

CURSO DE PYTHON DESDE CERO



🤔 ¿QUÉ ES EL CICLO FOR?

El ciclo for en Python se usa para recorrer elementos de una secuencia (lista, cadena, tupla, diccionario, rango, etc.) y ejecutar un bloque de código una vez por cada elemento.



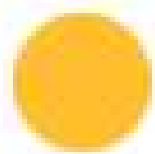
🤔 ¿QUÉ ES EL CICLO FOR?

A diferencia de otros lenguajes, en Python el for no está basado en índices, sino en iterar sobre elementos.





SINTAXIS DEL FOR



```
1  for variable in secuencia:  
2      # código a ejecutar por cada elemento
```

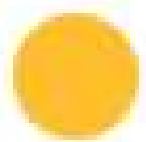
DONDE:

- variable → toma el valor del elemento actual
- secuencia → lo que vas a recorrer
(lista, string, range, etc.)



EJEMPLOS PRÁCTICOS

Ejemplo 1: Recorrer una lista

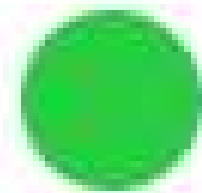
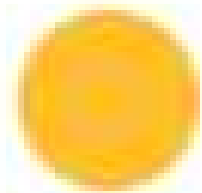
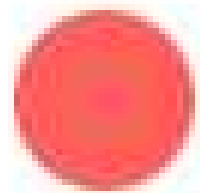


```
1  frutas = ["manzana", "pera", "uva"]  
2  
3  for fruta in frutas:  
4      print(fruta)
```



✓ EJEMPLOS PRÁCTICOS

Ejemplo 2: Recorrer una cadena



```
1 for letra in "Python":  
2     print(letra)
```



✓ EJEMPLOS PRÁCTICOS

Ejemplo 3: Usar range() para recorrer números
range() genera una secuencia de números.

```
1  for i in range(5):  
2      print(i)  
3  # Imprime: 0, 1, 2, 3, 4
```




✓ EJEMPLOS PRÁCTICOS

Ejemplo 4: Range con inicio y fin

```
1  for i in range(1, 6):  
2      print(i)  
3  # Imprime: 1, 2, 3, 4, 5
```



✓ EJEMPLOS PRÁCTICOS

Ejemplo 5: Range con salto

```
1  for i in range(0, 10, 2):  
2      print(i)  
3  # Imprime pares del 0 al 8.
```



EJEMPLOS PRÁCTICOS

Ejemplo 6: Recorrer un diccionario

```
1  persona = {"nombre": "Ana", "edad": 25}
2
3  for clave, valor in persona.items():
4      print(clave, valor)
```



EJEMPLOS PRÁCTICOS

Ejemplo 7: Uso de break para detener el ciclo

```
1  for numero in range(10):  
2      if numero == 5:  
3          break  
4      print(numero)
```




EJEMPLOS PRÁCTICOS

Ejemplo 8: Uso de continue para saltar una iteración

```
1  for i in range(5):  
2      if i == 2:  
3          continue  
4      print(i)  
5  # Imprime: 0, 1, 3, 4
```



EJEMPLOS PRÁCTICOS

Ejemplo 9: Ciclo for con else
El else se ejecuta solo si no hubo break.

```
1  for i in range(3):  
2      print(i)  
3  else:  
4      print("Ciclo completado")
```

BUENAS PRÁCTICAS DE FOR

- ✓ Usa for cuando sí sabes cuántas veces quieres repetir una acción.
- ✓ Usa nombres de variables significativos cuando recorres elementos.
- ✓ Evita modificar la lista mientras la recorres (puede causar errores).



RESUMEN DEL CICLO FOR

`for variable in secuencia:` Recorre secuencias elemento por elemento

`range():` Genera números para iterar

`break:` Rompe el ciclo

`continue:` Salta a la siguiente vuelta

**NOS VEMOS EN UN PRÓXIMO
VIDEO DE ESTE CURSO,
SALUDOS 🚀**

