

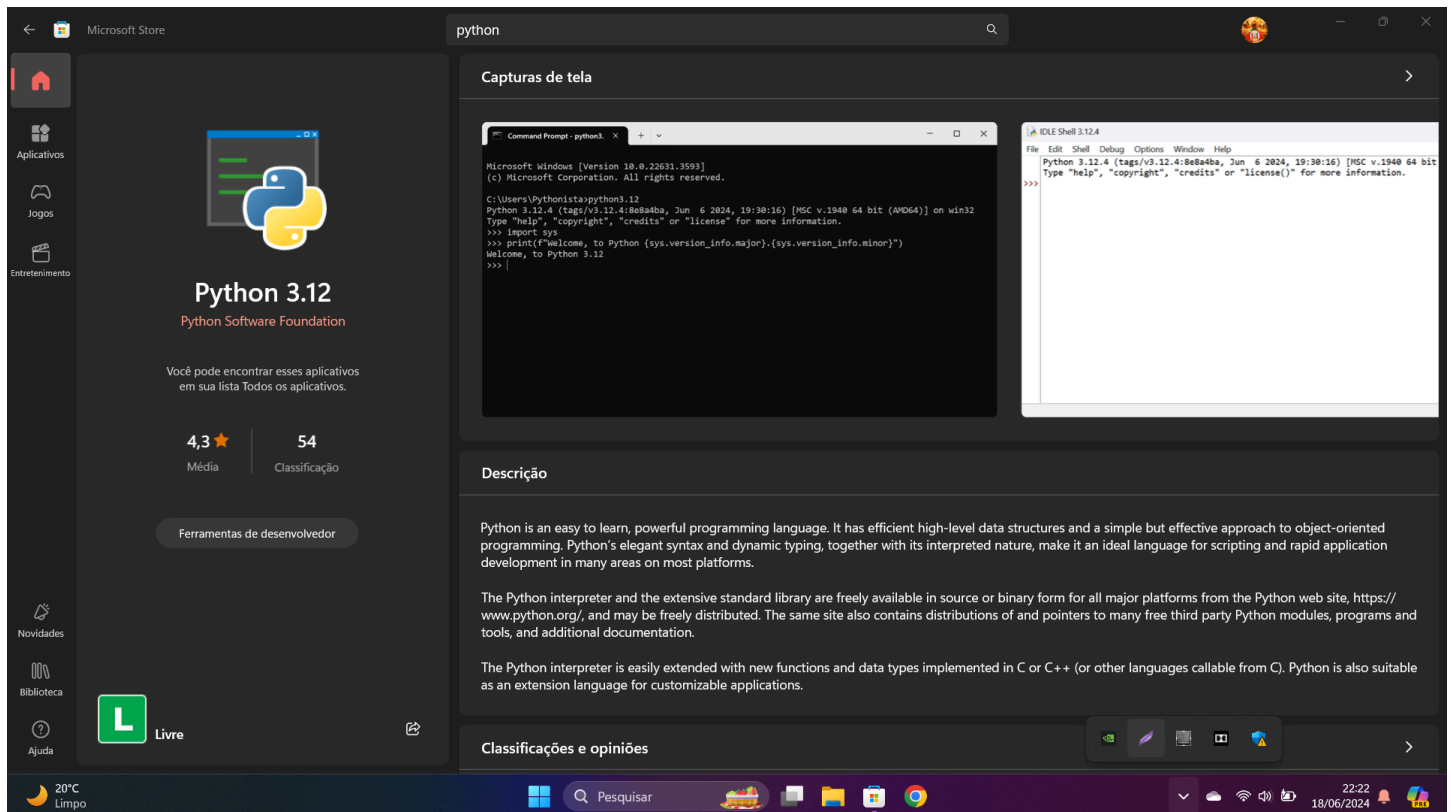
Evander Lucherris da Silva Rosa

MATRÍCULA: 202406091179

Missão Prática | Nível 1 | Mundo 1

Microatividades 1, 2, 3, 4 e 5.
Desvendando a Caixa Preta

Microatividade 1:



“Por este print iniciei o projeto, realizando o download do Interpretador Python pela Microsoft Store”.

```
Prompt de Comando
C:\Users\evand>python -V
Python 3.12.4
C:\Users\evand>
```

“Em seguida, realizei a verificação da versão do Interpretador pelo CMD, utilizando o comando: python -V”

Microatividade 2:

```
Prompt de Comando - pythor
Microsoft Windows [versão 10.0.22621.3737]
(c) Microsoft Corporation. Todos os direitos reservados.

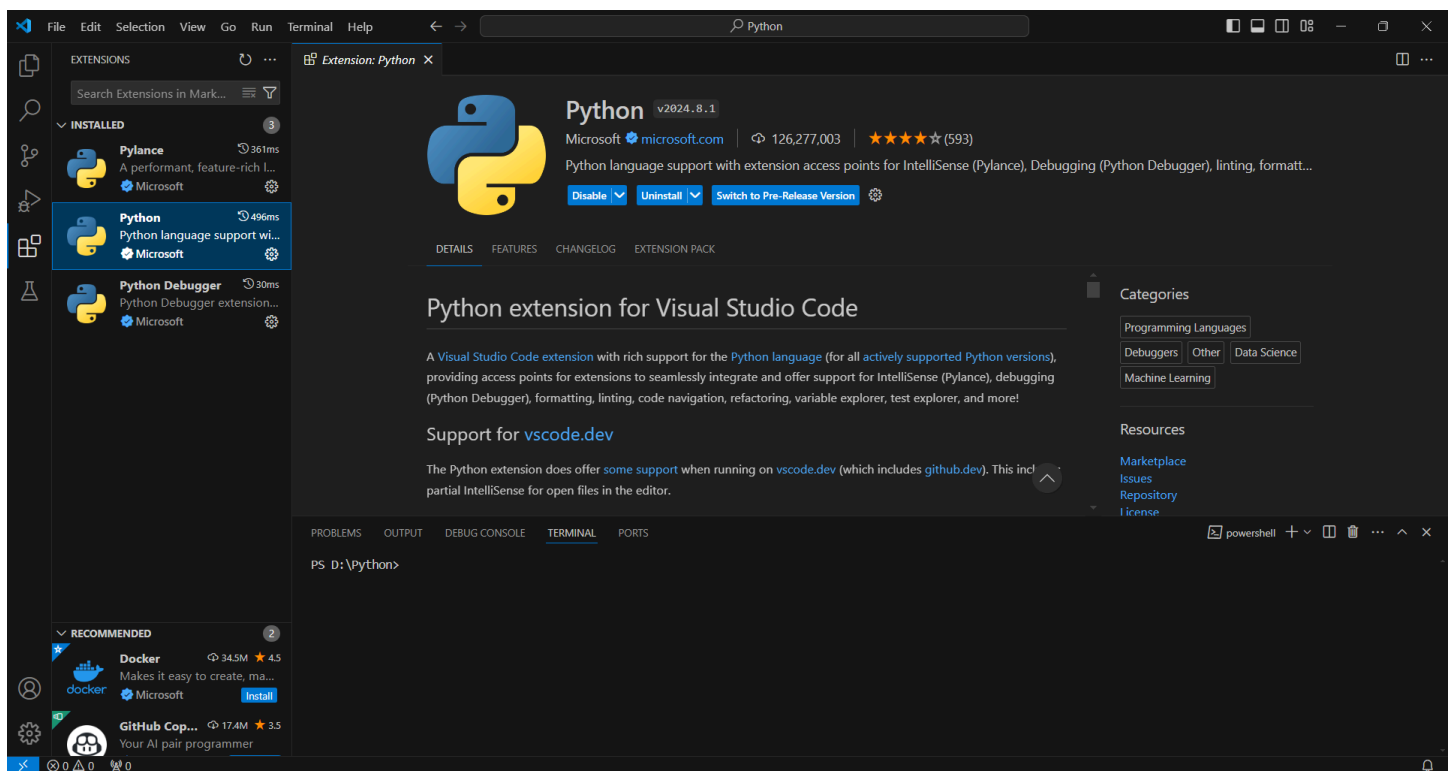
C:\Users\evand>python
Python 3.12.4 (tags/v3.12.4:8e8a4ba, Jun 6 2024, 19:30:16) [MSC v.1940 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello Python Code")
Hello Python Code
>>> print(1 + 1)
2
>>> print(10 + 10)
20
>>> print(100 + 100)
200
>>> |
```

*“Já com o acesso ao Interpretador Python pelo CMD, inserido através do comando: python
Realizei a primeira linha de instrução: print(“Hello Python Code”)
No qual permite imprimir o texto ‘Hello Python Code’ na tela.”*

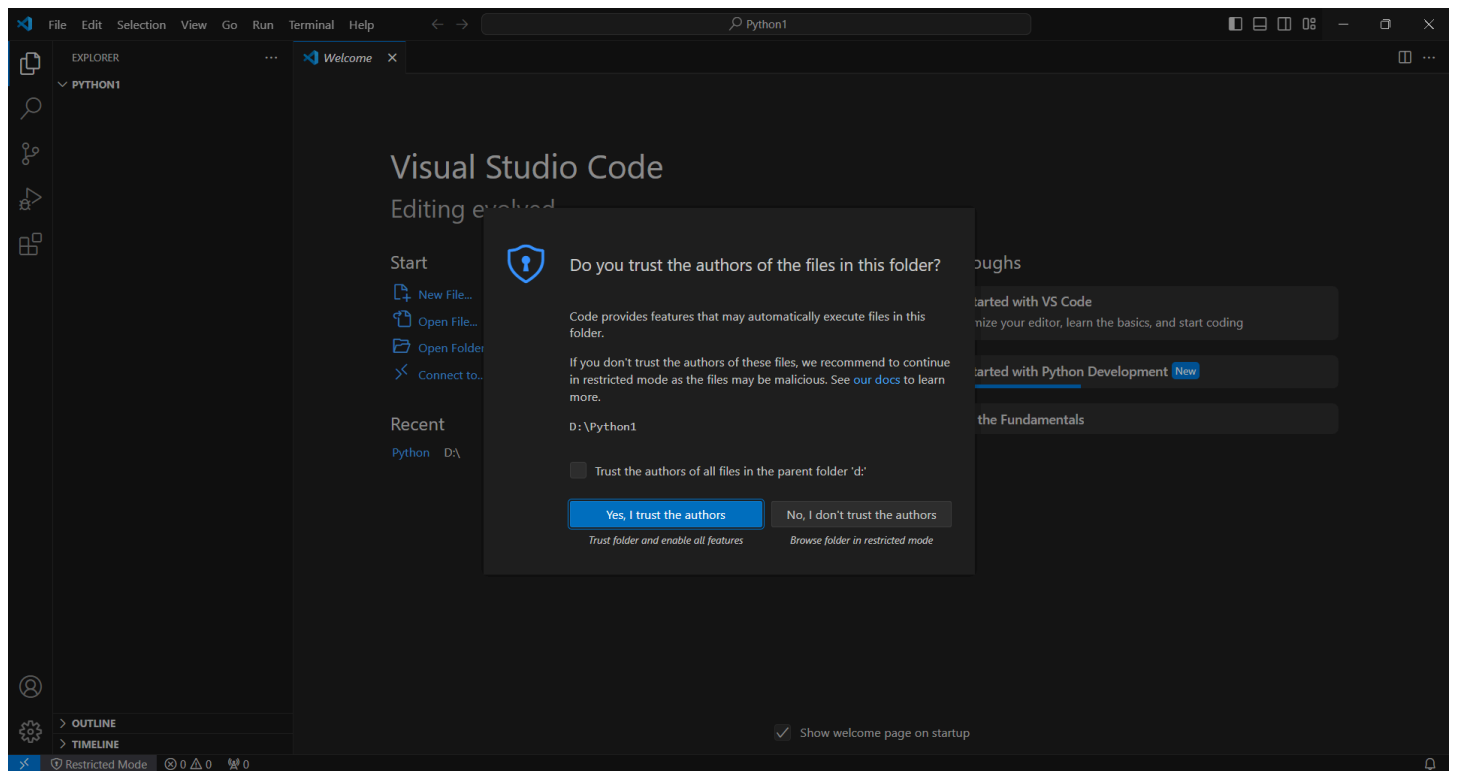
*Em seguida, adicionei a seguinte instrução: `print(1 + 1)`
Este comando solicita que a máquina(CPU) efetue o calculo,
e imprima o resultado na tela. (Isso ocorre de forma extremamente rápida).*

*Após isso, realizei mais testes com os comandos: `print(10 + 10)` e `print(100 + 100)`
aos quais também imprimiram os resultados 20 e 200.”*

Microatividade 3:



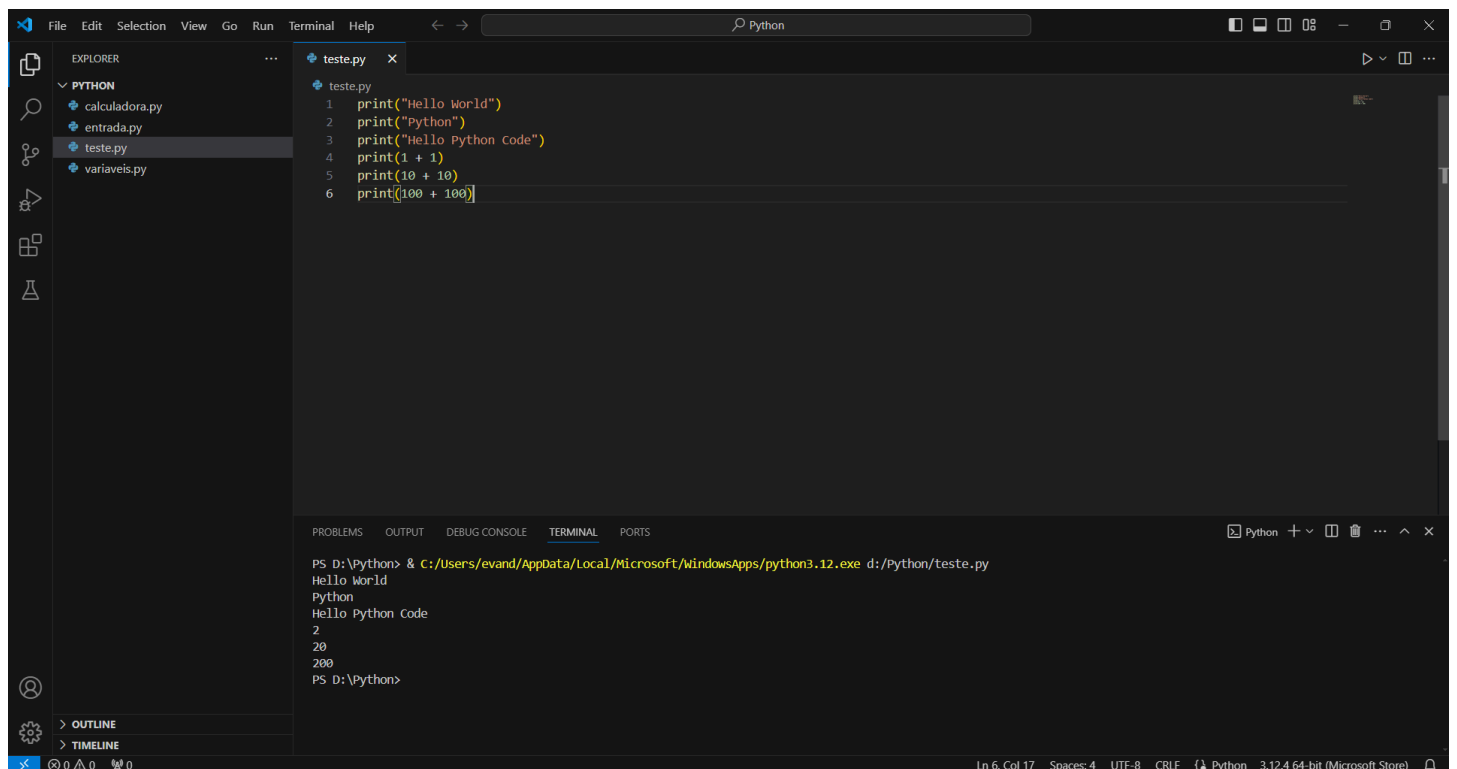
*“Já com a IDE VS Code instalada e em operação, instalei a extensão do Python.
Para isso, naveguei até o menu ‘Extensions’ e digitei ‘python’ na barra de pesquisa.
Na lista de resultados, selecionei a opção ‘Python do fornecedor/desenvolvedor Microsoft’.
Ao abrir a página da extensão, cliquei em ‘Install’ e após a conclusão da instalação,
fechei a janela da extensão.”*



“O próximo passo foi selecionar uma pasta no computador, onde será o local de armazenamento para os códigos python. Para isso, naveguei até a opção ‘Explorer’ e cliquei em ‘Open Folder’.

Ao selecionar a pasta escolhida, abriu uma janela Pop-Up(Imagem acima) questionando a minha confiança no autor da pasta.

Cliquei então no botão ‘Yes’, para que a pasta fosse exibida na janela ‘Explorer’.”



“Já na janela ‘Explorer’, criei um novo arquivo com o nome ‘teste.py’.

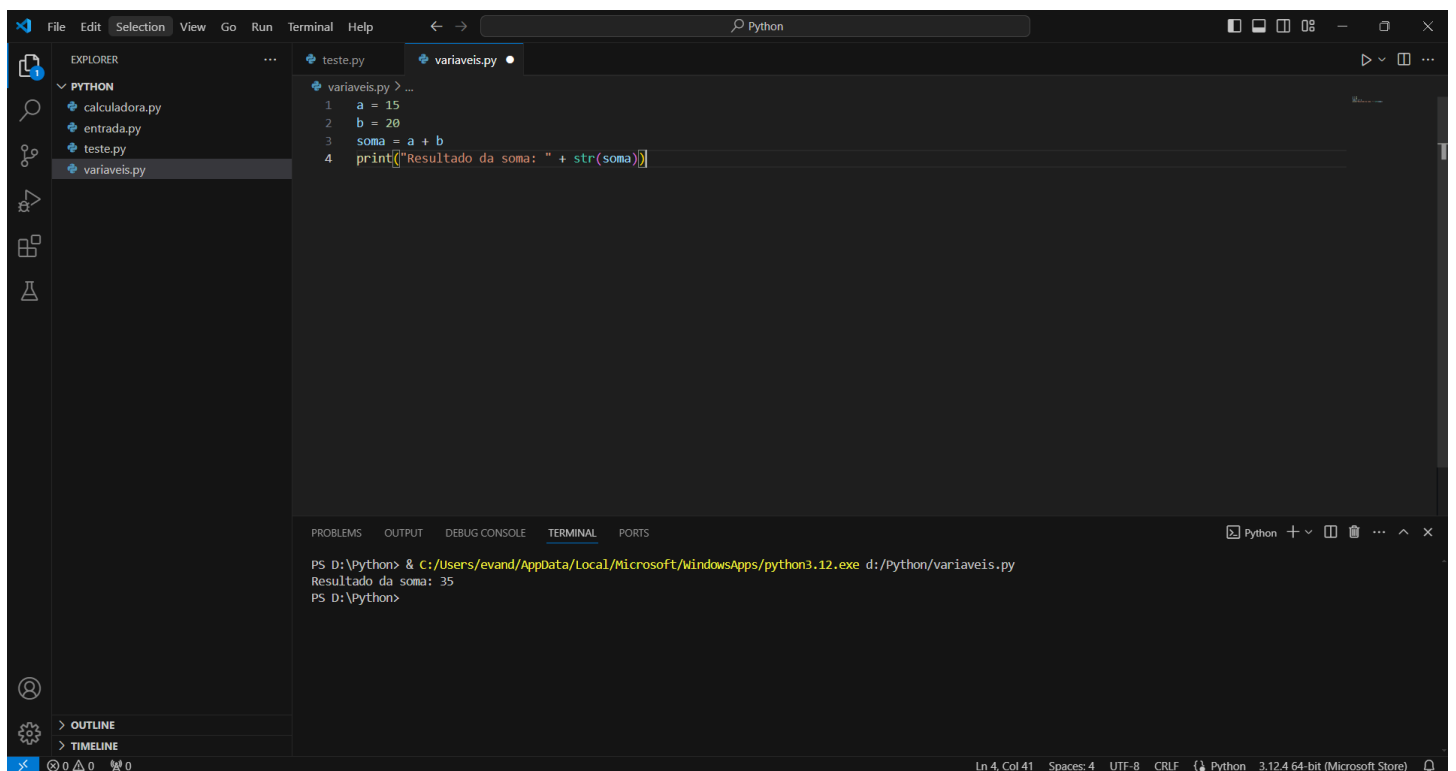
Como o próprio nome já sinaliza, testar o funcionamento do local de desenvolvimento.

Inseri então o comando ‘print(“Hello World”)’ e após salvar, cliquei no ícone em formato de triângulo deitado, na parte superior direita da janela, para que executasse instrução.

Abriu então a janela do terminal na parte inferior do VS Code, exibindo a saída do comando 'print'."

Em seguida, realizei testes com novas linhas no script, inserindo os mesmos comandos da micriatividade anterior, com o intuito de testar as funcionalidades do VS Code. Podendo assim, declarar novas instruções em sequecia e executar todas ao mesmo tempo."

Microatividade 4:



The screenshot shows the Visual Studio Code (VS Code) interface. On the left, the Explorer sidebar displays a file tree with the following files: `calculadora.py`, `entrada.py`, `teste.py`, and `variaveis.py`. The `variaveis.py` file is selected and open in the main editor. The code in the editor is as follows:

```
1 a = 15
2 b = 20
3 soma = a + b
4 print("Resultado da soma: " + str(soma))
```

Below the editor, the integrated terminal is open, showing the command prompt output:

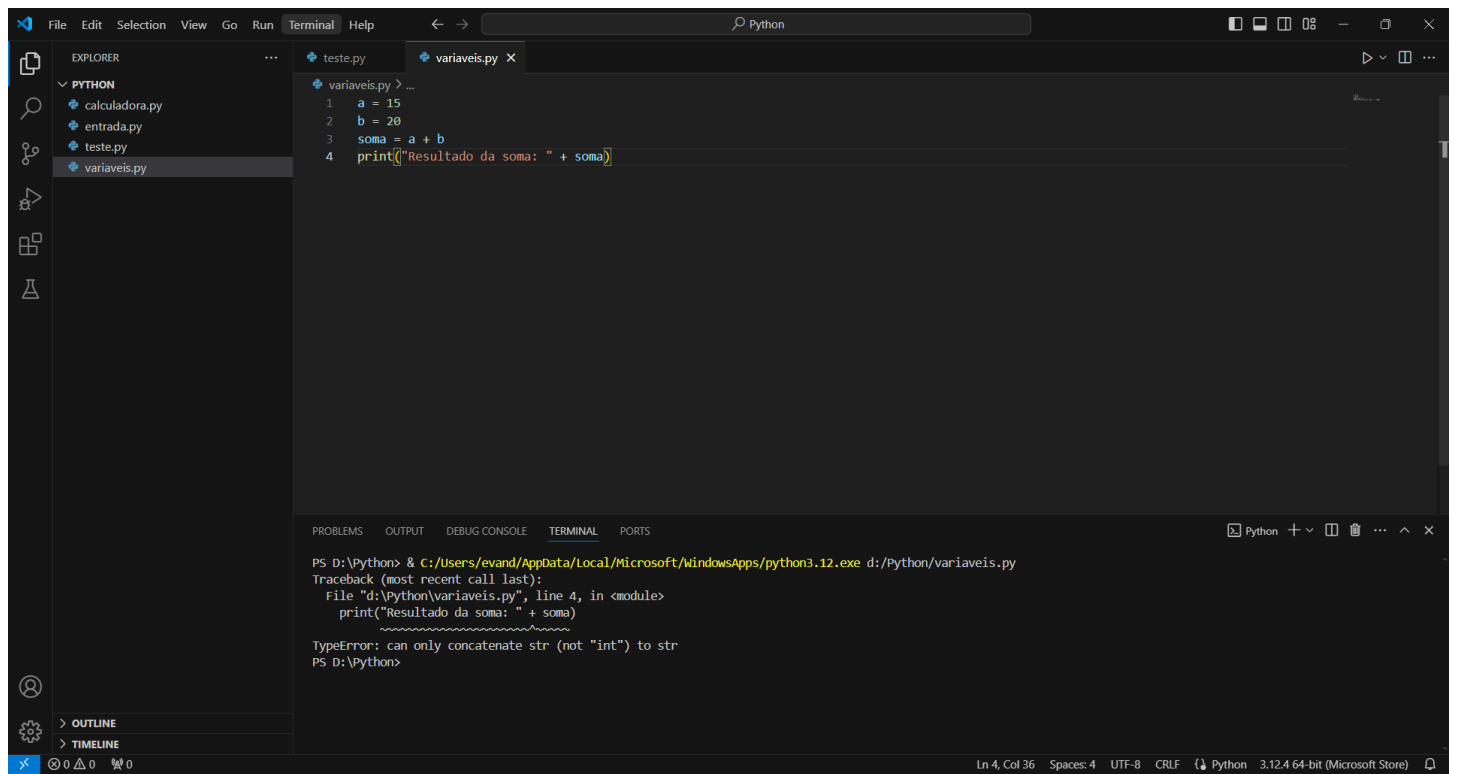
```
PS D:\Python> & C:\Users\evand\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Python/variaveis.py
Resultado da soma: 35
PS D:\Python>
```

The status bar at the bottom indicates the current line and column as 'Ln 4, Col 41', the encoding as 'UTF-8', and the file type as 'Python'.

“Ainda na mesma pasta do script da atividade anterior, criei um novo script como o nome: ‘váriaveis.py’. Com o objetivo de aprender a declarar variáveis e suas aplicações nas instruções.

Neste caso, foram declaradas 3 variáveis: a, b e soma, e atribuído valores para cada uma delas. Na ultima linha de instruções, utilizei o comando print para concatenar uma string e uma variável(soma), e exibir a combinação de ambas na tela.

Para converter a variável(soma) que é do tipo ‘inteiro’ para uma variável do tipo ‘string’, foi necessário preceder a variável ‘soma’ com a palavra ‘str’ na instrução print.”



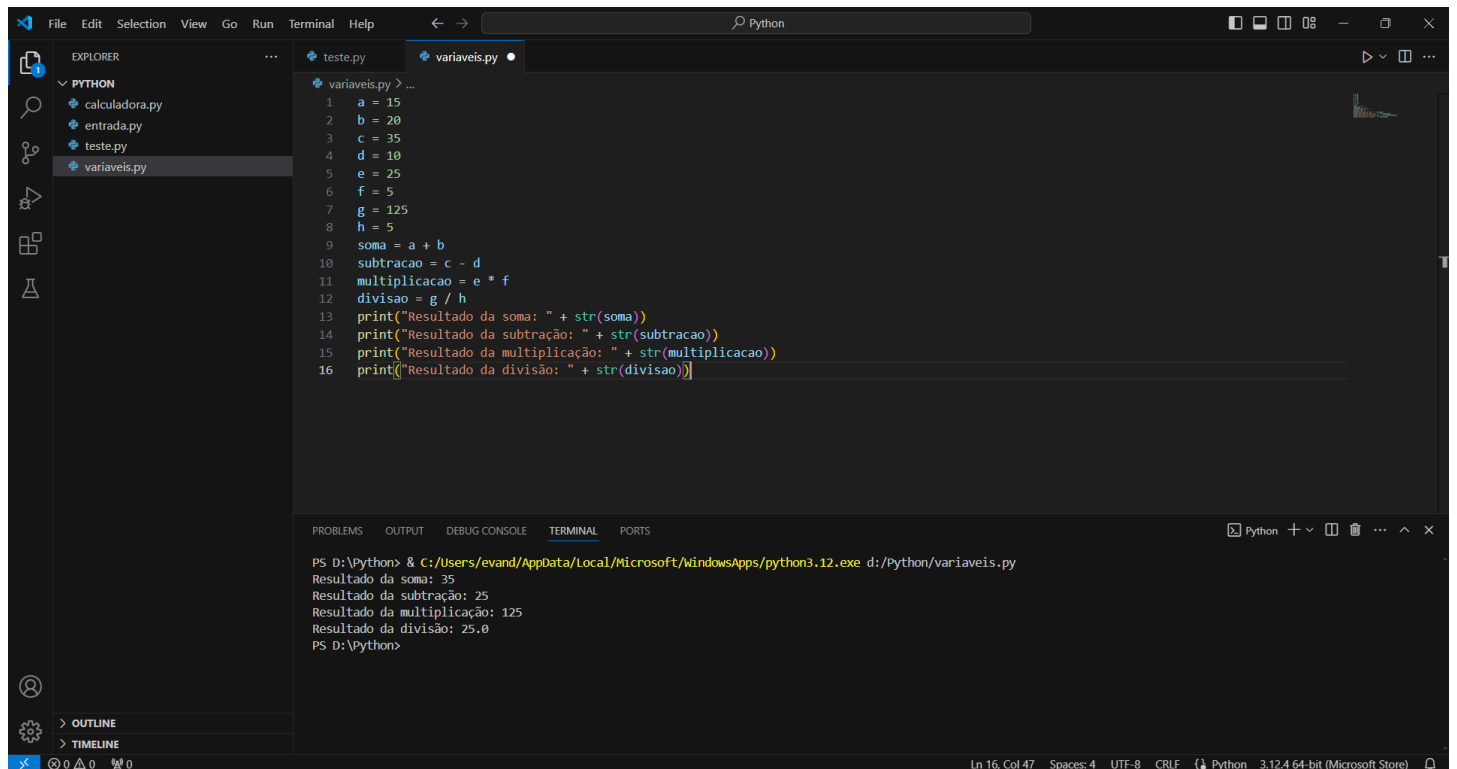
The screenshot shows the Visual Studio Code editor with a file named `variaveis.py` open. The code in the file is as follows:

```
1 a = 15
2 b = 20
3 soma = a + b
4 print("Resultado da soma: " + soma)
```

The terminal at the bottom shows the command `PS D:\Python> & C:/Users/evand/AppData/Local/Microsoft/WindowsApps/python3.12.exe d:/Python/variaveis.py` and the resulting error message:

```
Traceback (most recent call last):
  File "d:\Python\variaveis.py", line 4, in <module>
    print("Resultado da soma: " + soma)
    ~~~~~^~~~~~
TypeError: can only concatenate str (not "int") to str
PS D:\Python>
```

“No passo seguinte(imagem acima), testei a instrução ‘print’ retirando a palavra ‘str’ que precedia a variável ‘soma’. O terminal retornou um erro de concatenação. E que apenas seria possível concatenar ‘str’ com ‘str’. Isso se deu por conta da variável ‘soma’ ser do tipo int(Inteiro)”



The screenshot shows the Visual Studio Code editor with a file named `variaveis.py` open. The code in the file is as follows:

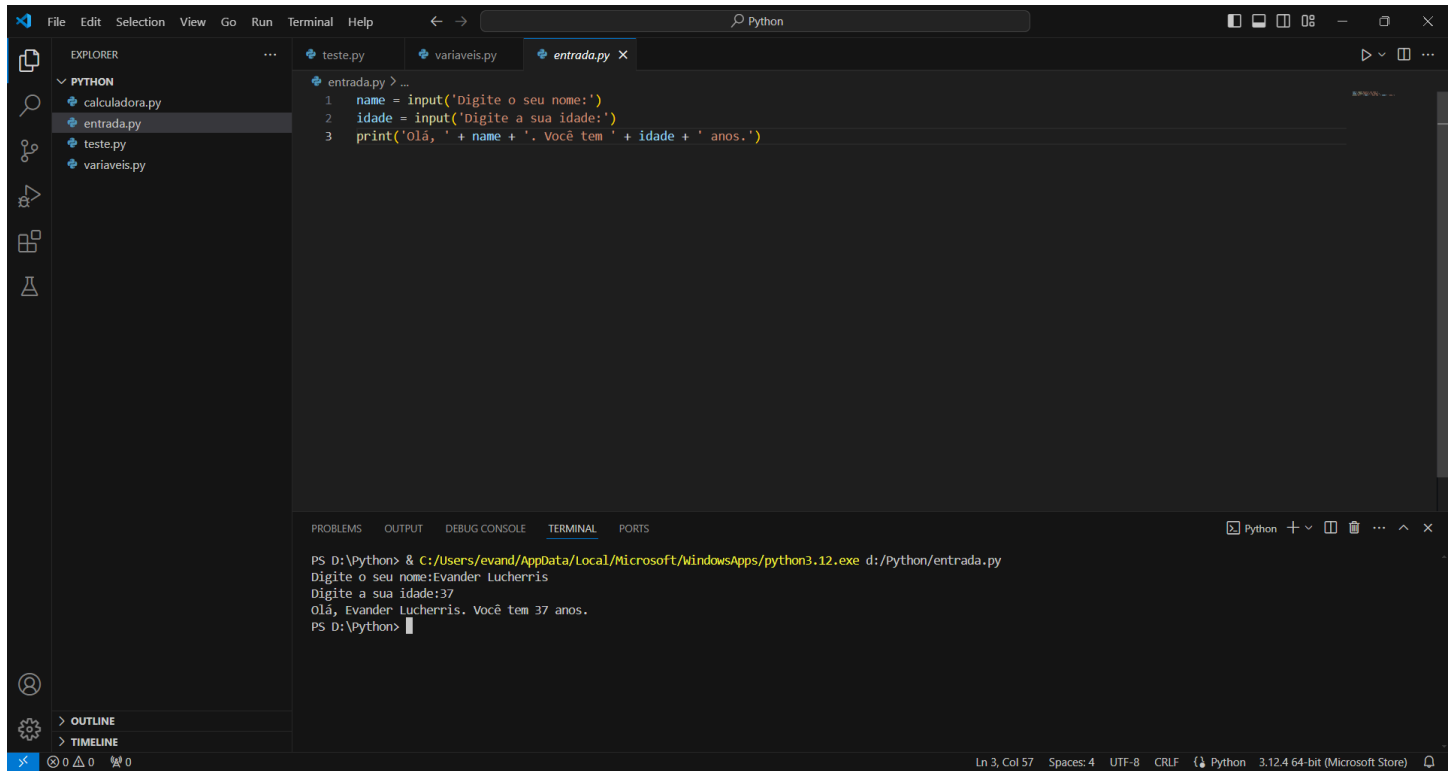
```
1 a = 15
2 b = 20
3 c = 35
4 d = 10
5 e = 25
6 f = 5
7 g = 125
8 h = 5
9 soma = a + b
10 subtracao = c - d
11 multiplicacao = e * f
12 divisao = g / h
13 print("Resultado da soma: " + str(soma))
14 print("Resultado da subtração: " + str(subtracao))
15 print("Resultado da multiplicação: " + str(multiplicacao))
16 print("Resultado da divisão: " + str(divisao))
```

The terminal at the bottom shows the command `PS D:\Python> & C:/Users/evand/AppData/Local/Microsoft/WindowsApps/python3.12.exe d:/Python/variaveis.py` and the resulting output:

```
Resultado da soma: 35
Resultado da subtração: 25
Resultado da multiplicação: 125
Resultado da divisão: 25.0
PS D:\Python>
```

“Após efetuar a correção das instruções, novos testes inserindo variáveis e operações aritméticas, bem como combinações dos diferentes tipos de variáveis foram realizados e executados com sucesso, e apresentados na janela do terminal.”

Microatividade 5:



The screenshot shows a Python IDE with a dark theme. The Explorer panel on the left lists files: calculadora.py, entrada.py, teste.py, and variaveis.py. The main editor window displays the code in 'entrada.py':

```
1 name = input('Digite o seu nome:')
2 idade = input('Digite a sua idade:')
3 print('Olá, ' + name + '. Você tem ' + idade + ' anos.')
```

The bottom panel shows the terminal output after running the script:

```
PS D:\Python> & C:\Users\evand\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Python/entrada.py
Digite o seu nome:Evander Lucherris
Digite a sua idade:37
Olá, Evander Lucherris. Você tem 37 anos.
PS D:\Python>
```

“Continuando, criei mais um script como o nome de ‘entrada.py’

Neste script, solicito ao usuário para que informe o seu nome e sua idade, através do comando ‘input’(Solicita inserção de dados, interagindo com usuários).

O mesmo então, informa seus dados através do terminal, primeiro o seu nome e confirma ‘Enter’, depois repete o processo na instrução seguinte confirmando com Enter’.

Durante a inserção, o sistema armazenou a primeira e segunda informação nas variáveis ‘name’ e ‘idade, em ordem de execução’. Feito isso, a instrução ‘print’ entra em cena apresentando os resultados na tela, concatenando e convertendo as variáveis do tipo ‘str’ e ‘int’.”

The screenshot shows a Python IDE with a dark theme. The Explorer panel on the left lists files: calculadora.py, entrada.py, teste.py, and variaveis.py. The main editor displays the contents of entrada.py, which contains a script for collecting user input and printing formatted output. The script prompts for name, age, weight, height, and sex, then prints a summary. The Terminal panel at the bottom shows the execution of the script, with the user's input (Evander Lucherris, 37, 90, 178, Masculino) and the corresponding output.

```
1 name = input('Digite o seu nome:')
2 idade = input('Digite a sua idade:')
3 peso = input('Digite o seu peso:')
4 altura = input('Digite a sua altura:')
5 sexo = input('Digite o seu sexo:')
6
7 print('Olá, ' + name + '. Você tem ' + idade + ' anos.')
8 print('Está pesando ' + peso + ' kilos e medindo ' + altura + ' de altura.')
9 print('Você é do sexo ' + sexo + '.')
```

Terminal output:

```
PS D:\Python> & C:\Users\evand\AppData\Local\Microsoft\WindowsApps\python3.12.exe d:/Python/entrada.py
Digite o seu nome:Evander Lucherris
Digite a sua idade:37
Digite o seu peso:90
Digite a sua altura:178
Digite o seu sexo:Masculino
Olá, Evander Lucherris. Você tem 37 anos.
Está pesando 90 kilos e medindo 178 de altura.
Você é do sexo Masculino.
PS D:\Python>
```

“Neste passo, adiciono mais linhas de instruções ‘input’ para solicitação de entrada de dados do usuário, seguindo basicamente os mesmos passos da instrução anterior. Todas as informações solicitadas ao usuário, preenchidas e apresentadas com sucesso no terminal, através do comando ‘print’.”

Missão Prática | Desvendando a Caixa Preta

The image shows a Visual Studio Code editor window with a Python file named `calculadora.py` open. The file contains a simple calculator script that takes two numbers as input and performs addition, subtraction, multiplication, and division. The script is as follows:

```
1 a = int(input("Primeiro número:"))
2 b = int(input("Segundo número:"))
3
4 adicao = (a + b)
5 subtracao = (a - b)
6 multiplicacao = (a * b)
7 divisao = (a / b)
8
9 print("Resultado da Adição: ",a,"+",b,"=",adicao)
10 print("Resultado da Subtração: ",a,"-",b,"=",subtracao)
11 print("Resultado da Multiplicação: ",a,"*",b,"=",multiplicacao)
12 print("Resultado da Divisão: ",a,"/",b,"=",divisao)
```

Below the editor, the terminal window shows the execution of the script. The user runs the command `python calculadora.py` and provides the inputs 10 and 10. The output shows the results of the four operations: 10 + 10 = 20, 10 - 10 = 0, 10 * 10 = 100, and 10 / 10 = 1.0. The user then runs the script again with inputs 30 and 5, and the output shows: 30 + 5 = 35, 30 - 5 = 25, 30 * 5 = 150, and 30 / 5 = 6.0. Finally, the user runs the script with inputs 75 and 23, and the output shows: 75 + 23 = 98, 75 - 23 = 52, 75 * 23 = 1725, and 75 / 23 = 3.260869565217391.

Esta missão, tem o objetivo de por em prática uma aplicação simples de uma calculadora, utilizando recursos aplicados nas microatividades anteriores. Permitindo também, calculos utilizando adição, subtração, multiplicação e divisão,e após isso apresentar os resultados na tela do terminal.

*Iniciei o projeto solicitando ao usuário para digitar o 1º e 2º número , através da interação via terminal, para que fossem armazenados nas variáveis 'a' e 'b'. em seguida inseri as instruções de calculo das 4 principais operações aritméticas +, -, *, / (Esta instrução ao ser executada, é realizada de forma extremamente rápida pela CPU) Ao final, adicionei o comando 'print' para que imprimisse na tela os resultados já calculados dos números solicitados ao usuário.*

Como esperado, as instruções foram executadas e apresentadas com sucesso.

Atenciosamente, Evander Lucherris.