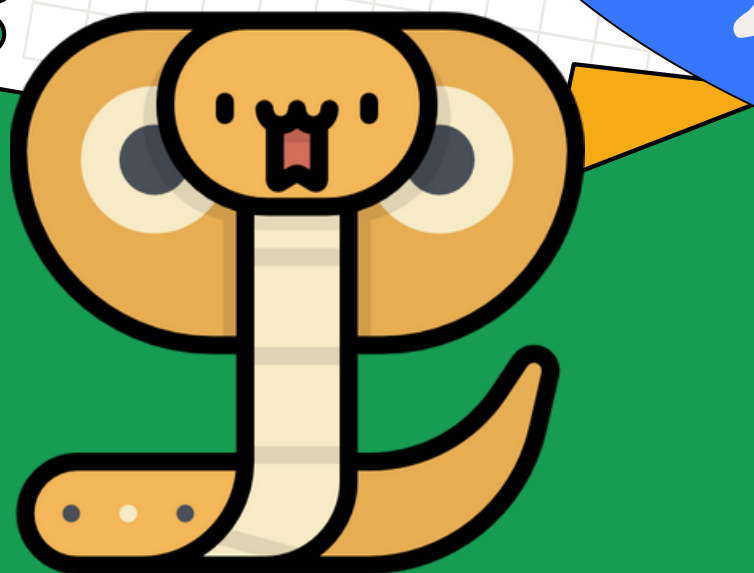


# PYTHON

## AULA 6





# BEM - VINDOS!



## AGENDA

- Hoje vamos praticar mais o uso do **for**

# ***IF***

O que é if em python?

- A função **if** é usado para o código tomar uma decisão.
- Ele executa uma parte do código somente se as condições forem atendidas.
- Pode ser usado o **else** junto ao **if** para executar parte do código caso a condição **não** seja atendida.

# IF

Exemplos de como usar if em python:

## Verdadeiro ou falso

```
Resultado = true  
if Resultado:  
    print("Verdade")
```

→ Saída: Verdade

## Verdadeiro ou falso

```
Resultado = false  
if Resultado:  
    print("Verdade")
```

→ Sem saída

## Idade

```
Idade = 20  
if Idade >= 18:  
    print("Maior de idade")  
else:  
    print("Menor de idade")
```

→ Saída: Maior de idade

## Idade

```
Idade = 16  
if Idade >= 18:  
    print("Maior de idade")  
else:  
    print("Menor de idade")
```

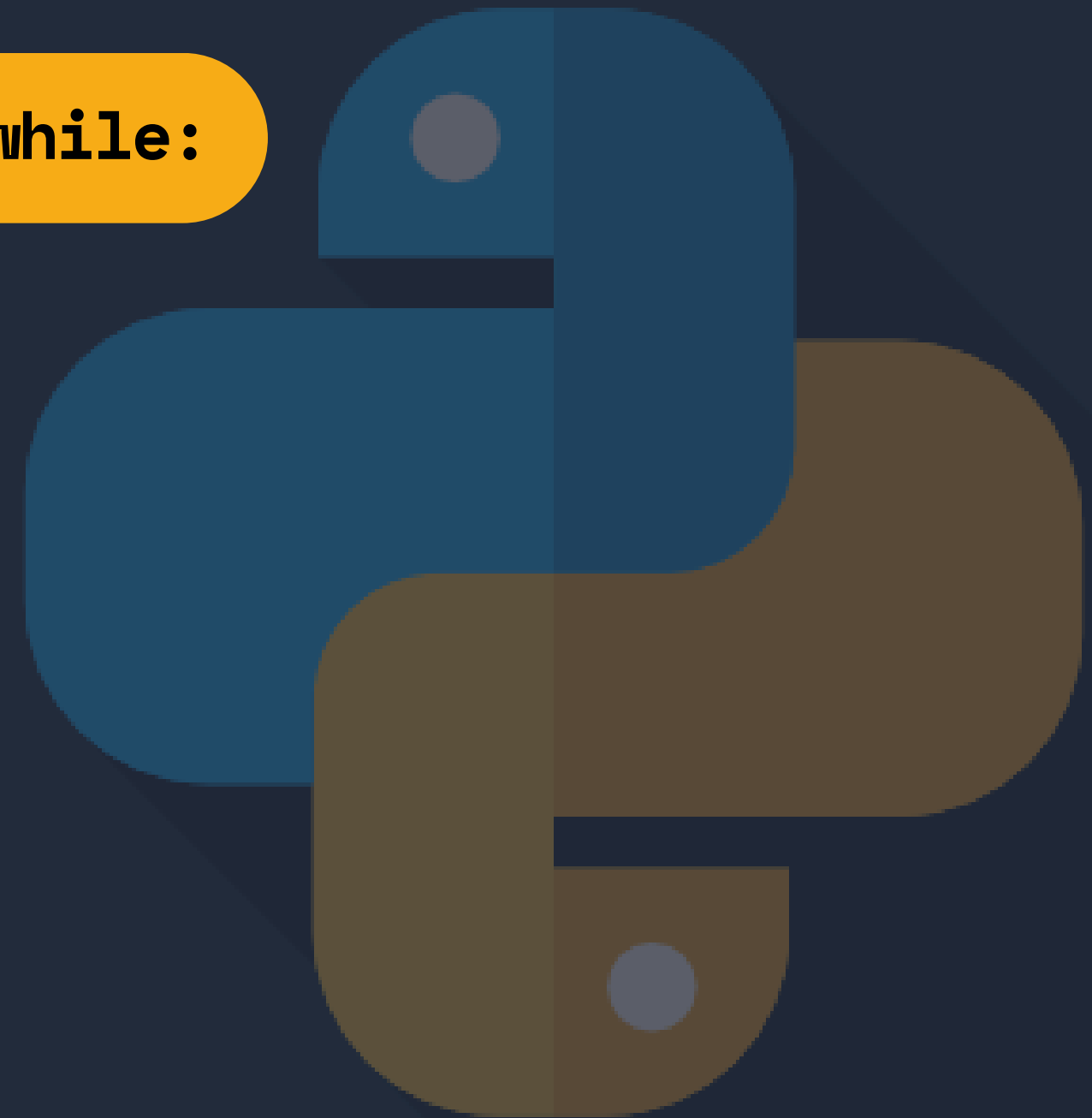
→ Saída: Menor de idade

# ***IF***

**ATENÇÃO!** Quando for usar funções como `if`, `else`, `for` `while`:

Deve sempre ter dois pontos `( : )` e indentação

```
if teste:  
    print(resultado)
```



# ***IF***

Desta forma o print está **fora** do if e será executado independente do resultado do if

```
if teste:  
    print(resultado)
```

Desta forma há erro de sintaxe e o código não rodará

```
if teste  
    print(resultado)
```

(serve também para outras funções como while e for)



# IF

## if com mais de uma condição

### Condição 1 e Condição 2

```
IdadeJoao = 16  
IdadeMaria = 19
```

```
if IdadeJoao >= 18 and Idade Maria >= 18:  
    print("Ambos são maior de idade")
```

### Condição 1 ou Condição 2

```
IdadeJoao = 16  
IdadeMaria = 19
```

```
if IdadeJoao >= 18 or Idade Maria >= 18:  
    print("Pelo menos um é maior de idade")
```

(serve também para outras funções como while e for)

# FOR

## Resumo:

For é uma estrutura de repetição bastante utilizada para tarefas repetitivas onde o código deve se repetir, como somar vários números em sequência.

início: onde começa a contagem (padrão é 0)  
fim: até onde vai (mas não inclui esse número)  
passo: de quanto em quanto conta (padrão é 1)



# FOR

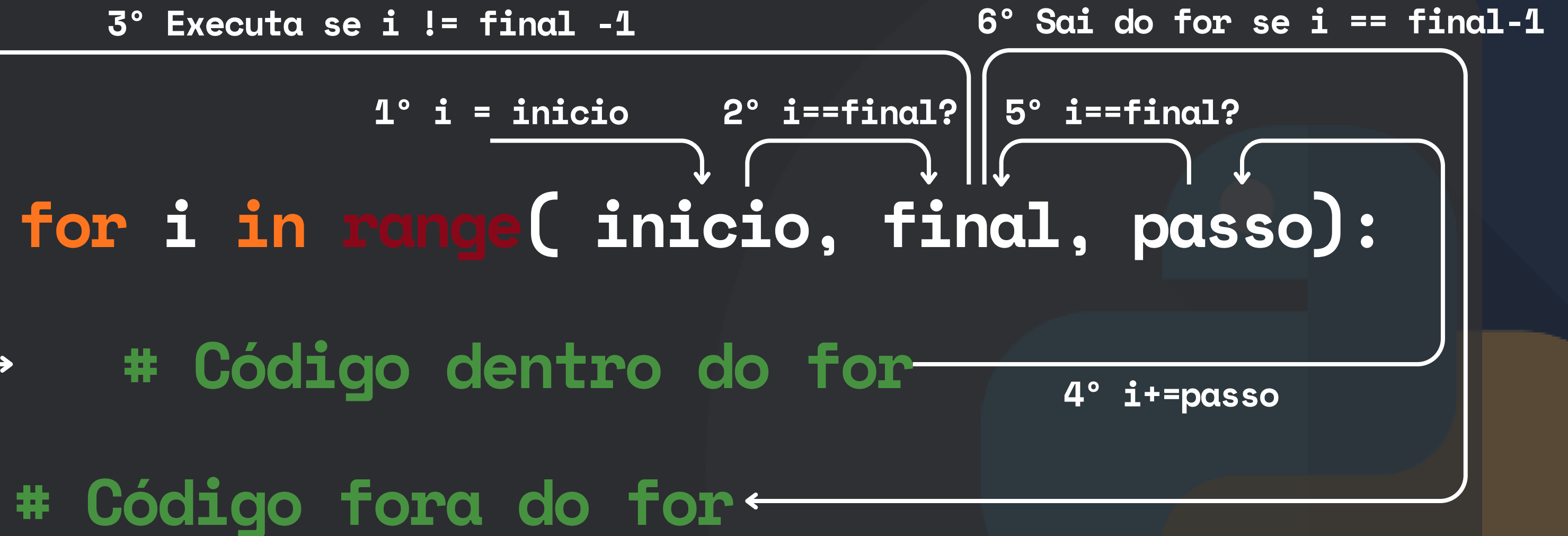
## Estrutura Básica:

```
for variavel in range(inicio, fim, passo):  
    # código a ser repetido
```

## Exemplo com repetições controladas:

```
for i in range(5):  
    print("Repetição número:", i)
```

# FOR



# FOR

Exemplo especificando início, meio e fim

```
for i in range(2, 10, 2):  
    print(i)
```

Exemplo Percorrendo lista

```
frutas = ["maçã", "banana", "uva"]  
for fruta in frutas:  
    print(fruta)
```

# FOR

início = 2

final = 10

passo = 2

{ 2, 4, 6, 8 }



→ **for** i **in** range( início, final, passo):

# Código dentro do for

**print**(i)

# Código fora do for

## Interação 1

Valores

i = 2

Saída

# FOR

início = 2

final = 10

passo = 2

for i in range( início, final, passo):

# Código dentro do for

print(i)

# Código fora do for

{ 2, 4, 6, 8 }



## Interação 1

Valores

i = 2

Saída

print(2) => 2

# FOR

início = 2

final = 10

passo = 2

{ 2, 4, 6, 8 }



$i += 2$

→ **for** i **in** range( início, final, passo):

# Código dentro do for

**print**(i)

# Código fora do for

## Interação 2

Valores

i = 4

Saída

print(2) => 2

# FOR

```
início = 2
```

```
final = 10
```

```
passo = 2
```

```
for i in range( início, final, passo):
```

```
    # Código dentro do for
```

```
    print(i)
```

```
# Código fora do for
```

{ 2, 4, 6, 8 }



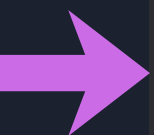
## Interação 2

Valores

i = 4

Saída

```
print(2) => 2  
print(4) => 4
```



# FOR

início = 2

final = 10

passo = 2

{ 2, 4, 6, 8 }



$i += 2$

→ **for** i **in** range( início, final, passo):

# Código dentro do for

**print**(i)

# Código fora do for

## Interação 3

Valores

i = 6

Saída

print(2) => 2  
print(4) => 4



# FOR

```
início = 2
```

```
final = 10
```

```
passo = 2
```

```
for i in range( início, final, passo):
```

```
    # Código dentro do for
```

```
    print(i)
```

```
# Código fora do for
```

{ 2, 4, 6, 8 }



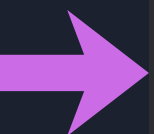
## Interação 3

Valores

i = 6

Saída

```
print(2) => 2  
print(4) => 4  
print(6) => 6
```



# FOR

início = 2

final = 10

passo = 2

{ 2, 4, 6, 8 }



$i += 2$

→ **for** i **in** range( início, final, passo):

# Código dentro do for

**print**(i)

# Código fora do for

## Interação 4

Valores

i = 8

Saída

```
print(2) => 2
print(4) => 4
print(6) => 6
```

# FOR

início = 2

final = 10

passo = 2

for i in range( início, final, passo):

# Código dentro do for

print(i)

# Código fora do for

{ 2, 4, 6, 8 }



## Interação 4

Valores

i = 8

Saída

```
print(2) => 2
print(4) => 4
print(6) => 6
print(8) => 8
```

# FOR

início = 2

final = 10

passo = 2

{ 2, 4, 6, 8 }



$i += 2$

→ **for** i **in** range( início, final, passo):

# Código dentro do for

**print**(i)

# Código fora do for

## Interação 5

Valores

i = 10 (?)

$i > \text{final} - 1$  ?

Saída

```
print(2) => 2
print(4) => 4
print(6) => 6
print(8) => 8
```

# FOR

Estaria fora do conjunto que definimos.

```
início = 2
```

```
final = 10
```

```
passo = 2
```

```
for i in range( início, final, passo):
```

```
    # Código dentro do for
```

```
    print(i)
```

```
→ # Código fora do for
```

{ 2, 4, 6, 8 }

## Interação 5

Valores

i = 10 (?)

Saída

```
print(2) => 2
print(4) => 4
print(6) => 6
print(8) => 8
```

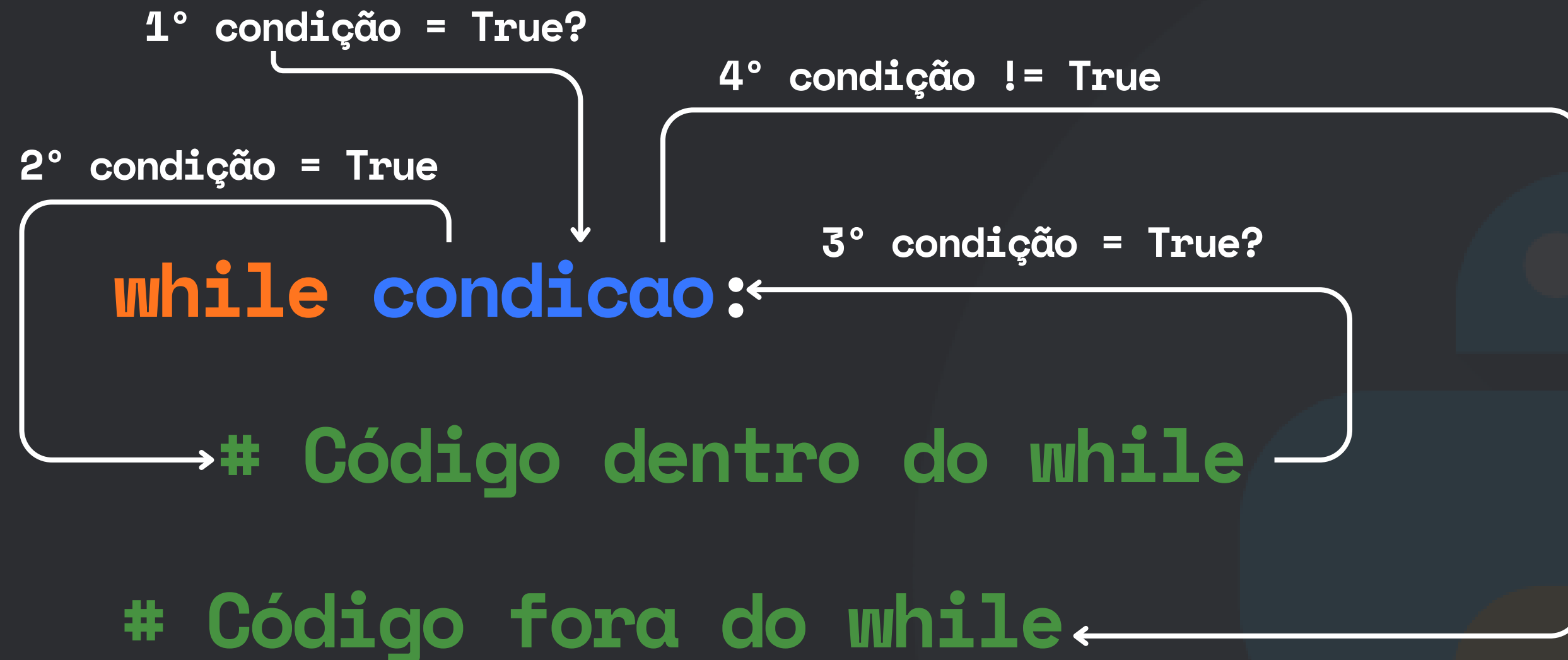
# WHILE

## Resumo

While é semelhante o for, porém, a diferença é que o For é usado quando você sabe quantas vezes quer repetir algo como repetir um número fixo de vezes ou percorrer uma sequência, enquanto o while é realizado quando não se sabe quantas vezes irá repetir, assim, repete enquanto uma condição for verdadeira

- O loop continua enquanto a condição for verdadeira.
- Se a condição nunca se tornar falsa, ele repete para sempre (loop infinito ⚠).

# WHILE



# WHILE

## Estrutura Básica:

```
while condição:  
    # código a ser repetido
```

## Exemplo Contagem simples

```
contador = 0  
  
while contador < 5:  
    print(contador)  
    contador += 1
```





# WHILE

## Exemplo interagindo com o usuário

```
senha = ""

while senha != "python123":
    senha = input("Digite a senha: ")

print("Acesso permitido!")
```

## Exemplo somando até digitar 0

```
soma = 0
num = int(input("Digite um número (0 para sair): "))

while num != 0:
    soma += num
    num = int(input("Digite outro número (0 para sair): "))

print(f"Soma total: {soma}")
```

# EXERCÍCIO 1

Escreva um programa que receba número inteiro positivo N e retorne 3 respostas diferentes:

- soma de todos os números de 1 até N;
- a soma dos números pares;
- a soma dos números ímpares.

UTILIZE FOR

Exemplos:

Entrada

10

Saída

55

30

25

Entrada

19

Saída

190

90

100

# EXERCÍCIO 1

```
1  n = int(input())
2
3  soma_total = 0
4  soma_pares = 0
5  soma_impares = 0
6
7  for i in range(1, n + 1):
8      soma_total = soma_total + i
9      if i % 2 == 0:
10         soma_pares = soma_pares + i
11     else:
12         soma_impares = soma_impares + i
13
14  print(soma_total)
15  print(soma_pares)
16  print(soma_impares)
17
18
```



# EXERCÍCIO 2

Calcule a potência de um número (base) elevado a um expoente, usando apenas multiplicação dentro de um laço. A base e o expoente devem ser digitados pelo usuário.

UTILIZE WHILE

Exemplos:

Entrada

2  
5

Saída

2 elevado a 5 = 32

Entrada

4  
3

Saída

4 elevado a 3 = 64

# EXERCÍCIO 2

```
1 base = int(input())
2 expoente = int(input())
3
4 resultado = 1
5 contador = 0
6
7 while contador < expoente:
8     resultado = resultado * base
9     contador = contador + 1
10
11 print(f"{base} elevado a {expoente} = {resultado}")
12
13
14
```

# EXERCÍCIO 2

Escreva um programa que receba um número inteiro positivo  $N$  e calcule:

- o produto (multiplicação) de todos os números de 1 até  $N$ .

Exemplos:

Entrada

5

Saída

120

UTILIZE FOR

# EXERCÍCIO 2

```
1 N = int(input("Digite um número inteiro positivo: "))
2
3 produto = 1 # começa em 1, pois será multiplicado
4
5 for i in range(1, N + 1): # percorre de 1 até N
6     produto *= i # multiplica o valor acumulado por i
7
8 print("Produto de 1 até", N, "=", produto)
```

# EXERCÍCIO 3

Exemplo:

Crie um programa que pergunta ao usuário o valor de 5 despesas fixas do mês.

O programa deve retornar:

- O gasto total do mês;
- O valor da despesa mais cara.

UTILIZE WHILE  
E FOR

Entrada

Despesa 1: 200

Despesa 2: 40

Despesa 3: 230

Despesa 4: 160

Despesa 5: 70

Saída

Gasto Total: R\$ 700

Despesa Mais Cara: R\$ 230



# EXERCÍCIO 3

```
1 despesas = [0, 0, 0, 0, 0]
2 contador = 0
3 total_de_despesas = 5
4
5 while contador < total_de_despesas:
6     valor = float(input(f"Despesa {contador + 1}: "))
7
8     despesas[contador] = valor
9     contador = contador + 1
10
11 gasto_total = 0
12 for valor in despesas:
13     gasto_total = gasto_total + valor
14
15 gasto_maximo = despesas[0]
16 for valor in despesas:
17     if valor > gasto_maximo:
18         gasto_maximo = valor
19
20 print(f"Gasto Total: R$ {gasto_total:.2f}")
21 print(f"Despesa Mais Cara: R$ {gasto_maximo:.2f}")
```

# EXERCÍCIO 4

Exemplo:

Saída

Nomes que começam com a  
letra A:  
['Ana', 'Amanda', 'Alice']

Dada uma lista de nomes,  
crie uma nova lista que  
contenha apenas os nomes que  
começam com a letra "A". Ao  
final, imprima a nova lista.

LISTA: [Ana, Bruno, Carlos,  
Amanda, Daniel, Alice]

UTILIZE `.append()`  
NO LAÇO FOR

# EXERCÍCIO 4

```
1 nomes = ["Ana", "Bruno", "Carlos", "Amanda", "Daniel", "Alice"]
2 nomes_com_a = []
3
4 for nome in nomes:
5     if nome[0] == "A":
6         nomes_com_a.append(nome)
7
8 print(f"Nomes que começam com a letra A: {nomes_com_a}")
9
10
```

# ***OBRIGADO!***

**Contem para a gente o que acharam da aula de hoje:**



**<https://forms.gle/Q1BYFnKxjyKuCC647>**

**CODELAB TEEN**