Insertar un teléfono
3. Eliminar un teléfono
4. Salir

Seleccione una opción: **1**

Nombre del cliente: **4404**

No existe ningún cliente con el nombre '4404'.



```
----- MENÚ -----
```

1. Obtener un teléfono
2. Insertar un teléfono
3. Eliminar un teléfono
4. Salir

Seleccione una opción: 3

Nombre del cliente: wjnsjfk

No se encontró ningún cliente con el nombre 'Wjnsjfk'.

```
----- MENÚ -----
```

1. Obtener un teléfono
2. Insertar un teléfono
3. Eliminar un teléfono
4. Salir

Seleccione una opción: 6

Debe seleccionar una opción de 1 a 4.

```
----- MENÚ -----
```

1. Obtener un teléfono
2. Insertar un teléfono
3. Eliminar un teléfono
4. Salir

Seleccione una opción: 4

Saliendo del menú

Process finished with exit code 0