

## Sesió 3. Funciones. Listas y Diccionarios

### Actividad Entrega Obligatoria

Este documento forma parte del curso [Programación con Python](#) del [CEFIRE CTEM](#).

Esta obra está sujeta a la licencia Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional de Creative Commons. Para ver una copia de esta licencia, visitad <http://creativecommons.org/licenses/by-nc-sa/4.0/>.

Autoría: José Luis Tomás Navarro

Revisión: M<sup>a</sup> Paz Segura Valero



### Actividad Entrega Obligatoria: Cálculo Notas Alumnos

En esta actividad obligatoria utilizaremos las funciones y las listas, además del resto de elementos que vimos en la sesión anterior como estructuras condicionales, repetitivas, variables, etc.

El objetivo es realizar unos cálculos sobre las notas de unos alumnos que estarán almacenadas en dos listas: **lista\_nombres\_alumnos**, **lista\_notas\_alumnos**.

Suponemos que cada alumno tiene una única nota, de manera que el componente n-ésimo de la lista de las notas será la nota obtenida por el alumno n-ésimo de la lista de los alumnos. En el siguiente esquema se aprecia mejor esta idea:

#### lista\_nombres\_alumnos

0	1	2	3	...
María García	Pedro Martínez	Carla Gil	Juan López	...

#### lista\_notas\_alumnos

0	1	2	3	...
8,5	6,75	9	4	...

Por ejemplo, el segundo componente de la lista de notas que tiene valor **6,75** (correspondiente al índice con valor 1) equivale a la nota del alumno **Pedro Martínez** que ocupa la segunda posición (índice 1) en la lista de los nombres.

En primer lugar solicitaremos al usuario que introduzca el número total de alumnos de utilizando la función **introduce\_total\_alumnos()**, la cual se encargará de solicitar el número de alumnos y de comprobar que este valor se encuentra ente 1 y 5, ambos inclusive. En caso de que el valor introducido no se encuentre en este intervalo, se visualizará un mensaje de error y la función volverá a solicitar el dato. Para realizar este control se empleará **una estructura repetitiva while ...**

La función `introduce_total_alumnos()` devolverá al programa principal (`return ...`) el número total de alumnos (`num_total_alumnos`).

A continuación, el programa principal solicitará al usuario que introduzca los datos relativos a los alumnos (**nombre y nota**) utilizando para ello **una estructura repetitiva for...**

Para introducir el nombre del alumno se empleará la función `introduce_nombre_alumno(...)`, la cual tendrá dos parámetros: el **número de alumno** que estamos introduciendo (1, 2, 3, etc.) y la lista con los **nombres** de los alumnos (`lista_nombres_alumnos`).

La función visualizará el número de alumno que estamos introduciendo y solicitará el nombre, según podemos ver:

**ALUMNO NÚMERO 1**

**Introduce el Nombre:**

Una vez que el usuario haya introducido el **nombre**, la función comprobará que dicho **nombre** de alumno no haya sido introducido ya, en cuyo caso visualizará un mensaje de error y solicitará un nuevo nombre. Emplearemos **una estructura repetitiva while...**, para realizar este control.

Finalmente, cuando el nombre sea correcto, la función devolverá al programa principal el nombre del alumno (`return...`), que insertará en la lista de nombres de alumnos.

Para introducir la nota del alumno se utilizará la función `introduce_nota_alumno()`, la cual solicitará la introducción de la nota como un dato real (float), según podemos ver a continuación:

**Introduce la Nota (0-10):**

Los valores deben encontrarse entre **0 y 10**, ambos inclusive. En caso de no ser así, la función mostrará un mensaje de error y volverá a solicitar la nota. Utilizaremos **una estructura repetitiva while...** para realizar este control.

Una vez que estemos seguros de que la nota es válida, se devolverá al programa principal (**return...**), quien la insertará en la lista de las notas.

Cuando ya tengamos introducidos todos los datos de los alumnos, el programa principal visualizará un menú con las siguientes opciones:

#### **MENÚ DE OPCIONES**

- 1.- Visualizar el alumno con mayor nota**
- 2.- Visualizar el alumno con menor nota**
- 3.- Calcular el número de Aprobados**
- 4.- Calcular el número de Notables**
- 5.- Salir**

Cada una de las cuatro primeras opciones será realizada por una función, mientras que con la última opción finalizará el programa. Emplearemos **una estructura repetitiva while...** para gestionar este menú.

En la siguiente tabla se muestran las funciones que llevarán a cabo el cometido asociado a cada opción de menú.

<b>Opción</b>	<b>Función</b>	<b>Cometido</b>
1	<b>visualiza_alumno_mayor_nota(...)</b>	Recibirá como parámetros las dos listas de datos de alumnos y buscará en la lista de notas cuál es la nota más alta. Finalmente visualizará tanto la nota como el nombre del alumno localizado.

2	<b>visualiza_alumno_menor_nota(...)</b>	Similar al anterior pero, en este caso, buscará el alumno con menor nota.
3	<b>calcula_num_aprobados(...)</b>	Esta función recibirá como parámetros las dos listas y calculará cuántos alumnos han aprobado. Finalmente visualizará el número de aprobados y el nombre de cada uno de los alumnos aprobados.
4	<b>calcula_num_notables(...)</b>	Esta función es similar a la anterior pero visualizará el número de alumnos que han obtenido notable (nota $\geq 7$ y nota $< 9$ ) y el nombre de los mismos.

En las dos últimas funciones se aconseja la utilización de **listas auxiliares**, que podemos utilizar para ir almacenando el nombre de los alumnos que cumplen los requisitos solicitados.

El esqueleto que se debería de seguir para realizar esta actividad obligatoria quedaría de la siguiente forma:

```
# Definición de las Funciones que se utilizarán en el Programa Principal
def introduce_total_alumnos():
    ...
def introduce_nombre_alumno(...):
    ...
def introduce_nota_alumno():
    ...
def visualiza_alumno_mayor_nota(...):
    ...
def visualiza_alumno_menor_nota(...):
    ...
def calcula_num_aprobados(...):
```

```
...
def calcula_num_notables(...):
    ...

# Programa Principal
# Llamamos a introduce_total_alumnos()
...
for ... # Para rellenar los datos de los alumnos
    ....

while ... # Gestión del menú con las cinco opciones
    ...
```

**Es importante seguir este orden, sobre todo en lo referente a la definición de todas las funciones al principio, antes del programa principal.**