

A woman with long dark hair, wearing glasses, is shown from the chest up, looking down at a laptop screen. The background is dark and slightly blurred.

IN

INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

AULA 02 - Condicionais



O QUE IREMOS APRENDER

- 01** O QUE SÃO OPERADORES
- 02** INDENTAÇÃO
- 03** ESTRUTURA CONDICIONAL

Revisão da Aula Anterior

- ALGORITMO
- VARIAVEIS
- FUNÇÕES INTERNAS
- OPERADORES ARITMETICOS
- F STRING



O QUE SÃO OPERADORES

Operadores são símbolos que instruem o compilador ou interpretador a realizar operações específicas sobre operandos.

Eles são essenciais em todas as linguagens de programação para manipular dados, realizar cálculos e comparações.

Na aula passada, vimos operadores aritméticos, porém existem vários tipos de operadores, incluindo:

Aritméticos;

Comparação;

Lógicos;

Atribuição;

Associação.

+	-	*	/	%	* *	//
==	!=	>	<	>=	<=	
and				or		not
=	+=	-=	*=	/=		
in						not in

OPERADORES DE COMPARAÇÃO

Comparam dois valores e retornam um valor booleano (True ou False).

Exemplos:



```
1 x = 5
2 y = 10
3 print(x == y) # False
4 print(x != y) # True
5 print(x > y) # False
6 print(x < y) # True
7 print(x >= y) # False
8 print(x <= y) # True
```

$==$	Verifica se dois valores são iguais.	$x == y$	# False
$!=$	Verifica se dois valores são diferentes	$x != y$	# True
$>$	Verifica se o primeiro valor é maior que o segundo.	$x > y$	# False
$<$	Verifica se o primeiro valor é menor que o segundo	$x < y$	# True
\geq	Verifica se o primeiro valor é maior ou igual ao segundo.	$x \geq y$	# False
\leq	Menor ou igual (\leq):	$x \leq y$	# True

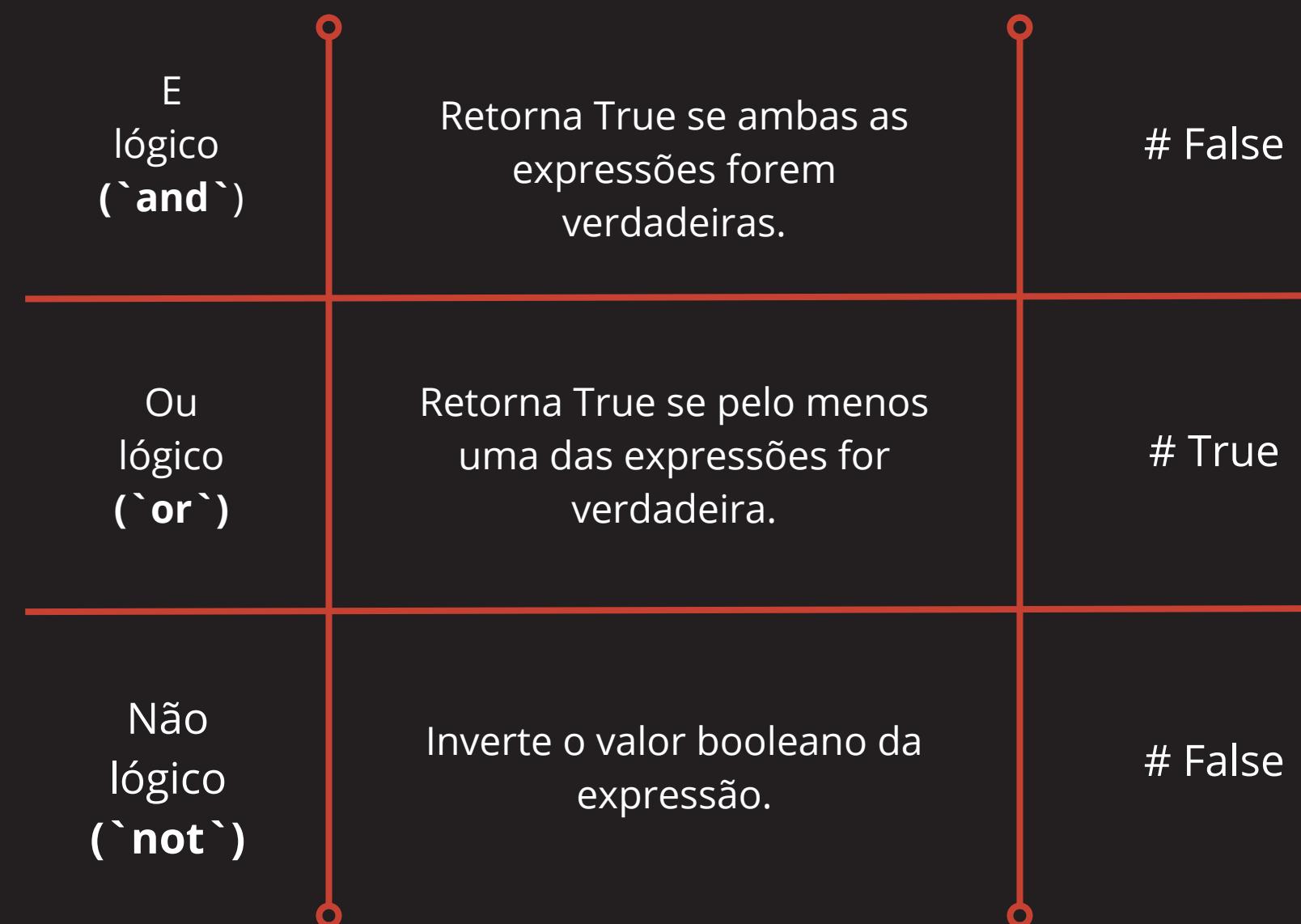
OPERADORES LÓGICOS

Combinam expressões booleanas e retornam um valor booleano.

Exemplos:



```
1 a = True
2 b = False
3
4 resultado = (a and b) # False
5
6 resultado = (a or b) # True
7
8 resultado = not a # False
```



OPERADORES DE ATRIBUIÇÃO

Atribuem valores a variáveis.

Exemplos:



1 `x = 10`

2

3 `x += 5 # x = x + 5`

4

5 `x -= 3 # x = x - 3`

6

7 `x *= 2 # x = x * 2`

8

9 `x /= 2 # x = x / 2`

Atribuição simples	(=)	Atribui o valor do lado direito à variável do lado esquerdo
Atribuição com adição	(+=)	Soma e atribui o resultado à variável
Atribuição com subtração	(-=)	Subtrai e atribui o resultado à variável.
Atribuição com multiplicação	(*=)	Multiplica e atribui o resultado à variável.
Atribuição com divisão	(/=)	Divide e atribui o resultado à variável.

OPERADORES DE ASSOCIAÇÃO

Verificam se um valor ou variável está presente em uma sequência.

Como uma string, lista, tupla, conjunto ou dicionário.

Exemplos:



```
1 fruta1 = "maça"
2 fruta2 = "banana"
3 fruta3 = "cereja"
4 resultado = "maça" in (fruta1, fruta2, fruta3)
5 print(resultado) # True
```



```
1 fruta1 = "maça"
2 fruta2 = "banana"
3 fruta3 = "cereja"
4 resultado = "laranja" not in (fruta1, fruta2, fruta3)
5 print(resultado) # True
```

'in'

Verifica se um valor
está presente na
sequência.

Retorna booleano
True or False

'not in'

Verifica se um valor
não está presente na
sequência.

Retorna booleano
True or False

IN

ATIVIDADE PRÁTICA

Atividade 01:

Comparação de Idades:

- Peça ao usuário duas idades e use operadores de comparação para verificar se a primeira idade é maior, menor ou igual à segunda.

Atividade 02:

Verificar Igualdade de Strings:

- Peça ao usuário duas palavras e use operadores de comparação para verificar se elas são iguais.

ATIVIDADE PRÁTICA

Atividade 03:

Verificação de Maioridade e Habilitação:

- Crie um programa que peça a idade do usuário e se ele possui habilitação (sim ou não). Use operadores lógicos para verificar se ele é maior de idade (≥ 18) e possui habilitação.

Atividade 04:

Verificação de Notas Aprovadas:

- Escreva um programa que peça duas notas de um aluno. Use operadores lógicos para verificar se as duas notas são maiores ou iguais a 6.

ATIVIDADE PRÁTICA

Atividade 05:

Desconto em Preço:

- Peça ao usuário para inserir o preço de um produto e, em seguida, use o operador de atribuição `-=` para aplicar um desconto de 5%.

Atividade 06:

Dobro do Valor:

- Solicite ao usuário um número e use o operador de atribuição `*=` para dobrar o valor e exibir o resultado.

ATIVIDADE PRÁTICA

Atividade 07:

Verificação de Caracteres em uma String:

- Crie um programa que peça ao usuário uma frase e um caractere.
Use o operador de associação `in` para verificar se o caractere está presente na frase.

Atividade 08:

Verificação de Palavras em uma Frase:

- Peça ao usuário para inserir uma frase e uma palavra.
Use `in` para verificar se a palavra está na frase.

INDENTAÇÃO

Em Python, a indentação é obrigatória e define a estrutura do código.

Explicação:

Importância: Facilita a leitura e manutenção do código, e é essencial para a sintaxe de controle de fluxo, como loops e funções.

Como Indentar: Use espaços ou tabulações de forma consistente, geralmente quatro espaços por nível de indentação. Indente após comandos com dois pontos (como if, for, while, def).

Regras: Mantenha a consistência na escolha de espaços ou tabulações e aumente a indentação em cada novo bloco de código.

01

Bloco 01

02

Bloco 02

03

Bloco 03

04

Bloco 02

05

Bloco 01

06

07

08

09

10

11

IN

ESTRUTURA CONDICIONAL

Estruturas condicionais em Python, como if, elif, e else, permitem que o código execute instruções baseadas em condições específicas.

Importância das Estruturas Condicionais:

Controle de Fluxo: Gerenciam a execução do programa.

Flexibilidade: Adaptam o código a diversas situações.

Decisões Lógicas: Facilitam a implementação de decisões, tornando o código dinâmico.

Conceito:

If = “SE”

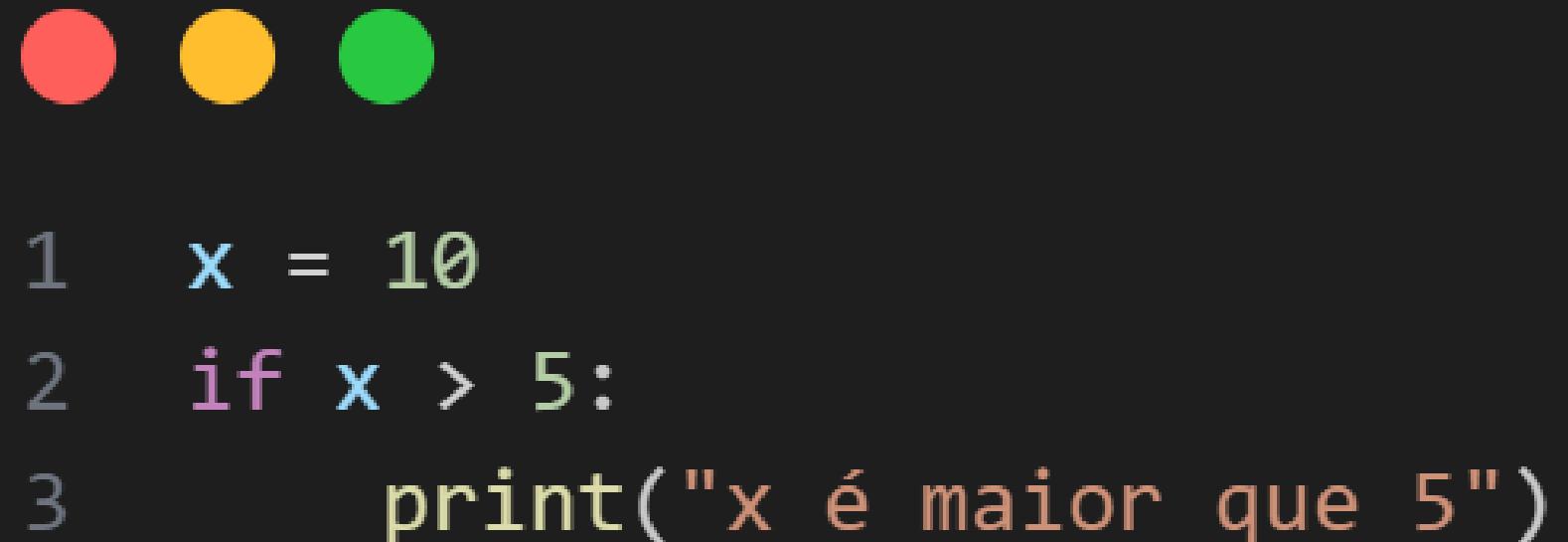
Elif = “Senão Se”

Else = Senão

ESTRUTURA CONDICIONAL `IF`

A estrutura `if` verifica uma condição e, se essa condição for verdadeira (True), o bloco de código dentro do `if` é executado.

Exemplo:



```
1 x = 10
2 if x > 5:
3     print("x é maior que 5")
```

Neste exemplo, a mensagem "x é maior que 5" será exibida porque a condição $x > 5$ é verdadeira.

ESTRUTURA CONDICIONAL 'ELIF'

A estrutura `elif`, que significa "else if", permite verificar múltiplas condições.

Se a condição `if` não for verdadeira, o Python verifica a condição do `elif`.

Você pode ter quantos `elif` forem necessários.

Exemplo:



```
1 x = 10
2 if x > 15:
3     print("x é maior que 15")
4 elif x > 5:
5     print("x é maior que 5, mas menor ou igual a 15")
```

Neste exemplo, a segunda mensagem será exibida porque $x > 5$ é verdadeira, mas $x > 15$ é falsa.

ESTRUTURA CONDICIONAL `ELSE`

A estrutura `else` é usada para executar um bloco de código quando nenhuma das condições anteriores (`if` ou `elif`) é verdadeira.

Exemplo:



```
1 idade = int(18)
2
3 if idade < 12:
4     print("Você é uma criança.")
5 elif 12 <= idade < 18:
6     print("Você é um adolescente.")
7 else:
8     print("Você é um adulto.")
```

Aqui, a mensagem "Você é um adulto" será exibida porque a condição idade igual ou maior 18 é verdadeira.

Estruturas Condicionais Aninhadas

Você pode aninhar estruturas condicionais dentro de outras para criar verificações mais complexas.

Exemplo:



```
1 nota = 49
2
3 if nota >= 90:
4     print("A")
5 elif nota >= 80:
6     print("B")
7 elif nota >= 70:
8     print("C")
9 elif nota >= 60:
10    print("D")
11 elif nota >= 50:
12    print("E")
13 else:
14     print("Reprovado")
```

Aqui, a mensagem "Reprovado" será exibida porque a condição nota é inferior ao if e aos elif

ATIVIDADE PRÁTICA

Atividade 09:

Verificar Par ou Ímpar:

- Crie um programa que peça ao usuário um número e use a estrutura condicional if para verificar se ele é par ou ímpar.

Atividade 10:

Verificar Nota para Aprovado:

- Crie um programa que peça a nota de um aluno e use if para verificar se ele foi aprovado (nota ≥ 6).

ATIVIDADE PRÁTICA

Atividade 11:

Verificar Par ou Ímpar e Positivo ou Negativo:

- Escreva um programa que peça um número e use if para verificar se ele é par ou ímpar e também se é positivo ou negativo.

Atividade 12:

Verificar Classificação de IMC:

- Crie um programa que calcule o Índice de Massa Corporal (IMC) e use if para classificar o resultado em "Abaixo do peso", "Peso normal", "Sobrepeso" e "Obesidade".

ATIVIDADE PRÁTICA

Atividade 13:

Sistema de Login Simples:

- Desenvolva um programa que peça ao usuário um nome de usuário e uma senha e use if para verificar se são iguais a "admin" e "1234", respectivamente.

Atividade 14:

Verificar Status de Taxa de Desconto:

- Crie um programa que peça ao usuário o preço original de um produto e a quantidade comprada. Use if para verificar se a quantidade é maior que 10 para aplicar um desconto de 10% sobre o total.

**SE LIGA NO
CONTEÚDO DA
PRÓXIMA AULA!**

AULA 03 Estrutura de repetição (WHILE)



INFINITY SCHOOL
VISUAL ART CREATIVE CENTER

Aula 03 - Estrutura de repetição (WHILE)

- ESTRUTURA DE REPETIÇÃO
- INTRODUÇÃO AO WHILE
- WHILE + CONDICIONAIS
- CONDIÇÃO DE TÉRMINO
- ANINHAMENTO DE WHILE
- LOOPS INFINITY

A woman with long dark hair, wearing glasses, is shown from the chest up, looking down at a laptop screen. The background is dark and slightly blurred.

IN

INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

AULA 02 - Condicionais