

# Ayudantia2

September 30, 2022

## 1 Ayudantía 2 29/09/2022

While (Mientras) tiene la gracia que recibe una condición y dependiendo de esa condición se repite un bloque de código

```
[1]: i = 0
    lista = [3,4,6,2]
    while i < len(lista):
        print("Hola")
        i += 1
```

Hola  
Hola  
Hola  
Hola

```
[2]: lista_nombres = ["Diego", "Vicente", "Javiera", "John"]
    i = 0
    while i < len(lista_nombres):
        elemento_actual = lista_nombres[i]
        j = 0
        while j < len(elemento_actual):
            if(elemento_actual[j] == "i"):
                print("Encontre la letra i")
            j += 1
        i += 1
    # ESTO NO ES PARTE DEL CICLO WHILE
```

Encontre la letra i  
Encontre la letra i  
Encontre la letra i

```
[3]: # Transformar elementos de una lista de String a Integers
    lista_strings = ["2", "3", "4"]

    i = 0
    while i < len(lista_strings):
        elemento_actual = type(int(lista_strings[i]))
```

```

    print(elemento_actual)
    i += 1
print("-----")
i = 0
while i < len(lista_strings):
    elemento_actual = type(lista_strings[i])
    print(elemento_actual)
    i += 1

```

```

<class 'int'>
<class 'int'>
<class 'int'>
-----
<class 'str'>
<class 'str'>
<class 'str'>

```

Podemos utilizar el método .split(“algo”) para separar un string y convertir los elementos en una lista. “algo” debe ser un string

```

[4]: strings = "Hola,que,tal,tu,dia"
    lista_elementos = strings.split(",")
    print(lista_elementos)

```

```
['Hola', 'que', 'tal', 'tu', 'dia']
```

Metodos que mañana podrian prohibir:

- max()
- min()
- reverse()
- sort()

```

[5]: # append

lista = []
lista.append("John")
lista.append("Camila")

posicion = lista.index("Camila")
print(lista)
print(posicion)

```

```
['John', 'Camila']
```

```
1
```

- append(): Agrega un elemento a la lista, agregandolo al final de la lista. Dentro de los parentesis debe colocarse el elemento a agregar

- `index()`: Obtiene la posición de un elemento de la lista. Dentro de los parentesis debe colocarse el elemento a buscar

`eval()`: Eval permite que Python “evalúe” el formato de la entrada para asignar un tipo de dato. Permite ahorrar problemas al utilizar `list` junto con `input` para ingresar una lista por entrada

```
[6]: entrada = eval(input("Ingrese algo: "))
      print(type(entrada))
      print(entrada)
```

```
Ingrese algo: [3,4,2]
<class 'list'>
[3, 4, 2]
```

```
[7]: entrada = list(input("Ingrese nuevamente algo: "))
      print(type(entrada))
      print(entrada)
```

```
Ingrese nuevamente algo: [3,4,2]
<class 'list'>
['[', '3', ',', '4', ',', '2', ']']
```

```
[8]: acumulador = 0
      lista_numeros = [3,4,6,2]
      i = 0
      while i < len(lista_numeros):
          acumulador = acumulador + lista_numeros[i]
          i += 1
      print(acumulador)
```

```
15
```

```
[9]: lista = [3,4,2,1]

      print(lista[0]) # 3
      print(lista[1]) # 4
      print(lista[2]) # 2
      print(lista[3]) # 1
```

```
3
4
2
1
```

## 2 Malas practicas

- `while(True)`
- `break`

[ ]: