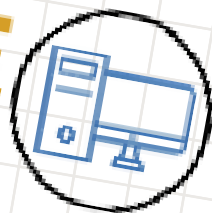


CODE  
LAB TEEN

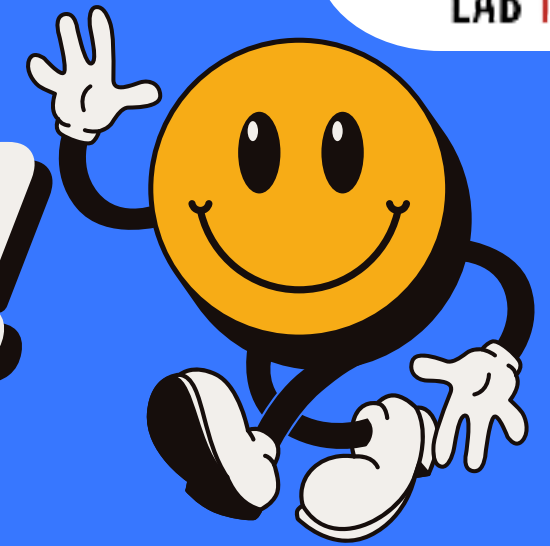


# PYTHON AULA 17





# ***BEM - VINDOS!***



## **AGENDA**

**Hoje vamos revisar o uso de for e de strings!**

# ***FOR-REVISÃO***

O for é usado para fazer algo repetidamente. Ele ajuda por exemplo, a percorrer uma lista, palavra, ou números e executa uma tarefa para cada item, sem você precisar repetir o código. Em vez de escrever a mesma coisa várias vezes, o for faz o trabalho por você!

# FOR-REVISÃO

Diferente do **while**, usado em repetições em que **não** sabemos quantas vezes o comando será feito e depende de uma **condição**, no **for**, o numero de repetições é determinado por uma **sequência**!

```
while i <= 3:
```

O loop **while** continua enquanto **i** for menor ou igual a 3.


```
for i in range(1, 4):
```

Nesse loop o **for** gera os números de 1 a 3.

# FOR-EXEMPLO

No **for**, o numero de repetições é determinado por uma **sequência**!


```
for numero in range(1, 4):  
    print(numero)
```

 Variável que vai ser atualizada cada vez que o for se repete

# FOR-EXEMPLO

No **for**, o numero de repetições é determinado por uma **sequência**!

```
for numero in range(1, 4):  
    print(numero)
```




Função usada  
para gerar uma  
sequência de  
números

# FOR-EXEMPLO

No **for**, o numero de repetições é determinado por uma **sequência**!

```
for numero in range(1, 4):  
    print(numero)
```



**Números  
que  
definem a  
repetição**

# FOR-EXEMPLO

No **for**, o numero de repetições é determinado por uma **sequência**!

```
for numero in range(1, 4):  
    print(numero)
```



O primeiro número  
é o valor inicial da  
variável



# FOR-EXEMPLO

No **for**, o numero de repetições é determinado por uma **sequência**!

```
for numero in range(1, 4):  
    print(numero)
```

O segundo número  
é onde a variável  
acaba e não inclui  
esse valor no  
resultado final

# FOR-EXEMPLO

No **for**, o numero de repetições é determinado por uma **sequência**!

Assim, esse programa terá  
como saída:

```
for numero in range(1, 4):  
    print(numero)
```

1

2

3

# ***STRING-REVISÃO***

Uma `string` é basicamente um texto. Quando você está trabalhando com textos, como nomes, frases ou qualquer coisa escrita, você usa `strings`.

# STRING-REVISÃO

Para mexer com strings, usamos funções que servem para realizar pequenas operações como:

**LEN** - calcula o comprimento da string

**IN** - verifica se uma palavra está dentro de uma frase

**LOWER** - transforma os caracteres da string em letra minúscula

**UPPER** - transforma os caracteres da string em letra maiúscula

**CAPITALIZE** - transforma o primeiro caractere da string para maiúscula e todos os outros caracteres para minúsculas.

**JOIN** - junta uma lista de strings

**FIND** - procura o índice que queremos de uma determinada string

# STRING-EXEMPLO

```
frase = "a programação é divertida e útil"

frase_upper = frase.upper()

comprimento = len(frase_upper)

palavra = 'DIVERTIDA'
posicao = frase_upper.find(palavra)

print("Frase em maiúsculas:", frase_upper)
print("Comprimento da frase formatada:", comprimento)
print("Posição da primeira ocorrência de 'DIVERTIDA':", posicao)
```

→ **TRANSFORMAR A FRASE EM MAIÚSCULO**

→ **CONTAR O NÚMERO TOTAL DE CARACTERES NA FRASE EM MAIÚSCULAS**

→ **PARA LOCALIZAR A POSIÇÃO DA PALAVRA 'DIVERTIDA' NA FRASE EM MAIÚSCULAS**

# STRING-EXEMPLO

```
frase = "a programação é divertida e útil"

frase_upper = frase.upper()

comprimento = len(frase_upper)

palavra = 'DIVERTIDA'
posicao = frase_upper.find(palavra)

print("Frase em maiúsculas:", frase_upper)
print("Comprimento da frase formatada:", comprimento)
print("Posição da primeira ocorrência de 'DIVERTIDA':", posicao)
```

**Assim, esse programa terá  
como saída:**

```
Frase em maiúsculas: A PROGRAMAÇÃO É DIVERTIDA E ÚTIL
Comprimento da frase formatada: 35
Posição da primeira ocorrência de 'DIVERTIDA': 19
```

# FOR E STRING-EXEMPLO

Palavra é uma string que contém "programação"

```
palavra = "PROGRAMACAO"
```

```
for letra in palavra:  
    print(letra.lower())
```

letra.lower() é um método que converte o caractere letra para minúscula, O método lower() é aplicado a cada caractere da string durante o loop

O loop for percorre cada caractere (letra) na string palavra. A cada iteração do loop, a variável letra recebe o próximo caractere da string

# FOR E STRING-EXEMPLO

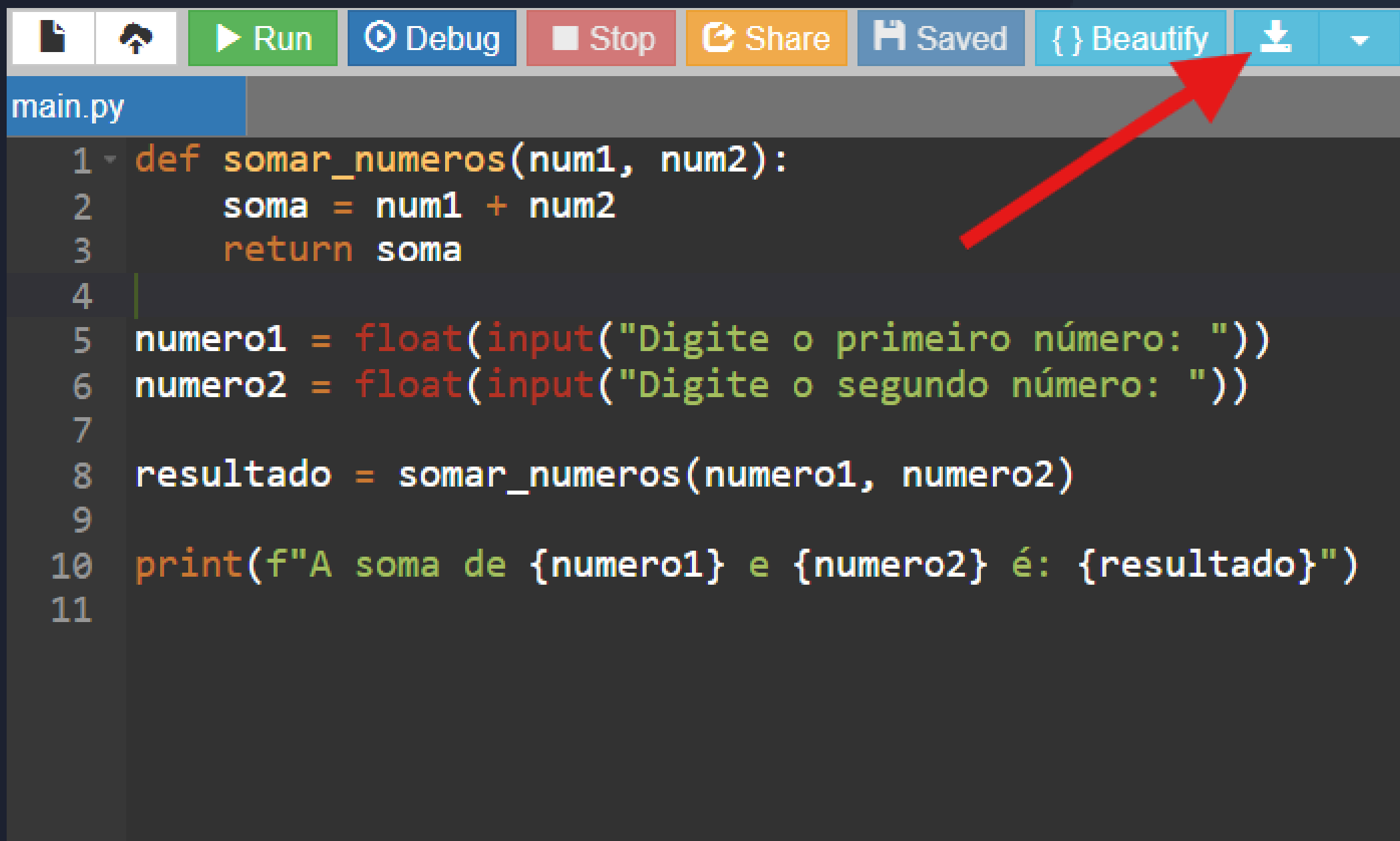
Assim, esse programa terá  
como saída:

```
palavra = "PROGRAMACAO"  
  
for letra in palavra:  
    print(letra.lower())
```

p  
r  
o  
g  
r  
a  
m  
a  
c  
a  
o



# BAIXAR CÓDIGO

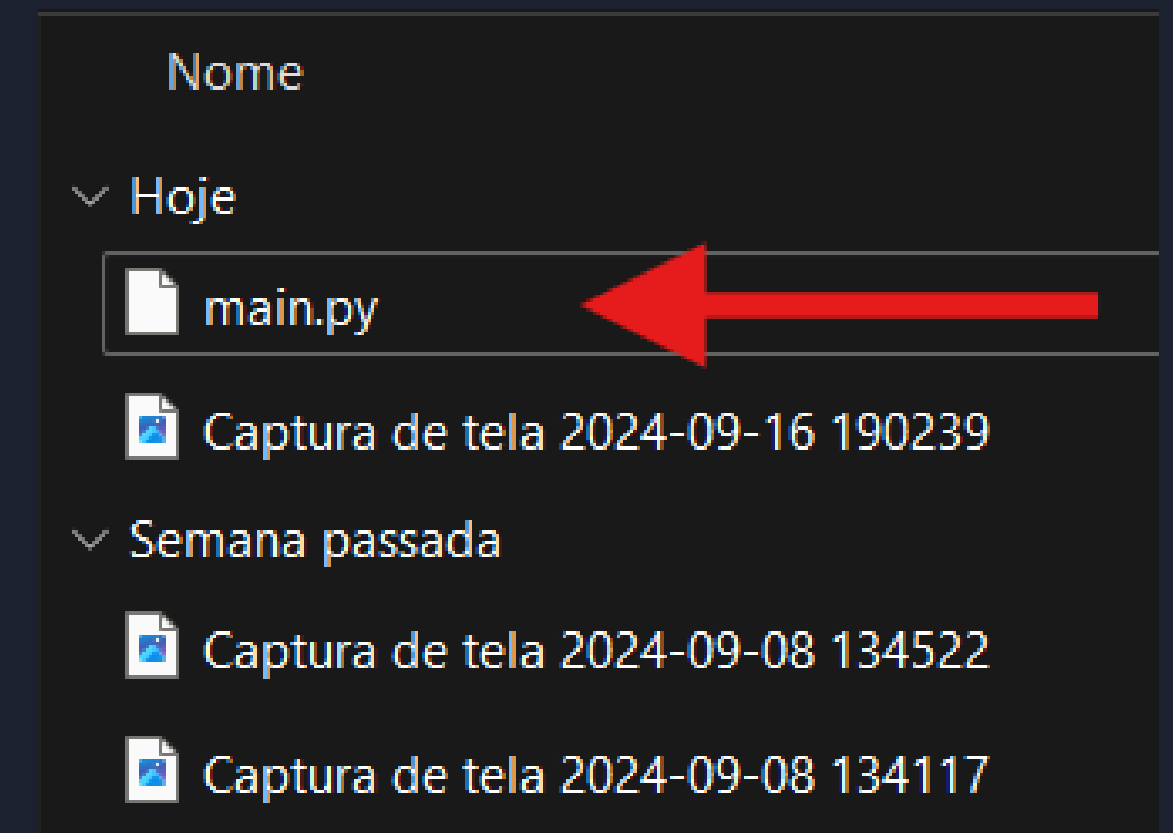
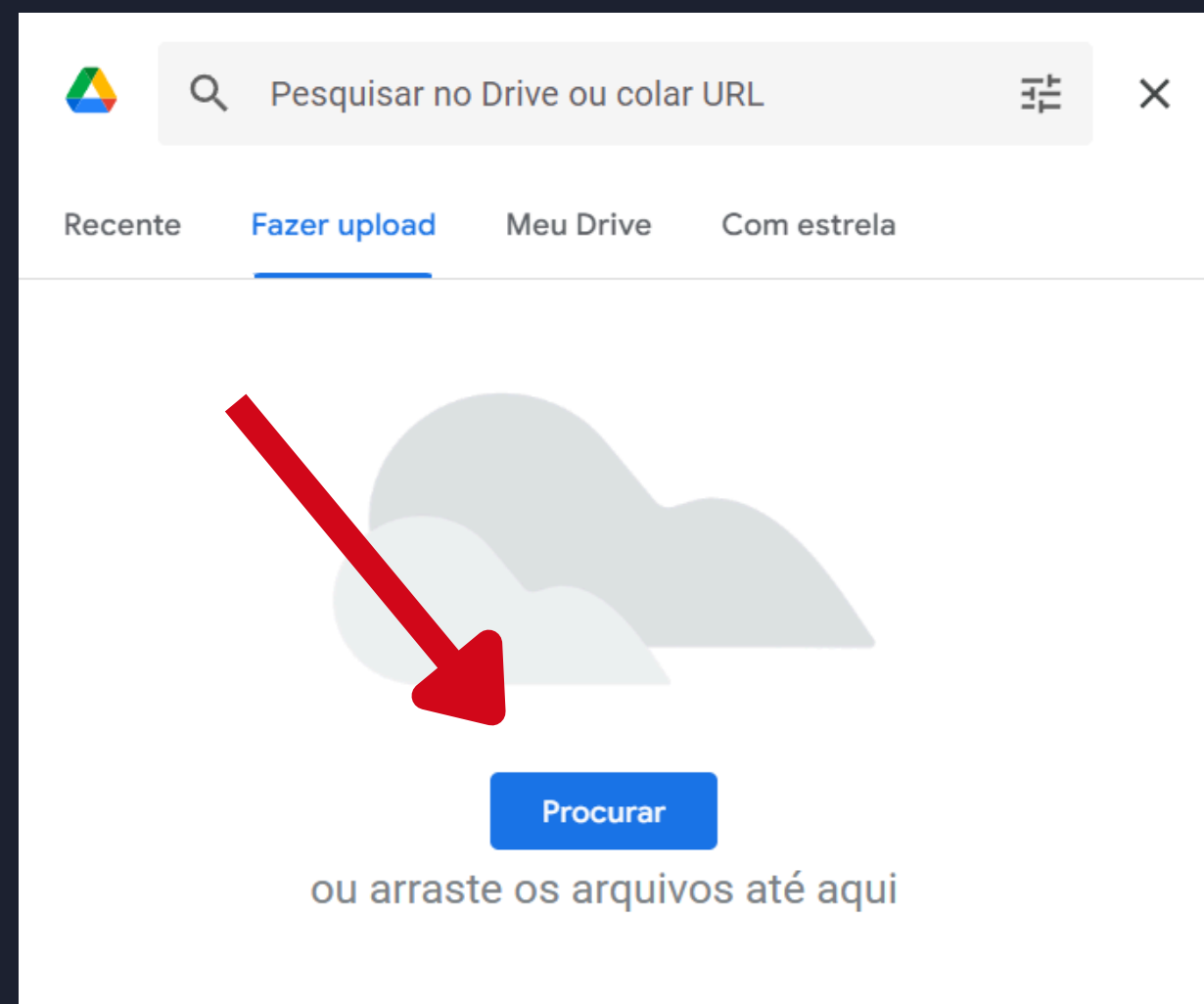
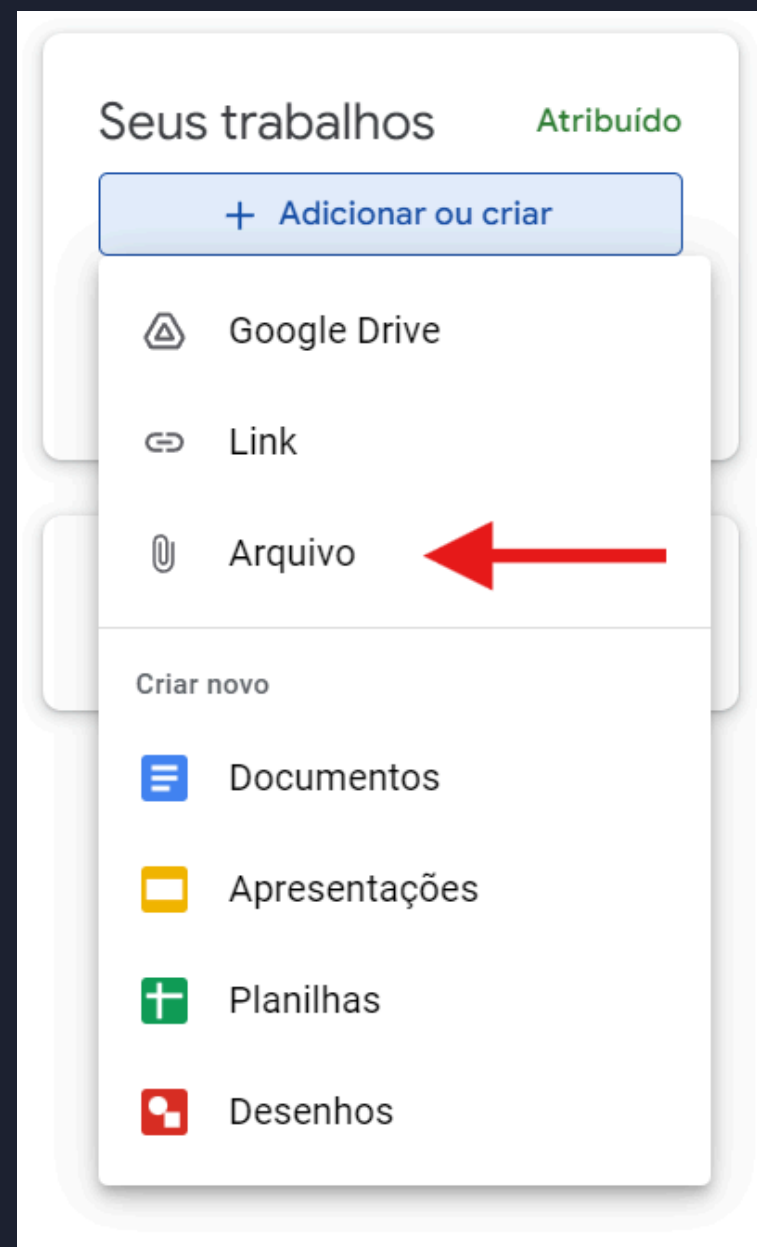


The screenshot shows a code editor window with a toolbar at the top. The toolbar includes buttons for 'Run' (green), 'Debug' (blue), 'Stop' (red), 'Share' (orange), 'Saved' (blue), 'Beautify' (light blue), and a download icon (blue) which is highlighted by a red arrow. Below the toolbar, the file name 'main.py' is displayed. The code editor contains the following Python code:

```
1 def somar_numeros(num1, num2):
2     soma = num1 + num2
3     return soma
4
5 numero1 = float(input("Digite o primeiro número: "))
6 numero2 = float(input("Digite o segundo número: "))
7
8 resultado = somar_numeros(numero1, numero2)
9
10 print(f"A soma de {numero1} e {numero2} é: {resultado}")
11
```

Por padrão o  
arquivo será  
baixado como  
**main.py**

# SUBIR NO GOOGLE- CLASSROOM



# ***EXERCÍCIO 1***

Escreva um código em Python que use um loop *for* para imprimir a tabuada do número 5, começando pelo multiplicador 1 e terminando em 10.

# ***EXERCÍCIO 1 - GABARITO***

```
for i in range(1, 11):  
    resultado = 5 * i  
    print(resultado)
```

# ***EXERCÍCIO 2***

O seu código vai utilizar um loop *for* para percorrer a string "CODELAB" e imprimir cada letra individualmente na tela. Como você faria isso?

# ***EXERCÍCIO 2 - GABARITO***

```
for letra in "CODELAB":  
    print(letra)
```

# ***EXERCÍCIO 3***

Faça um código que peça ao usuário para digitar uma palavra e uma letra. O código deve contar quantas vezes a letra aparece na palavra e imprimir o resultado.

Dica: Use o *for* para fazer essa contagem.

# ***EXERCÍCIO 3 - GABARITO***

```
palavra = input("Digite uma palavra: ")
letra = input("Digite uma letra: ")
contador = 0

for char in palavra:
    if char == letra:
        contador = contador + 1

print(f"A letra '{letra}' aparece {contador} vezes na
palavra '{palavra}'.")
```



# EXERCÍCIO 4

Faça um código que peça ao usuário para digitar uma palavra. O código deve contar quantas vogais (a, e, i, o, u) estão presentes na palavra e imprimir cada vogal encontrada usando *for*.

# EXERCÍCIO 4 - GABARITO

```
palavra = input("Digite uma palavra: ")
vogais = "aeiou"
contador_vogais = 0

print("Vogais encontradas:")
for letra in palavra.lower(): # Converte a palavra
    para minúsculas
    if letra in vogais:
        print(letra)
        contador_vogais = contador_vogais + 1

print(f"Total de vogais encontradas:
{contador_vogais}")
```

# ***EXERCÍCIO 5***

Escreva um código usando *for*, no qual o usuário digite um número, e vai imprimir o seguinte padrão:

Exemplo: usuário digita o valor 5

Saída:

1

22

333

4444

55555

# ***EXERCÍCIO 5 - GABARITO***

```
numero = int(input("Digite um número para realizar o padrão: "))  
for i in range(1, numero+1):  
    print(str(i) * i)
```

# ***EXERCÍCIO 6***

Peça ao usuário para digitar uma frase e conte quantas letras "a" existem na frase, utilizando um laço *for*.

# ***EXERCÍCIO 6 - GABARITO***

```
frase = input("Digite uma frase: ")
contador = 0

for letra in frase:
    if letra in 'a':
        contador += 1

print("A frase tem", contador, "letras a.")
```

# OBRIGADO!

Contem para gente o que você achou da aula de hoje:



<https://forms.gle/u2UwxodjNUrGzbnX9>

**CODELAB TEEN**