



# Estruturas Condicionais e de Repetição em Python

## A Estética do Código 🖋️

Você já brincou de organizar os lápis de cor em ordem ou de alinhar os brinquedos na prateleira? Isso deixa tudo mais bonito e fácil de encontrar, certo? No Python, organizar o código também é importante! 📖 ✨

Quando escrevemos um programa, precisamos "empurrar" algumas linhas para frente, como se estivéssemos fazendo parágrafos em uma redação. Isso se chama **identação**. Ela não só deixa o código mais bonito, mas também ajuda o computador a entender onde cada parte do programa começa e termina. 🧩✅

Se não organizarmos direitinho, o Python não vai entender o que queremos dizer, e o programa pode dar erro! 😱 Então, sempre lembre-se de manter tudo arrumado!

---

## O que são Estruturas Condicionais? 🤔

Imagine que você está jogando um jogo e precisa decidir: **se** pegar a chave, a porta abre; **se não** pegar a chave, a porta continua trancada. 🔑🔒

No Python, podemos escrever regras assim! Chamamos isso de **estrutura condicional**. Basicamente, o computador toma decisões baseado em certas

condições. Ele verifica **se** algo acontece e, dependendo disso, segue um caminho ou outro. 🏞️💻

Por exemplo:

```
idade = 10

if idade >= 18:
    print("Você pode dirigir! 🚗")
else:
    print("Você ainda não pode dirigir. 🚲")
```

Aqui, o programa verifica **se** a idade é 18 ou mais. Se for, mostra uma mensagem. Se não for, mostra outra. Fácil, né? 😊✨

---

## O que é o "if"? 🤔

O **if** é como se fosse uma escolha que fazemos no dia a dia. Imagine que sua mãe diz:

🔴 **Se** você terminar a lição de casa, pode jogar videogame. 🎮

🔴 **Se não**, nada de jogo!

No Python, funciona do mesmo jeito! O **if** verifica se uma condição é verdadeira e, se for, ele executa uma ação. Se não for, ele simplesmente ignora.

Veja um exemplo:

```
chovendo = True

if chovendo:
    print("Pegue um guarda-chuva! 🌧️")
```

Aqui, o programa verifica se **está chovendo**. Se estiver, ele manda você pegar um guarda-chuva! 🌧️

Simples, né? O **if** ajuda o computador a tomar decisões sozinho! 😊💡

## If/Else: O que é isso? 🤔

O **if/else** é como tomar decisões na vida real! Imagine que sua mãe diz:

🍪 **Se** você comer tudo no jantar, pode ganhar uma sobremesa! 🎉

🥬 **Senão**, nada de doce... 😞

No Python, funciona assim! O **if** verifica uma condição. **Se for verdadeira**, ele faz uma coisa. **Se não for**, ele faz outra.

Veja um exemplo:

```
idade = 16

if idade >= 18:
    print("Você pode dirigir! 🚗")
else:
    print("Você ainda não pode dirigir. 🚲")
```

Aqui, o programa verifica **se** a idade é 18 ou mais.

✅ **Se for**, ele diz que você pode dirigir.

❌ **Se não for**, ele diz que você ainda não pode.

O **if/else** ajuda o computador a tomar decisões sozinho! Fácil, né? 😊✨

---

## If/Elif/Else: Várias Escolhas! 🤔

O **if/elif/else** é como quando você escolhe uma roupa para sair! 👕👗

👁️ **Se** estiver calor, você veste uma camiseta.

☁️ **Se** estiver frio, você coloca um casaco.

❄️ **Se** estiver MUITO frio, você usa um casaco e um cachecol!

🧸 **Se não**, você escolhe qualquer roupa confortável.

No Python, podemos usar **elif** quando temos mais de duas opções!

Veja um exemplo:

```
temperatura = 10

if temperatura > 30:
```

```
print("Está muito quente! 🥵")
elif temperatura > 20:
    print("O clima está agradável! 😊")
elif temperatura > 10:
    print("Está um pouco frio! 🧥")
else:
    print("Está muito frio! ❄️")
```

✅ **Se** a temperatura for maior que 30, diz que está quente.

✅ **Se** for maior que 20, diz que está agradável.

✅ **Se** for maior que 10, diz que está frio.

❌ **Se não**, diz que está muito frio!

Isso ajuda o computador a escolher a melhor opção entre várias possibilidades!



## O que são Estruturas de Repetição? 🔄

Imagine que você está brincando de contar até 10. Você pode falar "1, 2, 3..." um por um, mas e se o computador tivesse que fazer isso? 🤔

Em Python, temos algo chamado **estrutura de repetição**! Elas ajudam o programa a repetir algo várias vezes sem precisar escrever tudo manualmente.



Por exemplo, podemos dizer ao computador:

✅ **Repita 5 vezes:** "Eu amo Python!" 💙🐍

Veja um exemplo:

```
for i in range(5):
    print("Eu amo Python! 💙🐍")
```

Isso imprime a frase **5 vezes** sem precisar escrever **print** várias vezes! 😊

Outra forma é repetir enquanto algo for verdadeiro:

```
contador = 1
```

```
while contador <= 5:  
    print(f"Contando: {contador}")  
    contador += 1
```

Aqui, o computador vai contar de 1 a 5 **sozinho!** 🧮🚀

As estruturas de repetição ajudam a fazer tarefas repetitivas de forma rápida e eficiente! Legal, né?

## O que é o "for"? 🔁

Imagine que você tem um saco de doces 🍬 e quer pegar **um por um** para comer. Você pode fazer isso manualmente, mas e se o computador pudesse te ajudar?

O **for** faz exatamente isso! Ele pega **cada item** de uma lista e executa uma ação com ele. 📦✅

Por exemplo, se queremos contar de 1 a 5:

```
for numero in range(1, 6):  
    print(f"Contando: {numero}")
```

Isso vai imprimir:

```
Contando: 1  
Contando: 2  
Contando: 3  
Contando: 4  
Contando: 5
```

Também podemos usar o **for** para percorrer uma lista:

```
frutas = ["🍏 Maçã", "🍌 Banana", "🍓 Morango"]  
  
for fruta in frutas:  
    print(f"Eu gosto de {fruta}!")
```

Isso vai imprimir:

Eu gosto de 🍏 Maçã!  
Eu gosto de 🍌 Banana!  
Eu gosto de 🍓 Morango!

O **for** é muito útil quando já sabemos **quantas vezes** precisamos repetir algo ou queremos percorrer listas e textos.

## O que é o for/else? ↺ →

Você já jogou um jogo de esconde-esconde? 🕵️ Imagine que você está procurando seu amigo:

👁️ Você olha atrás do sofá, da porta e da cortina.

✅ **Se encontrar, você grita: "Achei!"**

❌ **Se não encontrar em lugar nenhum, você diz: "Não achei..."**

No Python, o **for/else** funciona do mesmo jeito! O **for** percorre uma lista ou texto, e o **else** só acontece **se o for terminar sem interrupções**.

Veja um exemplo:

```
texto = input("Informe um texto: ")
VOGAIS = "AEIOU"

for letra in texto:
    if letra.upper() in VOGAIS:
        print(letra, end="") # Mostra só as vogais

else:
    print() # Adiciona uma quebra de linha no final
```

## O que acontece aqui? 🤔



🔍 O código verifica cada **letra** do texto digitado.

✅ **Se for uma vogal (A, E, I, O, U), imprime a letra.**

❌ **Quando todas as letras forem verificadas, o else adiciona uma quebra de linha.**

Isso pode ser útil para indicar quando um loop terminou sem interrupções!

## O que é a função range?

Imagine que você está subindo uma escada e conta os degraus: **1, 2, 3, 4...**  

A função **range** faz o computador contar números automaticamente, sem precisar escrever todos à mão! ✨

## Como funciona?

A função **range()** pode ter até **3 números** dentro dos parênteses:

```
range(início, fim, passo)
```

- ◆ **Início** (opcional) → Onde começa a contagem.
- ◆ **Fim** (obrigatório) → Onde termina (mas ele **não conta esse número**).
- ◆ **Passo** (opcional) → De quanto em quanto contar (pode ser positivo ou negativo).

## Exemplos:

### ✓ Contando de 0 a 4:

```
for i in range(5):  
    print(i) # Imprime: 0, 1, 2, 3, 4
```

### ✓ Contando de 1 a 5:

```
for i in range(1, 6):  
    print(i) # Imprime: 1, 2, 3, 4, 5
```

### ✓ Contando de 2 em 2:

```
for i in range(0, 10, 2):  
    print(i) # Imprime: 0, 2, 4, 6, 8
```

### ✓ Contando para trás!

```
for i in range(10, 0, -2):  
    print(i) # Imprime: 10, 8, 6, 4, 2
```

A função **range()** é super útil para repetir coisas sem precisar escrever todos os números! 🚀💡

## O que é o while? 🔄

O **while** é como um jogo que só acaba quando você decide parar! 🎮

Imagine que você está brincando de perguntas e respostas com um amigo:

❓ Ele continua perguntando até que você diga "**chega!**".

No Python, o **while** faz o computador repetir algo **enquanto** uma condição for verdadeira. ✅

## Exemplo do Caixa Eletrônico 🏦

O código da imagem representa um **menu interativo**, onde o usuário pode escolher uma opção:

```
opcao = -1 # Começa com um valor qualquer  
  
while opcao != 0: # Enquanto a opção for diferente de 0, o loop continua!  
    opcao = int(input("[1] Sacar \n[2] Extrato \n[0] Sair \n: "))  
  
    if opcao == 1:  
        print("Sacando... 💰")  
    elif opcao == 2:  
        print("Exibindo o extrato... 📄")
```

## Como funciona? 🤔

- 1 O programa pede uma opção ao usuário.
- 2 Se ele escolher 1, aparece "Sacando...".
- 3 Se ele escolher 2, aparece "Exibindo o extrato...".
- 4 Se ele escolher 0, o programa para! 🛑



Esse tipo de código é útil para menus e sistemas interativos! 🚀 ✨

**By MChiodi**