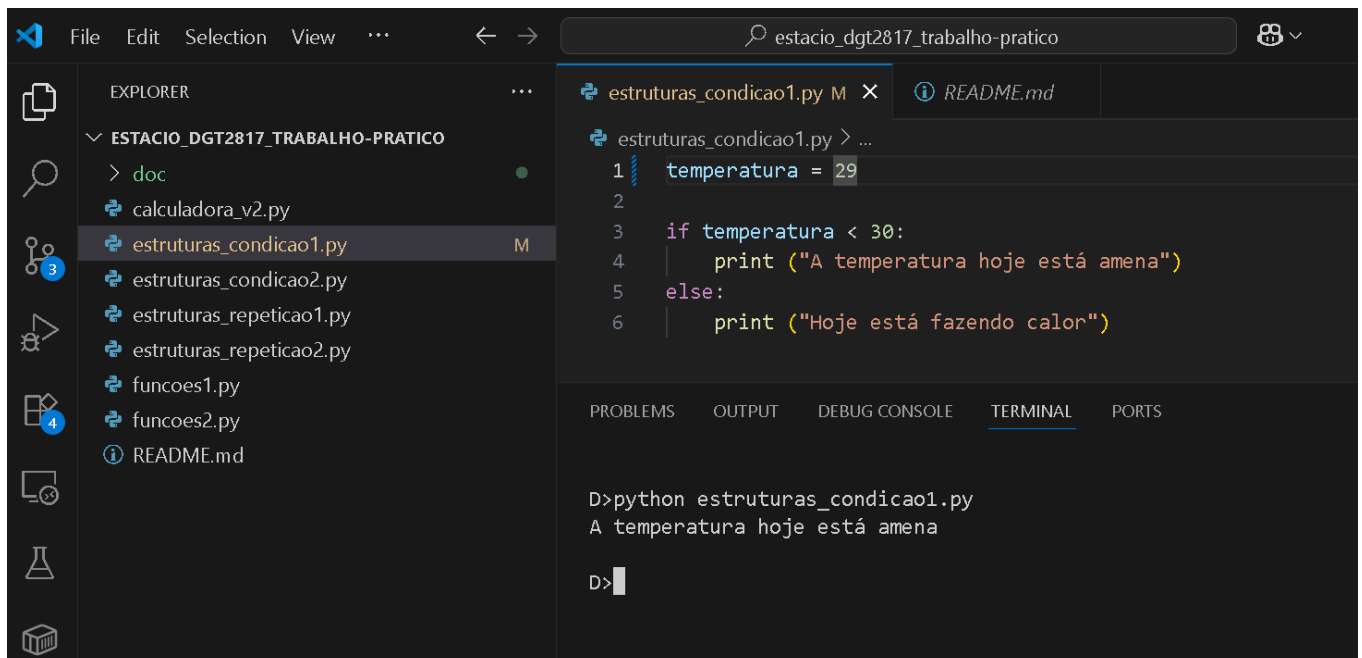


# Trabalho prático

## 1. Micro atividade 1

**Objetivo:** Descrever a utilização das estruturas de condição *if* e *else* em Python

- Temperatura = 29

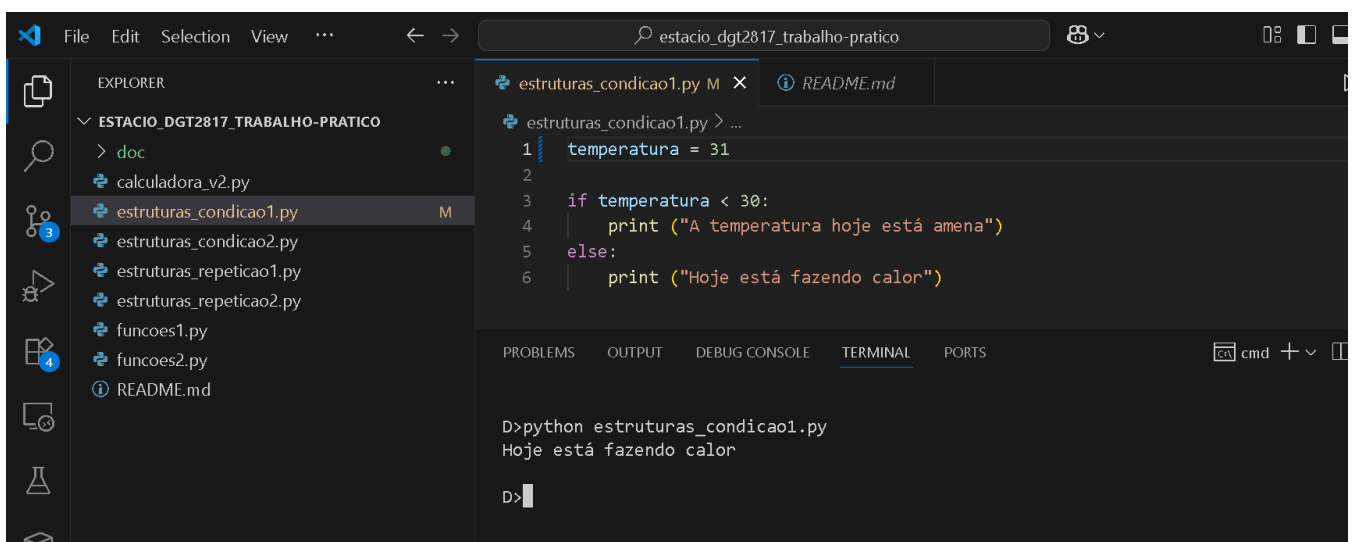


The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays a project named 'ESTACIO\_DGT2817\_TRABALHO-PRATICO' with files including 'doc', 'calculadora\_v2.py', 'estruturas\_condicao1.py' (selected), 'estruturas\_condicao2.py', 'estruturas\_repeticao1.py', 'estruturas\_repeticao2.py', 'funcoes1.py', 'funcoes2.py', and 'README.md'. The main editor shows the code for 'estruturas\_condicao1.py':

```
1 temperatura = 29
2
3 if temperatura < 30:
4     print ("A temperatura hoje está amena")
5 else:
6     print ("Hoje está fazendo calor")
```

The TERMINAL panel at the bottom shows the command 'D>python estruturas\_condicao1.py' and its output 'A temperatura hoje está amena'.

- Temperatura = 31



The screenshot shows the Visual Studio Code interface with the same project. The main editor shows the code for 'estruturas\_condicao1.py' with 'temperatura' set to 31:

```
1 temperatura = 31
2
3 if temperatura < 30:
4     print ("A temperatura hoje está amena")
5 else:
6     print ("Hoje está fazendo calor")
```

The TERMINAL panel shows the command 'D>python estruturas\_condicao1.py' and its output 'Hoje está fazendo calor'.

**Aluno:** Christian Marques de Oliveira Silva

**Matrícula:** 202505154772

**Disciplina:** DGT2817 Lógica, Algoritmos e Programação de Computadores

**Data:** 01/07/2025



## 2. Micro atividade 2

**Objetivo:** Descrever a utilização da estrutura de condição *else if (elif)* em Python

- Tempo de Experiência = 5

The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays a file tree for 'ESTACIO\_DGT2817\_TRABALHO-PRATICO'. The file 'estruturas\_condicao2.py' is selected. The main editor window shows the code for 'estruturas\_condicao2.py' with the following content:

```
1 tempoExperiencia = 5
2
3 if tempoExperiencia < 2:
4     print("Nível de conhecimento júnior.")
5 elif (tempoExperiencia >= 2 and tempoExperiencia < 5):
6     print("Nível de conhecimento pleno.")
7 else:
8     print("Nível de conhecimento sênior.")
```

The TERMINAL panel at the bottom shows the command 'D>python estruturas\_condicao2.py' and the output 'Nível de conhecimento sênior.'

- Tempo de Experiência = 3

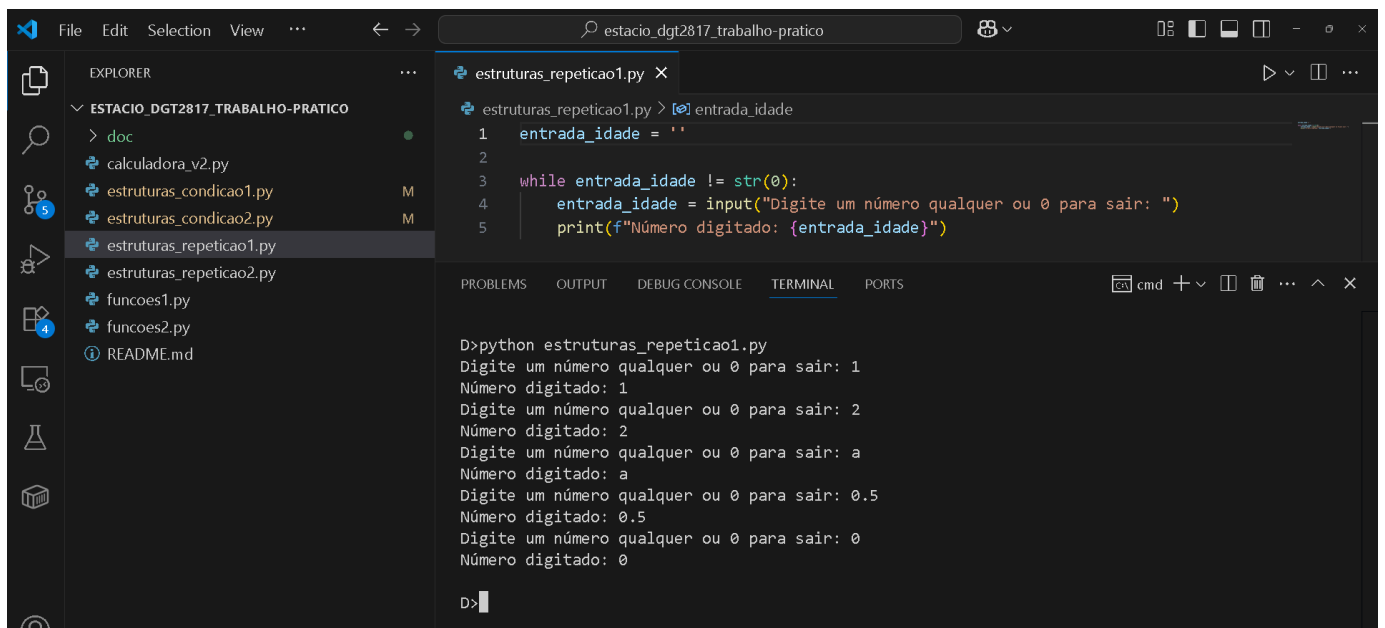
The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays a file tree for 'ESTACIO\_DGT2817\_TRABALHO-PRATICO'. The file 'estruturas\_condicao2.py' is selected. The main editor window shows the code for 'estruturas\_condicao2.py' with the following content:

```
1 tempoExperiencia = 3
2
3 if tempoExperiencia < 2:
4     print("Nível de conhecimento júnior.")
5 elif (tempoExperiencia >= 2 and tempoExperiencia < 5):
6     print("Nível de conhecimento pleno.")
7 else:
8     print("Nível de conhecimento sênior.")
```

The TERMINAL panel at the bottom shows the command 'D>python estruturas\_condicao2.py' and the output 'Nível de conhecimento pleno.'

## 3. Micro atividade 3

**Objetivo:** Descrever a utilização da estrutura de repetição *while* em Python



```
File Edit Selection View ... < -> estacio_dgt2817_trabalho-pratico
```

EXPLORER

- ESTACIO\_DGT2817\_TRABALHO-PRATICO
  - doc
  - calculadora\_v2.py
  - estruturas\_condicao1.py M
  - estruturas\_condicao2.py M
  - estruturas\_repeticao1.py
  - estruturas\_repeticao2.py
  - funcoes1.py
  - funcoes2.py
  - README.md

estruturas\_repeticao1.py X

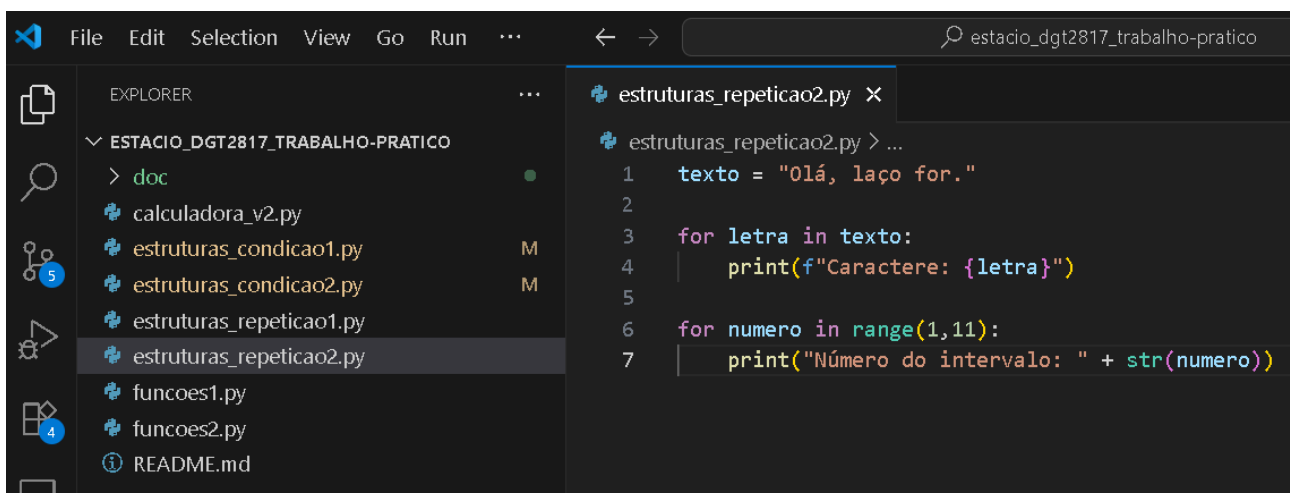
```
estruturas_repeticao1.py > entrada_idade
1 entrada_idade = ''
2
3 while entrada_idade != str(0):
4     entrada_idade = input("Digite um número qualquer ou 0 para sair: ")
5     print(f"Número digitado: {entrada_idade}")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

D>python estruturas\_repeticao1.py  
Digite um número qualquer ou 0 para sair: 1  
Número digitado: 1  
Digite um número qualquer ou 0 para sair: 2  
Número digitado: 2  
Digite um número qualquer ou 0 para sair: a  
Número digitado: a  
Digite um número qualquer ou 0 para sair: 0.5  
Número digitado: 0.5  
Digite um número qualquer ou 0 para sair: 0  
Número digitado: 0  
D>

## 4. Micro atividade 4

**Objetivo:** Descrever a utilização da estrutura de repetição *for* em Python



```
File Edit Selection View Go Run ... < -> estacio_dgt2817_trabalho-pratico
```

EXPLORER

- ESTACIO\_DGT2817\_TRABALHO-PRATICO
  - doc
  - calculadora\_v2.py
  - estruturas\_condicao1.py M
  - estruturas\_condicao2.py M
  - estruturas\_repeticao1.py
  - estruturas\_repeticao2.py
  - funcoes1.py
  - funcoes2.py
  - README.md

estruturas\_repeticao2.py X

```
estruturas_repeticao2.py > ...
1 texto = "Olá, laço for."
2
3 for letra in texto:
4     print(f"Caractere: {letra}")
5
6 for numero in range(1,11):
7     print("Número do intervalo: " + str(numero))
```

**Aluno:** Christian Marques de Oliveira Silva

**Matrícula:** 202505154772

**Disciplina:** DGT2817 Lógica, Algoritmos e Programação de Computadores

**Data:** 01/07/2025



```
File Edit Selection View Go Run ...
estruturas_repeticao2.py
1 texto = "Olá, laço for."
2
D>python estruturas_repeticao2.py
Caractere: 0
Caractere: 1
Caractere: á
Caractere: ,
Caractere: 1
Caractere: a
Caractere: ç
Caractere: o
Caractere:
Caractere: f
Caractere: o
Caractere: r
Caractere: .
Número do intervalo: 1
Número do intervalo: 2
Número do intervalo: 3
Número do intervalo: 4
Número do intervalo: 5
Número do intervalo: 6
Número do intervalo: 7
Número do intervalo: 8
Número do intervalo: 9
Número do intervalo: 10
D>
```

## 5. Micro atividade 5

**Objetivo:** Descrever a utilização de *funções* em Python

```
File Edit Selection View Go Run ...
funcoes1.py
1 def imprimir_variavel():
2     texto = "Olá, funções em Python"
3     print(texto)
4
5 imprimir_variavel()
D>python funcoes1.py
Olá, funções em Python
D>
```

**Aluno:** Christian Marques de Oliveira Silva

**Matrícula:** 202505154772

**Disciplina:** DGT2817 Lógica, Algoritmos e Programação de Computadores

**Data:** 01/07/2025



## 6. Micro atividade 6

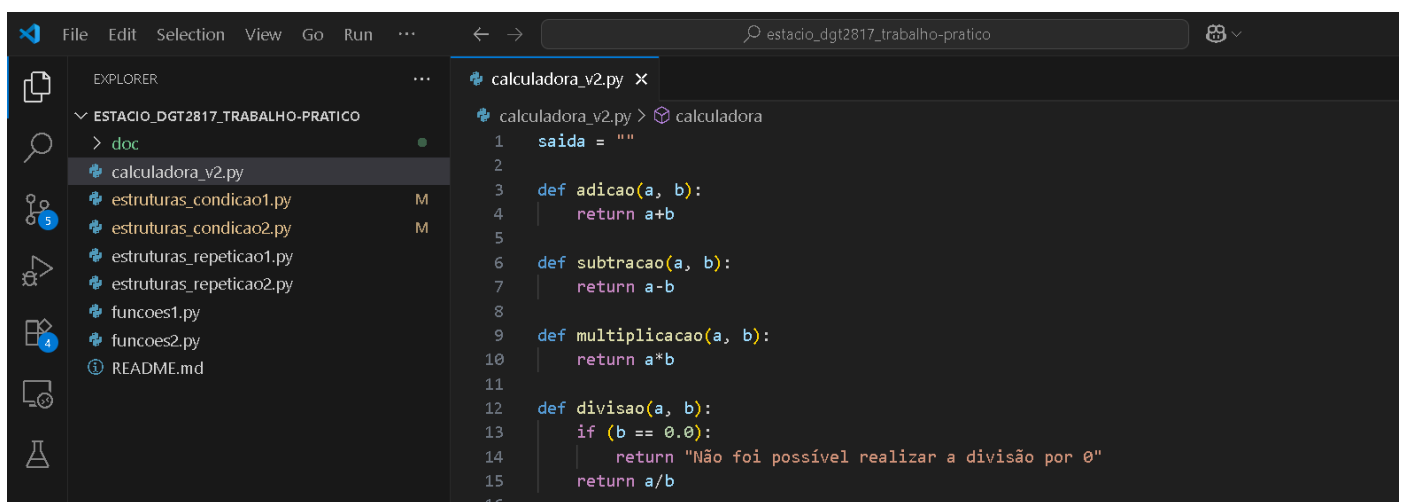
**Objetivo:** Descrever a utilização de *argumentos de funções* em Python

```
File Edit Selection View Go Run ... estado_dgt2817_trabalho-pratico
EXPLORER
ESTACIO_DGT2817_TRABALHO-PRATICO
  > doc
  calculadora_v2.py
  estruturas_condicao1.py
  estruturas_condicao2.py
  estruturas_repeticao1.py
  estruturas_repeticao2.py
  funcoes1.py
  funcoes2.py
  README.md
funcoes2.py > loginUsuario
1 def loginUsuario(perfil):
2     if (perfil.lower() == "admin"):
3         print("Bem-vindo, Administrador")
4     else:
5         print("Bem-vindo, Usuário")
6
7 loginUsuario("Admin")
8 loginUsuario("admin")
9 loginUsuario("User")
10 loginUsuario("usuário")
11 loginUsuario("etc")
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
D>python funcoes2.py
Bem-vindo, Administrador
Bem-vindo, Administrador
Bem-vindo, Usuário
Bem-vindo, Usuário
Bem-vindo, Usuário
D>
```

## 7. Trabalho Prático

**Objetivo:** Refazer a calculadora utilizando estrutura condicional e funções

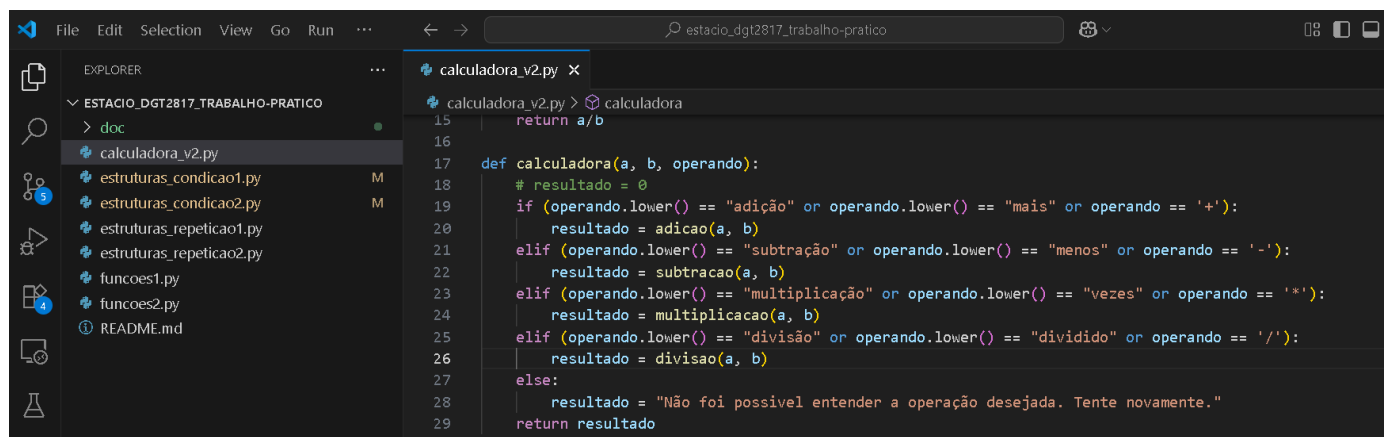
- Código:



```

1  saida = ""
2
3  def adicao(a, b):
4      return a+b
5
6  def subtracao(a, b):
7      return a-b
8
9  def multiplicacao(a, b):
10     return a*b
11
12 def divisao(a, b):
13     if (b == 0.0):
14         return "Não foi possível realizar a divisão por 0"
15     return a/b
16

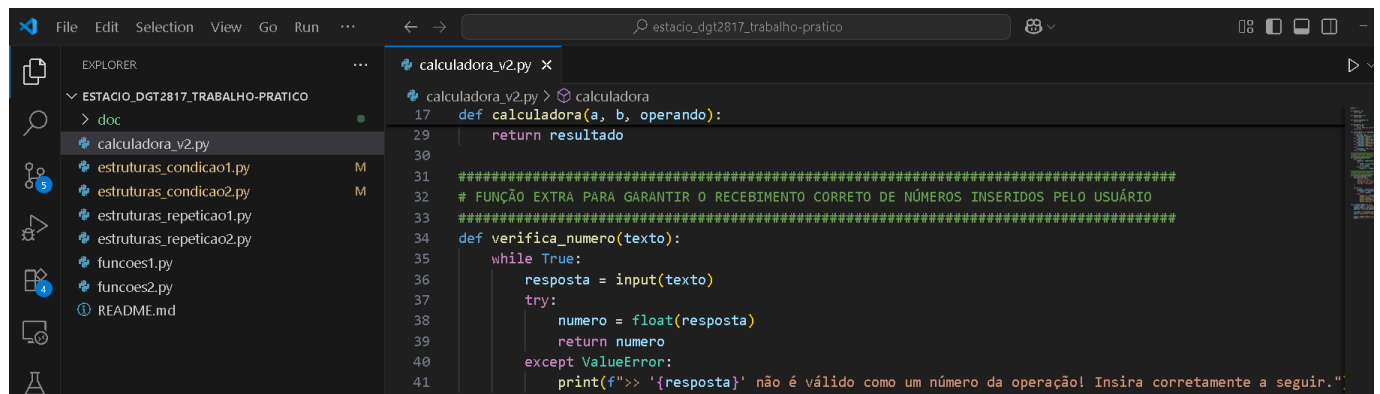
```



```

15     return a/b
16
17 def calculadora(a, b, operando):
18     # resultado = 0
19     if (operando.lower() == "adição" or operando.lower() == "mais" or operando == '+'):
20         resultado = adicao(a, b)
21     elif (operando.lower() == "subtração" or operando.lower() == "menos" or operando == '-'):
22         resultado = subtracao(a, b)
23     elif (operando.lower() == "multiplicação" or operando.lower() == "vezes" or operando == '*'):
24         resultado = multiplicacao(a, b)
25     elif (operando.lower() == "divisão" or operando.lower() == "dividido" or operando == '/'):
26         resultado = divisao(a, b)
27     else:
28         resultado = "Não foi possível entender a operação desejada. Tente novamente."
29     return resultado
30

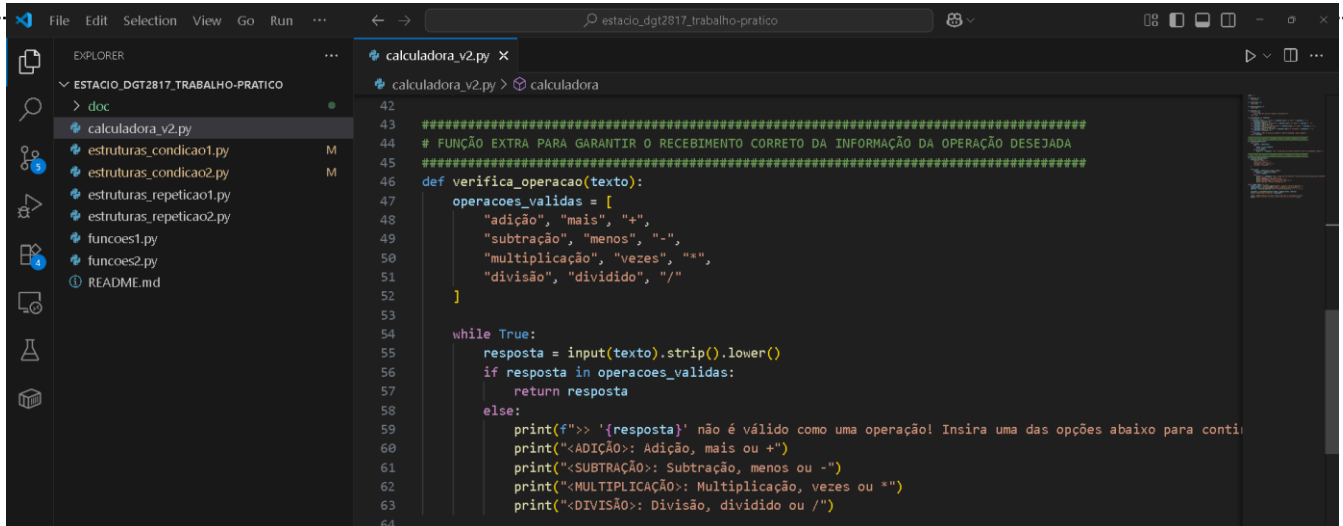
```



```

17 def calculadora(a, b, operando):
18     return resultado
19
20 #####
21 # FUNÇÃO EXTRA PARA GARANTIR O RECEBIMENTO CORRETO DE NÚMEROS INSERIDOS PELO USUÁRIO
22 #####
23 def verifica_numero(texto):
24     while True:
25         resposta = input(texto)
26         try:
27             numero = float(resposta)
28             return numero
29         except ValueError:
30             print(f">> '{resposta}' não é válido como um número da operação! Insira corretamente a seguir.")
31

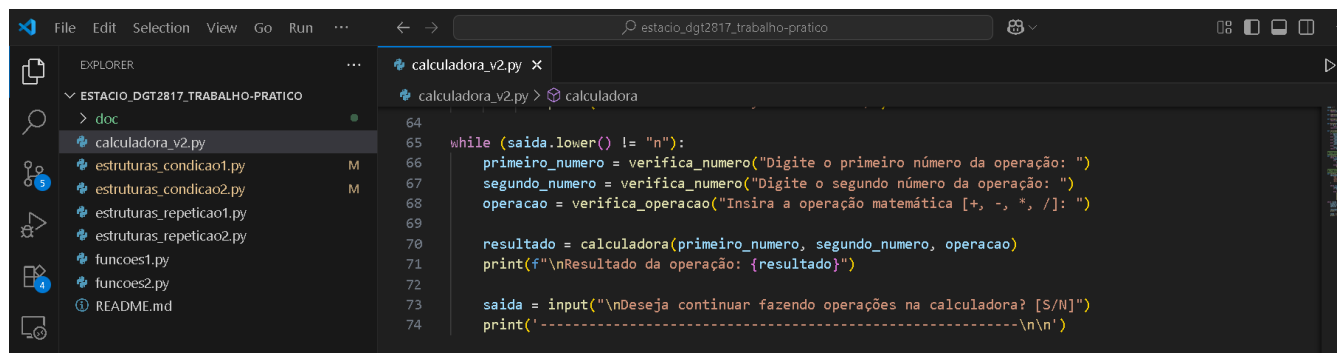
```



```

42
43 #####
44 # FUNÇÃO EXTRA PARA GARANTIR O RECEBIMENTO CORRETO DA INFORMAÇÃO DA OPERAÇÃO DESEJADA
45 #####
46 def verifica_operacao(texto):
47     operacoes_validas = [
48         "adição", "mais", "+",
49         "subtração", "menos", "-",
50         "multiplicação", "vezes", "*",
51         "divisão", "dividido", "/"
52     ]
53
54     while True:
55         resposta = input(texto).strip().lower()
56         if resposta in operacoes_validas:
57             return resposta
58         else:
59             print(f">>> '{resposta}' não é válido como uma operação! Insira uma das opções abaixo para conti
60             print("<ADIÇÃO>: Adição, mais ou +")
61             print("<SUBTRAÇÃO>: Subtração, menos ou -")
62             print("<MULTIPLICAÇÃO>: Multiplicação, vezes ou *")
63             print("<DIVISÃO>: Divisão, dividido ou /")
64
65

```

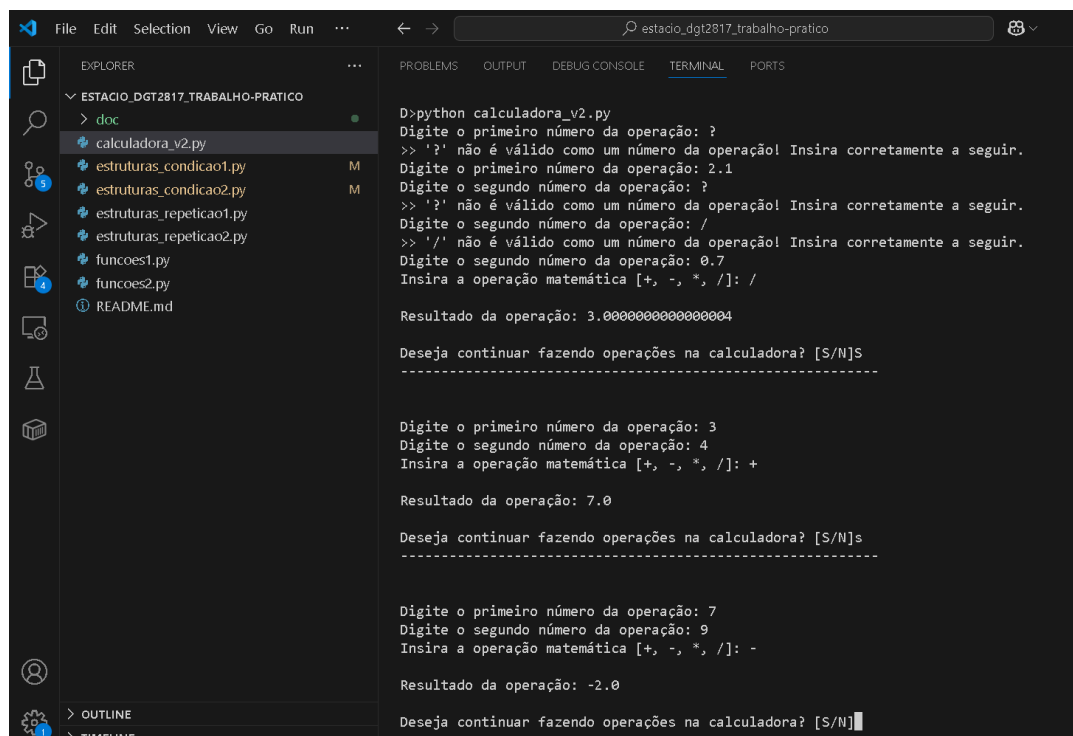


```

64
65 while (saida.lower() != "n"):
66     primeiro_numero = verifica_numero("Digite o primeiro número da operação: ")
67     segundo_numero = verifica_numero("Digite o segundo número da operação: ")
68     operacao = verifica_operacao("Insira a operação matemática [+ , -, *, /]: ")
69
70     resultado = calculadora(primeiro_numero, segundo_numero, operacao)
71     print(f"\nResultado da operação: {resultado}")
72
73     saida = input("\nDeseja continuar fazendo operações na calculadora? [S/N]")
74     print('-----\n\n')

```

## • Iterações:



```

D>python calculadora_v2.py
Digite o primeiro número da operação: ?
>> '?' não é válido como um número da operação! Insira corretamente a seguir.
Digite o primeiro número da operação: 2.1
Digite o segundo número da operação: ?
>> '?' não é válido como um número da operação! Insira corretamente a seguir.
Digite o segundo número da operação: /
>> '/' não é válido como um número da operação! Insira corretamente a seguir.
Digite o segundo número da operação: 0.7
Insira a operação matemática [+ , -, *, /]: /

Resultado da operação: 3.0000000000000004

Deseja continuar fazendo operações na calculadora? [S/N]S
-----

Digite o primeiro número da operação: 3
Digite o segundo número da operação: 4
Insira a operação matemática [+ , -, *, /]: +

Resultado da operação: 7.0

Deseja continuar fazendo operações na calculadora? [S/N]s
-----

Digite o primeiro número da operação: 7
Digite o segundo número da operação: 9
Insira a operação matemática [+ , -, *, /]: -

Resultado da operação: -2.0

Deseja continuar fazendo operações na calculadora? [S/N]

```

# UNIVERSIDADE ESTÁCIO DE SÁ

**Aluno:** Christian Marques de Oliveira Silva

**Matrícula:** 202505154772

**Disciplina:** DGT2817 Lógica, Algoritmos e Programação de Computadores

**Data:** 01/07/2025



```
D>python calculadora_v2.py
Digite o primeiro número da operação: 3
Digite o segundo número da operação: 8
Insira a operação matemática [+ , - , * , /]: vezes

Resultado da operação: 24.0

Deseja continuar fazendo operações na calculadora? [S/N]n
-----
D>
```

```
D>python calculadora_v2.py
Digite o primeiro número da operação: 2
Digite o segundo número da operação: 0
Insira a operação matemática [+ , - , * , /]: dividir
>> 'dividir' não é válido como uma operação! Insira uma das opções abaixo para continuar:
<ADIÇÃO>: Adição, mais ou +
<SUBTRAÇÃO>: Subtração, menos ou -
<MULTIPLICAÇÃO>: Multiplicação, vezes ou *
<DIVISÃO>: Divisão, dividido ou /
Insira a operação matemática [+ , - , * , /]: divisão

Resultado da operação: Não foi possível realizar a divisão por 0

Deseja continuar fazendo operações na calculadora? [S/N]
```