# AULA 17

## • Revisão da aula 15 e 16

#### o Aula 15

- Explicação do projeto final e seus critérios técnicos e de layout
- Entrega:
  - Até a aula 20
  - Até 22h (horário de Brasília)

#### o Aula 16

- Divulgação da premiação e seu regulamento
- Abertura do formulário na semana da aula 18 pelas monitoras
- Nova data para a formatura 26/10
- Início de Python
  - Possibilidades de carreiras
  - Exemplo de projeto com automação em Python
  - Adaptar o vscode para trabalhar com Python

- Tema Drácula
- Exemplo de print e programa para calcular a idade

#### Estrutura condicional if else

- Permite que um programa tome decisões com base em condições específicas
- Executa um bloco de códigos se uma condição for verdadeira
- E outro bloco de códigos se a condição for falsa
- Sintaxe if else no PYTHON

```
if condição:
    # Código a ser executado se a condição for verdadeira
else:
    # Código a ser executado se a condição for falsa
```

Sintaxe if else no JAVASCRIPT

```
if (condição) {
    // Código a ser executado se a condição for verdadeira
} else {
    // Código a ser executado se a condição for falsa
}
```

 Programa para verificar se um número é par ou ímpar

```
numero = int(input('Digite um número inteiro: '))
if numero % 2 == 0:
    print('O número ' + str(numero) + ' é par!')
else:
    print('O numero ' + str(numero) + ' é ímpar!')
```

- Programa para verificar se um número é par ou ímpar, e maior que 10
  - String formatada

```
numero = int(input('Digite um número inteiro: '))

par = numero % 2 == 0

maior_que_10 = numero > 10

if par:
    tipo_numero = 'par'

else:
    tipo_numero = 'impar'
```

```
if maior_que_10:
    status = 'maior que 10'
else:
    status = 'não é maior que 10'

'''
Utilização de string formatada
Facilita a inserção de variáveis
'''
print(f'O número {numero} é {tipo_numero} e {status}.')
```

## Outro exemplo que funciona, mas sem string formatada:

```
numero = int(input('Digite um número inteiro: '))

if numero % 2 == 0:
    if numero > 10:
        print('O número ' + str(numero) + ' é par e maior

que 10!')
    else:
        print('O número ' + str(numero) + ' é par, mas

não é maior que 10!')

else:
    if numero > 10:
        print('O número ' + str(numero) + ' é ímpar e

maior que 10!')
    else:
        print('O número ' + str(numero) + ' é ímpar, mas

não é maior que 10!')
```

## Estrutura de repetição for

- Permite que um programa execute um bloco de códigos quantas vezes forem definidas
- Automatiza tarefas que envolvem repetição
- Sintaxe for no PYTHON

```
for elemento in sequência:
# Código a ser executado para cada elemento na sequência
```

- Programa que mostra números de 1 a
   10
  - Por qual número quer iniciar? (1)
  - Até que número (índice)? (10)
    - O índice 10, corresponde ao elemento 11 na tabela

```
for numero in range(1, 11):
    print(numero)
```

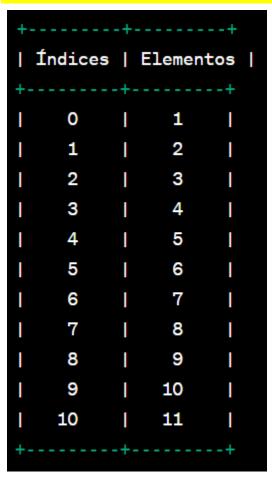
- Programa que imprime os número de
   0 a 6 com incremento de 2
  - Por qual número quer iniciar? (0)

- Até que número (índice)? (6)
  - O índice 6, corresponde ao elemento 7

```
for numero in range(0, 7, 2):

print(numero)
```

#### Índices x Elementos



#### Estrutura de repetição while

 Permite que um bloco de códigos seja executado repetidas vezes, enquanto

# uma determinada condição for verdadeira

Sintaxe while no PYTHON

```
while condição:
    # Código a ser repetido enquanto a condição for verdadeira
```

- Programa que calcula o fatorial de um número escolhido pelo usuário
  - Explicar o que é fatorial
    - Fatorial de 4
    - 4 \* 3 \* 2 \* 1 = 24

```
fatorial = 1
contador = 1

numero = int(input("Digite um número para calcular o
fatorial: "))

while contador <= numero:
         fatorial = fatorial * contador
         contador = contador + 1

print(f'O fatorial de {numero} é {fatorial}')</pre>
```

- o Fazer o teste de mesa
- o O que é teste de mesa?

É um procedimento feito à mão para verificar a coerência de um algoritmo específico

```
Passo 1:
contador (1) <= numero (4) SIM
fatorial = 1 * 1 = 1
contador = 1 + 1 = 2
Passo 2:
contador (2) <= numero (4) SIM
fatorial = 1 * 2 = 2
contador = 2 + 1 = 3
Passo 3:
contador (3) <= numero (4) SIM
fatorial = 2 * 3 = 6
contador = 3 + 1 = 4
Passo 4:
contador (4) <= numero (4) SIM
fatorial = 6 * 4 = 24
contador = 4 + 1 = 5
Passo 5:
contador (5) <= numero (4) NÃO
```

Qual a diferença entre FOR e WHILE?

- For: usado para repetições com um número conhecido de iterações, geralmente com uma variável de controle que segue um intervalo fixo
- While: usado quando a condição de repetição é avaliada a cada iteração e não é conhecida antecipadamente.
   Continua enquanto a condição for verdadeira

## **DESAFIO**

# Postar a parte 3 do projeto final

## TAREFA: SEÇÕES 12 A



Programação Python do Zero ao Avançado + 32 Projetos Reais