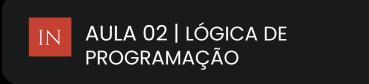
IN

INFINITY SCHOOL

VISUAL ART CREATIVE CENTER



01

02 Elif

03 Else

04 Operadores lógicos

05 Match



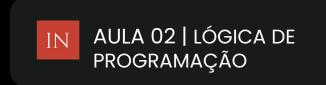




O que é uma condicional?

As estruturas condicionais em Python permitem que o programa tome decisões com base em condições específicas. Essas estruturas verificam uma condição e executam um bloco de código se a mesma for verdadeira, caso contrário, executa outro bloco. Essas estruturas ajudam a controlar o fluxo de execução do programa, permitindo ações diferentes com base em diferentes condições.

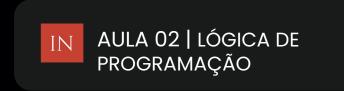
```
00 08 00 0 @OCLffffLG@@00 0 08
             CfffffffttttttttttttL0 80
          00 @fff1ftftttttttttttttG 80 00 80@ 80
          00 8fff@@Ottttttttttttt1@00
      00 00 00fffftttttttttttttt1110
      00 00
            00fftttttttttttttttttt11110
          00 8t111111111tttttt11111@00 00
     00 008G088G080G88GG8ttttt11118008800
      OGtffffftttttttttttttttttt11111GGGGGGGG0 00
      Gfffffffttttttttttttttttt1111110GGGGGGGC080
     00fffffffttttttttttttttttt11111118GGGGGGGG
   00 Lfffffftttttttttttttttt111111110GGGGGGGC80
     Offffftttttttttttttttl1111111118GGGGGGGGG
     Offfftttttttttttttttt1111111111CLGGGGGGGGGG8000
    *Lffttttttttttt00%000@0800080GGGGGGGCCCC0@00
    @LfftttttttttC8CGG0GGGGGGGGGGGGGGGCCCC0
    000ftttttttttt00GGGGGGGGGGGGGGGGGGGGCCCCCCG0
     00 fttttttt1GGGGGGGGGGGGGGGGGGGCCCCCCCCCCC
     001ttttt18GGGGGGGGGGGGGGCCCCCCCCCCCCCCCC
      00f111110ggggggggggggcccccccccccc08 80
          00 @GGGGGGGGGG808@00@@00 00
            00GGGGGGGGGGCCCCCCCC
      00 00
            @GGGGGGGCCCCCCCCGLCCC@OO
             8GGGGGGGCCCCCCC
                          @CCL800
      00 00
            00cgggcccccccc
                          @CCCO
      80 08
            00GGCCCCCCCCCC8@ CCC0
              00GCCCCCCCCCCCCCG8 80
      00 00
            00 00ccccccccccc0 0
             08 88@8GG00G0800
                           08 80
00 00 00 00 00 00 00@@80@@0@ 00 00 00 00 00
```





Qual o melhor momento para se usar uma condicional?

• Quando é preciso tornar seu código dinâmico e inteligente através de tomadas de decisão, fazer filtros, validações e muito mais.



LP

- Condicionais

A estrutura condicional "if" em Python permite que o programa execute um bloco de código se uma determinada condição for verdadeira, é uma forma de tomar decisões com base em informações específicas durante a execução do programa.

```
num1 = 10
num2 = 10

# 0 bloco if só executa se a condição for verdadeira
if num1 == num2:
    print(f'{num1} é igual a {num2}')

print('Fim do programa')
```



ELIF

estrutura condicional "elif" permite testar novas condições se a condição anterior for falsa, com o "elif" é possível criar fluxos de decisão diferentes dos anteriores de maneira organizada. Em resumo, o "elif" nos ajuda tomar decisões a específicas com base em múltiplas condições, melhorando a clareza e eficiência do código.

```
num1 = 5
num2 = 10
if num1 == num2: # Falso
    print(f'{num1} é igual a {num2}')
# O bloco elif só executa se o bloco anterior for falso
elif num1 < num2:
    print(f'{num1} é menor que {num2}')
```



ELSE

estrutura condicional "else" permite definir uma ação alternativa quando todas as condições anteriores não forem atendidas.

O "else" é opcional e é utilizado para lidar com situações em que há duas possibilidades distintas, dependendo do resultado das condições anteriores, tanto "if" quanto "elif".

```
num1 = 5
num2 = 10
if num1 == num2: # Falso
    print(f'{num1} é igual a {num2}')
elif num1 > num2: # Falso
    print(f'{num1} é maior que {num2}')
# O bloco else só executa
# se todos os blocos anteriores forem falsos
else:
    print(f'{num2} é maior que {num1}')
```



OPERADORES LÓGICOS

AND

р	q	p^q
٧	V	V
٧	F	F
F	V	F
F	F	F

OR

р	q	pvq
٧	٧	V
٧	F	V
F	٧	V
F	F	F

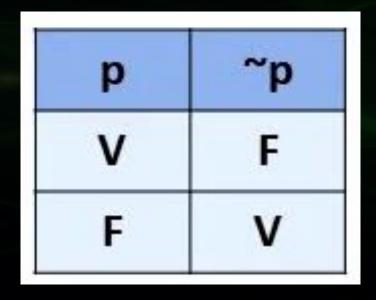
A tabela verdade é usada na lógica e programação para mostrar os resultados possíveis de uma expressão lógica.

operador lógico "and" verdadeiro somente se ambas entradas forem verdadeiras, Já operador "or" retorna verdadeiro se pelo entradas das for uma menos verdadeira, a tabela verdade é usada avaliar expressões e tomar para decisões com base nos valores de entrada.



OPERADORES LÓGICOS

NOT



O operador "not" é um operador lógico em Python usado para inverter o valor de uma expressão lógica.

Se a expressão é verdadeira, o "not" retorna falso, se a expressão é falsa, o "not" retorna verdadeiro.

Basicamente, o "not" nega o valor lógico da expressão, esse operador é útil para alterar ou inverter o fluxo de execução do programa com base em determinadas condições.

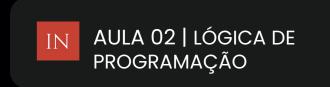


AND

O operador lógico "and" retorna verdadeiro apenas se ambos operandos forem OS verdadeiros, caso contrário o resultado será falso.

usado para verificar de forma simultânea se duas ou mais condições SÃO verdadeiras.

```
num1 = 5
num2 = 10
num3 = 20
''' O operador and só retorna verdadeiro
    se todas as condições forem verdadeiras'''
if num1 < num2 and num2 < num3: # Verdadeiro
    print(f'{num3} é maior que {num1} e {num2}')
```



OR

O operador lógico "or" retorna verdadeiro se pelo menos uma das condições forem verdadeiras e retorna falso apenas se ambas as condições forem falsas. É usado para verificar se pelo menos uma das condições é verdadeira.

```
num1 = 5
num2 = 10
num3 = 20
    O operador or retorna verdadeiro
    se pelo menos uma das condições
    for verdadeira'''
if num1 < num2 or num2 == num3: # Verdadeiro
    print(f'{num1} é menor que {num2}')
```



LP

- Condicionais

NOT

O operador lógico "not" inverte o valor de uma expressão lógica, ele retorna verdadeiro se a expressão for falsa e retorna falso se a expressão for verdadeira.

```
num1 = 5
num2 = 10

# 0 operador not inverte o resultado da expressão

if not num1 < num2: # Falso
    print(f'{num1} é menor que {num2}')
else:
    print('Expressão negada')</pre>
```



MATCH

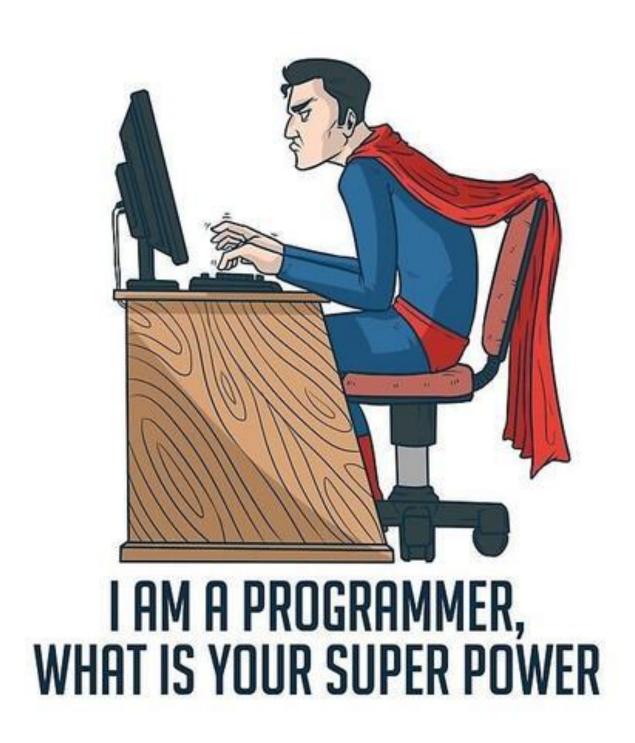
O "match case" é uma estrutura condicional que permite comparar uma expressão com uma série de padrões e executar o código associado ao primeiro padrão correspondente. Ele simplifica a lógica condicional, torna o código mais legível e ajuda a desenvolver algoritmos mais expressivos.

```
nome = input ("Digite seu nome: ")
'''A condicional match testa uma variável
   e executa um bloco se o case for compatível'''
match nome:
    case 'Fulano':
        print (f'Seja bem vindo, {nome}')
    case 'Beltrano':
        print(f'Seja bem vindo, {nome}')
    case 'Joãozinho':
        print (f'Seja bem vindo, {nome}')
    case 'José':
        print (f' Seja bem vindo, {nome}')
```



EXERCICIOS

- Faça um Programa que peça dois números e imprima o maior deles.
- Faça um Programa que peça um valor e mostre na tela se o valor é positivo ou negativo.
- 3. Faça um Programa que verifique se uma letra digitada é "F" ou "M". Conforme a letra escrever: F - Feminino, M - Masculino, Sexo Inválido.
- Faça um Programa que verifique se uma letra digitada é vogal ou consoante.
- Faça um programa para a leitura de duas notas parciais de um aluno. O programa deve calcular a média alcançada por aluno e apresentar:
 - o A mensagem "Aprovado", se a média alcançada for maior ou igual a sete;
 - A mensagem "Reprovado", se a média for menor do que sete;
 - o A mensagem "Aprovado com Distinção", se a média for igual a dez.
- Faça um Programa que leia três números e mostre o maior deles.
- Faça um Programa que leia três números e mostre o maior e o menor deles.
- Faça um programa que pergunte o preço de três produtos e informe qual produto você deve comprar, sabendo que a decisão é sempre pelo mais barato.
- Faça um Programa que leia três números e mostre-os em ordem decrescente.
- 10. Faça um Programa que pergunte em que turno você estuda. Peça para digitar M-matutino ou V-Vespertino ou N- Noturno. Imprima a mensagem "Bom Dia!", "Boa Tarde!" ou "Boa Noite!" ou "Valor Inválido!", conforme o caso.



IN

INFINITY SCHOOL

VISUAL ART CREATIVE CENTER