



PROFESSOR: JEOFTON COSTA

# Duvidas da Aula Anterior?





## Objetivos

- Estrutura de Repetição **for**;
- Função **range()**;
- Declaração **break** e **continue**.





DEFINITE

FOR

1 to 10

BREAK

CONTINUE

1... 2... 3... 4... 5... 6... ..... 10

1  
2  
3  
4  
5  
6  
7  
8  
9  
10

END

Real Python

```
1  
2 """ Implemente um programa que lê uma frase e conta quantas letras "a"  
3 existem na frase digitada. """
```

```
4  
5 frase = input("Digite uma frase e tecla enter").lower()
```

```
6 aux = 0
```

```
7  
8 for letra in frase:
```

```
9     if letra == "a":
```

```
10         aux+=1
```

```
11  
12 if aux>0:
```

```
13     print(f'Na frase:\n "{\033[3m{frase.capitalize()}\033[0m}"\nA letra "a" aparece {aux} vezes')
```

```
14 else:
```

```
15     print(f'INCRÍVEL!!!\nVocê conseguiu digitar uma frase sem a letra "a".')
```

O comando **for** é ideal para percorrer sequencias finitas de dados.

Esta variável serve apenas como auxiliar. Ela não tem o mesma função da variável contadora do comando **while**.

```
15     print(f'INCRÍVEL!!!\nVocê conseguiu digitar uma frase sem a letra "a".')
```

```
16
```

```
17 |
```

Run: laco\_for ×

▶ ↑ C:\Users\jeoft\AppData\Local\Programs\Python\Python312\python.exe D:\00\_UNIPE\_F

⚙ ↓ Digite uma frase e tecle enter *0 rato roue a roupa do rei de roma*

↺ Na frase:

↻ *"0 rato roue a roupa do rei de roma"*

☰ A letra "a" aparece 4 vezes

🖨

🌟 Process finished with exit code 0

🗑



## Exercitando

Implemente um algoritmo que recebe uma palavra e separa todas as letras da palavra digitada.

Implemente um algoritmo que recebe uma palavra e separa todas as letras da palavra digitada horizontalmente.

Implemente um algoritmo que recebe uma palavra, conta todas as letras e informa a letra que apareceu mais vezes.

Implemente um algoritmo que recebe uma frase e verifica se a letra “e” apareceu na frase e quantas vezes apareceu a letra “i”.



## Função range()

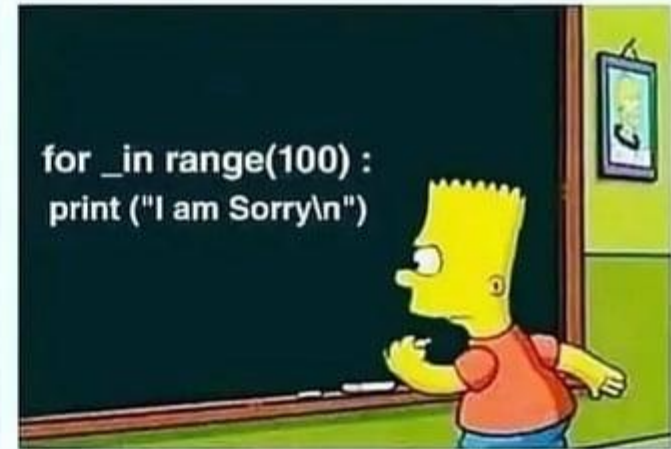
Teacher : Write " I am Sorry " 100 times as punishment .

Normal Students



Programmer

```
for _ in range(100):  
    print ("I am Sorry\n")
```





```
for n in range(6):  
    print(n)
```

Saída.

0  
1  
2  
3  
4  
5

**range()** retorna uma sequência de números, começando por zero (0) e incrementando por 1 até o número especificado.

```
for n in range(2, 6):  
    print(n)
```

Saída.

2  
3  
4  
5

Podemos indicar um outro valor para o início da função **range()**.


```
for n in range(2, 6, 2):  
    print(n)
```

Saída.

2  
4

Podemos determinar um passo a ser obedecido na **range()**.



```
1  
2 """ Implemente um programa que recebe um número e  
3  devolve a tabuada de subtração do número digitado. """  
4
```

```
5 numero = int(input('Digite um número\n'))  
6 for n in range(numero + 1, numero + 11):  
7     subtracao = n - numero  
8     print(f"{n} - {numero} = {subtracao}")  
9
```

A função **range()** serve para percorrer um determinado número de vezes.



Implemente um programa que, recebendo um valor digitado pelo usuário, retorne a respectiva tabuada de adição.

Faça um programa que mostre na tela uma contagem regressiva até zero, a partir do número digitado pelo usuário.

Escreva um programa que receba um valor numérico digitado pelo usuário e informe a quantidade de números pares e ímpares.

## Fortalecendo o aprendizado



Implemente um programa que, recebendo um valor digitado pelo usuário e cada três números, escreva um número contido no intervalo entre 1 e o número digitado.

Implemente um programa que receba um número digitado pelo usuário e realize a soma dos números pares existentes no respectivo intervalo, entre o valor *default* da função `range()` e o número digitado.



# for

Possível determinar o fim da  
contagem ou iteração

# while

O encerramento depender de  
uma ação do usuário.



# Sintaxe do **for** em outras linguagens

## Java

```
for(int i = 0; i < 10; i++){  
    System.out.println(i);  
}
```

## Javascript

```
JS script.js > ...  
1  
2  ✓ for(let i = 1; i <=9; i++){  
3    |   console.log(i)  
4    |   }  
5
```

## PHP

```
for ($x = 0; $x <= 10; $x++) {  
    echo "The number is: $x <br>";  
}
```

## C#

```
for(int i = 0; i < 10; i++)  
{  
    Console.WriteLine(i);  
}
```



## Continue V/s break statement

```
while (condition)
statement
statement
continue
statement
statement
```

```
while ( condition)
statement
statement
break
statement
statement
```



DECLARAÇÕES **break** E **continue**

```

1  from time import sleep
2
3  for n in range(1, 10):
4      print(f"O valor de número na {n}ª passagem: {n}")
5      sleep(2)
6      if n%6 == 0:
7          print(f'O resto de {n}/6 é 0. Se 0 é igual a 0, a condição do "if" ficou true.')
8          print('Isto fez a execução do algoritmo entrar no bloco "if".')
9          print('Fazendo acionar a declaração "break", e parar a sequência.')
10         break
11     print(f"Fim do programa após número igual a {n}.\n\nLinha while")
12
13     i = 0
14     while i < 7:
15         print(f"O valor de i na {i+1}ª passagem: {i}")
16         sleep(2)
17         if i == 4:
18             print(f'O número {i} fez a condição do "if" ficar verdadeira.')
19             print('Isto fez a execução do algoritmo entrar no bloco "if".')
20             print('Fazendo acionar a declaração "break", e parar a sequência.')
21             break
22         i += 1
23     print(f"Fim do programa após número igual a {n}.\n\n")
24

```

**Sintaxe da declaração break.**



```

1  from time import sleep
2
3  for n in range(1, 10):
4      sleep(1)
5      if n%6 == 0:
6          print(f'0 resto de {n}/6 é 0. Se 0 é igual a 0, a condição do "if" ficou true.')
7          print('Isto fez a execução do algoritmo entrar no bloco "if".')
8          print('Fazendo acionar a declaração "continue". O trecho de código abaixo não será exibido.')
9          continue
10         print(f"0 valor de número na {n}ª passagem: {n}")
11     print(f"Fim do programa após número igual a {n}.\n\nLinha while")
12
13     i = 0
14     while i < 7:
15         sleep(1)
16         i += 1
17         if i == 4:
18             print(f'0 número {i} fez a condição do "if" ficar verdadeira.')
19             print('Isto fez a execução do algoritmo entrar no bloco "if".')
20             print('Fazendo acionar a declaração "continue". O trecho de código abaixo não será exibido.')
21             continue
22         print(f"0 valor de i na {i}ª passagem: {i}")
23     print(f"Fim do programa após número igual a {i}.\n\n")
24

```

**Sintaxe da declaração `continue`.**

Implemente um programa que leia uma frase contendo a palavra “feio”. Quando o programa identificar esta palavra, o programa deve parar a leitura da frase.

Desenvolva um algoritmo que receba um valor digitado pelo usuário, sempre que o programa encontrar um múltiplo de 11 (onze) ele deve escrever algo divertido e seguir para o próximo número.

Escreva um programa que leia uma frase e sempre que for encontrada uma vogal, ele deve seguir para a próxima iteração.

## Exercitando





## USO DE IMAGENS DE PERSONAGENS DE QUADRINHOS PARA FINS ESTRITAMENTE ACADÊMICOS

"As imagens de personagens de quadrinhos utilizadas neste material têm o propósito exclusivo de enriquecer o conteúdo acadêmico da apresentação. O uso está em conformidade com as leis de direitos autorais, pois se enquadra no contexto educacional. Respeitamos os direitos dos detentores dessas propriedades intelectuais."

