

*Creado por:*

*Isabel Maniega*

## Tema 4 - Bucles-

Principales Estructuras de datos (se verán en otro archivo un poco mejor):

- Tuplas ()
- numpy arrays `np.array ([ ])`
- Listas `[ ]`
- Diccionarios `{'clave': valor} / {'key': value}`
- dataframes (matriz de datos)

Signos en Python:

- Mayor que `>`
- Menor que `<`
- Mayor o igual que `>=`
- Menor o igual que `<=`

### -2.1- For

```
In [1]: ventas = [100, 20, 30, 40, 50]
ventas
```

```
Out[1]: [100, 20, 30, 40, 50]
```

```
In [2]: for venta in ventas:
        print(venta)
```

```
100
20
30
40
50
```

```
In [3]: len(ventas)
```

```
Out[3]: 5
```

```
In [4]: # Posicion de izquierda a derecha: 0, 1, 2, ...
        # Posición de derecha a izquierda: -1, -2, -3, ...
        listado = [10, 20, 30, 40, 50, "Hola Mundo"]
        listado
```

```
Out[4]: [10, 20, 30, 40, 50, 'Hola Mundo']
```

```
In [5]: listado[0]
```

```
Out[5]: 10
```

```
In [6]: listado[0], listado[-1]
```

```
Out[6]: (10, 'Hola Mundo')
```

```
In [7]: listado[-1]
```

```
Out[7]: 'Hola Mundo'
```

```
In [8]: len(listado)
```

```
Out[8]: 6
```

```
In [9]: for value in listado:  
        print(value)
```

```
10  
20  
30  
40  
50  
Hola Mundo
```

### **Estructura abreviada**

```
In [10]: listado1 = [10, 20, 30, 40]  
listado1
```

```
Out[10]: [10, 20, 30, 40]
```

```
In [11]: for numero in listado1:  
        print(numero)
```

```
10  
20  
30  
40
```

```
In [12]: # Paso 1: Comienza una lista []  
# Paso 2: Añade el primer valor de la listado1 --> [10]  
# Paso 3: Añade el segundo valor de la listado1 --> [10, 20]  
# Paso 4: Añade el tercer valor de la listado1 --> [10, 20, 30]  
# Paso 5: Añade el cuarto valor de la listado1 --> [10, 20, 30, 40]  
  
[numero for numero in listado1]
```

```
Out[12]: [10, 20, 30, 40]
```

```
In [13]: x = [numero for numero in listado1]  
x
```

```
Out[13]: [10, 20, 30, 40]
```

```
In [14]: # No es posible usarlo con tuplas crea un objeto:
```

```
(numero for numero in listado1)
```

Out[14]: <generator object <genexpr> at 0x7f945807add0>

### Creamos la lista vacia desde 0

```
In [15]: listado1
```

Out[15]: [10, 20, 30, 40]

```
In [16]: listado2 = []
for numero in listado1:
    listado2.append(numero)
listado2
```

Out[16]: [10, 20, 30, 40]

```
In [17]: max(listado2), min(listado2)
```

Out[17]: (40, 10)

```
In [ ]: # Calcular el minimo:

# minimo_lista = 1000000
# for para recorrer la lista
# if el numero < minimo_lista:
#     # el numero_minimo = numero
# if-else
# if - elif...elif -else
# minimo_lista --> 10
```

### Condicionales:

- si es esto haz .. sino haz...: if -else
- si es esto haz ... si es esto otro haz ... sino haz ..: if - elif - else

```
In [18]: for numero in listado2:
    if numero == 10:
        print('El valor es 10')
    elif numero == 20:
        print('El valor es 20')
    elif numero == 30:
        print('El valor es 30')
    else:
        print('El valor es 40')
```

El valor es 10  
El valor es 20  
El valor es 30  
El valor es 40

```
In [19]: # Apendizar:
listado2.append(1000)
listado2
```

Out[19]: [10, 20, 30, 40, 1000]

```
In [20]: # Modificar un valor, tendremos en cuenta la posición:  
listado2[1] = 50
```

```
In [21]: listado2
```

```
Out[21]: [10, 50, 30, 40, 1000]
```

```
In [22]: import numpy as np  
print(np.array(listado2))
```

```
[ 10  50  30  40 1000]
```

```
In [23]: x = [-2, -1, 0, 1, 2]  
  
# -2, -1, 0, 1, 2  
for n in x:  
    if n < 0:  
        print(f'El valor {n} es negativo')  
    else:  
        print(f'El valor {n} es positivo')
```

```
El valor -2 es negativo  
El valor -1 es negativo  
El valor 0 es positivo  
El valor 1 es positivo  
El valor 2 es positivo
```

```
In [24]: x = [-2, -1, 0, 1, 2]  
  
# -2, -1, 0, 1, 2  
for n in x:  
    if n < 0:  
        print(f'El valor {n} es negativo')  
    elif n == 0:  
        print(f'El valor {n} es neutro')  
    else:  
        print(f'El valor {n} es positivo')
```

```
El valor -2 es negativo  
El valor -1 es negativo  
El valor 0 es neutro  
El valor 1 es positivo  
El valor 2 es positivo
```

### Forma abreviada

```
In [36]: L = [1, 2, 3, 4, 5, 6, 7, 8, 9]  
L
```

```
Out[36]: [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [37]: for i in L:  
    print(i)
```

1  
2  
3  
4  
5  
6  
7  
8  
9

```
In [38]: listadoNew = [i for i in L]
listadoNew
```

```
Out[38]: [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [39]: # Modificamos un valor no se ve afectado el listado L
listadoNew[0] = 0
listadoNew
```

```
Out[39]: [0, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [40]: L
```

```
Out[40]: [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [41]: ##; OJO !
# Realiza una copia si uno se modifica (ListadoNew) se modifica el otro (
ListadoNew = L
```

```
In [42]: # Mezclar bucle con condicional
for numero in L:
    if numero < 5:
        print(numero) # 1, 2, 3, 4
    # else:
    # print(numero) # imprime 5, 6, 7, ...
```

1  
2  
3  
4

```
In [43]: print(L)
# Mezclar bucle con condicional
for numero in L:
    if numero < 5:
        print("*** Valores if: ", numero) # 1, 2, 3, 4
    else:
        print(numero) # imprime 5, 6, 7, ...
```

[1, 2, 3, 4, 5, 6, 7, 8, 9]

\*\*\* Valores if: 1  
\*\*\* Valores if: 2  
\*\*\* Valores if: 3  
\*\*\* Valores if: 4

5  
6  
7  
8  
9

```
In [44]: print(L)
# Mezclar bucle con condicional
for numero in L:
    if numero < 5:
        print("*** Valores if: ", numero) # 1, 2, 3, 4
        continue
    #else:
    print(numero) # imprime 5, 6, 7, ...
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9]
*** Valores if:  1
*** Valores if:  2
*** Valores if:  3
*** Valores if:  4
5
6
7
8
9
```

### Forma abreviada for y if

```
In [2]: L = [1, 2, 3, 4, 5, 6, 7, 8, 9]
L
```

```
Out[2]: [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [51]: listado3 = []
for numero in L:
    if numero < 5:
        print(numero)
        listado3.append(numero)
listado3
```

```
1
2
3
4
```

```
Out[51]: [1, 2, 3, 4]
```

```
In [47]: # Lista menores de 5
newList = [numero for numero in L if numero < 5]
newList
```

```
Out[47]: [1, 2, 3, 4]
```

```
In [48]: # Lista mayores de 5
superiores = [numero for numero in L if numero > 5]
superiores
```

```
Out[48]: [6, 7, 8, 9]
```

```
In [49]: x = [i for i in L if i < 5]
x
```

```
Out[49]: [1, 2, 3, 4]
```

## -2.2- While

```
In [52]: # Condición while
# Mientras se cumpla la condición el while esta funcionando,
# mientras que la i sea menor a 5 muestrame (print) los valores:
i = 1

while i < 5:
    print(i) # Print(1)... print(2)... print(3)...print(4)
    i = i + 1 # suma 1 valor pasa a tener valor 2 ... pasa a valor 3... p
# llega al valor 5 se para

1
2
3
4
```

```
In [53]: # Condición while
# Mientras se cumpla la condición el while esta funcionando,
# mientras que la i sea menor a 5 muestrame (print) los valores:
i = 1
data = []

while i < 5:
    print(i) # Print(1)... print(2)... print(3)...print(4)
    data.append(i)
    i = i + 1 # suma 1 valor pasa a tener valor 2 ... pasa a valor 3... p

# llega al valor 5 se para
data

1
2
3
4
```

Out[53]: [1, 2, 3, 4]

```
In [1]: # While true actividad continua...
# n = 0
# while True:
#     n += 1
#     print(f'El valor de n es {n}')
```

## Indentación

```
In [54]: # Error en posición del código:
for i in L:
print(i)
```

```
Cell In[54], line 3
    print(i)
    ^
IndentationError: expected an indented block
```

```
In [55]: L = [1, 2, 3, 4, 5]
```

```
L
```

```
Out[55]: [1, 2, 3, 4, 5]
```

```
In [56]: # Indentación con 4 golpes de la barra espaciadora  
for i in L:  
    print(i)
```

```
1  
2  
3  
4  
5
```

```
In [57]: # Indentación con un golpe de tabulador  
for i in L:  
    print(i)
```

```
1  
2  
3  
4  
5
```

```
In [58]: # Indentación con detección de python simplemente  
# nos detecta en que posición debe ir el código y pulsamos enter  
for i in L:  
    print(i)
```

```
1  
2  
3  
4  
5
```

*Creado por:*

*Isabel Maniega*