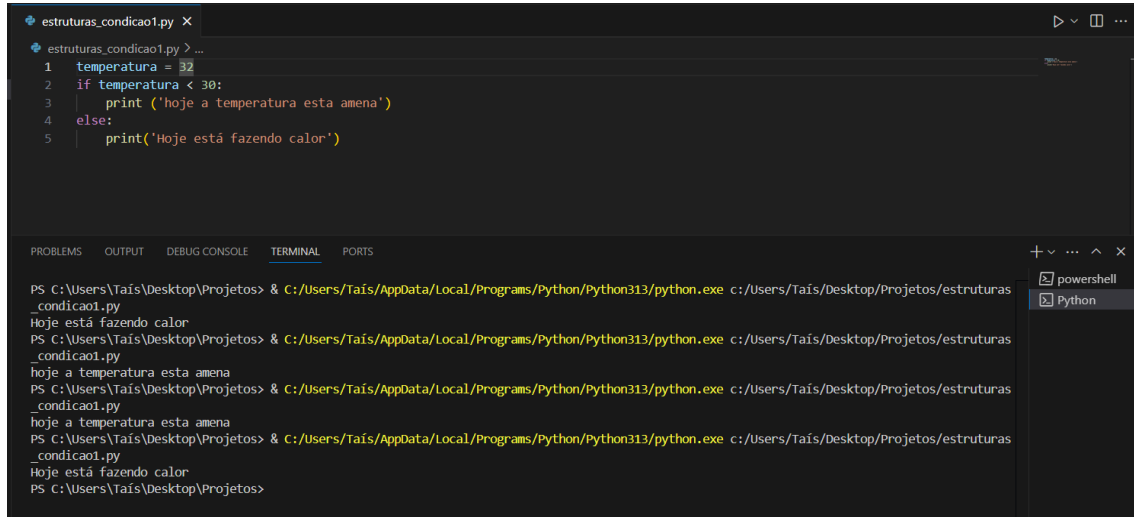


# Microatividade 1

Implementado um if/else simples que lê a variável temperatura e exibe uma mensagem diferente conforme o valor ( $< 30$  ou  $\geq 30$ ).

Para testar outro caso, altere temperatura para 31 e execute de novo.



The screenshot shows a code editor with a file named `estruturas_condicao1.py`. The code is as follows:

```
1 temperatura = 32
2 if temperatura < 30:
3     print('hoje a temperatura esta amena')
4 else:
5     print('Hoje está fazendo calor')
```

Below the code editor is a terminal window with the following output:

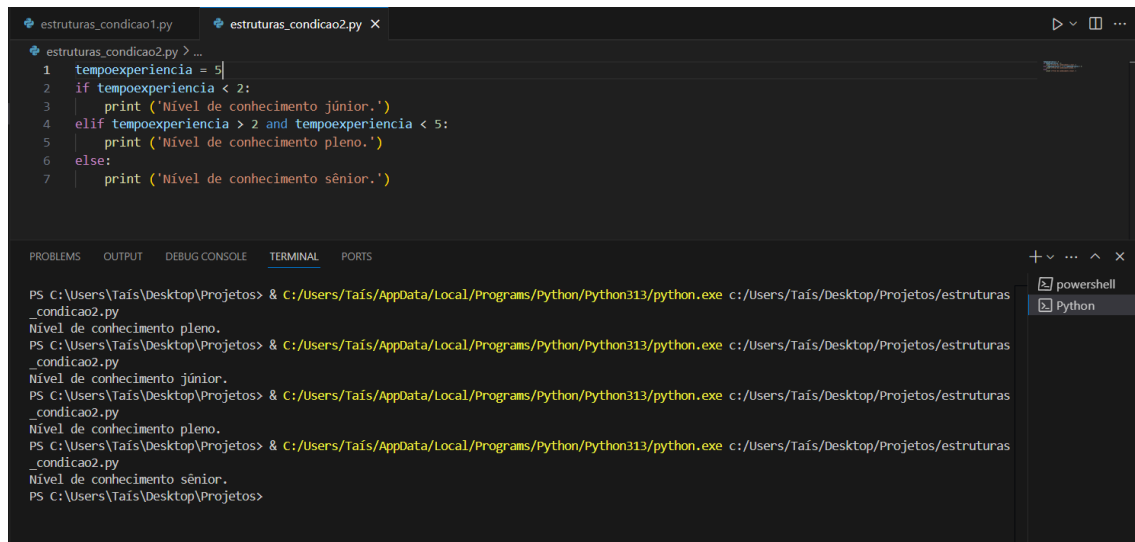
```
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/estruturas_condicao1.py
Hoje está fazendo calor
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/estruturas_condicao1.py
hoje a temperatura esta amena
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/estruturas_condicao1.py
hoje a temperatura esta amena
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/estruturas_condicao1.py
Hoje está fazendo calor
PS C:\Users\Taís\Desktop\Projetos>
```

On the right side of the terminal, there is a dropdown menu with 'powershell' and 'Python' options.

## Microatividade 2

Criada uma cadeia if/elif/else que classifica o tempo de experiência (júnior, pleno ou sênior) conforme o valor de tempoExperiencia.

Alterando o valor de tempoExperiencia para 1 e depois para 3 e 5, salvando e executando novamente, para verificar cada resultado.



The image shows a Visual Studio Code editor with two tabs: `estruturas_condicao1.py` and `estruturas_condicao2.py`. The active tab, `estruturas_condicao2.py`, contains the following Python code:

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

Below the editor, the TERMINAL panel is open, showing the execution of the script. The terminal output is as follows:

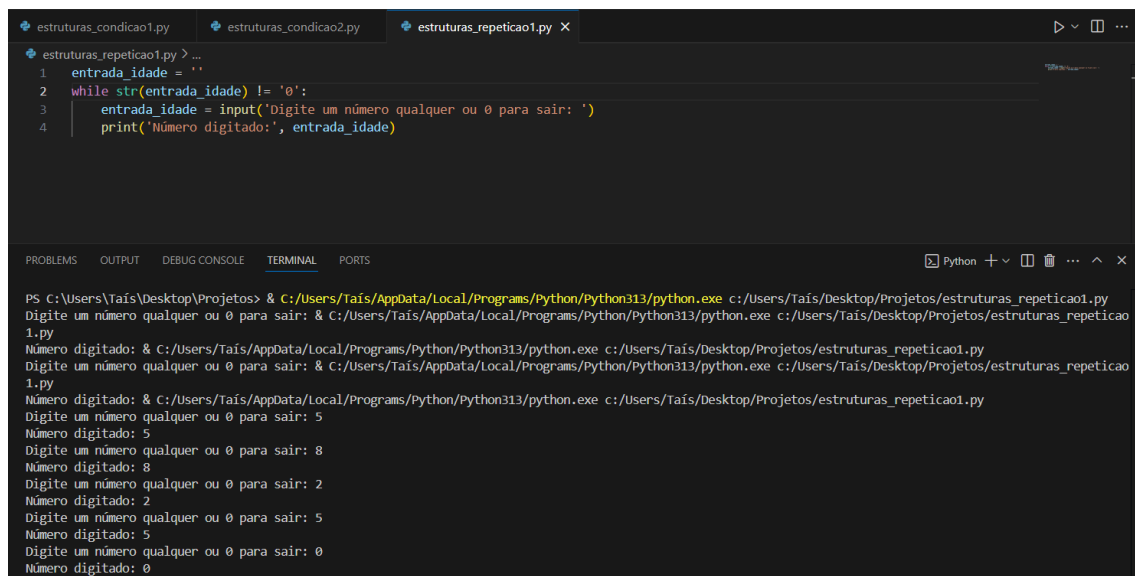
```
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Taís/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Taís/Desktop/Projetos/estruturas_condicao2.py
Nível de conhecimento pleno.
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Taís/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Taís/Desktop/Projetos/estruturas_condicao2.py
Nível de conhecimento júnior.
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Taís/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Taís/Desktop/Projetos/estruturas_condicao2.py
Nível de conhecimento pleno.
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Taís/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Taís/Desktop/Projetos/estruturas_condicao2.py
Nível de conhecimento sênior.
PS C:\Users\Taís\Desktop\Projetos>
```

The terminal window also shows tabs for `powershell` and `Python`.

## Microatividade 3

Construído um laço while que pede números ao usuário até que ele digite 0. Cada entrada é ecoada no formato “Número digitado: ...”.

Deve digitar valores diversos; o programa termina quando você digitar 0



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

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

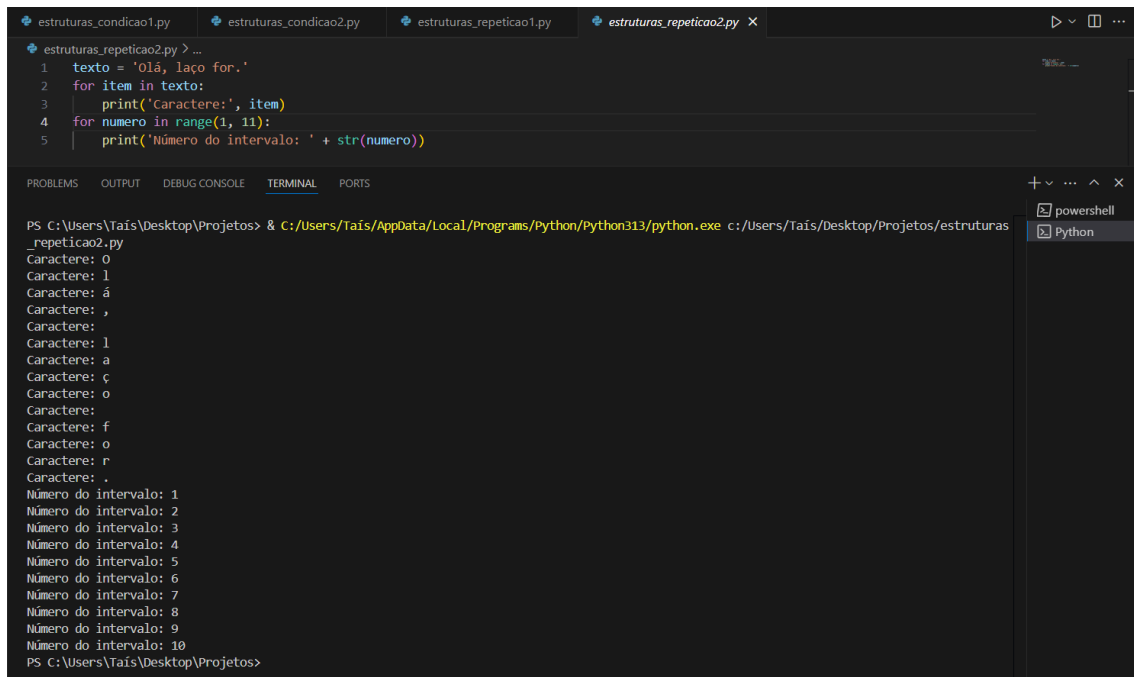
Python + - [ ] ... ^ x

PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/estruturas\_repeticao1.py  
Digite um número qualquer ou 0 para sair: & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/estruturas\_repeticao1.py  
Número digitado: & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/estruturas\_repeticao1.py  
Digite um número qualquer ou 0 para sair: 5  
Número digitado: 5  
Digite um número qualquer ou 0 para sair: 8  
Número digitado: 8  
Digite um número qualquer ou 0 para sair: 2  
Número digitado: 2  
Digite um número qualquer ou 0 para sair: 5  
Número digitado: 5  
Digite um número qualquer ou 0 para sair: 0  
Número digitado: 0

## Microatividade 4

For que percorre a string texto, imprimindo caractere a caractere e o segundo for usando range(1, 11) para listar os números de 1 a 10.

Lista de caracteres e, em seguida, os números do intervalo.



The screenshot shows a Python IDE with four tabs: `estruturas_condicao1.py`, `estruturas_condicao2.py`, `estruturas_repeticao1.py`, and `estruturas_repeticao2.py`. The active tab is `estruturas_repeticao2.py`, which contains the following code:

```
1 texto = 'Olá, laço for.'
2 for item in texto:
3     print('Caractere:', item)
4 for numero in range(1, 11):
5     print('Número do intervalo: ' + str(numero))
```

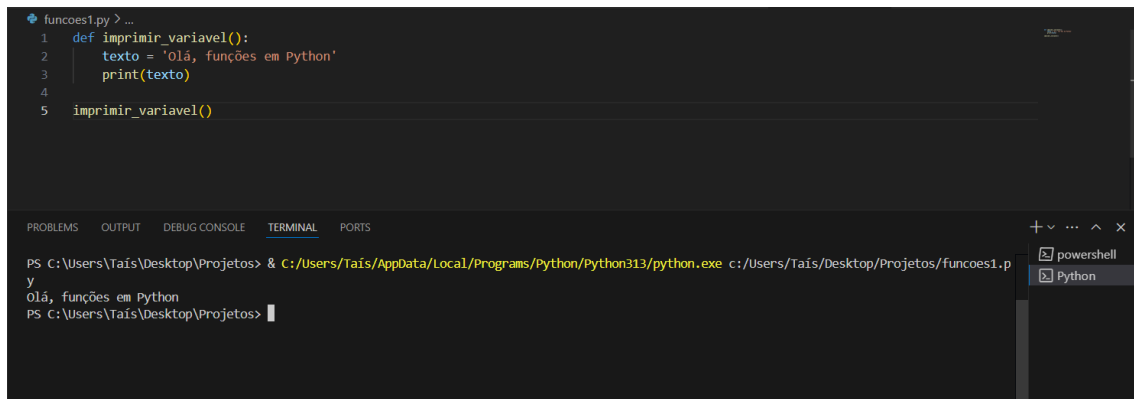
The bottom panel shows the output of the script, with tabs for `PROBLEMS`, `OUTPUT`, `DEBUG CONSOLE`, `TERMINAL`, and `PORTS`. The `TERMINAL` tab is active, displaying the output of the script:

```
PS C:\Users\Tais\Desktop\Projetos> & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/estruturas_repeticao2.py
Caractere: O
Caractere: l
Caractere: á
Caractere: ,
Caractere: 
Caractere: l
Caractere: a
Caractere: ç
Caractere: o
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
PS C:\Users\Tais\Desktop\Projetos>
```

## Microatividade 5

Declarada a função `imprimir_variavel`, que cria a variável `texto` dentro do escopo e imprime seu valor; a função é chamada uma única vez no final do script.

A mensagem “Olá, funções em Python” aparece no console.



```
funcoes1.py > ...
1 def imprimir_variavel():
2     texto = 'Olá, funções em Python'
3     print(texto)
4
5 imprimir_variavel()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/funcoes1.py

Olá, funções em Python

PS C:\Users\Taís\Desktop\Projetos> |

+ ... ^ x

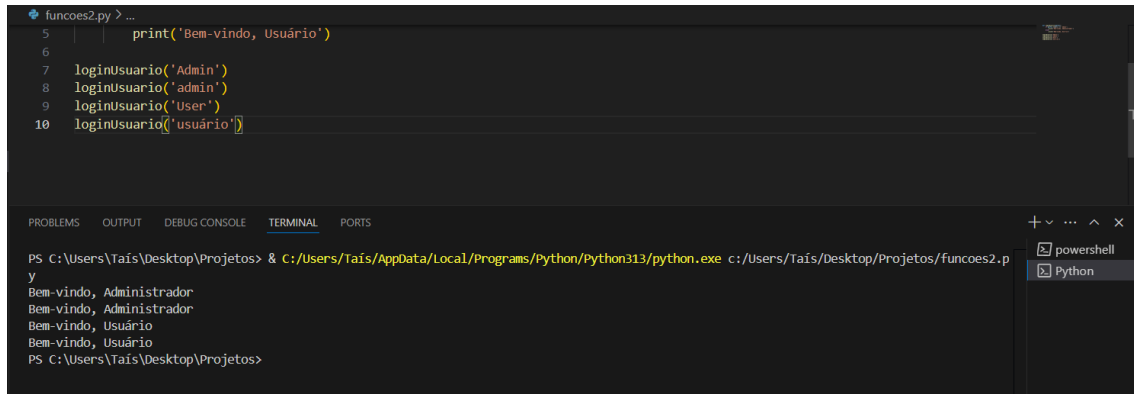
powershell

Python

## Microatividade 6

Criada a função `loginUsuario(perfil)` que usa `perfil.lower()` para distinguir administradores de usuários comuns. O script chama a função quatro vezes com argumentos diferentes para demonstrar ambos os caminhos.

“Bem-vindo, Administrador” e “Bem-vindo, Usuário” aparece no console.



```
funcoes2.py > ...
5 |         print('Bem-vindo, Usuário')
6 |
7 |     loginUsuario('Admin')
8 |     loginUsuario('admin')
9 |     loginUsuario('User')
10 |    loginUsuario('usuário')

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\Taís\Desktop\Projetos> & C:/Users/Tais/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Tais/Desktop/Projetos/funcoes2.py
Bem-vindo, Administrador
Bem-vindo, Administrador
Bem-vindo, Usuário
Bem-vindo, Usuário
PS C:\Users\Taís\Desktop\Projetos>
```

# Trabalho Prático

Refatoração completa da calculadora original.

Criação de quatro funções de operação (adicao, subtracao, multiplicacao, divisao) para modularizar o código.

Implementação da função calculadora que decide, via if/elif, qual operação executar e devolve o resultado ou a mensagem “Operação inválida”.

Tratamento de divisão por zero dentro de divisao.

Laço while que mantém o programa ativo até o usuário digitar N/n.

Validação de entrada numérica com try/except para evitar que o programa quebre quando o usuário digitar algo que não seja número.

```
Logica-Algoritmos-e-Programacao-de-Computadores > calculadora_v2.py > [?] saida
1  saida = 'n'
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:
14         return 'Não foi possível realizar a divisão por 0'
15     return a / b
16
17 def calculadora(num1, num2, operacao):
18     operacao = operacao.lower()
19     if operacao in ('+', 'adicao', 'adição'):
20         resultado = (function) def multiplicacao(
21             a: Any,
22             b: Any
23         ) -> Any
24         resultado = multiplicacao(num1, num2)
25     elif operacao in ('/', 'divisao', 'divisão'):
26         resultado = divisao(num1, num2)
27     else:
28         resultado = 'Operação inválida'
29     return resultado
30
31 while saida.lower() == 'n':
32     num1 = float(input('Digite o primeiro número: '))
33     num2 = float(input('Digite o segundo número: '))
34     operacao = input('Digite a operação (+, -, *, / ou nome): ')
35     resultado = calculadora(num1, num2, operacao)
36     print('Resultado da operação: ' + str(resultado))
37     saida = input('Deseja sair? (S/N): ')
38
Logica-Algoritmos-e-Programacao-de-Computadores > calculadora_v2.py > ...
12 def divisao(a, b):
13     if b == 0:
14         return 'Não foi possível realizar a divisão por 0'
15     return a / b
16
17 def calculadora(num1, num2, operacao):
18     operacao = operacao.lower()
19     if operacao in ('+', 'adicao', 'adição'):
20         resultado = multiplicacao(num1, num2)
21
PS C:\Users\Taís\Desktop\Projetos> & C:\Users\Taís\AppData\Local\Programs\Python\Python313\python.exe c:\Users\Taís\Desktop\Projetos\Logica-Algoritmos-e-Programacao-de-Computadores/calculadora_v2.py
Digite o primeiro número: 56
Digite o segundo número: 54
Digite a operação (+, -, *, / ou nome): /
Resultado da operação: 1.037037037037037
Deseja sair? (S/N): n
Digite o primeiro número: 54
Digite o segundo número: 87
Digite a operação (+, -, *, / ou nome): *
Resultado da operação: 4698.0
Deseja sair? (S/N): n
Digite o primeiro número: 58
Digite o segundo número: 65
Digite a operação (+, -, *, / ou nome): -
Resultado da operação: -7.0
Deseja sair? (S/N): n
Digite o primeiro número: 35
Digite o segundo número: 65
Digite a operação (+, -, *, / ou nome): +
Resultado da operação: 100.0
Deseja sair? (S/N): s
PS C:\Users\Taís\Desktop\Projetos>
```