

## Aula 04

### Estrutura Condicional

Operadores relacionais - São símbolos que indicam a comparação entre dois valores.

OPERADOR	FUNÇÃO
>	Maior que
<	Menor que
>=	Maior ou igual
<=	Menor ou igual
!=	Diferente
==	Igual

O resultado da comparação entre dois valores só pode retornar um valor do **tipo Boolean**, ou seja, **true (V)** ou **false (F)**.

Operadores lógicos - Os operadores lógicos são usados quando se deseja comparar duas condições ou mais. Para isto é necessário usar os operadores AND, OR ou NOT.

Condições 1 e 2		Resultados		
C1	C2	C1 and C2	C1 or C2	not C2
V	V	V	V	F
V	F	F	V	V
F	V	F	V	
F	F	F	F	

O resultado da comparação entre duas condições só pode retornar um valor do **tipo Boolean**, ou seja, **true (V)** ou **false (F)**.

### Estrutura Condicional

Estrutura Condicional é aquela caracterizada pela tomada de uma decisão. Esta decisão é tomada a partir de um teste lógico, onde uma condição é testada e como resultado só há duas possibilidades de respostas: True (Verdadeiro) ou False (Falso). Independente da condição ser verdadeira ou falsa, um conjunto de comandos será executado pelo programa. A seguir são apresentados os tipos de estruturas condicionais: Estrutura condicional simples, Estrutura condicional composta e Estrutura case

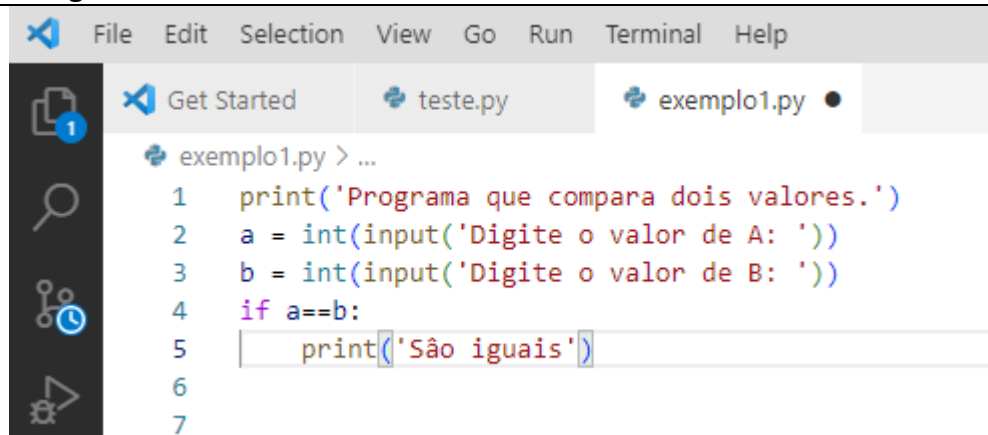
## Estrutura condicional simples

É utilizada quando se deseja executar um conjunto de comandos somente quando o resultado do teste lógico de uma condição for verdadeiro.

Sintaxe 1	Sintaxe 2
<b>if</b> condição : Comando1	<b>if</b> condição : Comando1 Comando2
<b>Execução:</b> Caso a condição seja verdadeira, o comando 1 será executado; Caso a condição seja falsa, o comando 1 não será executado.	<b>Execução:</b> Caso a condição seja verdadeira, o comando 1 e o comando 2 serão executados; Caso a condição seja falsa, nem o comando 1 e nem o comando 2 serão executados.

## Exemplos

### Código-fonte 1



```

File Edit Selection View Go Run Terminal Help
Get Started teste.py exemplo1.py
exemplo1.py > ...
1 print('Programa que compara dois valores.')
2 a = int(input('Digite o valor de A: '))
3 b = int(input('Digite o valor de B: '))
4 if a==b:
5     print('São iguais')
6
7
  
```

### Execução

Se a condição é verdadeira a instrução é executada.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

PS C:\Users\Cibele\Desktop\Python_arq> python exemplo1.py
Programa que compara dois valores.
Digite o valor de A: 2
Digite o valor de B: 2
são iguais
PS C:\Users\Cibele\Desktop\Python_arq>
  
```

Se a condição é falsa a instrução não é executada.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

PS C:\Users\Cibele\Desktop\Python_arq> python exemplo1.py
Programa que compara dois valores.
Digite o valor de A: 3
Digite o valor de B: 2
PS C:\Users\Cibele\Desktop\Python_arq>
  
```

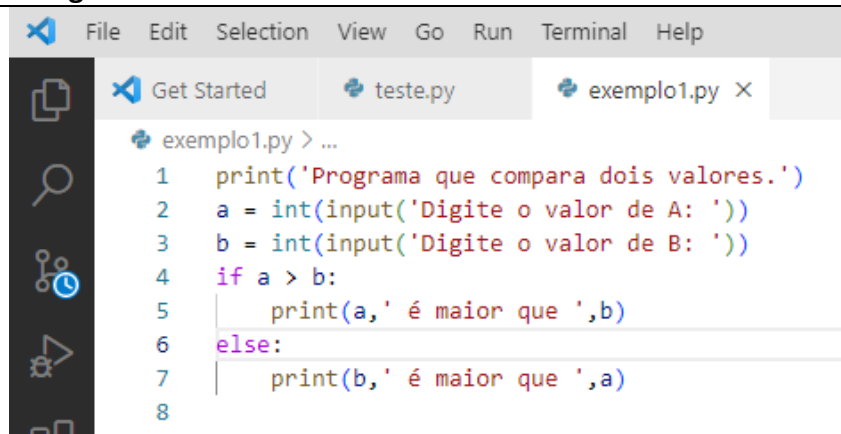
## Estrutura condicional composta

É utilizada para executar um conjunto de comandos quando o resultado da condição for verdadeiro e outro conjunto diferente de comandos quando o resultado da condição for falso.

Sintaxe 1	Sintaxe 2
<b>if</b> condição: Comando1 <b>else</b> Comando2	<b>if</b> condição: Comando1 Comando2  <b>else</b> Comando3 Comando4

## Exemplos

### Código-fonte



```

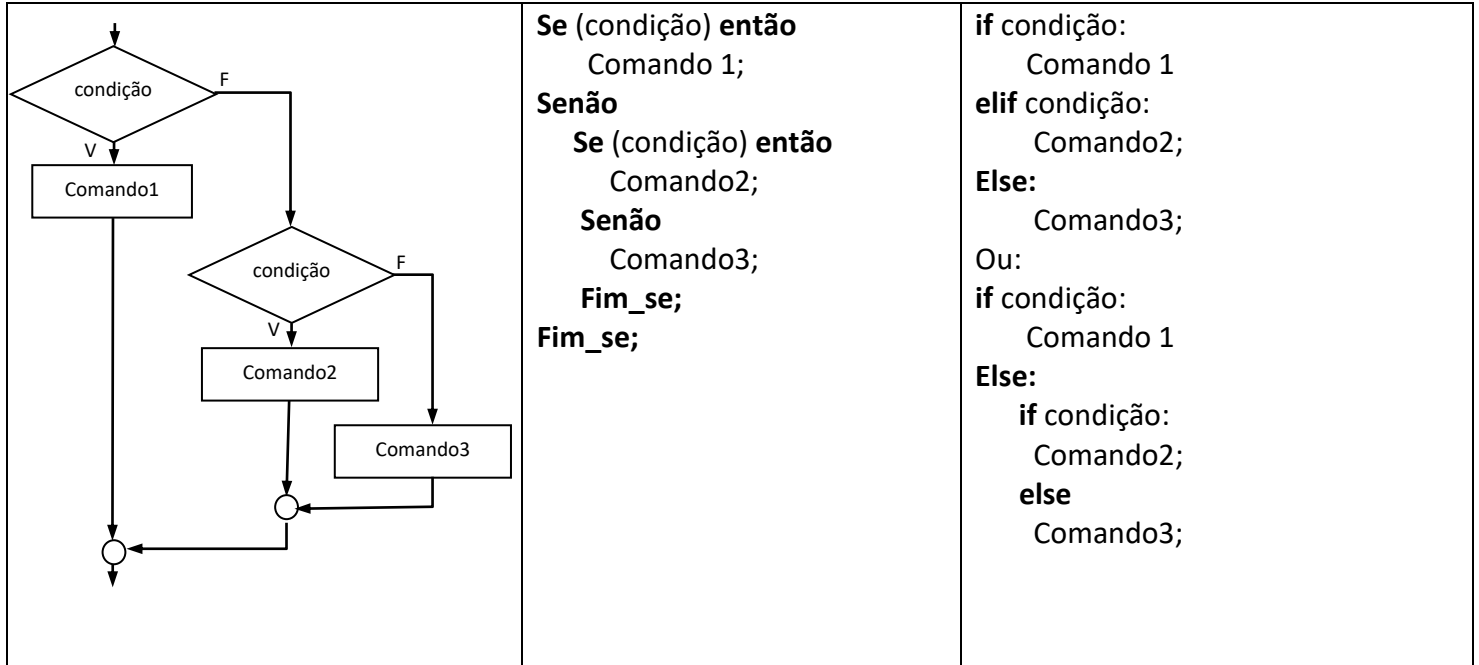
1  print('Programa que compara dois valores.')
2  a = int(input('Digite o valor de A: '))
3  b = int(input('Digite o valor de B: '))
4  if a > b:
5      print(a, ' é maior que ',b)
6  else:
7      print(b, ' é maior que ',a)
8

```

### Execução

PROBLEMS	OUTPUT	DEBUG CONSOLE	TERMINAL
			PS C:\Users\Cibele\Desktop\Python_arq> python exemplo1.py Programa que compara dois valores. Digite o valor de A: 3 Digite o valor de B: 6 6 é maior que 3 PS C:\Users\Cibele\Desktop\Python_arq>
			PS C:\Users\Cibele\Desktop\python\Aulas\aula1> python exemplo1.py Programa que compara dois números Digite o primeiro número: 11 Digite o primeiro número: 5 11 é maior que 5 PS C:\Users\Cibele\Desktop\python\Aulas\aula1>

## Estrutura condicional aninhada (ou encadeada)



### Exemplo 1:

```
static void Main(string[] args)
{
    int n;

    Console.Write("Digite um número: ");
    n = int.Parse(Console.ReadLine());

    if (n > 20)
        Console.WriteLine("O número que você digitou é maior do que 20.");
    else
        if (n < 20)
            Console.WriteLine("O número que você digitou é menor do que 20.");
        else
            Console.WriteLine("O número que você digitou é igual a 20.");
    Console.ReadKey();
}
```

### Exemplo 2:

```

File Edit Selection View Go Run Terminal Help
exemplo1.py - aula1 - Visual Studio Code

EXPLORER
AULA1
  exemplo1.py
  teste.py

exemplo1.py > ...
1 print("Pograma que verifica se um número é maior, menor ou igual a 20")
2 n=int(input("Digite o primeiro número: "))
3 if n > 20:
4     print(n," é maior que 20.")
5 elif n < 20:
6     print(n," é menor que 20.")
7 else:
8     print(n," é igual a 20.")
    
```