



Ejercicio 1: Conteo condicional en lista

Dada la siguiente lista:

```
numeros = [5, -3, 12, 0, 15, 8, 22, -10, 33, 11, 19, -7]
```

Cuenta cuántos **números positivos mayores que 10** hay y cuántos son **negativos múltiplos de 3**.

Muestra ambos totales. Al final, calcula el porcentaje que representa cada grupo respecto al total de elementos.

Ejercicio 2: Suma condicionada hasta número especial

Solicita al usuario que introduzca números enteros positivos uno por uno. Suma los valores que sean **mayores que 5 y múltiplos de 3**. Detén el ciclo cuando el usuario introduzca un número **negativo par** o cuando se hayan introducido más de 10 números válidos. Muestra la suma acumulada.

Ejercicio 3: Múltiplos excluidos y conteo

Recorre los números del 1 al 100.

- **Suma** los múltiplos de 5.
 - Usa `continue` para omitir los demás.
 - Muestra cuántos se sumaron y cuál fue el total.
-

Ejercicio 4: Verificación y creación de usuarios

Dada la tupla de usuarios:

```
usuarios_registrados = ("ana", "luis", "maria", "carlos")
```

Y la lista:

```
usuarios_nuevos = ["pedro", "ana", "maria", "sofia", "lucas", "Luis"]
```

recorre ambas estructuras e imprime los nombres de usuario que se repiten



Ejercicio 5: Filtrado de números con restricción

Imprime los números del 1 al 100, **omitiendo** aquellos que:

- Contengan el dígito 7, **y**
- Sean múltiplos de 3.

(Pista: Convierte el número a str y usa continue si cumple ambas condiciones)

Ejercicio 6: Conteo por categoría con validación

Dada una lista de productos categorizados:

```
categorias = ["ropa", "electro", "error1", "alimento", "otros", "electro", "ropa", "error",  
"ropa", "Electro", "ALIMENTO", "error2", "aeronauticos"]
```

- Crea un conteo por categoría (ropa, electro, alimento, otros) sin distinguir mayúsculas/minúsculas.
 - Ignora elementos inválidos (los que contienen la palabra error) usando continue.
 - Muestra el total de elementos de la lista excluyendo el elemento error.
-

Ejercicio 7: Login con intentos y advertencias

Permite que un usuario introduzca una contraseña. Tiene **3 intentos**.

- Si acierta, muestra "Acceso concedido" y detén el bucle.
- Si falla 2 veces, muestra un mensaje de advertencia.
- Si falla 3 veces, muestra "Acceso denegado, su cuenta ha sido bloqueada".



Ejercicio 8: Números primos con control de cantidad

Solicita un número n al usuario. Busca los **primeros 10 números primos menores que n** , usando una combinación de `for` y `while`.

- Si $n < 10$, muestra un mensaje de error y termina.
- Muestra la lista de primos encontrados.

Ejercicio 9: Clasificación de palabras complejas

Dada una lista de palabras:

```
palabras = ["casa", "ordenador", "mesa", "código123", "sol", "ventilador", "mouse", "teclado", "pantalla7"]
```

- Ignora palabras que contengan números (continue).
- Las que tienen **más de 6 letras** colócalas imprimelas.
- Imprime la cantidad total de palabras que contienen más de 6 letras.

Ejercicio 10: Verificador de palíndromos

Escribe un programa que reciba una palabra e indique si es o no un palíndromo.

- ✓ Solo puedes usar un único ciclo, ya sea `for` o `while`, para comparar letra por letra.

Una palabra palíndroma es aquella que se lee igual de izquierda a derecha que de derecha a izquierda.

Por ejemplo:

-  "reconocer" es un palíndromo porque al invertirla se obtiene "reconocer".

Condiciones: Solo se podrá utilizar `for` o `while`, cadenas, condicionales, lógica booleana, no se puede utilizar funciones de Python que inviertan cadenas.