

```
<!-- INF 129 | Paralelo 1 -->
```

# Ayudantía()

```
<Por="Mauricio Fernández"/>
```

}



# Contenidos

01

Listas

02

¿Mutable?

03

Operadores

04

Métodos

05

Errores Clásicos

# Listas{

- Es un conjunto ordenado de elementos que se pueden guardar en una variable

```
lista1 = ["Manzana", 4, True]
lista2 = list(["Manzana", 4, True])

lista3 = []
lista4 = list()
```

}

## Operador Suma (+){

- El operador suma junta los elementos de la lista 1 y la lista 2 en una única lista.

¡IMPORTA EL ORDEN!

```
[8] lista1 = ["Manzana", 4, False]
    lista2 = ["Pera", 2, True]

[9] lista1 + lista2
... ['Manzana', 4, False, 'Pera', 2, True]

[13] lista2 + lista1
... ['Pera', 2, True, 'Manzana', 4, False]
```

}

## Operador producto (\*) {

- El operador producto "copia" una misma lista n veces en una única lista

```
[16] lista1 = ["Manzana", 4, False]
```

```
[17] lista1 * 3
```

```
... ['Manzana', 4, False, 'Manzana', 4, False, 'Manzana', 4, False]
```

```
[18] lista1 + lista1 + lista1
```

```
... ['Manzana', 4, False, 'Manzana', 4, False, 'Manzana', 4, False]
```

}

# Operador in y del{

- El operador in verifica si un elemento esta dentro de una lista

```
[ ] lista1 = ["Manzana", 4, False]
```

```
[19] "Manzana" in lista1
```

```
... True
```

```
[20] False in lista1
```

```
... True
```

```
[21] "Pera" in lista1
```

```
... False
```

- El operador del elimina un elemento de una lista por indice

```
[23] lista1 = ["Manzana", 4, False]  
del lista1[1]
```

```
[24] lista1
```

```
... ['Manzana', False]
```

```
}
```

## Funciones Útiles {

len()

Retorna la cantidad de elementos de una lista

sum()

Suma todos los elementos de una lista y retorna la suma

min()

Obtiene el menor elemento de la lista

max()

Obtiene el mayor elemento de una lista

}

Métodos de listas{

A diferencia de una función, llamamos métodos a aquellas acciones que se aplican desde "dentro" de la lista

}



```
lista1.append(x) {
```

- .append(x) se encarga de agregar al final de la lista un elemento x a la lista

```
28]         lista1 = ["Manzana", 4, False]
           lista1.append(2)

29]         lista1

..         ['Manzana', 4, False, 2]
```

```
}
```

```
lista1.count(x) {
```

- .count(x) se encarga de contar cuántas veces aparece un elemento x dentro de la lista

```
lista1 = ["Manzana", "Pera", "Frutilla", "Frutilla", "Manzana"]
lista1.count("Manzana")

[30]
... 2
```

```
}
```

```
lista1.index(x) {
```

- .index(x) retorna el índice en donde se encuentra el elemento x. Si x se encuentra varias veces, retorna el primer índice en donde se encuentra

```
[31] lista1 = ["Manzana", "Frutilla", "Manzana"]
      lista1.index("Manzana")
...  0

[32] lista1.index("Pera")
...  -----
      ValueError                                Traceback (most recent call last)
      Cell In[32], line 1
      ----> 1 lista1.index("Pera")

      ValueError: 'Pera' is not in list
```

```
}
```

```
lista1.remove(x) {
```

- .remove(x) elimina la primera aparición de x en la lista

```
lista1 = ["Manzana", "Frutilla", "Manzana"]
lista1.remove("Manzana")

[34] lista1
... ['Frutilla', 'Manzana']

▶ ▾ lista1.remove("Pera")
[35]
... -----
ValueError                                Traceback (most recent call last)
Cell In[35], line 1
----> 1 lista1.remove("Pera")

ValueError: list.remove(x): x not in list
```

```
}
```

```
lista1.reverse() {
```

- `.reverse()` invierte el orden de los elementos de la lista (modifican la original)

```
lista1 = ["Manzana", "Frutilla", "Pera"]
[36] lista1
... ['Manzana', 'Frutilla', 'Pera']

lista1.reverse()
[37] lista1
... ['Pera', 'Frutilla', 'Manzana']
```

```
}
```

```
lista1.sort() {
```

- .sort() ordena los elementos de la lista de forma ascendente.
- Si quisieramos un orden descendente lo indicamos como en el ejemplo

```
lista1 = [3, 2, 4, 7, 9]
lista1.sort()
lista1
[40]
... [2, 3, 4, 7, 9]

lista1.sort(reverse=True)
lista1
[41]
... [9, 7, 4, 3, 2]
```

```
}
```

```
lista1.insert(i, x) {
```

- .insert(i, x) inserta el elemento x en la posición i de la lista

```
lista1 = [1, 2, 3, 4, 6]
lista1.insert(4, 5)
lista1
[50]
... [1, 2, 3, 4, 5, 6]

lista1.insert(6, 7)
lista1
[51]
... [1, 2, 3, 4, 5, 6, 7]

▷ ▾ lista1.insert(8, 9) #Ojopiojo
[52]
... [1, 2, 3, 4, 5, 6, 7, 9]
```

```
}
```

## Recorrido Listas {

```
lista1 = ["Manzana", 1, True]

[54] for elemento in lista1:
      print(elemento)
```

```
... Manzana
    1
    True
```

```
lista2 = [3, 1, 2, 5]

[55] for num in lista2:
      print(num)
```

```
... 3
    1
    2
    5
```

```
lista1 = ["Manzana", 1, True]
i = 0
```

```
[56] while i < len(lista1):
      print(lista1[i])
      i += 1
```

```
... Manzana
    1
    True
```

}



## Copia de Listas{

```
#Caso 1
lista1 = [2, 5, 8]
lista2 = lista1 #Copia de referencia (memoria)
[59] lista2

... [2, 5, 8]

lista2.append(4)
[60] lista1

... [2, 5, 8, 4]
```

```
#Caso 2
lista1 = [2, 5, 8]
lista2 = list(lista1) #Copia
[61] lista2

... [2, 5, 8]

lista2.append(4)
[62] lista1

... [2, 5, 8]
```

}

## Referencia de listas en funciones{

```
def cambio_divisa(precios_dolares):  
    for i in range(len(precios_dolares)):  
        precios_dolares[i] *= 1000  
  
    return precios_dolares  
  
precios_dolares = [2, 5, 10, 0.5]  
precios_clp = cambio_divisa(precios_dolares)
```

```
[70]      precios_clp  
...      [2000, 5000, 10000, 500.0]  
  
[71]      precios_dolares  
...      [2000, 5000, 10000, 500.0]  
  
[72]      precios_clp[0] = 4000  
          precios_dolares  
...      [4000, 5000, 10000, 500.0]
```

}

## Referencia de listas en funciones{

```
def cambio_divisa(precios_dolares):  
    for i in range(len(precios_dolares)):  
        precios_dolares[i] *= 1000  
  
    return precios_dolares  
  
precios_dolares = [2, 5, 10, 0.5]  
precios_clp = cambio_divisa(list(precios_dolares))
```

```
[89]     precios_clp  
...     [2000, 5000, 10000, 500.0]
```

```
[90]     precios_dolares  
...     [2, 5, 10, 0.5]
```

```
         precios_clp[0] = 4000  
[91]     precios_clp  
...     [4000, 5000, 10000, 500.0]
```

```
[92]     precios_dolares  
...     [2, 5, 10, 0.5]
```

}

# Listas de listas {

```
lista1 = ["Manzana", 3, False], ["Pera", 1, False], ["Manzana", 2, True], "hola"]
```

```
for elemento in lista1:  
    print(elemento)
```

```
['Manzana', 3, False]  
['Pera', 1, False]  
['Manzana', 2, True]  
hola
```

```
lista1 = ["Manzana", 3, False], ["Pera", 1, False]
```

```
for stock in lista1:  
    print("Producto: ")  
    for elemento in stock:  
        print(elemento)  
  
    print("\n\n")
```

```
Producto:  
Manzana  
3  
False
```

```
Producto:  
Pera  
1  
False
```

```
}
```

KAHOOT {

}

```
<!-- INF 129 | Paralelo 1 -->
```

Gracias {

```
<Por="Mauricio Fernández"/>
```

}