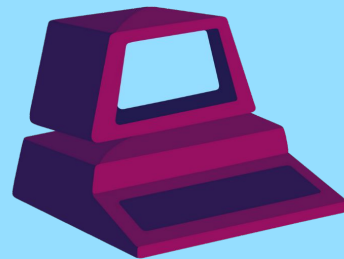


Aula 3 - Turma Avançado

- Estruturas Condicionais

PET



COMPUTAÇÃO

2026

1.

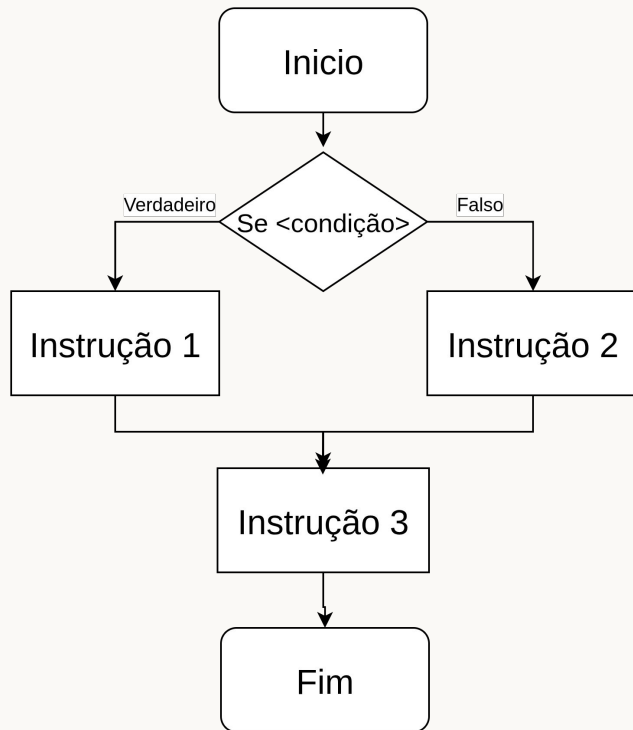
Entendendo as Estruturas

Estruturas Condicionais

- Podemos pensar nelas como “caminhos” que determinarão o fluxo de execução do nosso código.



Fluxo de execução



2.

**Quais Estruturas Condicionais
existem?**

if, elif e else

- São as principais estruturas condicionais que permitem executar determinados blocos do código, se baseando em determinada condição.



```
1  if condicao_1:
2      #bloco 1
3  elif condicao_2:
4      #bloco 2
5  else:
6      #bloco 3
```

Match case

- É uma estrutura de controle de fluxo que permite comparar um valor com múltiplos padrões de forma mais elegante e legível.



```
1  expressao = "valor"
2
3  match expressao:
4      case x:
5          #bloco de codigo
6      case y:
7          #bloco de codigo
8      case z:
9          #bloco de codigo
10     case _:
11         #caso padrao
```

Indentação

- É o uso de espaços ou tabulações que permite deixar o código mais organizado e legível, separando claramente os blocos de código;
- **Obrigatória no Python!**
- Geralmente, os editores de texto fazem a indentação automaticamente, mas caso seja necessário arrumar use o Tab.



Indentação

Código bem indentado



```
1 a = 5
2 b = input("Digite um número: ")
3 if a == b:
4     print("Acertou!")
5 else:
6     print("Errou...")
```

Código mal indentado



```
1 a = 5
2 b = input("Digite um número: ")
3 if a == b:
4 print("Acertou!")
5 else:
6 print("Errou...")
```

3.

Operadores lógicos e de
comparação

Operadores de comparação

- == — igual
- != — diferente
- > — maior que
- < — menor que
- >= — maior ou igual que
- <= — menor ou igual que



```
1 comparacao = a == b
2 comparacao = a != b
3 comparacao = a > b
4 comparacao = a < b
5 comparacao = a >= b
6 comparacao = a <= b
```

Operadores de comparação

- == — igual
- != — diferente
- > — maior que
- < — menor que
- >= — maior ou igual que
- <= — menor ou igual que

Mas qual é o resultado dessas operações?



Operadores de comparação

- == — igual
- != — diferente
- > — maior que
- < — menor que
- >= — maior ou igual que
- <= — menor ou igual que

Mas qual é o resultado dessas operações?

Verdadeiro ou **falso!**



1	a	==	b
2	a	!=	b
3	a	>	b
4	a	<	b
5	a	>=	b
6	a	<=	b

Operadores lógicos

- Variáveis e expressões booleanas só podem assumir o valor **True** ou **False** (verdadeiro ou falso).
- Usando operadores lógicos podemos **modificar** ou **combinar** essas expressões para criar condições mais complexas.

Operadores lógicos

- **not** — inverte o valor lógico da expressão;
- **and** — verdadeiro se, e somente se, todas as expressões forem verdadeiras;
- **or** — verdadeiro se pelo menos uma das expressões for verdadeira.

Operadores lógicos

Exemplos:

- **not** — Hoje **não** é domingo
- **and** — Hoje é sábado **e** está chovendo
- **or** — Hoje é sábado **ou** está chovendo (**não exclusivo**)

Tabela verdade

A	B	not A	not B	A and B	A or B
V	V	F	F	V	V
V	F	F	V	F	V
F	V	V	F	F	V
F	F	V	V	F	F

4.

Trabalhando com Estruturas Condicionais

Trabalhando com if, elif e else



```
1  A = True
2  B = False
3
4  if B and A:
5      print("False and True")
6  elif not (A and not B):
7      print("not (True and not B)")
8  else:
9      print("Nenhuma das condições anteriores")
```

Trabalhando com match case



```
1 nivel = "basico"
2
3 match nivel:
4     case "basico":
5         print("Este é o nível basico")
6     case "intermediario":
7         print("Este é o nível intermediario")
8     case "avancado":
9         print("Este é o nível avanzado")
10    case _:
11        print("Este é o caso padrão")
```

5.

Exercícios

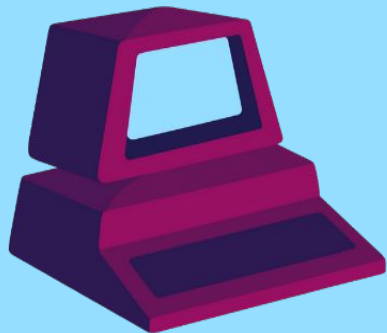
Exercício 1

- Crie um programa em Python que:
 - Peça ao usuário para digitar dois números;
 - Armazene esses números em variáveis;
 - Utilize uma estrutura com if, elif e else para comparar os dois números;
 - Imprima se a é maior que b, se a é menor que b ou se a é igual a b.
- Utilize o site OnlineGDB para rodar seu código:
https://www.onlinegdb.com/online_python_compiler

Exercício 2

- Crie um programa em Python que:
 - Simule um menu de um Sistema de Cadastro de Alunos com as opções:
 1. Cadastrar aluno
 2. Listar alunos
 3. Remover aluno
 4. Sair
 - Peça ao usuário para digitar uma opção;
 - Utilize uma estrutura condicional (if, elif e else ou match case) para tratar cada opção digitada;
 - Imprima uma mensagem correspondente à opção escolhida (ex: "Aluno cadastrado com sucesso!").
- Utilize o site OnlineGDB para rodar seu código:
https://www.onlinegdb.com/online_python_compiler

Obrigado!



PET
COMPUTAÇÃO

pet.inf.ufpr.br
pet@inf.ufpr.br
@petcompufpr