



Ejercicios para practicar ciclos en Python

Ejercicio 1: Contador de positivos

Plantea un programa que permita ingresar 10 números y cuente cuántos de ellos son mayores que cero.

Ejercicio 2: Sumatoria hasta alcanzar un mínimo

Solicita al usuario que ingrese números. Sigue sumándolos hasta que la suma sea mayor a 100. Usa un `while`.

Ejercicio 3: Generador de tabla de multiplicar

Pide al usuario un número y muestra su tabla de multiplicar del 1 al 10 utilizando `for` y `range()`.

Ejercicio 4: Contador de intentos

Diseña un programa donde el usuario debe adivinar un número secreto entre 1 y 10. Límitale a un máximo de 5 intentos usando `while`.

Ejercicio 5: Verificador de edad

Crea un menú interactivo (`while True`) donde se pregunte continuamente la edad del usuario hasta que una edad mayor o igual a 18 sea ingresada. Usa `break` adecuadamente.

Ejercicio 6: Conteo de números pares en un rango

Pide dos números al usuario (inicio y fin) y cuenta cuántos números pares hay entre ellos usando `for` y `if`.

Ejercicio 7: Contador de letras 'a'

Solicita una palabra al usuario y cuenta cuántas veces aparece la letra 'a'. Utiliza un `for` para recorrer la cadena.



Ejercicio 8: Calculadora de factorial

Pide un número al usuario y calcula su factorial usando un ciclo `for`.

Ejercicio 9: Sumar números hasta cancelar

Permite al usuario ingresar números de manera indefinida. Usa un `while True` que termina si el usuario escribe "salir", acumulando los números ingresados.

Ejercicio 10: Cuenta regresiva personalizada

Solicita al usuario un número inicial y muestra una cuenta regresiva hasta llegar a 0 usando `for` con `range(inicio, -1, -1)`.

Ejercicio 11: Contar números positivos

Crea un programa que pida al usuario ingresar 5 números. El programa debe contar cuántos de esos números son positivos.

- Resolver primero con `while`.
- Resolver luego con `for`.

Ejercicio 12: Mostrar los múltiplos de 3

Crea un programa que muestre todos los múltiplos de 3 entre 1 y 30.

- Resolver primero con `while`.
- Resolver luego con `for` con `range()`.