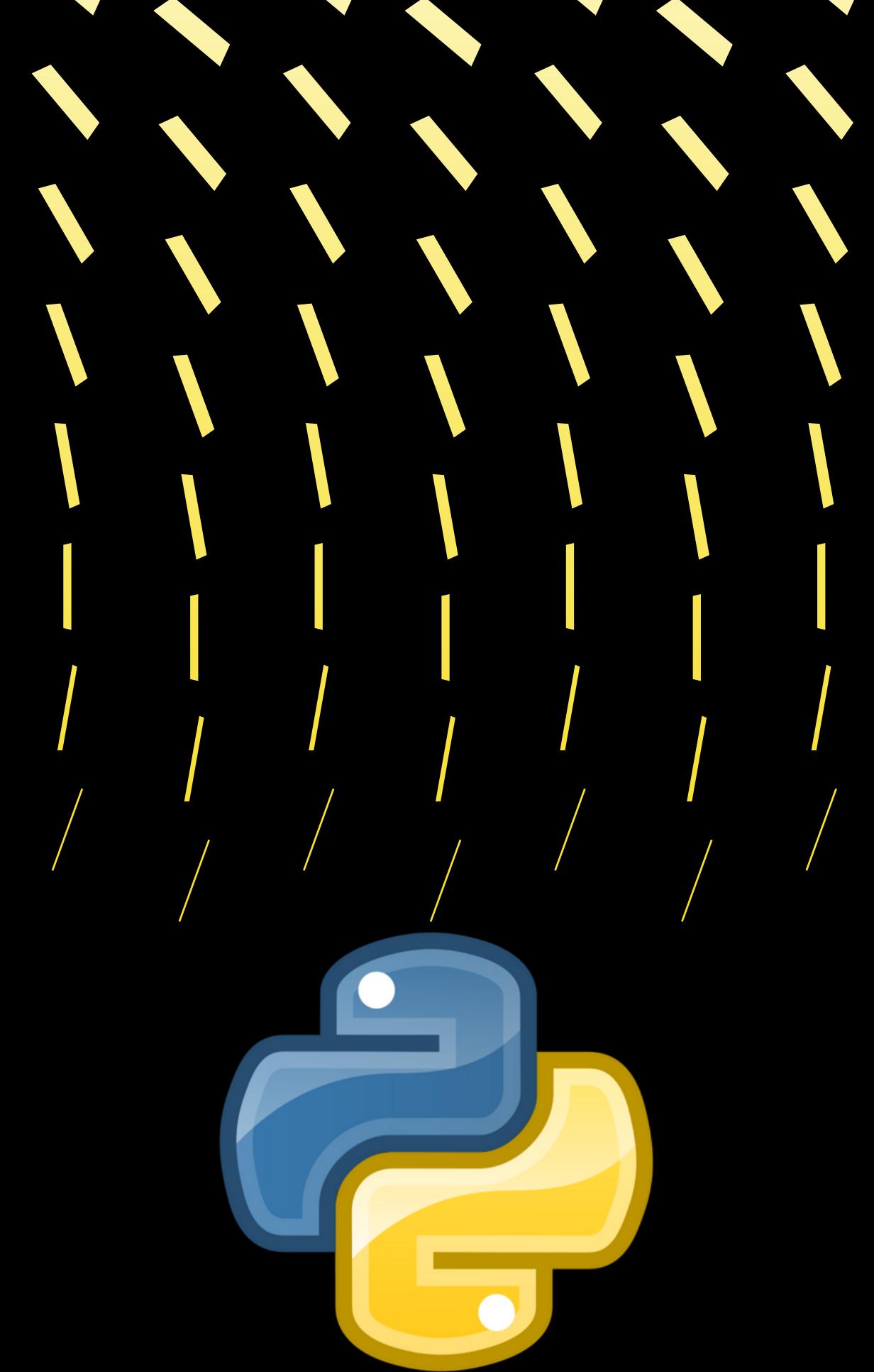


Python

Lenguaje de
programación de alto
nivel

Instructora: Kattherine Hernandez



Día 5: Estructuras condicionales

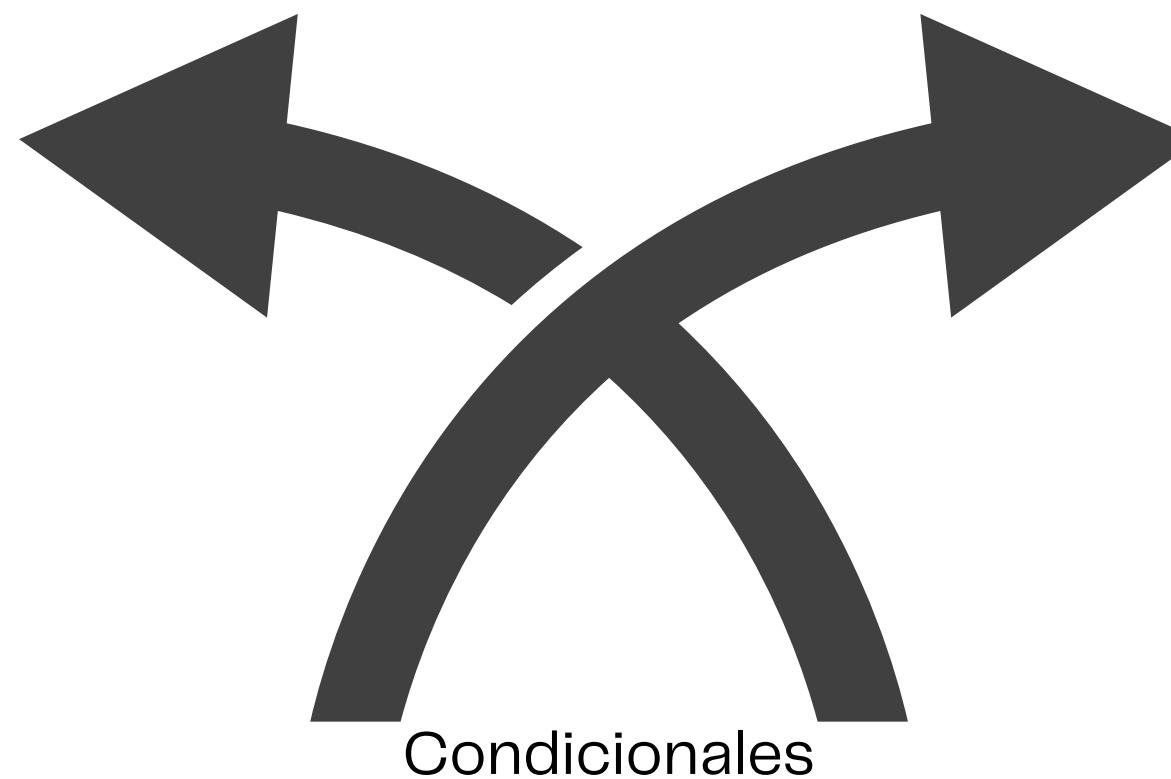


Instr: Kattherine Hernandez

Estructuras de control

Son herramientas fundamentales en cualquier lenguaje de programación, incluyendo Python, que permiten dirigir el flujo de ejecución de un programa.

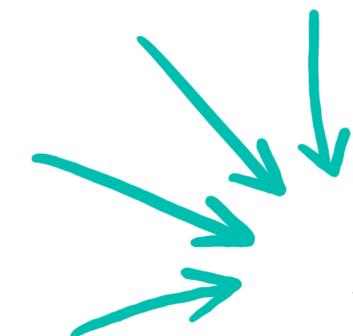
Estas estructuras determinan cómo se ejecutan las instrucciones en función de ciertas condiciones o repeticiones.



Bucles

Condicionales

Estructuras condicionales



Sintaxis

```
if expresión_condicional:  
    ejecutar_esto_si_cumple  
  
if expresión_condicional:  
    ejecutar_esto_si_cumple  
else:  
    ejecutar_esto_si_no_se_cumple  
  
if expresión_condicional:  
    ejecutar_esto_si_cumple  
elif expresión_condicional:  
    ejecutar_esto_si_se_cumple  
else:  
    ejecutar_esto_si_no_se_cumple
```

Estas permiten tomar decisiones basadas en condiciones. Según si una condición es verdadera o falsa, el programa ejecutará un bloque de código u otro.

Ejercicio de la semana

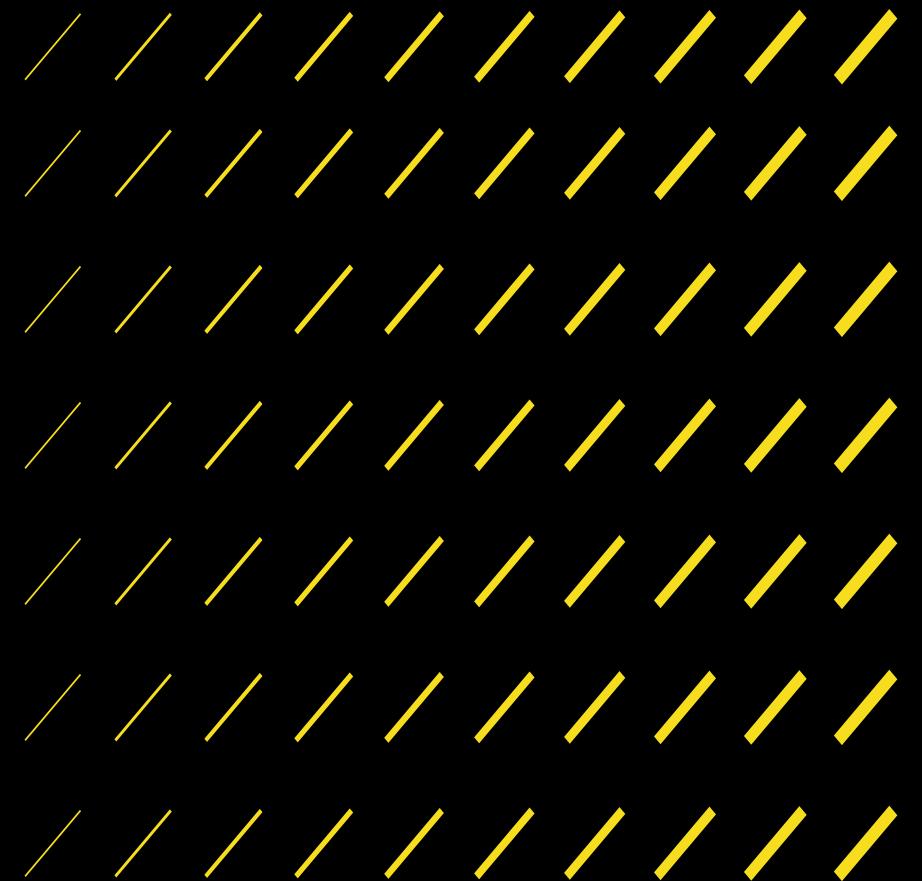
Crea una calculadora que realice operaciones básicas (suma, resta, multiplicación, división). El usuario podrá realizar múltiples cálculos hasta que decida terminar. Al finalizar, mostrará un mensaje con el total de operaciones realizadas.

Requisitos:

- Solicitar al usuario dos números y el tipo de operación (suma, resta, multiplicación o división).
- Realizar la operación correspondiente usando funciones.
- Repetir el proceso tantas veces como el usuario quiera utilizando un bucle.
- Al finalizar, mostrar el total de operaciones realizadas con un mensaje al usuario

</>

Día 6: Bucles for



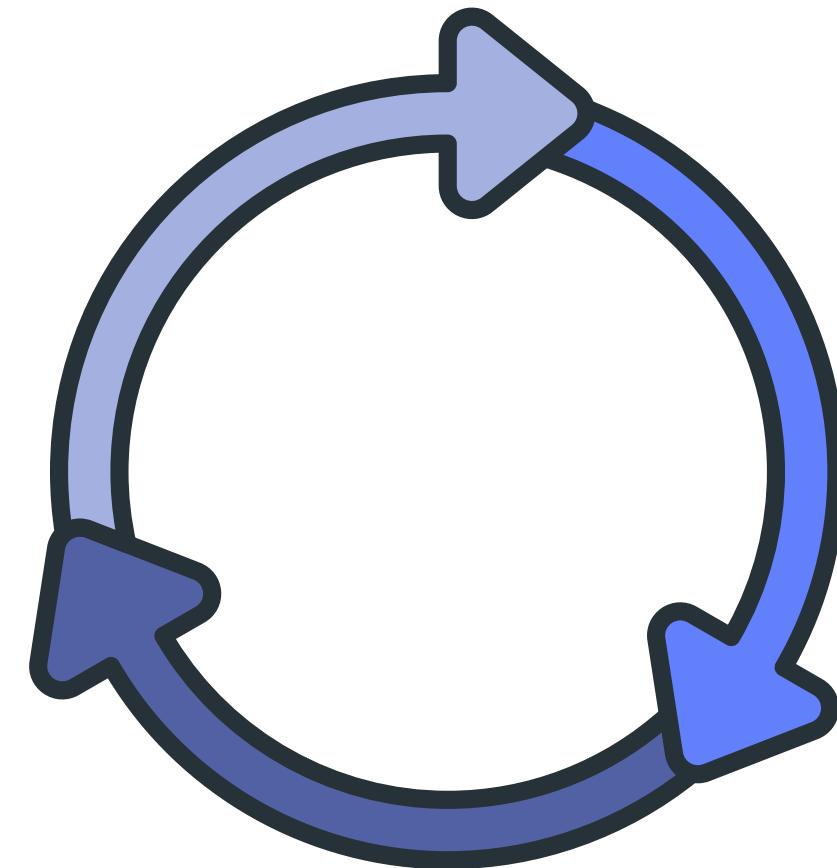
Instr: Kattherine Hernandez

Bucles

son estructuras de control que permiten repetir un bloque de código varias veces, ya sea mientras se cumpla una condición específica o para recorrer elementos de una secuencia como una lista, cadena o rango.

Son esenciales para automatizar tareas repetitivas y hacer los programas más eficientes.

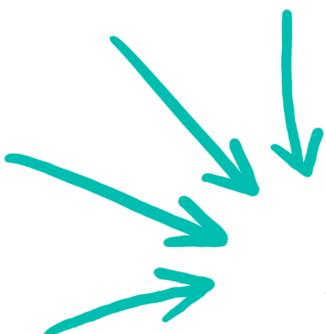
Bucles **for** y **while**.



Bucles for

Se utiliza para iterar sobre una secuencia
(listas, cadenas, tuplas, conjuntos,
diccionarios o rangos)

Con cada iteración, se obtiene un
elemento de la secuencia



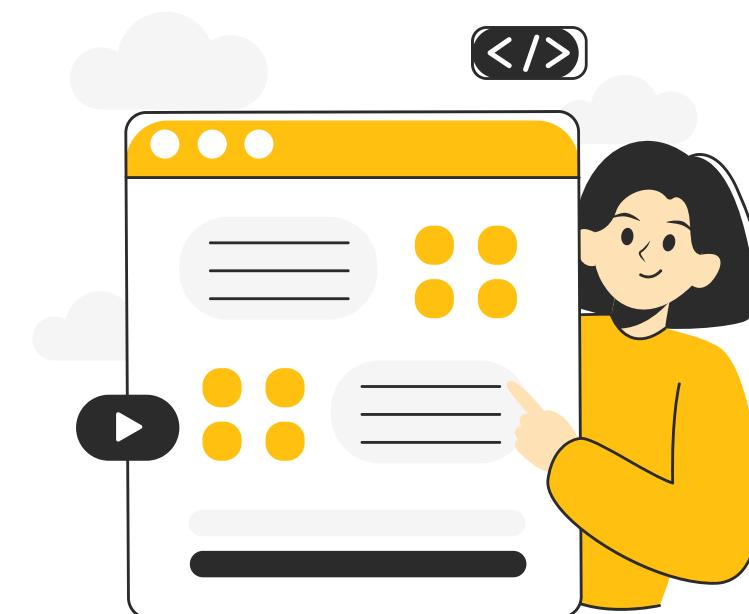
Sintaxis

```
for i in range(rango):  
    # cualquier_cosa()  
    pass
```

f-strings

Las f-strings (formatted string literals) permiten insertar expresiones dentro de cadenas utilizando llaves {} y un prefijo f antes de la cadena.

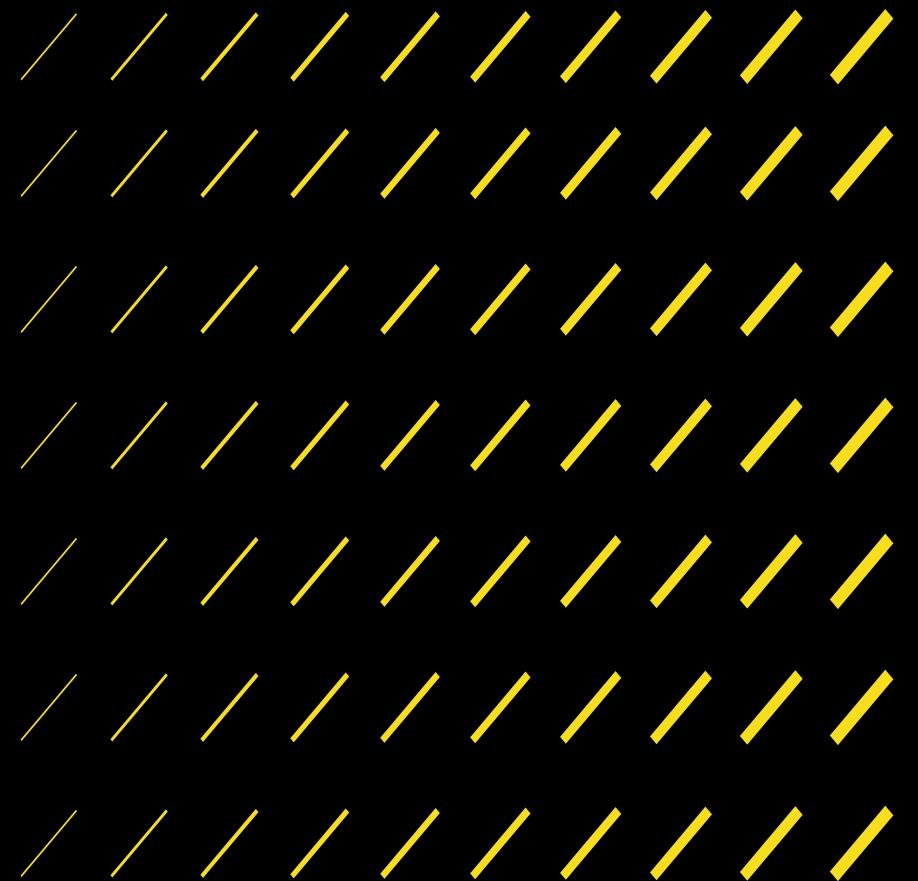
Esto facilita la interpolación de variables y expresiones directamente en el texto, haciendo que sea más limpio y legible.



```
variable = "Python"
print(f"Estoy aprendiendo {variable}")
# Salida: Estoy aprendiendo Python
```

</>

Día 7: Bucles while



Instr: Kattherine Hernandez

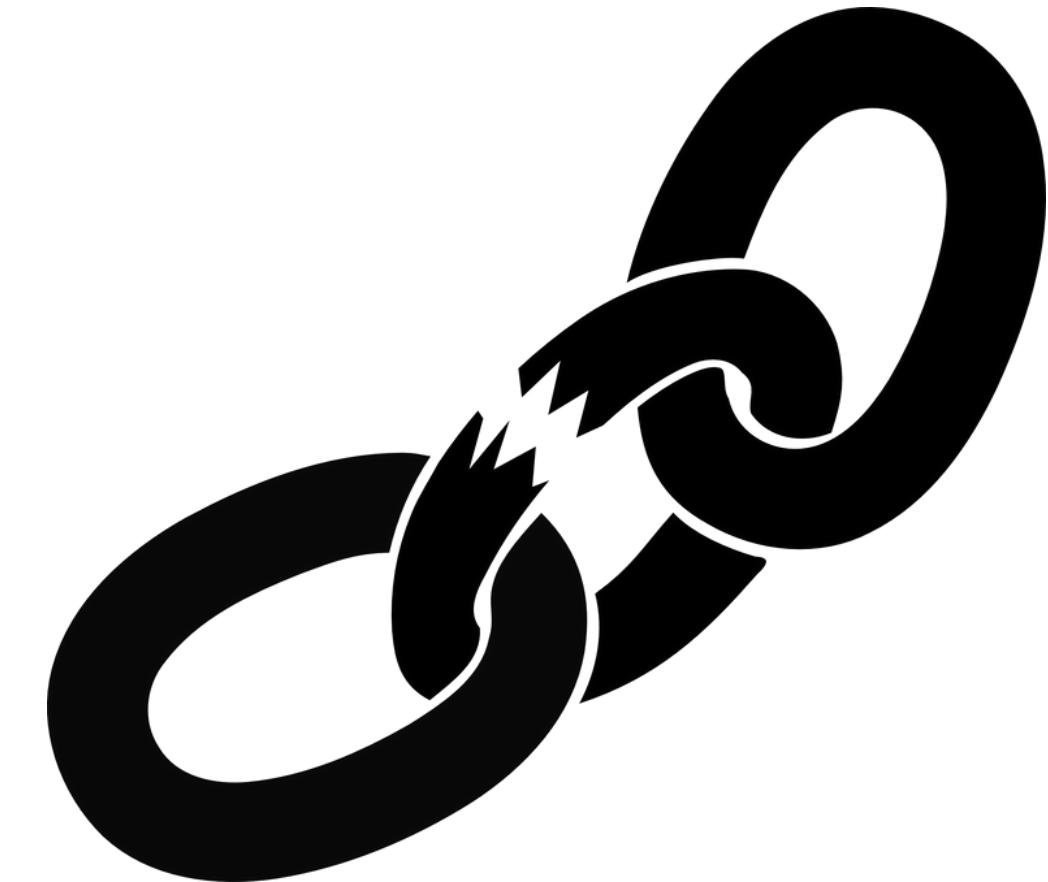
Bucles **while**

Cuando se cumple la condición, **while** repite la ejecución siempre que la condición se evalúe como **True**.

```
while expresion_condicional:  
    |     instrucciones...
```

break

sale del bucle inmediatamente, e incondicionalmente termina la operación del bucle; el programa comienza a ejecutar la instrucción más cercana después del cuerpo del bucle.

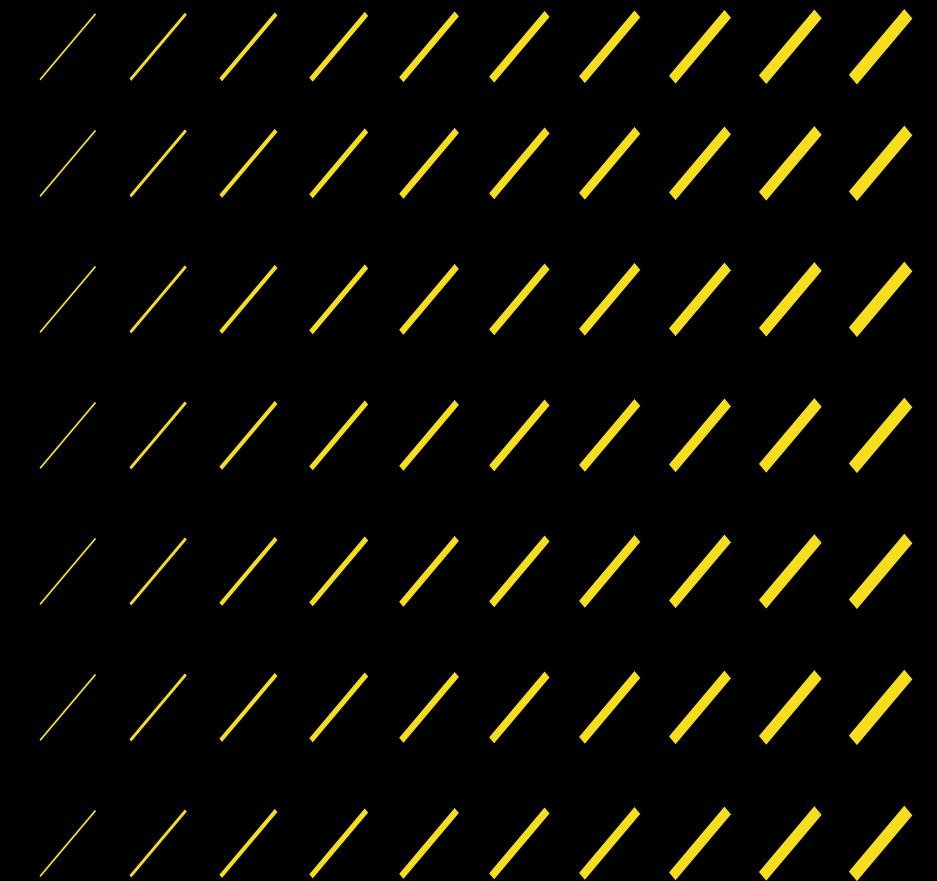


continue

se comporta como si el programa hubiera llegado repentinamente al final del cuerpo; el siguiente turno se inicia y la expresión de condición se prueba de inmediato.

</>

Día 8: Práctica de estructuras de control



Instr: Kattherine Hernandez

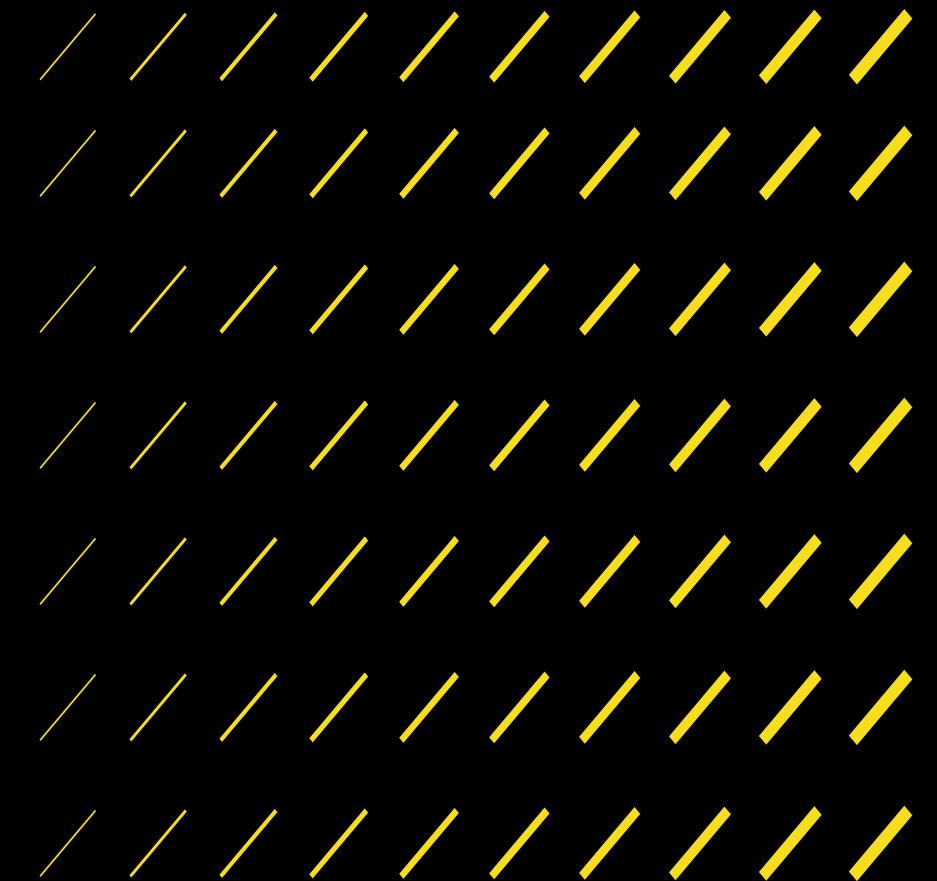
Ejercicio de repaso

Escribe un programa que permita ingresar las calificaciones de un grupo de estudiantes y determine:

1. Si cada estudiante aprobó o reprobó (nota mínima para aprobar es 70).
2. El promedio del grupo.
3. Cuántos estudiantes aprobaron y cuántos reprobaron.

</>

Ejercicio de la Semana #2



Instr: Kattherine Hernandez

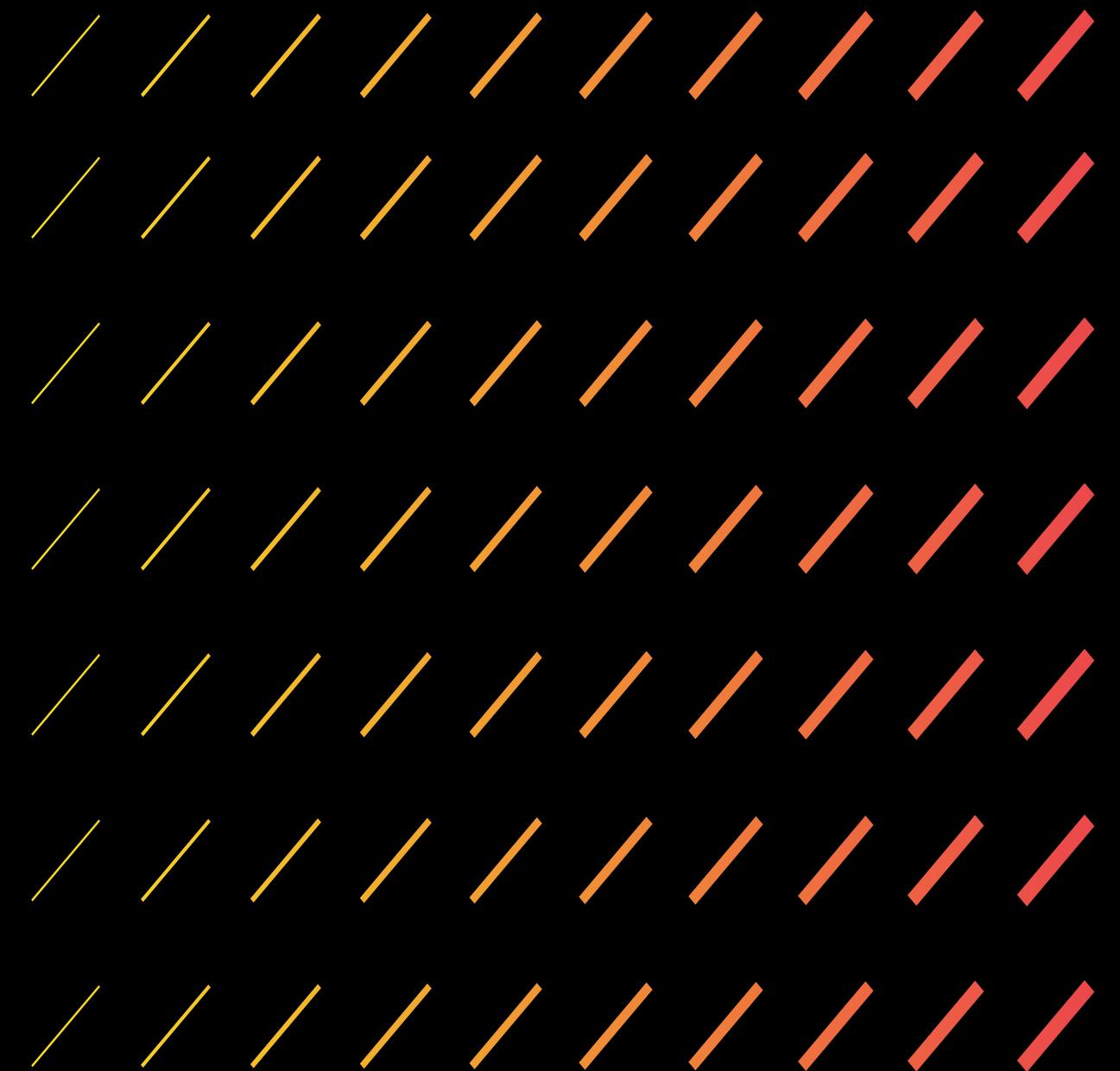
Ejercicio de la semana

Crea una calculadora que realice operaciones básicas (suma, resta, multiplicación, división). El usuario podrá realizar múltiples cálculos hasta que decida terminar. Al finalizar, mostrará un mensaje con el total de operaciones realizadas.

Requisitos:

- Solicitar al usuario dos números y el tipo de operación (suma, resta, multiplicación o división).
- Realizar la operación correspondiente usando funciones.
- Repetir el proceso tantas veces como el usuario quiera utilizando un bucle.
- Al finalizar, mostrar el total de operaciones realizadas con un mensaje al usuario

¡Éxitos en la resolución del ejercicio de la semana!



Instr. Kattherine Hernandez