



El futuro digital
es de todos

MinTIC



EJERCICIOS PROPUESTOS

Listas y Diccionarios



Ejercicios de Listas

Ejercicio 1

Diseñe un algoritmo que lea por teclado las 5 notas obtenidas por un alumno (comprendidas entre 0 y 10). A continuación, debe mostrar todas las notas, la nota media, la nota más alta que ha sacado y la menor.

```
import math #importamos la libreria matematicas
notas=[] #Lista vacia de notas
for indice in range(1,6):
    nota=float(input(f'Digite nota {indice}:'))
    if nota>=0 and nota<=10:
        notas.append(nota)
    else:
        print('Nota fuera del rango')

#print(notas)
for indice in notas:
    print(indice)

print('Nota media:',sum(notas)/len(notas))
print('Nota máxima:',max(notas))
print('Nota mínima:',min(notas))
```

Ejercicio 2

Diseñe un algoritmo que pida un número al usuario un número de mes (por ejemplo, el 5) y diga cuántos días tiene (por ejemplo, 31) y el nombre del mes. Debes usar listas. Para simplificarlo vamos a suponer que febrero tiene 28 días.

```
meses=['Enero','Febrero','Marzo','Abril','Mayo','Junio','Julio',
'Agosto','Septiembre','Octubre','Noviembre','Diciembre']
dias=[31,28,31,30,31,30,31,31,30,31,30,31]
while True:
    mes=int(input('Digite el mes (1-12):'))
    if mes>=1 and mes<=12:
        print('El mes',mes,'es:',meses[mes-1],'tiene',dias[mes-1],'días')
    else:
        print('El mes esta fuera del rango')
    if mes==0:
        break
```

Ejercicio 3

Diseñar un programa que tenga los siguientes requerimientos:

- Crea una tabla (lista con dos dimensiones) de 5x5 enteros.
- Carga la tabla con valores numéricos enteros.
- Suma todos los elementos de cada fila y todos los elementos de cada columna visualizando los resultados en pantalla.

Ejercicio 4

Escriba un algoritmo que permita crear una lista de palabras y que, a continuación

de tres opciones:

- • **Contar:** Me pide una cadena, y me dice cuantas veces aparece en la lista
- • **Modificar:** Me pide una cadena, y otra cadena a modificar, y modifica todas las apariciones de la primera por la segunda en la lista.
- • **Eliminar:** Me pide una cadena, y la elimina de la lista.
- • **Mostrar:** Muestra la lista de cadenas
- • **Terminar**



Ejercicios de Diccionarios

Ejercicio 1

Escribe un programa python que pida un número por teclado y que cree un diccionario cuyas claves sean desde el número 1 hasta el número indicado, y los valores sean los cuadrados de las claves.

Ejercicio 2

Vamos a crear un programa en python donde vamos a declarar un diccionario para guardar los precios de las distintas frutas. El programa pedirá el nombre de la fruta y la cantidad que se ha vendido y nos mostrará el precio final de la fruta a partir de los datos guardados en el diccionario. Si la fruta no existe nos dará un error. Tras cada consulta el programa nos preguntará si queremos hacer otra consulta.

```
precios={
    'fresa':4600,
    'maracuya':2500,
    'lulo':3000,
    'tomate de árbol':1786,
    'fruta de dragón':8000
}
while True:
    fruta=input('Digita la fruta:').lower()
    if fruta not in precios:
        print('Fuera de rango')
    else:
        cantidad=float(input('Digite cantidad:'))
        #print('El precio de ',fruta, 'es:', cantidad*precios[fruta])
        print(f'El precio de {fruta} es: {cantidad*precios[fruta]} pesos')
        salir=input('¿Quiere hacer otra consulta? s/n')
        if salir=='n':
            break
        else:
            continue
```

Ejercicio 3

Codifica un programa en python que nos permita guardar los nombres de los alumnos de una clase y las notas que han obtenido. Cada alumno puede tener distinta cantidad de notas. Guarda la información en un diccionario cuya claves serán los nombres de los alumnos y los valores serán listas con las notas de cada alumno.

El programa pedirá el número de alumnos que vamos a introducir, pedirá su nombre e irá pidiendo sus notas hasta que introduzcamos un número negativo. Al final el programa nos mostrará la lista de alumnos y la nota



media obtenida por cada uno de ellos. Nota: si se introduce el nombre de un alumno que ya existe el programa nos dará un error.

Ejercicio 4

Escribir un programa que implemente una agenda. En la agenda se podrán guardar nombres y números de teléfono. El programa nos dará el siguiente menú:

- **Añadir/modificar:** Nos pide un nombre. Si el nombre se encuentra en la agenda, debe mostrar el teléfono y, opcionalmente, permitir modificarlo si no es correcto. Si el nombre no se encuentra, debe permitir ingresar el teléfono correspondiente.
- **Buscar:** Nos pide una cadena de caracteres, y nos muestra todos los contactos cuyos nombres comiencen por dicha cadena.
- **Borrar:** Nos pide un nombre y si existe nos preguntará si queremos borrarlo de la agenda.

Listar: Nos muestra todos los contactos de la agenda.