

Browser

Programação Básica em Python

Revisão

O QUE REPETIÇÕES

FOR

WHILE

NOVO ASSUNTO

NEXT 

Funções



Funções

$f(x)$



Uma função é uma sequência de comandos que executa alguma tarefa. Usamos geralmente para quebrar o código em blocos ou executar determinada tarefa mais de uma vez.

Na linha 1 criamos uma função que nomeamos de “soma” e atribuímos dois parâmetros (ou argumentos) e finalizamos com “:”.

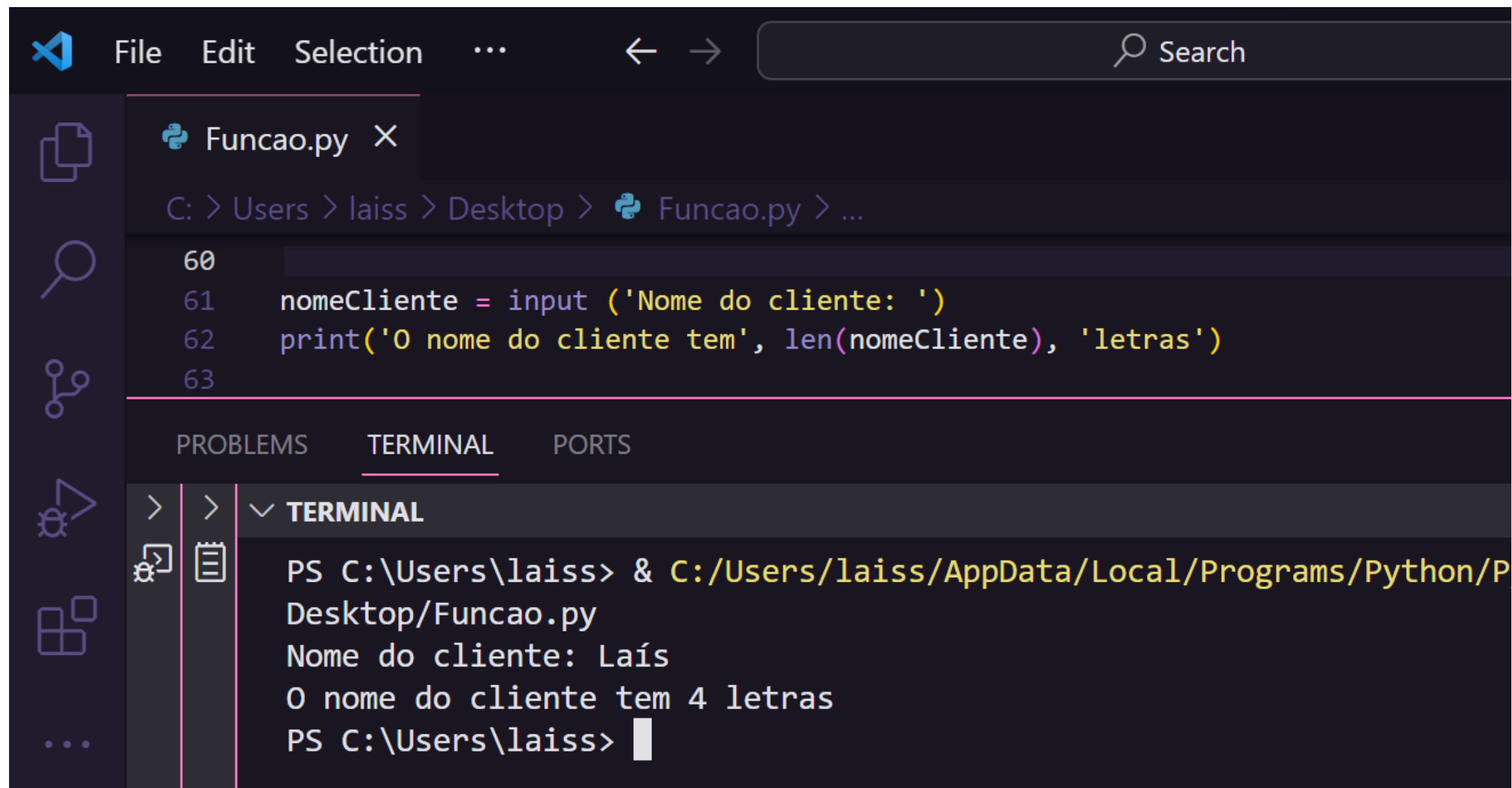
O compilador vai começar a executar o seu código na linha 5 e 6, onde temos as duas variáveis a=2 e b=3.

```
1  def soma(numero1, numero2):  
2      soma = numero1 + numero2  
3      return soma  
4  
5  a = 2  
6  b = 3  
7  resultado = soma(a, b)  
8  print(resultado)
```

A variável **resultado** recebe o retorno da função ou chama a função. O que essa função faz? Ela recebe os dois **parâmetros numero1 e numero2** e faz a operação de soma, logo na linha 3 ela retorna a variável soma para a **função principal**, e em seguida mostra na tela o resultado, na linha 8.

Exemplo de funções útil

Len () -> É uma função que tem o objetivo de retornar o comprimento do item inserido.



The screenshot shows a code editor with a file named 'Funcao.py'. The code in the editor is as follows:

```
60  
61 nomeCliente = input ('Nome do cliente: ')  
62 print('O nome do cliente tem', len(nomeCliente), 'letras')  
63
```

Below the code editor, the 'TERMINAL' tab is active, showing the execution of the script. The output is:

```
PS C:\Users\laiss> & C:/Users/laiss/AppData/Local/Programs/Python/P  
Desktop/Funcao.py  
Nome do cliente: Laís  
O nome do cliente tem 4 letras  
PS C:\Users\laiss>
```

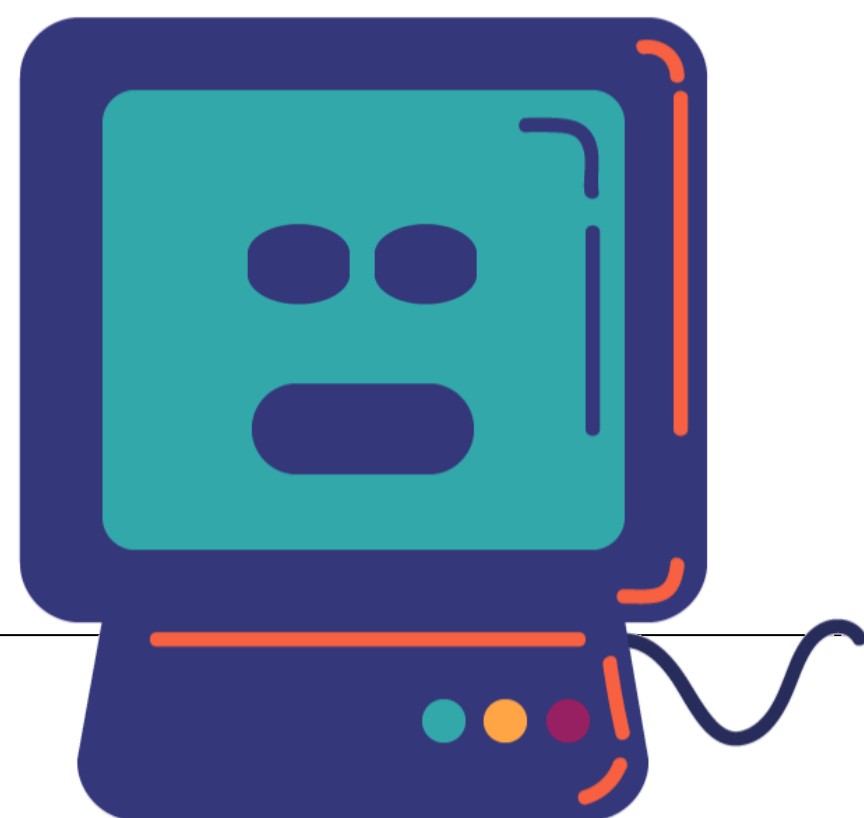
Exercício de Fixação



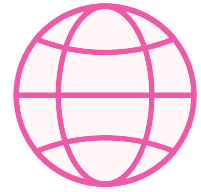
- Faça um programa, com uma função que necessite de três argumentos, e que forneça a soma desses três argumentos.
- Faça um programa com uma função chamada `somaImposto`. A função possui dois parâmetros formais: `taxaImposto`, que é a quantia de imposto sobre vendas expressa em porcentagem e `custo`, que é o custo de um item antes do imposto. A função “altera” o valor de `custo` para incluir o imposto sobre vendas.

Por hoje é só):

Nos vemos na próxima aula



• ° : * **Dúvidas?** ✧ • *



Browser

Programação Básica em Python

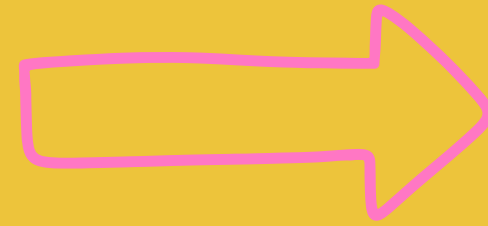
Listas e Tuplas

```
python_tutorial_46.py
1 frutas = ["Laranja", "Abacaxi", "Pera"]
2 print(frutas)
3 frutas[1] = "Morango"
4 print(frutas)
5 frutas[0] = "Banana"
6 print(frutas)
7
```



Listas

Uma lista é uma coleção de valores indexada, em que cada valor é identificado por um índice.



O primeiro item na lista está no índice 0, o segundo no índice 1, etc!

```
1 | programadores = ['Victor', 'Juliana', 'Samuel', 'Caio', 'Luana']
2 | print(type(programadores)) # type 'list'
3 | print(len(programadores)) # 5
4 | print(programadores[4]) # Luana
```

Manipulando listas

Alterando e adicionando
itens de listas

`programadores[1] = 'Lais'`



alteração por index

`programadores.append('Larissa')`



adiciona ao final da lista

`programadores.insert(1, 'Rafael')`



adiciona pelo index (não remove)

Removendo itens da lista

`programadores.remove('Lara')`



remoção pelo valor informado no parâmetro

`programadores.pop(0)`



remoção pelo índice do elemento na lista

Tuplas

São semelhantes as listas. No entanto, possuem a característica de serem imutáveis.
Ou seja, após uma tupla ser criada, ela não pode ser alterada

```
8      0      1      2      3      4      5      6
9  tuple_frutas = ("manzana", "plátano", "cereza", "piña", "limón", "fresa", "uva")
10  print(tuple_frutas[2:5])
11  print(tuple_frutas[-7:-3])
12
```

input

```
('cereza', 'piña', 'limón')
('manzana', 'plátano', 'cereza', 'piña')
```

Exercício de Fixação

- Crie um programa que vai ler vários números e colocar em uma lista.
Depois disso, mostre:

a) Quantos números foram digitados.

b) A lista de valores, ordenados de forma decrescente.

c) Se o valor 5 foi digitado e está ou não na lista.

Exercício de Fixação

Escreva um programa em Python que solicite ao usuário uma lista de números inteiros. Em seguida, verifique se a lista está vazia. Se estiver vazia, exiba uma mensagem informando. Caso contrário, verifique se a lista possui algum número repetido e exiba uma mensagem adequada.

- **Exemplo de saída esperada:**

**>> Digite uma lista de números inteiros
separados por vírgula: 1, 2, 3, 4, 2
>> A lista possui números repetidos.**

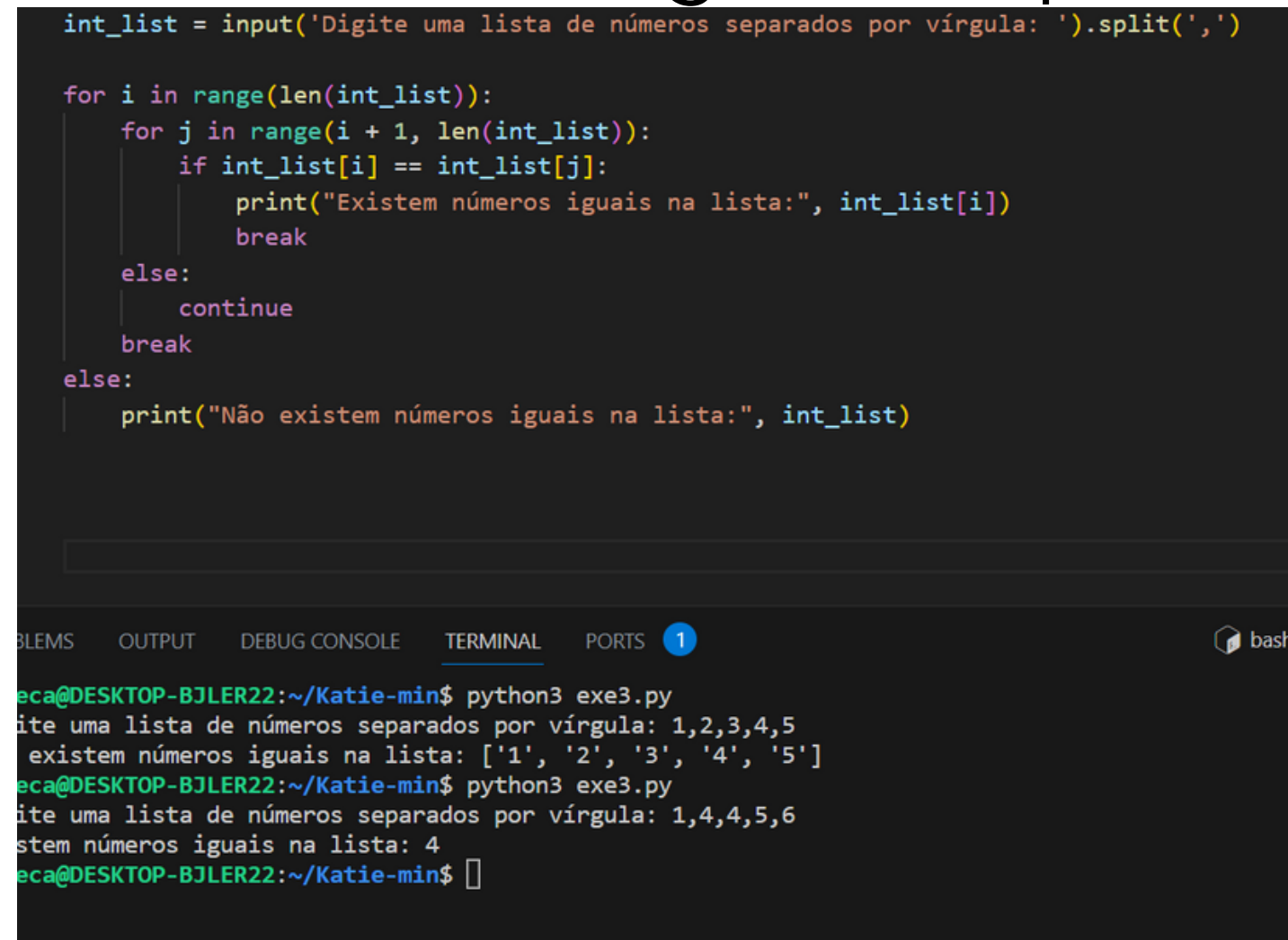
Exercício de Fixação

Escreva um programa em Python que solicite ao usuário uma lista de números inteiros. Em seguida, verifique se a lista está vazia. Se estiver vazia, exiba uma mensagem informando. Caso contrário, verifique se a lista possui algum número repetido e exiba uma mensagem adequada.

- Resolução

```
int_list = input('Digite uma lista de números separados por vírgula: ').split(',')

for i in range(len(int_list)):
    for j in range(i + 1, len(int_list)):
        if int_list[i] == int_list[j]:
            print("Existem números iguais na lista:", int_list[i])
            break
        else:
            continue
    else:
        break
else:
    print("Não existem números iguais na lista:", int_list)
```



The screenshot shows a code editor with the Python code above. Below the code, there is a terminal window with the following output:

```
eca@DESKTOP-BJLER22:~/Katie-min$ python3 exe3.py
Digite uma lista de números separados por vírgula: 1,2,3,4,5
Existem números iguais na lista: ['1', '2', '3', '4', '5']
eca@DESKTOP-BJLER22:~/Katie-min$ python3 exe3.py
Digite uma lista de números separados por vírgula: 1,4,4,5,6
Existem números iguais na lista: 4
eca@DESKTOP-BJLER22:~/Katie-min$
```


Exercício de Fixação

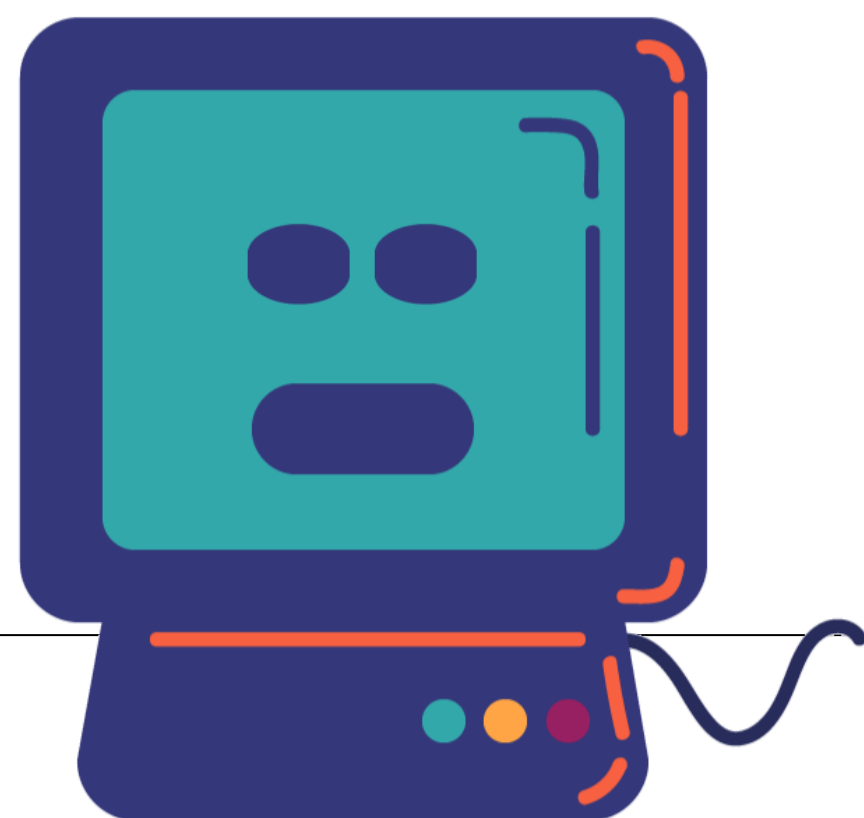


1. Faça um programa, que realize a soma dos elementos da tupla mostre os elementos da tupla e logo após o resultado da soma.
- Dica: Você pode utilizar duas funções para a resolução dessa questão. Funções: `sum()` e `list()`
 - **Exemplo de saída esperada:**

```
>> Elementos da Tupla: (5, 7, 10, 1, 8, 3)  
>> A soma dos elementos: 34
```

Por hoje é só):

Nos vemos na próxima aula



• ° : * Dúvidas? ✧ • *