



# Aula 4: Laços de Repetição

FUNDAMENTOS DE PROGRAMAÇÃO

CodeRoots

# Aprendendo a repetir ações automaticamente com “for” e “while”

Na programação, muitas vezes precisamos executar o mesmo bloco de código várias vezes.  
Em vez de escrever o mesmo comando repetidamente, usamos estruturas de repetição.

## Exemplo do dia a dia:

““Enquanto tiver roupa suja, continue lavando.”  
Isso é um loop!

# O que são laços de repetição

Um laço de repetição (ou loop) permite que o programa repita instruções enquanto uma condição for verdadeira ou até percorrer todos os itens de uma sequência.

- Um laço de repetição (ou loop) é uma estrutura que permite ao programa executar um conjunto de instruções várias vezes, de forma automática.
- Ele continua repetindo o código enquanto uma condição for verdadeira ou até percorrer todos os itens de uma sequência.
- Isso é muito útil quando precisamos fazer tarefas repetitivas, como mostrar uma lista de nomes, somar números ou pedir dados várias vezes.

## Em resumo:

O programa repete uma ação enquanto a condição for verdadeira.

Quando ela deixa de ser verdadeira, o loop para automaticamente.

# Tipos de laços em Python

Python possui dois tipos principais de laços:

Tipo	Quando usar	Exemplo de ideia
for	Quando <b>sabemos quantas vezes</b> queremos repetir	“Repita 5 vezes”
while	Quando <b>não sabemos quantas vezes</b> , mas há uma <b>condição</b>	“Enquanto tiver fichas, jogue novamente”

# O laço `for`

O `for` é usado para percorrer uma sequência (como uma lista, texto ou intervalo de números).

## Sintaxe (Como deve ser escrito):



```
for variável in sequência:  
    # código a ser repetido
```

## Exemplo:



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



- `range(5)` gera os números 0, 1, 2, 3 e 4.
- O Python repete o `print` 5 vezes.
- A variável `i` muda a cada repetição.

# Usando range()

A função **range()** é muito usada com o **for**.

Exemplo	Descrição	Saída
range(5)	de 0 até 4	0,1,2,3,4
range(1,6)	de 1 até 5	1,2,3,4,5
range(0,10,2)	de 0 a 8 pulando de 2 em 2	0,2,4,6,8

## Exemplo:



```
for numero in range(1, 6):
    print("Número:", numero)
```

# O laço **while**

O **while** significa “enquanto”.

Ele repete o bloco de código enquanto a condição for verdadeira.

## Sintaxe (Como deve ser escrito):

```
● ● ●  
while condição:  
    # código repetido
```

## Exemplo:

```
● ● ●  
contador = 1  
while contador <= 5:  
    print("Contagem:", contador)  
    contador += 1
```



- Enquanto contador for menor ou igual a 5, o bloco será executado.
- Quando contador vira 6, o while para.

# Cuidado com loops infinitos

Se a condição nunca ficar falsa, o programa nunca para.

## Exemplo com erro:

```
● ● ●  
  
x = 1  
while x <= 5:  
    print(x)  
    # esquecemos de aumentar o x!
```

## Correção:

```
● ● ●  
  
x = 1  
while x <= 5:  
    print(x)  
    x += 1
```

- Sempre verifique se o valor da condição **muda dentro do loop**.

# Comparando for e while

Situação	Melhor usar
Você sabe quantas vezes quer repetir	for
A repetição depende de uma condição	while
Vai percorrer uma lista, string ou range	for
Vai esperar algo acontecer (ex: input)	while

## Exemplo prático:

- for → “Repita 5 vezes.”
- while → “Enquanto o usuário não digitar ‘sair’.”

# Boas práticas

- Use nomes claros para variáveis (contador, i, numero).
- Evite loops infinitos acidentais.
- Teste seu código com diferentes valores.
- Mantenha indentação correta dentro do laço.



Obrigado por participar  
da aula de hoje.

CodeRoots