



Temas:

- **Bucle while**
- **Contadores y acumuladores**
- **Bucle for**
- **.....**



Bucle While

En Python el bucle while se utiliza para ejecutar un bloque de código mientras se cumpla una condición.

Por ejemplo si quisiéramos contar de 1 hasta 5 podríamos escribir el siguiente programa:

```
contador = 1
while contador <= 5: #Mientras el contador sea menor o igual a 5
    print("Contador:", contador) #Muestro el valor del contador
    contador += 1 #Actualizo el valor del contador sumando 1
```

Por pantalla veremos lo siguiente:

```
Contador: 1
Contador: 2
Contador: 3
Contador: 4
Contador: 5
```



Bucle While

Tener en cuenta lo siguiente:

- La **condición** se evalúa en cada **iteración**
- Si la **condición** nunca se vuelve falsa, el **bucle** puede ser **infinito**
- Se puede utilizar **break** para salir del bucle o **continue** para saltar a la siguiente **iteración**
- Podemos agregar un bloque de código cuando finalicen las iteraciones utilizando la cláusula **else**

Bucle While - Break

Es una instrucción que se utiliza para cortar la ejecución del bucle.

Ejemplo:

```
contador = 1
while contador <= 5: #Mientras el contador sea menor o igual a 5
    print("Contador:", contador) #Muestro el valor del contador
    contador += 1 #Actualizo el valor del contador sumando 1
    if contador == 3: #Si el contador es igual a 3
        print("¡finalizó el bucle!")
        break #Se corta el bucle
```

Salida por pantalla:

```
Contador: 1
Contador: 2
¡finalizó el bucle!
```



Bucle While - Continue

Es una instrucción que se utiliza para “saltar” a la siguiente iteración. **Ejemplo:**

```
contador = 1
while contador <= 5: #Mientras el contador sea menor o igual a 5
    if contador == 3: #Si el contador es igual a 3 no lo muestro
        print("¡Se salto la iteración!")
        contador += 1
        continue #Finalizo la iteración, pero no el bucle
    else: #Si el contador no es igual a 3 entonces lo muestro
        print("Contador:", contador)
        contador += 1 #Actualizo el valor del contador sumando 1
```

Salida por pantalla:

```
Contador: 1
Contador: 2
¡Se salto la iteración!
Contador: 4
Contador: 5
```

Bucle While - Else

También podremos ejecutar un bloque de código cuando finalicen las iteraciones utilizando la cláusula else.

Ejemplo:

```
contador = 1
while contador <= 5: #Mientras el contador sea menor o igual a 5
    print("Contador:", contador) #Muestro el valor del contador
    contador += 1 #Actualizo el valor del contador sumando 1
else: #Si el contador no es menor o igual a 5 entonces el bucle finaliza
    print("El bucle while finalizó")
```

Salida por pantalla:

```
Contador: 1
Contador: 2
Contador: 3
Contador: 4
Contador: 5
El bucle while finalizó
```



Contadores y Acumuladores

Contador:

- Se usa para contar cuántas veces ocurre algo.
- En general se incrementa en cada iteración.
- Ejemplo típico: contador += 1

Acumulador:

- Se usa para sumar (o acumular) valores en cada iteración.
- Ejemplo típico: acumulador += valor

¿Por qué se usan con bucles?

- Porque necesitamos llevar un control de lo que sucede en cada iteración
- Para contar las iteraciones de un bucle
- Para almacenar valores que se ingresan o se calculan
- Calcular promedios, totales, estadísticas, etc.



Contadores y Acumuladores

Veamos un ejemplo donde un programa cuenta los números desde 1 hasta 5 y luego muestra por pantalla la suma de todos ellos.

```
contador = 1
acumulador = 0
while contador <= 5:
    print("Contador:", contador)
    acumulador += contador # 1 + 2 + 3 + 4 + 5 = 15
    contador += 1
else:
    print("La suma de todos los valores es: ", acumulador)
```

```
Contador: 1
Contador: 2
Contador: 3
Contador: 4
Contador: 5
La suma de todos los valores es: 15
```