



IN

INFINITY SCHOOL

V I S U A L A R T C R E A T I V E C E N T E R

LP – Condicionais

01 If

02 Elif

03 Else

04 Operadores lógicos

05 Match



LP – Condicionais

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.

```
08 0 00 00 08 00 0 @0CLffffLG@00 0 08 00 00 00 0
00 0 00 00 00 00@8ffffffttttttttttC 0 00 00 00 00 0
00 00 00 0 00 CffffffttttttttttL0 80 08 0 80 00 0
08 80 00 @0 00 @ffff1ftttttttttttG 80 00 80@ 80 00 8
00 0 00 00 00 00fff @ftttttttttttt10 00 00 00 00 0
00 00 00 0 00 @ffL @ttttttttttttt1@00 00 0 00 00 0
08 00 00 0 00 8fff@0ttttttttttttt1@00 08 0 00 80 0
00 0 00 00 00 00ffffttttttttttttt1110 00 00 00 00 0
00 0 80 00 00 00fftttttttttttttt11110 08 00 80 80 0
00 00 00 0 00 8t1111111ittttttt1111@00 00 0 00 00 0
00 00 00 0@8G088G080G88GG8ttttttt11118008800 0 00 00 0
00 0 00 0Gtffffffttttttttttttttt11111GGGGGGGG0 00 00 0
08 0 00 Gffffffttttttttttttttt1111110GGGGGGGGC@80 80 0
00 00 00ffffffttttttttttttttt1111118GGGGGGGG0 00 00 0
08 @0 00 Lffffffttttttttttttt11111110GGGGGGGGC80 80 0@
00 0 00 fffffffttttttttttttt111111110GGGGGGGGG00 00 0
08 00 0ffffffttttttttttttt11111111i8GGGGGGGGG 00 80 0
00 00 0ffftttttttttttttt11111111CLGGGGGGGGG8@00 00 0
08 0 008ffftttttttttttttt1t111111118GGGGGGGGGGG8 80 0
00 0 00Cffftttttttttttttt11111111f@GGGGGGGGGGCG0 00 0
00 00 Lffttttttttttt00 @00@@08@0080GGGGGGGGCCCC0@00 00 0
08 00 @LffttttttttttC8CGG0GGG0GG0GGGGGGGGGGGGCCCC0 00 00 8
00 0 000fttttttttt00GGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCG0 00 0
08 80 0ftttttttttLGGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCC0@80 00 8
00 00 0ttttttttt80GGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCC8 00 00 0
08 0 00 ttttttttt0GGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCC00 00 0
00 0 00 ftttttttt1GGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCCCCCC80 00 0
00 00 0Ltttttttt18GGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCCCCCCCG 00 00 0
00 00 001ttttttt18GGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCCCCCCCCCC0 80 00 0
00 0 00 Gtttttt11GGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCCCCCCCCCC0 00 00 0
08 0 80 00f111110GGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCCCCCCCCCC08 80 00 0
00 00 00 0 00 @GGGGGGGGGG808@00@00 00 00 0 00 00 0
00 0 00 88 00 00GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCC0 00 08 80 80 0@
00 0 00 00 00 00GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCC0 00 00 00 00 0
00 00 00 0 00 @GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCCGLCCC@00 00 0 00 00 0
08 00 00 0 08 8GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCC @CCL800 00 0 00 00 0
00 0 00 00 00 00CGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCC @CCC0 00 00 00 00 0
08 0 00 80 08 00GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGCCCCCCCC8@ CCC0 00 08 80 80 0
00 00 00 0 00 GGCCCCCCCCCCCCCCCCCCCCGG 00 00 0 00 00 0
08 80 00 @0 08 00GGCCCCCCCCCCCCCCCCGG 80 08 @0 80 80 8
00 0 00 00 00 00 00CCCCCCCCCCCCCCCCG0 0 00 00 00 00 0
08 80 00 0 08 08 88@8GG00G0800 08 80 08 0 00 00 8
00 00 00 0 00 00 00@0@80@0@0 00 00 00 00 0 00 00 0
```


LP – Condicionais



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

IF

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
```

```
# O bloco if só executa se a condição for verdadeira
```

```
if num1 == num2:
```

```
    print(f'{num1} é igual a {num2}')
```

```
print('Fim do programa')
```


LP – Condicionais

ELIF

A 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 a tomar decisões 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}')
```

LP – Condicionais

ELSE

A 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}')
```

LP – Condicionais

OPERADORES LÓGICOS

AND

p	q	$p \wedge q$
V	V	V
V	F	F
F	V	F
F	F	F

OR

p	q	$p \vee q$
V	V	V
V	F	V
F	V	V
F	F	F

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

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

LP – Condicionais

OPERADORES LÓGICOS

NOT

p	$\sim p$
V	F
F	V

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.

LP – Condicionais

AND

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

É 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}')
```


LP – Condicionais

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
```

```
# O operador not inverte o resultado da expressão
```

```
if not num1 < num2: # Falso  
    print(f'{num1} é menor que {num2}')
```

```
else:  
    print('Expressão negada')
```


LP – Condicionais

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}')
```

LP – Condicionais

EXERCICIOS

1. Faça um Programa que peça dois números e imprima o maior deles.
2. 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.
4. Faça um Programa que verifique se uma letra digitada é vogal ou consoante.
5. Faça um programa para a leitura de duas notas parciais de um aluno. O programa deve calcular a média alcançada por aluno e apresentar:
 - A mensagem "Aprovado", se a média alcançada for maior ou igual a sete;
 - A mensagem "Reprovado", se a média for menor do que sete;
 - A mensagem "Aprovado com Distinção", se a média for igual a dez.
6. Faça um Programa que leia três números e mostre o maior deles.
7. Faça um Programa que leia três números e mostre o maior e o menor deles.
8. 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.
9. 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.



**I AM A PROGRAMMER,
WHAT IS YOUR SUPER POWER**



IN

INFINITY SCHOOL

V I S U A L A R T C R E A T I V E C E N T E R