

## Bucles

### 1. Bucles condicionales (**while**)

```
while condición:
    bloque código
```

Repite la ejecución del bloque de código mientras la expresión lógica condición sea cierta.

Se puede interrumpir en cualquier momento la ejecución del bloque de código con la instrucción `break`.

*El bloque de código debe estar indentado por 4 espacios.*

---

**Ejercicio de Inducción:** Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> # Pregunta al usuario por un número hasta que introduce 0.
>>> num = None
>>> while num != 0:
...     num = int(input('Introduce un número: '))
...
Introduce un número: 2
Introduce un número: 1
Introduce un número: 0
>>>
```

---

Alternativa:

---

**Ejercicio de Inducción:** Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> # Pregunta al usuario por un número hasta que introduce 0.
>>> while True:
...     num = int(input('Introduce un número: '))
...     if num == 0:
...         break
...
Introduce un número: 2
Introduce un número: 1
Introduce un número: 0
>>>
```

---

### 2. Bucles iterativos (**for**)

```
for i in secuencia:
    bloque código
```

Repite la ejecución del bloque de código para cada elemento de la secuencia `secuencia`, asignado dicho elemento a `i` en cada repetición.

Se puede interrumpir en cualquier momento la ejecución del bloque de código con la instrucción `break` o saltar la ejecución para un determinado elemento de la secuencia con la instrucción `continue`.

*El bloque de código debe estar indentado por 4 espacios.*

Se utiliza fundamentalmente para recorrer colecciones de objetos como cadenas, listas, tuplas o diccionarios.

A menudo se usan con la instrucción `range`:

- `range(fin)`: Genera una secuencia de números enteros desde 0 hasta `fin-1`.
- `range(inicio, fin, salto)`: Genera una secuencia de números enteros desde `inicio` hasta `fin-1` con un incremento de `salto`.

---

**Ejercicio de Inducción:** Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> palabra = 'Python'
>>> for letra in palabra:
...     print(letra)
...
P
y
t
h
o
n
```

---

**Ejercicio de Inducción:** Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
...     print(i, end=" ", " ")
...
1, 3, 5, 7, 9, >>>
```

---