

19 - Ferramenta de debug.mp4

```
21 i = 0
22 while i < 5:
23     print('Olá tudo bem ?')
24     print('Como vai você ?')
25     i = i + 2
```



Roda o código lembra que estou rodando no Windows.

```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repetição_while.py
Olá tudo bem ?
Como vai você ?
```

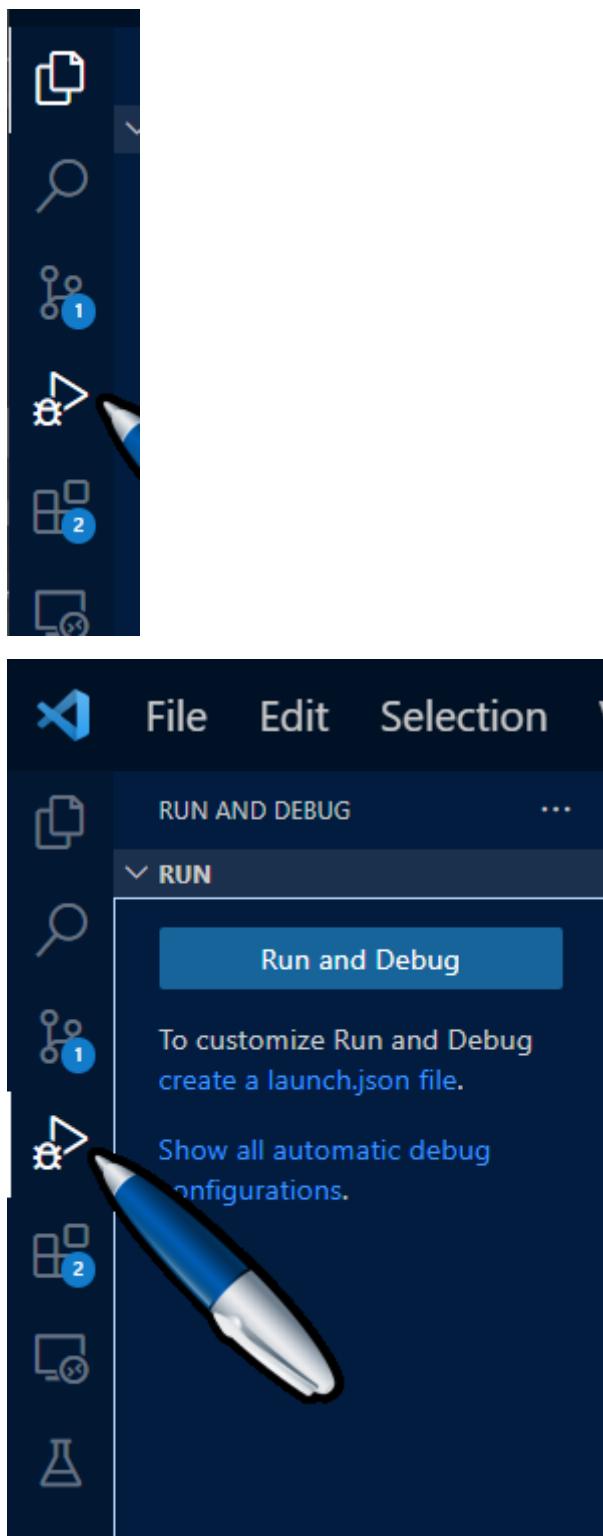


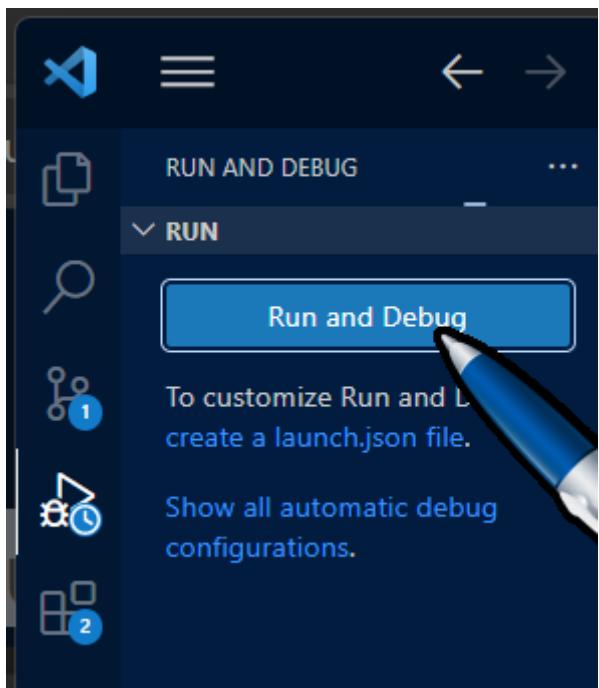
```
29 i = 0
30 j = 0
31 while i < 10 and j < 30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```



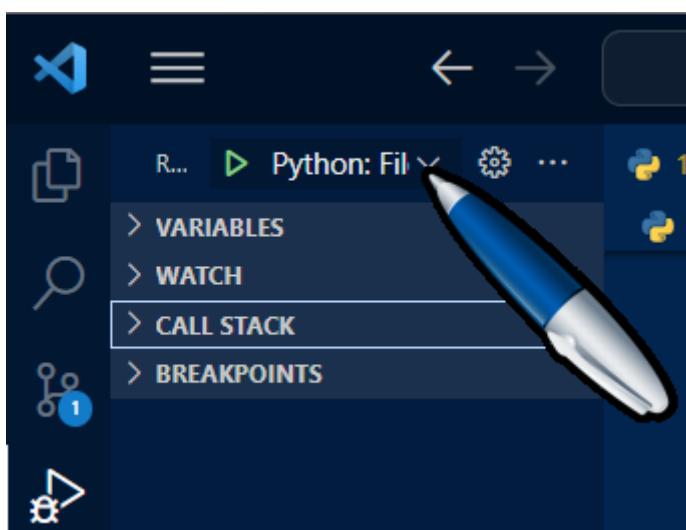
```
Como vai você ?  
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repetição_while.py  
Olá tudo bem ?  
Como vai você ?  
Olá tudo bem ?  
Como vai você ?  
Olá tudo bem ?  
Como vai você ?  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

Debug

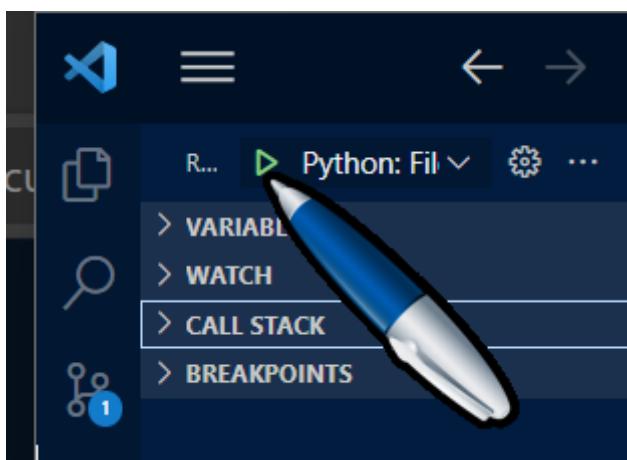




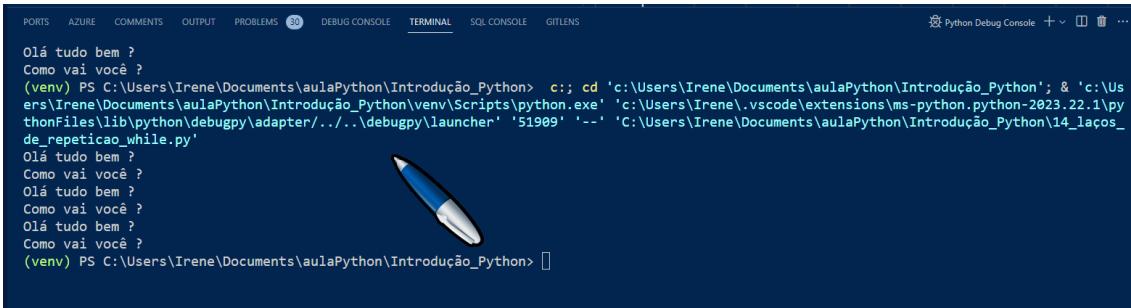
Selecciona PYTHON



Clica nesta seta verde para rodar

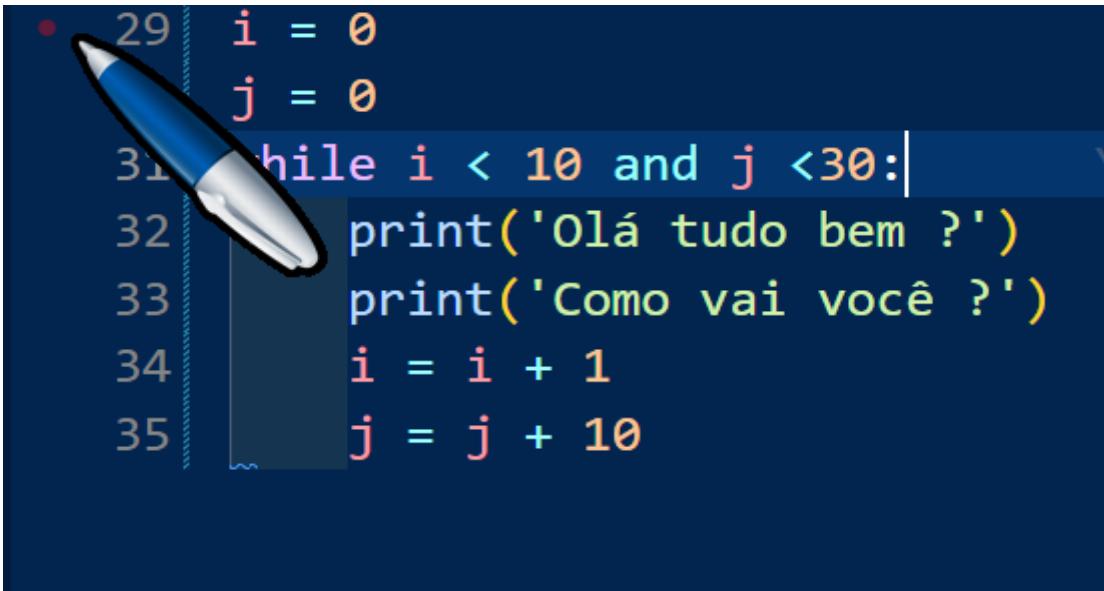


Depois de ter clicado na seta verde para rodar vai aparecer como print abaixo



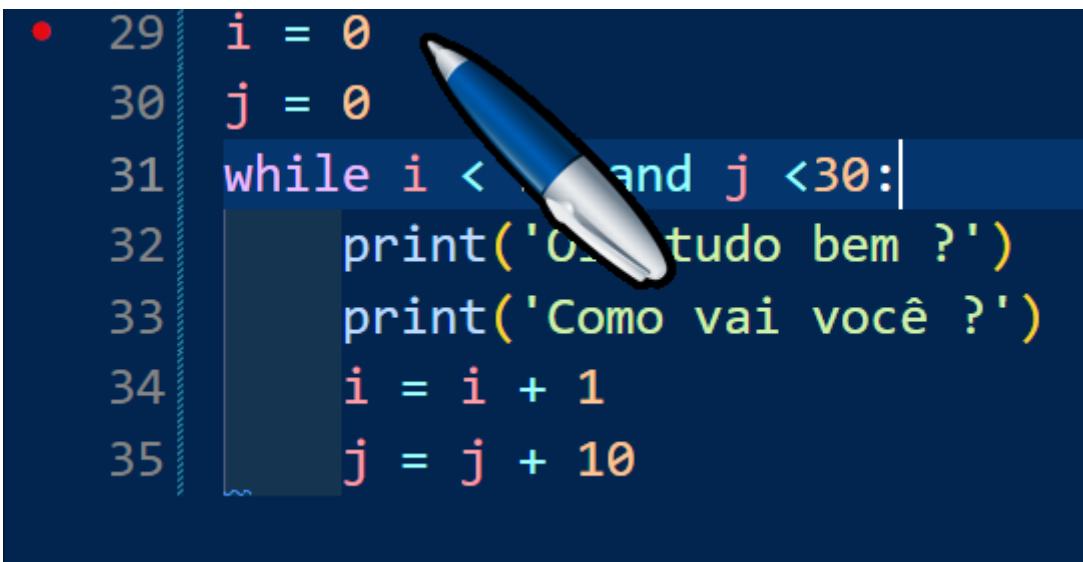
```
POROS AZURE COMMENTS OUTPUT PROBLEMS 30 DEBUG CONSOLE TERMINAL SQL CONSOLE GIT LENS Python Debug Console + ⌂ ⌂ ⌂ ...  
Olá tudo bem ?  
Como vai você ?  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> cd 'c:\Users\Irene\Documents\aulaPython\Introdução_Python'; & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' '51909' '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\14_lacos_de_repeticao_while.py'  
Olá tudo bem ?  
Como vai você ?  
Olá tudo bem ?  
Como vai você ?  
Olá tudo bem ?  
Como vai você ?  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

Seleciona um breakpoint, passa o mouse quando aparecer uma bolinha vermelha clica que focará selecionado.



```
• 29 i = 0  
j = 0  
31 while i < 10 and j < 30:  
32     print('Olá tudo bem ?')  
33     print('Como vai você ?')  
34     i = i + 1  
35     j = j + 10
```

Verifica que a bolinha ficará vermelha



```
• 29 i = 0  
j = 0  
31 while i < 10 and j < 30:  
32     print('Olá tudo bem ?')  
33     print('Como vai você ?')  
34     i = i + 1  
35     j = j + 10
```

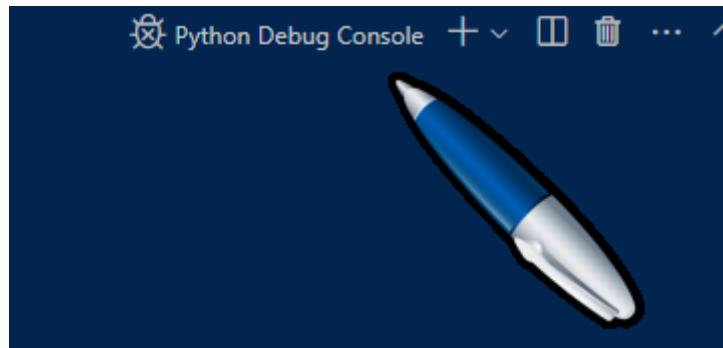
Fecha este debug no terminal e roda o código novamente



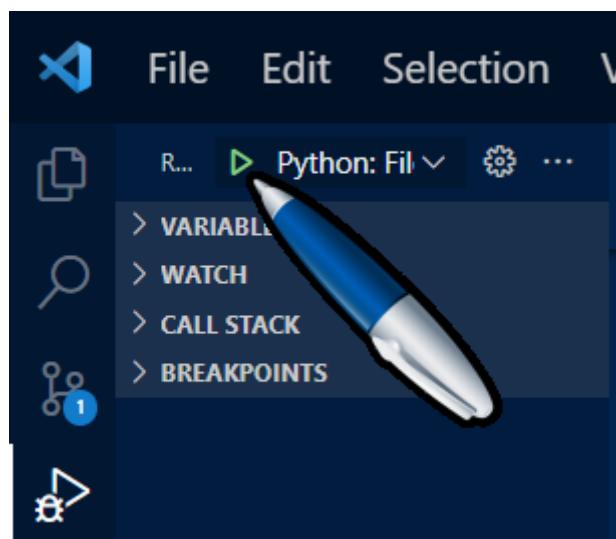
```
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

Olá tudo bem ?
Como vai você ?
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> c; cd 'c:\Users\Irene\Documents\aulaPython\Introdução_Python'; pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '51909' '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\14_lendo_arquivos_de_repeticao_while.py'
Olá tudo bem ?
Como vai você ?
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> []
```

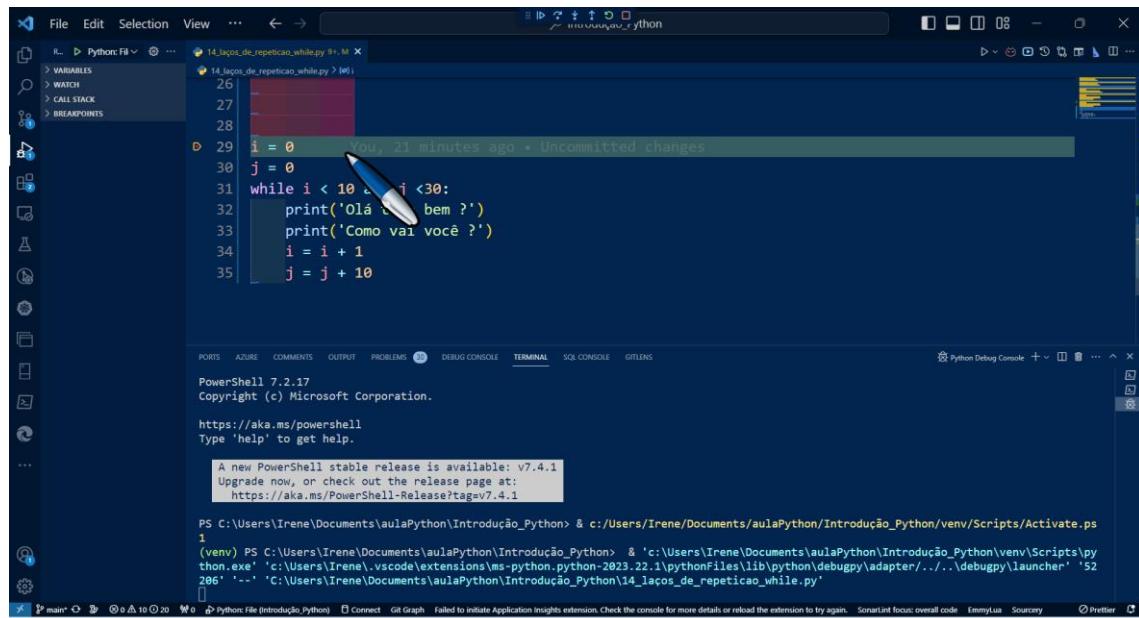
Após ter fechado ficará como print abaixo, isto vai depender do terminal que está usando mas vai fechar conforme print acima



Roda novamente o código



Após rodar o debug novamente vai aparecer conforme print abaixo



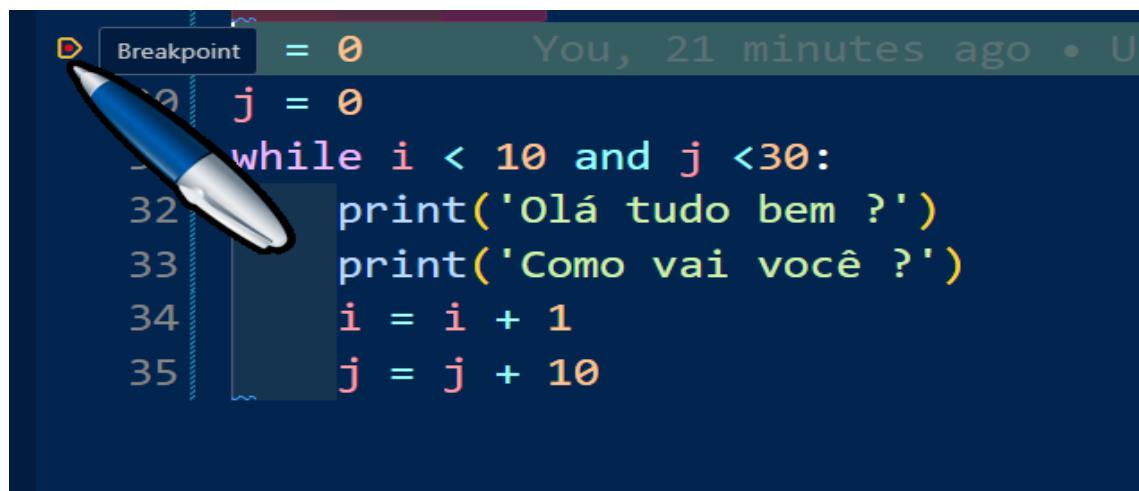
```
14_lacos_de_repeticao_while.py 91 M
26
27
28
29 i = 0 You, 21 minutes ago * Uncommitted changes
30 j = 0
31 while i < 10 and j <30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
PowerShell 7.2.17
Copyright (c) Microsoft Corporation.
https://aka.ms/powershell
Type 'help' to get help.

A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps
1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' '52
206' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'
[...]
```

Esta seta mostra em qual linha nosso código está.

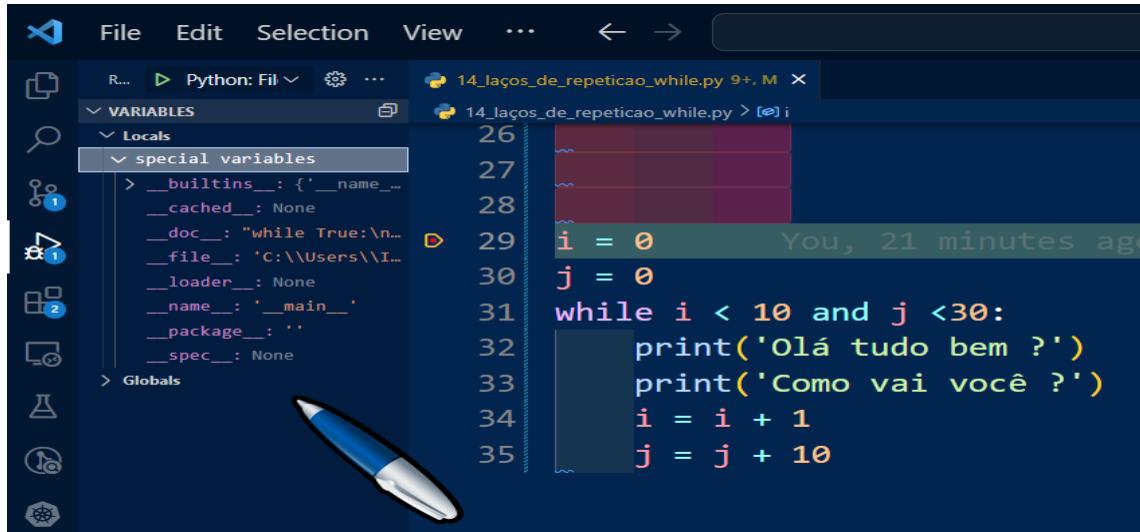


```
Breakpoint = 0 You, 21 minutes ago • U
29 j = 0
30 while i < 10 and j <30:
31     print('Olá tudo bem ?')
32     print('Como vai você ?')
33     i = i + 1
34     j = j + 10
```

Quando abrirmos variáveis vamos saber quais variáveis estão no sistema



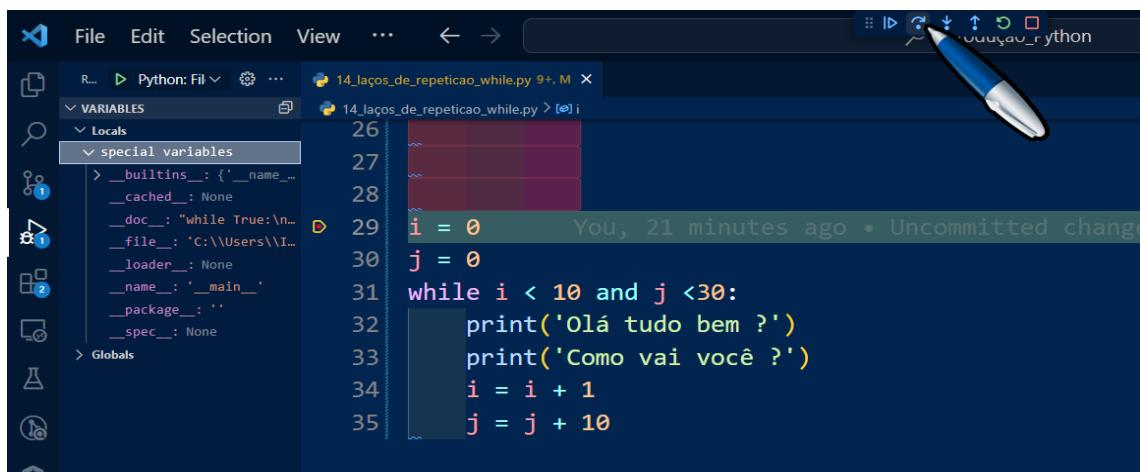
Quando abrirmos variáveis vamos saber quais variáveis estão no sistema. Variáveis que o python já tem por conta própria.



The screenshot shows the Visual Studio Code interface with a Python file named '14_laços_de_repeticao_while.py' open. The code contains a while loop that prints two messages and increments two variables, i and j. The Variables sidebar on the left shows the current state of the 'Locals' scope, which includes special variables like __builtins__, __cached__, __doc__, __file__, __loader__, __name__, __package__, and __spec__. The code editor shows the following:

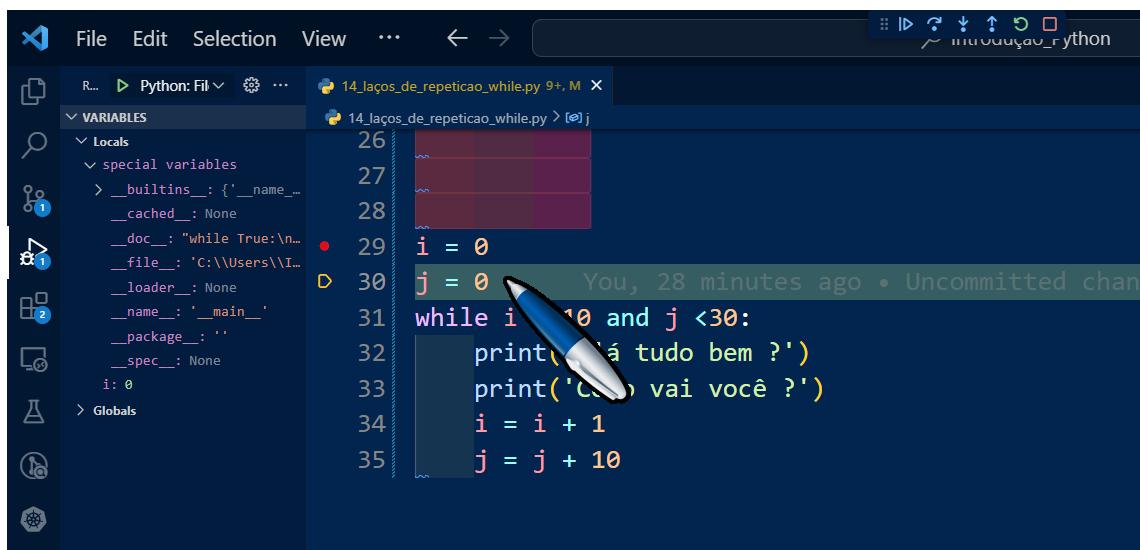
```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```

Vamos clicar conforme print abaixo para mudar de linha, vai aparecer linha J



The screenshot shows the Visual Studio Code interface with the same Python file open. A cursor is placed on the first 'print' statement. The Variables sidebar remains the same. The code editor shows the following:

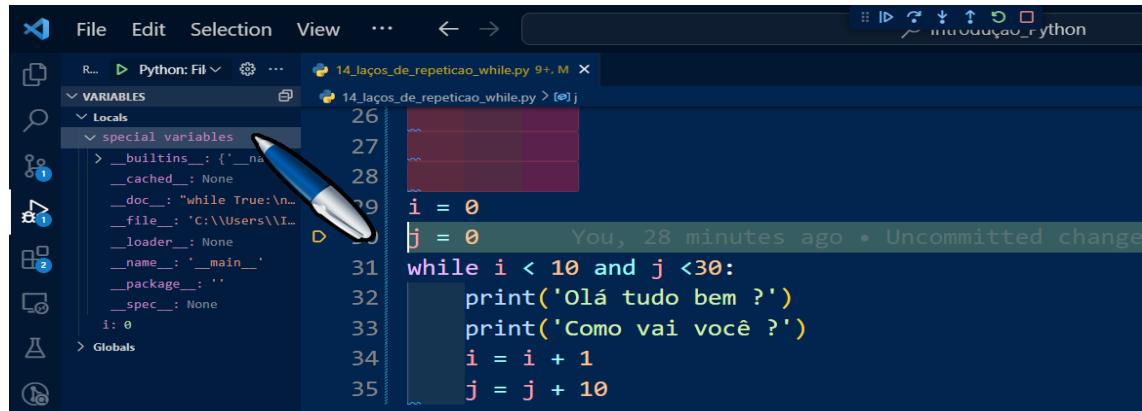
```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```



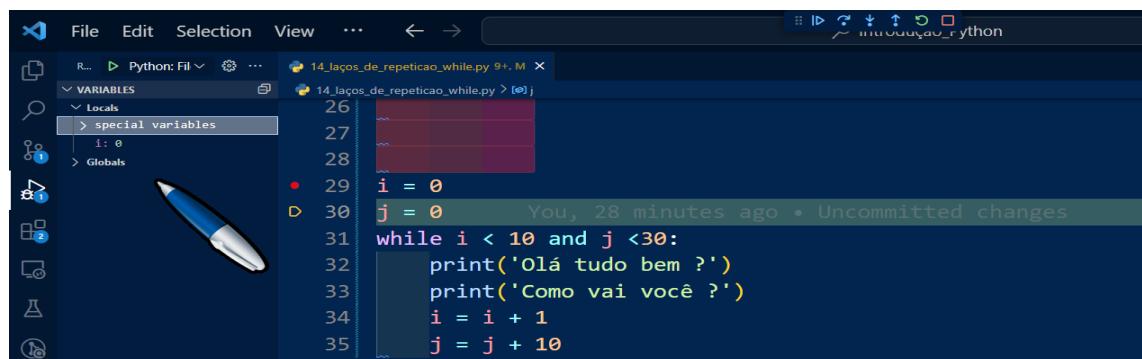
The screenshot shows the Visual Studio Code interface with the same Python file open. The cursor has moved to the second 'print' statement. The Variables sidebar remains the same. The code editor shows the following:

```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```

Vou fechar este special variables e vai ficar apenas a variável que foi criada

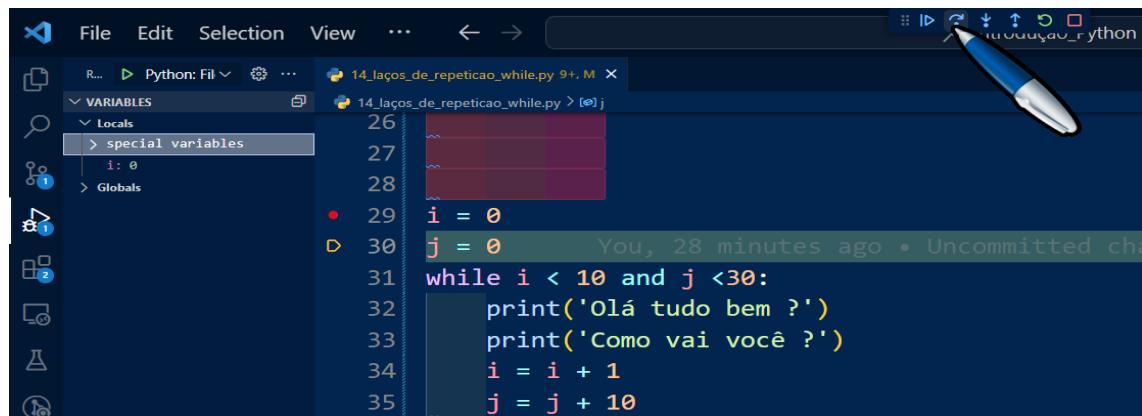


```
14_laços_de_repetição_while.py 9+, M
26
27
28
29 i = 0
30 j = 0 You, 28 minutes ago • Uncommitted change
31 while i < 10 and j <30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```



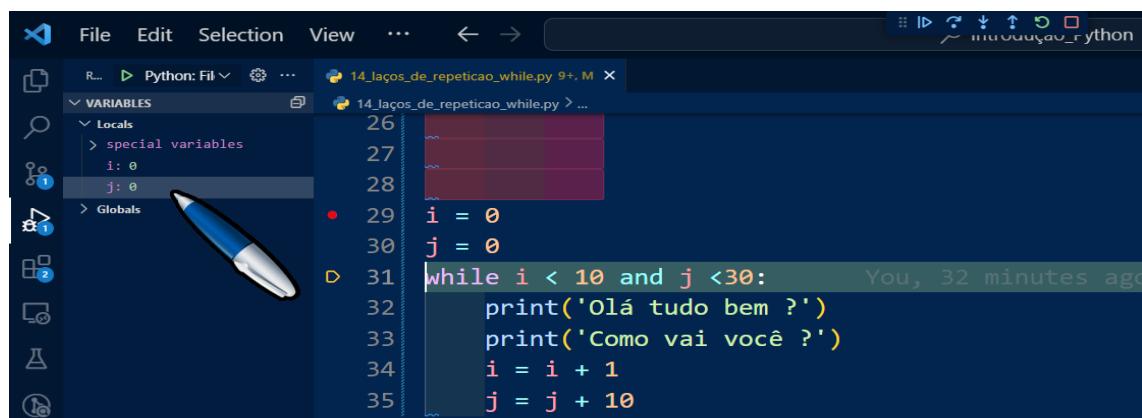
```
14_laços_de_repetição_while.py 9+, M
26
27
28
29 i = 0
30 j = 0 You, 28 minutes ago • Uncommitted changes
31 while i < 10 and j <30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```

Quando rodar novamente



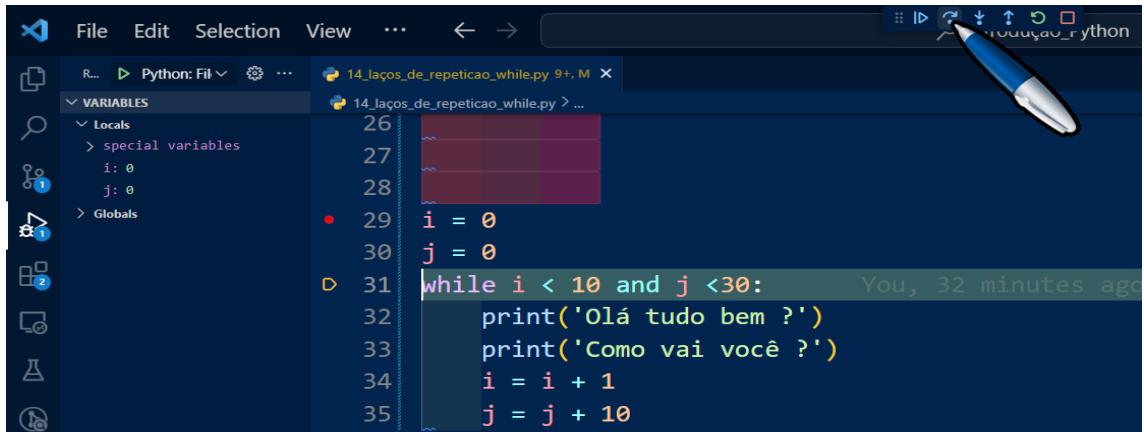
```
14_laços_de_repetição_while.py 9+, M
26
27
28
29 i = 0
30 j = 0 You, 28 minutes ago • Uncommitted change
31 while i < 10 and j <30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```

Vai aparecer a variável J



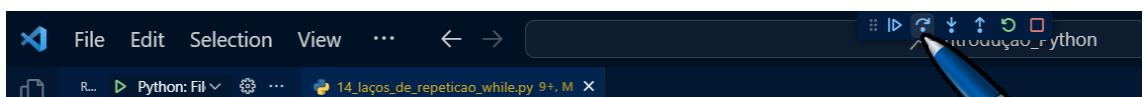
```
14_laços_de_repetição_while.py 9+, M
26
27
28
29 i = 0
30 j = 0
31 while i < 10 and j <30: You, 32 minutes ago
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```

Vair rodando o código e observa que a seta vai descendo e executando o código

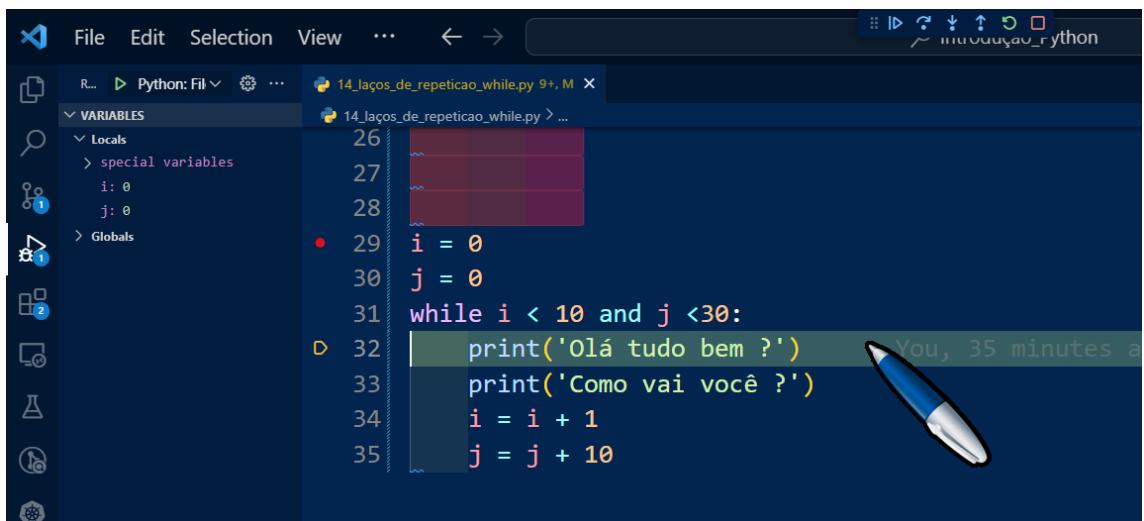


```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```

Clica na seta acima



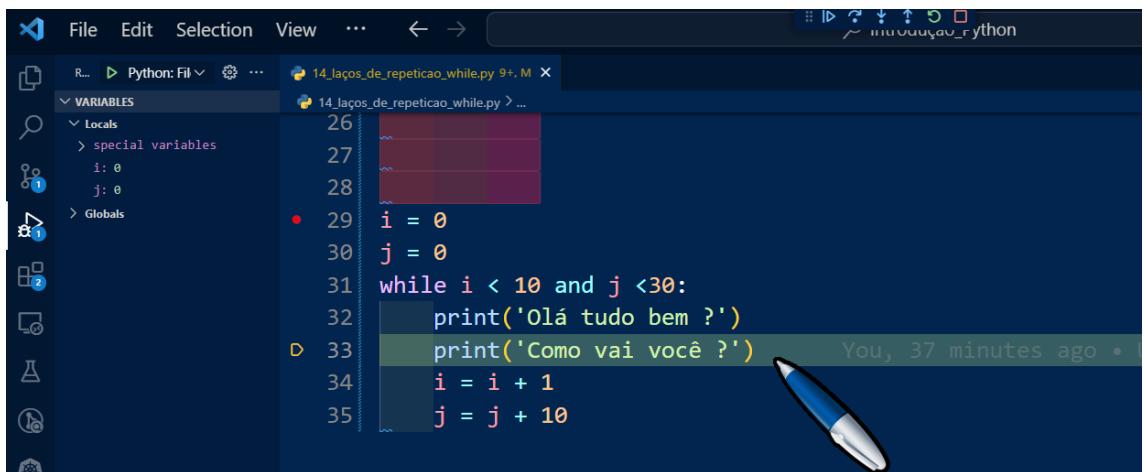
```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```



```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```

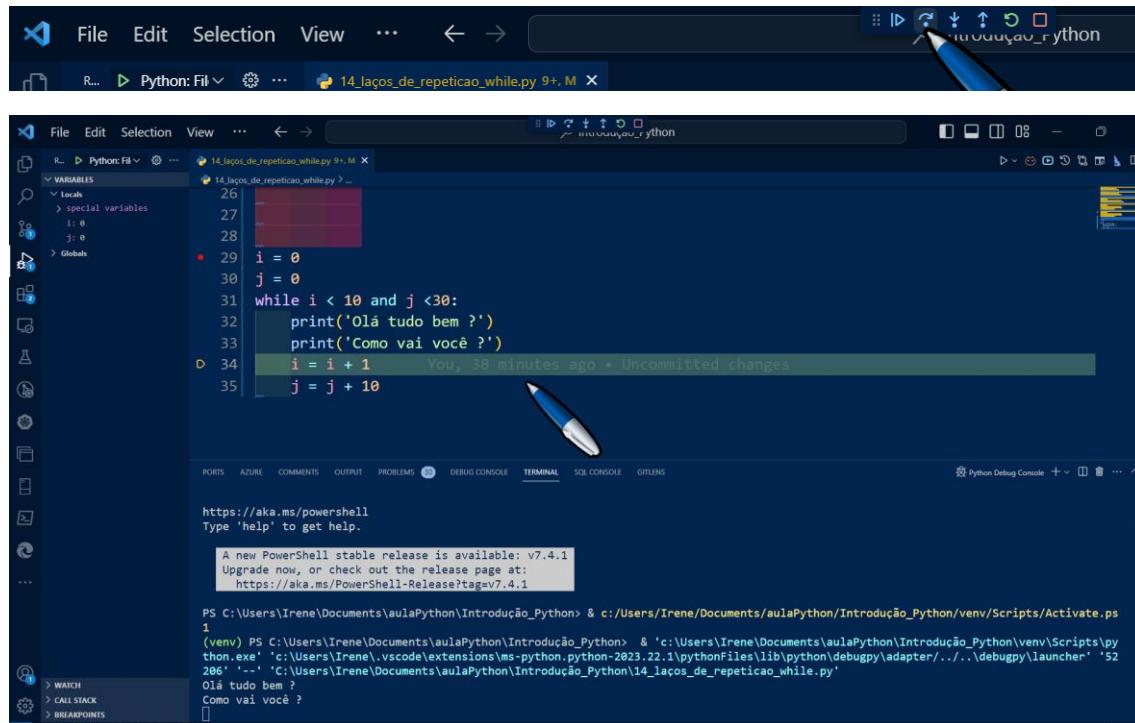


```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```



```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```

Clica na seta acima

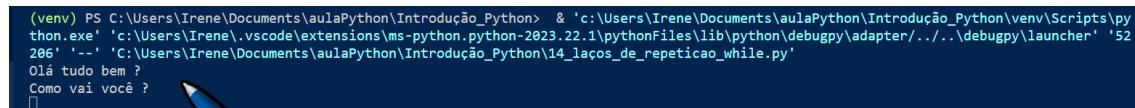


```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```

https://aka.ms/powershell
Type 'help' to get help.
A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
<https://aka.ms/PowerShell-Release?tag=v7.4.1>

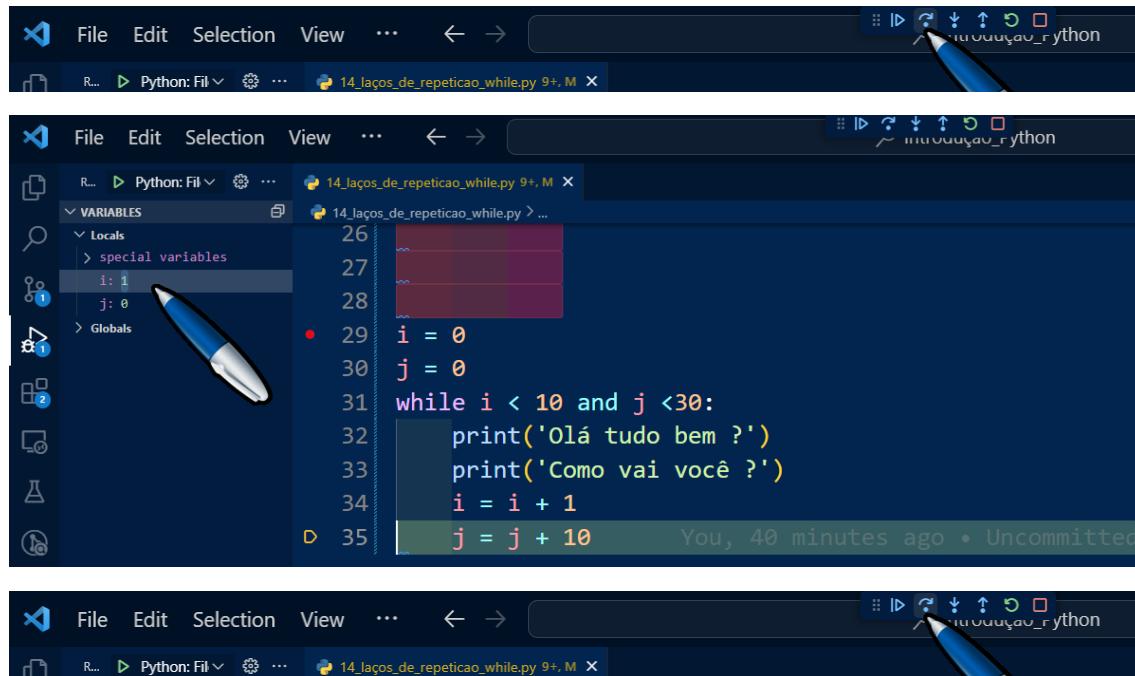
```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps
1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' '52
206' '' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
Olá tudo bem ?
Como vai você ?
```

Observa abaixo no terminal



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' '52
206' '' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
Olá tudo bem ?
Como vai você ?
```

Após clicar na seta o i vai valer 1

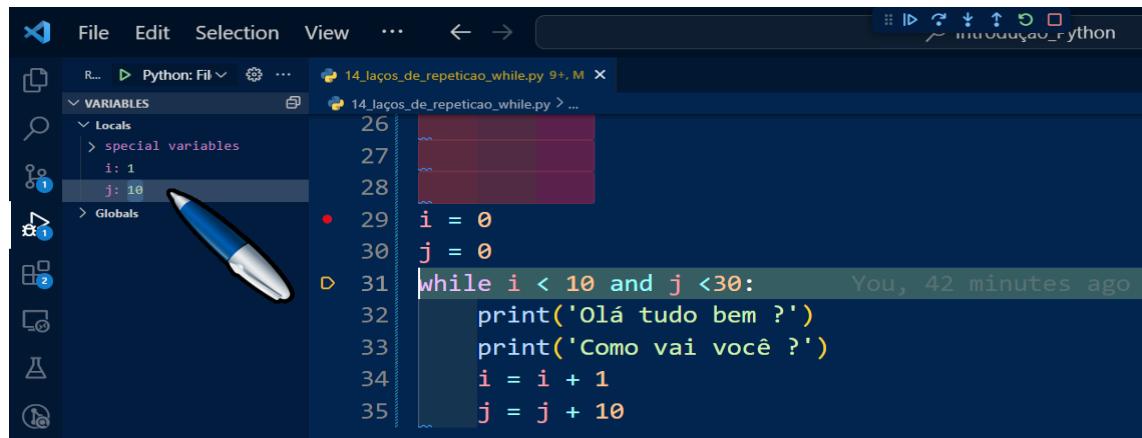


i = 0

i = 1

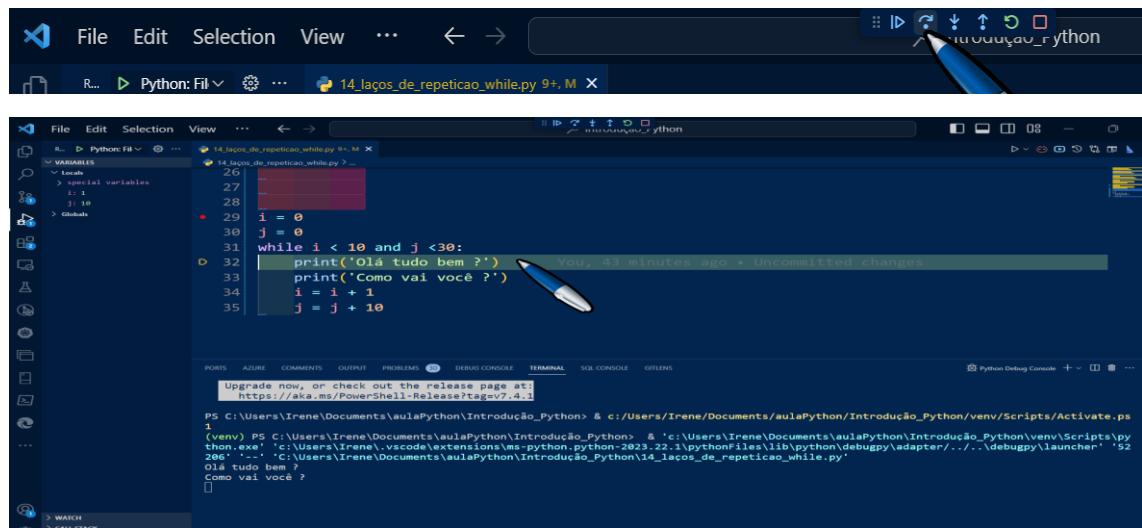
i = 2

O i vai valer 10



```
14_laços_de_repetição_while.py 9+, M
26
27
28
29 i = 0
30 j = 0
31 while i < 10 and j < 30: You, 42 minutes ago
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```

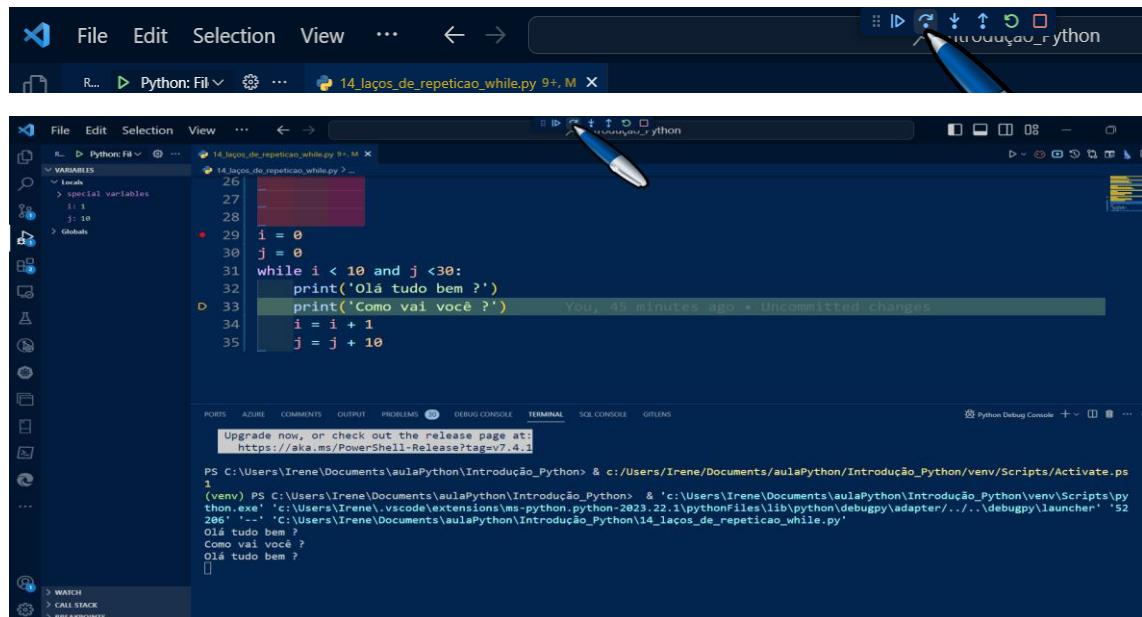
Vai observando as mudanças no terminal



```
14_laços_de_repetição_while.py 9+, M
26
27
28
29 i = 0
30 j = 0
31 while i < 10 and j < 30:
32     print("Olá tudo bem ?") You, 43 minutes ago * Uncommitted changes
33     print("Como vai você ?")
34     i = i + 1
35     j = j + 10
```

Upgrading now, or check out the release page at: https://aka.ms/PowerShell-Release?tag=v7.4.1

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps
1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/../../debugpy/launcher' '52
206' '-- -- 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
```



```
14_laços_de_repetição_while.py 9+, M
26
27
28
29 i = 0
30 j = 0
31 while i < 10 and j < 30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?') You, 45 minutes ago * Uncommitted changes
34     i = i + 1
35     j = j + 10
```

Upgrading now, or check out the release page at: https://aka.ms/PowerShell-Release?tag=v7.4.1

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps
1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/../../debugpy/launcher' '52
206' '-- -- 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Olá tudo bem ?
```



```
14_laços_de_repetição_while.py 9+, M
```

File Edit Selection View ...

14_lacos_de_repeticao_while.py 9+ M

VARIABLES

Locals

i: 1
j: 10

Globals

```
26
27
28
29 i = 0
30 j = 0
31 while i < 10 and j < 30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```

Upgrades now, or check out the release page at: <https://aka.ms/PowerShell-Release?tag=v7.4.1>

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' ./debugpy/launcher' '52 206' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'

Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Como vai você ?

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

Python Debug Console +

File Edit Selection View ...

14_lacos_de_repeticao_while.py 9+ M

VARIABLES

Locals

i: 1
j: 10

Globals

```
26
27
28
29 i = 0
30 j = 0
31 while i < 10 and j < 30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```

Upgrades now, or check out the release page at: <https://aka.ms/PowerShell-Release?tag=v7.4.1>

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' ./debugpy/launcher' '52 206' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'

Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Como vai você ?

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

Python Debug Console +

File Edit Selection View ...

14_lacos_de_repeticao_while.py 9+ M

VARIABLES

Locals

i: 2
j: 20

Globals

```
26
27
28
29 i = 0
30 j = 0
31 while i < 10 and j < 30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```

Upgrades now, or check out the release page at: <https://aka.ms/PowerShell-Release?tag=v7.4.1>

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' ./debugpy/launcher' '52 206' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'

Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Como vai você ?

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

Python Debug Console +

```
File Edit Selection View ... ← → 
R... Python: Fil ... 14_laços_de_repetição_while.py 9+, M X

File Edit Selection View ... ← → 
14_laços_de_repetição_while.py 9+, M X

VARIABLES
Locals > special variables
i: 2
j: 20
> Globals
i = 0
j = 0
while i < 10 and j < 30: You, 47 minutes ago * Uncommitted changes
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
Upgrade now, or check out the release page at: https://aka.ms/PowerShell-Release?tag=v7.4.1
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps
1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/../../debugpy/launcher' '52
206' '' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\14_laços_de_repetição_while.py'
Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Como vai você ?
```

```
File Edit Selection View ... ← → 
R... Python: Fil ... 14_laços_de_repetição_while.py 9+, M X

File Edit Selection View ... ← → 
14_laços_de_repetição_while.py 9+, M X

VARIABLES
Locals > special variables
i: 2
j: 20
> Globals
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
Upgrade now, or check out the release page at: https://aka.ms/PowerShell-Release?tag=v7.4.1
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps
1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/../../debugpy/launcher' '52
206' '' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\14_laços_de_repetição_while.py'
Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Como vai você ?
```

```
File Edit Selection View ... ← → 
R... Python: Fil ... 14_laços_de_repetição_while.py 9+, M X
```

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_laços_de_repetição_while.py' with the following code:

```
i = 0
j = 0
while i < 10 and j < 30:
    print('Olá tudo bem ?')
    print('Como vai você ?')
    i = i + 1
    j = j + 10
```

The code is highlighted in green, indicating it is valid Python. Below the code, the terminal window shows the output of running the script:

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps
1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' '52
206' '--' 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
Como vai você ?
Olá tudo bem ?
```

A second screenshot of the VS Code interface, showing the same Python file and terminal output as the first one. The code and its execution results are identical.

A third screenshot of the VS Code interface, showing the same Python file and terminal output as the previous ones. The code and its execution results are identical.

The screenshot shows the Visual Studio Code interface with three windows open:

- Top Window:** Shows the Python file `14_laços_de_repetição_while.py` with code demonstrating a while loop. The code prints "Olá tudo bem ?" and increments variables `i` and `j`. The status bar indicates "Uncommitted changes".
- Middle Window:** A terminal window showing the command `python 14_laços_de_repetição_while.py` being run. The output shows the program's interaction with the user, printing "Olá tudo bem ?" five times.
- Bottom Window:** Another instance of the Python file `14_laços_de_repetição_while.py`, showing the same code and terminal output.

Ai termina a linha porque não foi mais executado dentro da linha do wile

A screenshot of the Visual Studio Code interface. The top bar shows the title "Introdução_Python". The left sidebar has icons for File, Edit, Selection, View, and others. A search bar is at the top right. The main area shows a Python file named "14_laços_de_repetição_while.py" with the following code:

```
26
27
28
29 • i = 0
30 j = 0
31 while i < 10 and j < 30:
32     print('Olá tudo bem ?')
33     print('Como vai você ?')
34     i = i + 1
35     j = j + 10
```

A blue pen icon is positioned over the line "i = 0". A tooltip from a debugger shows the value "You, 4 minutes ago * Uncommitted changes". The bottom navigation bar includes tabs for PORTS, AZURE, COMMENTS, OUTPUT, PROBLEMS, DEBUG CONSOLE, TERMINAL, SQL CONSOLE, and GIT LENS. The DEBUG CONSOLE tab is active, showing command-line output. The status bar at the bottom indicates the current path: "PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\Activate.ps1". The terminal output shows the execution of the script, printing "Olá tudo bem ?" and "Como vai você ?" ten times each.

20 - Estrutura de repetição wh Vamos interromper um laço

```
37 i = 0
38
39 while i < 10:
40     print(i)
41     i = i + 1
```

```
37 i = 0
38
39 while i < 10:
40     print(i)
41     i = i + 1
```



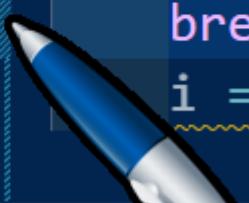
Vamos rodar o código

```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repeticao_while.py
```

```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repeticao_while.py
0
1
2
3
4
5
6
7
8
9
```

Agora vamos interromper o laço de repetição com um break

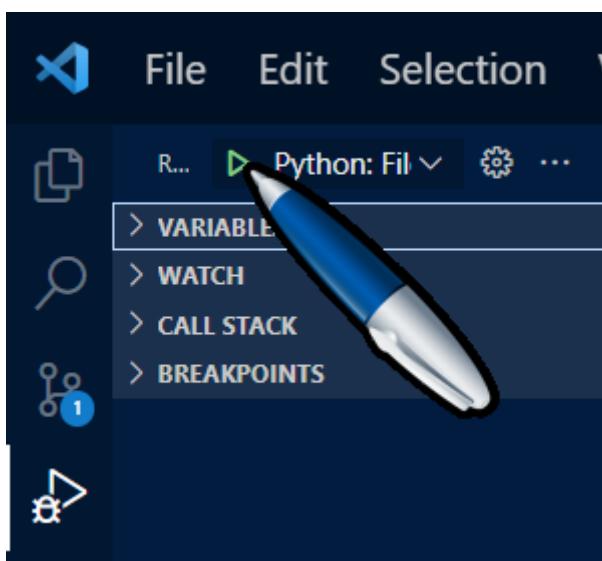
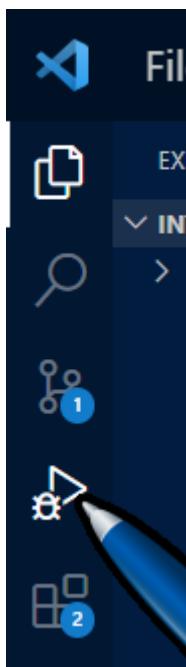
```
43 i = 0
44
45 while i < 10:
46     print(i)
47     break
48     i = i + 1
49
```

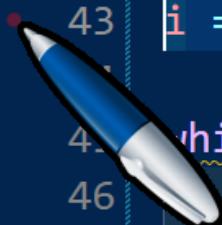


Vamos rodar o código

```
9
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repeticao_while.py
0
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

Você pode verificar no debug como passo a passo acima printado.





A screenshot of a Python code editor showing a script named '14_laços_de_repeticao_while.py'. The code contains a while loop that prints integers from 0 to 9. A breakpoint is set on the line 'i = i + 1' at line 42. The status bar at the bottom right shows 'You, 6 m'. A blue pen is drawn over the line 'i = i + 1'.

```
14_laços_de_repeticao_while.py
37     i = 0
38
39 while i < 10:
40     print(i)
41     i = i + 1
42
43 i = 0
44
45 while i < 10:
46     print(i)
47     break
48     i = i + 1
```



A screenshot of a Python code editor showing the same script '14_laços_de_repeticao_while.py'. The code is identical to the first screenshot. A breakpoint is set on the line 'i = i + 1' at line 42. The status bar at the bottom right shows 'You, 6 minutes ago'. A blue pen is drawn over the line 'i = i + 1'.

```
14_laços_de_repeticao_while.py
37     i = 0
38
39 while i < 10:
40     print(i)
41     i = i + 1
42
43 i = 0
44
45 while i < 10:
46     print(i)
47     break
48     i = i + 1
```



A screenshot of the Python Breakpoints tool window in Visual Studio Code. The window title is "Python: File". It shows a code editor with the file "14_laços_de_repetição_while.py". Line 37 has a breakpoint set at the start of the loop. The status bar at the bottom says "Breakpoint = 0". A blue pen is placed over the breakpoint icon on line 37.

```
37     i = 0
38
39 while i < 10:
40     print(i)
41     i = i + 1'''
```

A screenshot of the Python Breakpoints tool window in Visual Studio Code. The window title is "Python: File". It shows the same code editor as the previous screenshot. The breakpoint at line 37 is now removed. The status bar at the bottom says "i = 0". A blue pen is placed over the variable "i" in the status bar.

```
37     i = 0
38
39 while i < 10:
40     print(i)
41     i = i + 1'''
```

The screenshot shows the Microsoft Visual Studio Code interface. On the left is the sidebar with icons for Variables, Watch, Call Stack, and Breakpoints. The main editor window displays a Python file named '14_lacos_de_repeticao_while.py' with the following code:

```
14_lacos_de_repeticao_while.py
1  i = 0
2
3  while i < 10:
4      print(i)
5      i = i + 1
6
7  i = 0
8
9  while i < 10:
10     print(i)
11     break
12     i = i + 1
```

A status bar at the bottom indicates "You, 8 minutes ago * Uncommitted changes". Below the editor is a PowerShell terminal window titled "PowerShell 7.2.17" showing the following output:

```
PowerShell 7.2.17
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1
```

The bottom of the screen shows the Windows taskbar with various pinned icons.

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder structure for 'aulaPython' containing 'Introdução_Python' and '14_lacos_de_repeticao_while.py'.
- Code Editor:** Displays the Python script '14_lacos_de_repeticao_while.py'. The code uses a while loop to print numbers from 0 to 9.

```
1 i = 0
2
3 while i < 10:
4     print(i)
5     i = i + 1
6
7 i = 0
8
9 while i < 10:
10    print(i)
11    break
12    i = i + 1
```
- Output Panel:** Shows a message from PowerShell: "A new PowerShell stable release is available: v7.4.1 Upgrade now, or check out the release page at: https://aka.ms/PowerShell-Release?tag=v7.4.1".
- Bottom Status Bar:** Shows the path 'PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & c:/Users/Irene/Documents/aulaPython\Introdução_Python\venv\Scripts\Activate.ps1' and the command 'python.exe'.

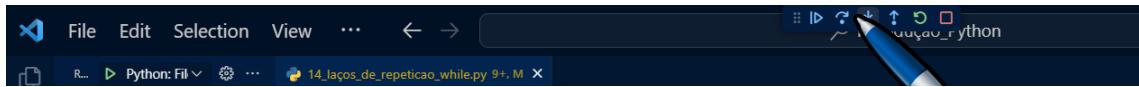
The screenshot shows the Visual Studio Code interface with the following details:

- File Structure:** The left sidebar shows a tree view with '14_lacos_de_repeticao_while.py' selected.
- Code Editor:** The main editor shows the following Python code:

```
14_lacos_de_repeticao_while.py
 1 i = 0
 2
 3 while i < 10:
 4     print(i)
 5     i = i + 1
 6
 7 i = 0
 8
 9 while i < 10:
10     print(i)
11     break
12     i = i + 1
```
- Terminal:** The bottom terminal window displays a PowerShell session:

```
Copyright (c) Microsoft Corporation.
https://aka.ms/powershell
Type 'help' to get help.

A new PowerShell stable release is available: v7.4.1
Update now! Check the release page at
https://aka.ms/PowerShell-Release?tag=v7.4.1
```
- Status Bar:** The status bar indicates 'You 12 minutes ago' and '1 uncommitted changes'.
- Side Panels:** The 'PROBLEMS' panel shows one error: 'SyntaxError: invalid syntax'. The 'OUTPUT' panel shows log messages from the PowerShell session.



Acabou o programa porque saiu do laço



The screenshot shows a Python script named `14_lacos_de_repeticao_while.py` in a code editor. The code contains two `while` loops. The first loop prints integers from 0 to 9. The second loop prints integers from 0 to 9, but includes a `break` statement which exits the loop after the first iteration. A red dot in the gutter indicates a breakpoint at line 42. A red box highlights the `break` keyword. The status bar at the bottom shows a copyright notice from Microsoft and a PowerShell upgrade message.

```
14_lacos_de_repeticao_while.py
  1 i = 0
  2
  3
  4 while i < 10:
  5     print(i)
  6     i = i + 1
  7
  8 i = 0
  9
 10 while i < 10:
 11     print(i)
 12     break
 13     i = i + 1

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
Copyright (c) Microsoft Corporation.
https://aka.ms/powershell
Type 'help' to get help.

A new PowerShell stable release is available! - v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1
```

Se colocarmos um if

```
43     i = 0
44
45 while i < 10:
46     print(i)
47     break
48     i = i + 1
49
50
51 i = 0
52
53 while i < 10:
54     print(i)
55     if i >= 5:
56         break
57     i = i + 1
```

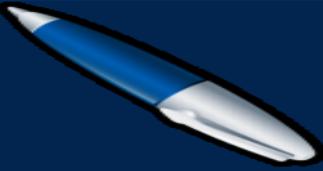


1 second ago • Uncommi

Rodar o código

```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repetição_while.py
0
1
2
3
4
5
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> 
```

Vamos colocar $i > 2$



```
'''i = 0
while i < 10:
    print(i)
    if i >= 5:
        break
    i = i + 1'''

i = 0

while i < 10:
    print(i)
    if i >= 2:          You, 1 seco
        break
    i = i + 1
```



```
5
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repeticao_while.py
0
1
2
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

Aqui você pode debugar conforme aprendido acima

A screenshot of a Python code editor interface. The menu bar includes File, Edit, Selection, View, ..., and a toolbar with icons for file operations. The left sidebar has a 'VARIABLES' section with options for VARIABLES, WATCH, CALL STACK, and BREAKPOINTS. The main area shows a script named '14_laços_de_repetição_while.py'. The code contains a while loop that prints values of 'i' from 0 to 4, then breaks, and then prints values from 0 to 9 again. A red rectangular selection highlights the line 'i = i + 1' at the end of the first loop iteration. A blue pen icon is visible in the top right corner.

```
54 print(i)
55 if i >= 5:
56     break
57 i = i + 1''''
58
59
60 i = 0 You, 4 minutes ago • Uncommitted changes
61
62 while i < 10:
63     print(i)
64     if i >= 2:
65         break
66     i = i + 1
```

A screenshot of the same Python code editor interface, showing the state of variables. The 'VARIABLES' sidebar now shows the 'Locals' section, which includes 'special variables' and 'Globals'. Under 'special variables', 'i' is listed with a value of 0. The code window remains the same, with the red selection and blue pen icon.

```
54 print(i)
55 if i >= 5:
56     break
57 i = i + 1''''
58
59
60 i = 0
61
62 while i < 10: You, 5 minutes ago • Uncommitted changes
63     print(i)
64     if i >= 2:
65         break
66     i = i + 1
```

A screenshot of the code editor interface, showing the 'VARIABLES' section collapsed in the sidebar. The code window and other interface elements remain the same.

```
54 print(i)
55 if i >= 5:
56     break
57 i = i + 1''''
58
59
60 i = 0
61
62 while i < 10: You, 5 minutes ago • Uncommitted changes
63     print(i)
64     if i >= 2:
65         break
66     i = i + 1
```

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_lacos_de_repeticao_while.py' with the following code:

```
print(i)
if i >= 5:
    break
i = i + 1

i = 0
while i < 10:
    print(i)
    if i >= 2:
        break
    i = i + 1
```

The code editor has a dark theme. A blue pen icon is positioned above the code. The status bar at the bottom indicates 'You, 6 minutes ago * Uncommitted changes'. The terminal tab is active, showing the command line output:

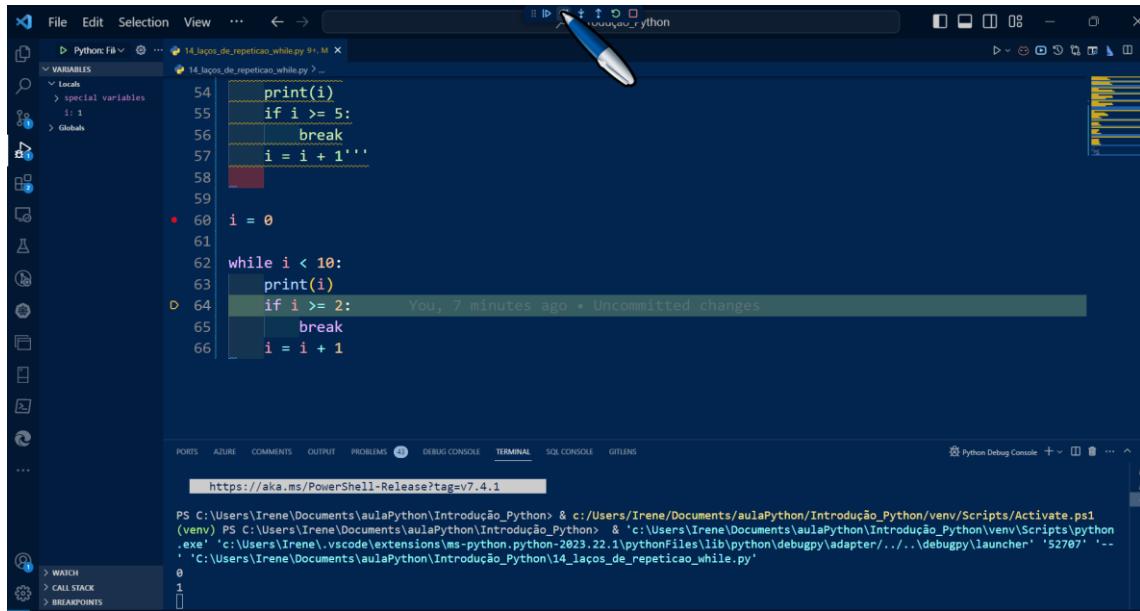
```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/..../debugpy/launcher' '52707' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'
0
```

A second screenshot of the VS Code interface, showing the same Python file '14_lacos_de_repeticao_while.py'. The code is identical to the first screenshot. The status bar at the bottom indicates 'You, 6 minutes ago * Uncommitted changes'. The terminal tab is active, showing the command line output:

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/..../debugpy/launcher' '52707' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'
0
```

A third screenshot of the VS Code interface, showing the same Python file '14_lacos_de_repeticao_while.py'. The code is identical to the previous screenshots. The status bar at the bottom indicates 'You, 6 minutes ago * Uncommitted changes'. The terminal tab is active, showing the command line output:

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/..../debugpy/launcher' '52707' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'
0
```



A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python script named `14_laços_de_repetição_while.py`. A blue pen icon is positioned over the code editor. The script contains the following code:

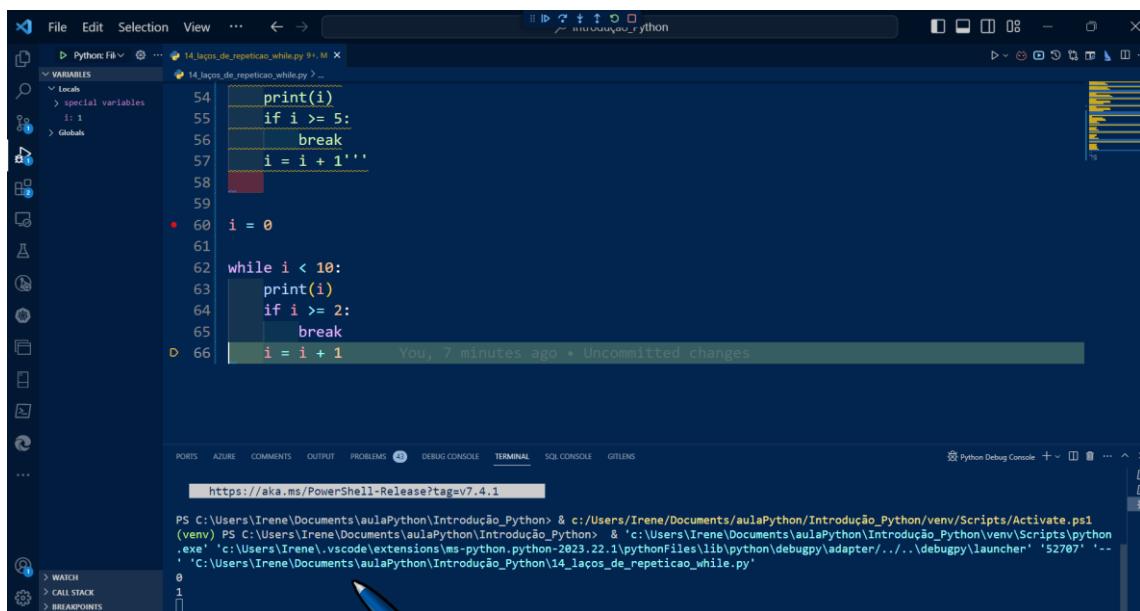
```
print(i)
if i >= 5:
    break
i = i + 1

• 60 i = 0
61
62 while i < 10:
63     print(i)
64     if i >= 2: You, 7 minutes ago * Uncommitted changes
65         break
66     i = i + 1
```

The code editor has a dark theme. The status bar at the bottom shows the URL `https://aka.ms/PowerShell-Release?tag=v7.4.1`. The terminal tab is active, displaying the command to run the script:

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1 (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython\Introdução_Python/venv/Scripts/python .exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher' '52707' '-- 'C:/Users/Irene/Documents/aulaPython\Introdução_Python\14_laços_de_repetição_while.py'
```

The left sidebar includes a 'WATCH' section with a variable `i` set to 0, and a 'CALL STACK' and 'BREAKPOINTS' section.



A second screenshot of the VS Code interface, showing the same Python script and environment. The watch panel now shows the variable `i` with a value of 1, indicating the script has executed one iteration of the loop.

```
File Edit Selection View ... < > ⓘ Python: File 14_laços_de_repetição_while.py M
VARIABLES
locals
> special variables
i: 2
> Globals
WATCH CALL STACK BREAKPOINTS
14_laços_de_repetição_while.py ...
54 print(i)
55 if i >= 5:
56     break
57 i = i + 1...
58
59
60 i = 0
61
62 while i < 10: You, 7 minutes ago * Uncommitted changes
63     print(i)
64     if i >= 2:
65         break
66     i = i + 1
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
https://aka.ms/PowerShell-Release?tag=v7.4.1
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter'..../debugpy/launcher' '52707' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
1
```

```
File Edit Selection View ... < > ⓘ Python: File 14_laços_de_repetição_while.py M
VARIABLES
locals
> special variables
i: 2
> Globals
WATCH CALL STACK BREAKPOINTS
14_laços_de_repetição_while.py ...
54 print(i)
55 if i >= 5:
56     break
57 i = i + 1...
58
59
60 i = 0
61
62 while i < 10: You, 7 minutes ago * Uncommitted changes
63     print(i)
64     if i >= 2:
65         break
66     i = i + 1
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
https://aka.ms/PowerShell-Release?tag=v7.4.1
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter'..../debugpy/launcher' '52707' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
1
```

```
File Edit Selection View ... < > ⓘ Python: File 14_laços_de_repetição_while.py M
VARIABLES
locals
> special variables
i: 2
> Globals
WATCH CALL STACK BREAKPOINTS
14_laços_de_repetição_while.py ...
54 print(i)
55 if i >= 5:
56     break
57 i = i + 1...
58
59
60 i = 0
61
62 while i < 10: You, 8 minutes ago * Uncommitted changes
63     print(i)
64     if i >= 2:
65         break
66     i = i + 1
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter'..../debugpy/launcher' '52707' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
1
2
```

A screenshot of the Visual Studio Code interface. The main area shows a Python file named '14_laços_de_repetição_while.py'. The code contains a while loop that prints numbers from 0 to 4. A break statement is present at line 65, which stops the loop. The status bar at the bottom indicates 'You, 9 minutes ago * Uncommitted changes'. The terminal tab is active, showing the command line path and the execution of the script.

```
54     print(i)
55     if i >= 5:
56         break
57     i = i + 1...
58
59
60 i = 0
61
62 while i < 10:
63     print(i)
64     if i >= 2:
65         break
66     i = i + 1
```

Fechou

A screenshot of the Visual Studio Code interface. The main area shows the same Python file '14_laços_de_repetição_while.py'. The code is identical to the previous screenshot. The status bar at the bottom indicates 'You, 9 minutes ago * Uncommitted changes'. The terminal tab is active, showing the command line path and the execution of the script.

```
54     print(i)
55     if i >= 5:
56         break
57     i = i + 1...
58
59
60 i = 0
61
62 while i < 10:
63     print(i)
64     if i >= 2:
65         break
66     i = i + 1
```

Vamos mandar o laço de repetição para cima COTINI (continue)em vez de interromper

A screenshot of the Python code in a code editor. The code has been modified to use the 'continue' keyword instead of 'break'. The 'break' statement at line 65 has been replaced by 'continue'. The code now prints numbers from 0 to 9. The status bar at the bottom indicates 'You, 1 minute ago * Uncommitted changes'.

```
60     i = 0
61
62 while i < 10:
63     print(i)
64     if i >= 2:
65         continue
66     i = i + 1...
67
68 i = 0
69
70 while i < 10:
71     print(i)
72     i = i + 1
```

Rodar o código

```
● (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repeticao_while.py
0
1
2
3
4
5
6
7
8
9
○ (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```



Continue vai virar laço infinito

```
67
68     '''i = 0
69
70     while i < 10:
71         print(i)
72         i = i + 1'''
73
74 i = 0      You, 1 second ago • Uncommitted
75
76 while i < 10:
77     print(i)
78     continue
79     i = i + 1
```



Roda o código vai executar o 0



```
74 i = 0      You, 1 second ago • Uncommitted changes
75
76 while i < 10:
77     print(i)
78     continue
79     i = i + 1
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS python + v

```
0
0
0
0
0
0
0
0
0
0
```

Control c para interromper Ctrl+c



```
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
```

```
0
0
0
0
0
0
0
0
0
0
```

```
Traceback (most recent call last):
  File "C:\Users\Irene\Documents\aulaPython\Introdução_Python\14_laços_de_repetição_while.py", line 77, in <module>
    print(i)
KeyboardInterrupt
[venv] PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

Quando a linha do continue é executada ele vai subir de volta para laço



```
74 '''i = 0
75
76 while i < 10:
77     print(i)
78     continue
79     i = i + 1'''
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1
```

1 second ago •

Roda o código

```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 14_laços_de_repeticao_while.py
0
1
3
5
7
9
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```



Vamos debugar este código conforme aprendemos acima





```
78     print(i)
79     continue
80     i = i + 1''' You, 1 second ago
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1
```

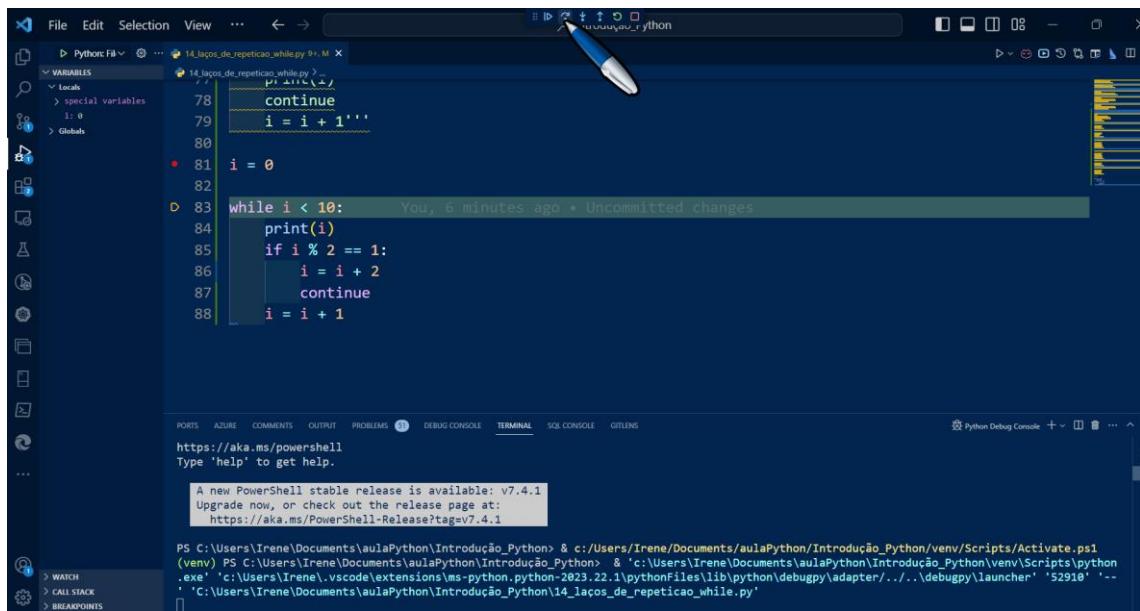


```
78     print(i)
79     continue
80     i = i + 1''' You, 1 second ago • Un
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1
```

A screenshot of the Visual Studio Code (VS Code) interface. The title bar shows "File Edit Selection View ... ← →". The left sidebar has icons for file operations like Open, Save, Find, and Debug. A large blue pen icon is overlaid on the sidebar. The main editor area shows a Python file named "14_laços_de_repetição_while.py". The code contains a while loop that prints odd numbers from 1 to 9. The code is as follows:

```
78     print(i)
79     continue
80     i = i + 1''' You, 1 second ago
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1
```

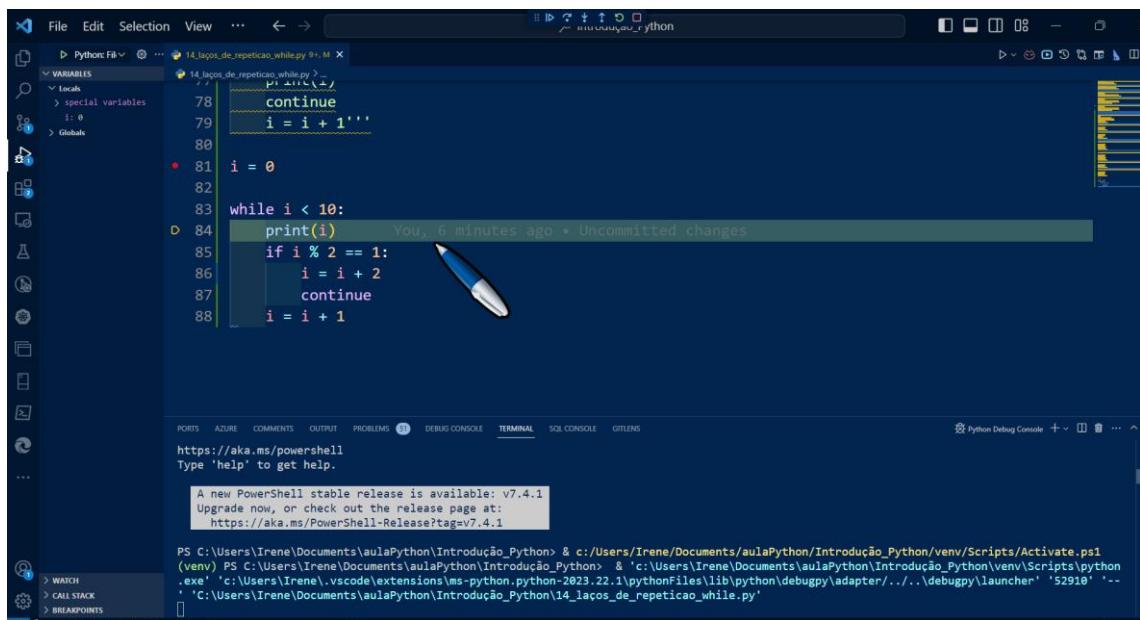
A screenshot of the Visual Studio Code interface, similar to the one above, but with a PowerShell terminal at the bottom. The terminal window shows the command "PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1 (venv)" and the output "PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv\Scripts\python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy\adapter/..\debugpy\launcher' '52910' '-- C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'". The terminal also displays a message about a new PowerShell stable release available at <https://aka.ms/powershell>. The code in the editor remains the same as the first screenshot.



A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_laços_de_repetição_while.py'. A blue pen icon is placed over the code at line 81, which contains the assignment 'i = i + 1'. The code itself is as follows:

```
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1
```

The status bar at the bottom indicates 'You, 6 minutes ago * Uncommitted changes'. The terminal tab is active, showing a PowerShell session with the command 'python 14_laços_de_repetição_while.py' entered. The output of the command is displayed in the terminal.



A second screenshot of the VS Code interface, showing the same Python script '14_laços_de_repetição_while.py'. The blue pen icon has moved to line 84, specifically over the 'print(i)' statement. The code remains the same as in the first screenshot.

A screenshot of the Visual Studio Code interface during a Python debugging session. The main area shows a code editor with a script named '14_laços_de_repetição_while.py'. The code contains a while loop that prints even numbers from 0 to 10. A break point is set at line 81. The status bar indicates 'You, 7 minutes ago * Uncommitted changes'. The bottom right corner shows a 'Python Debug Console' tab.

```
14_laços_de_repetição_while.py 9+ M
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:    You, 7 minutes ago * Uncommitted changes
86         i = i + 2
87         continue
88     i = i + 1
```

Type 'help' to get help.

```
A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1
```

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/../../debugpy/launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'

WIDGETS: WATCH, CALL STACK, BREAKPOINTS

A second screenshot of the Visual Studio Code interface during a Python debugging session. The code editor shows the same script '14_laços_de_repetição_while.py'. A break point is set at line 81. The status bar indicates 'You, 7 minutes ago * Uncommitted changes'. The bottom right corner shows a 'Python Debug Console' tab. A blue pen icon is overlaid on the interface.

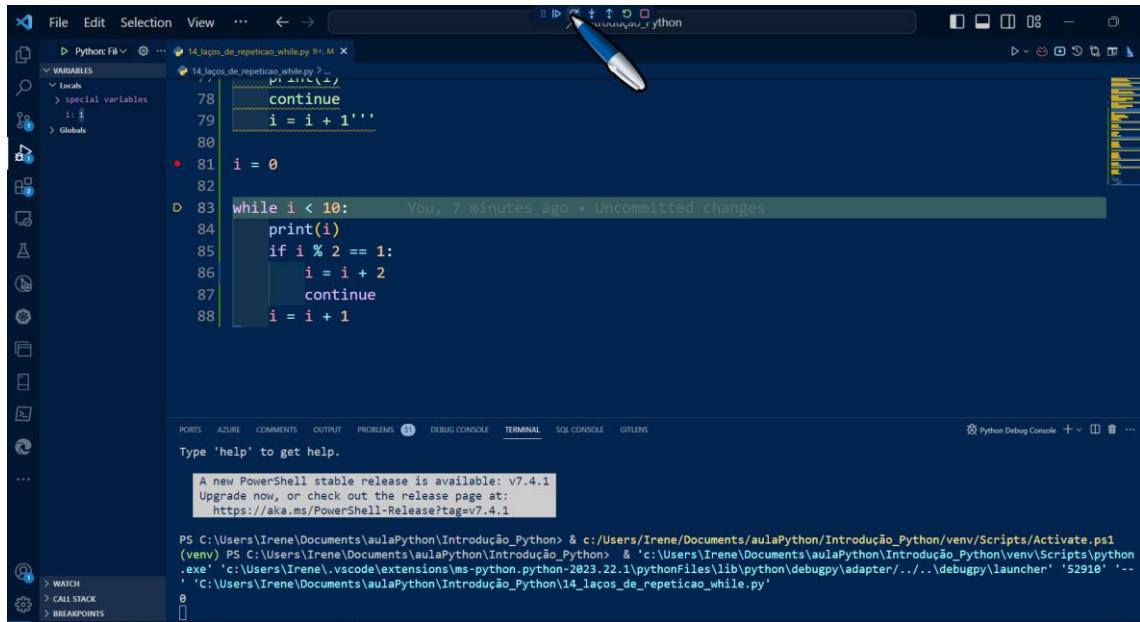
```
14_laços_de_repetição_while.py 9+ M
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1    You, 7 minutes ago * Uncommitted changes
```

Type 'help' to get help.

```
A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1
```

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/../../debugpy/launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'

WIDGETS: WATCH, CALL STACK, BREAKPOINTS

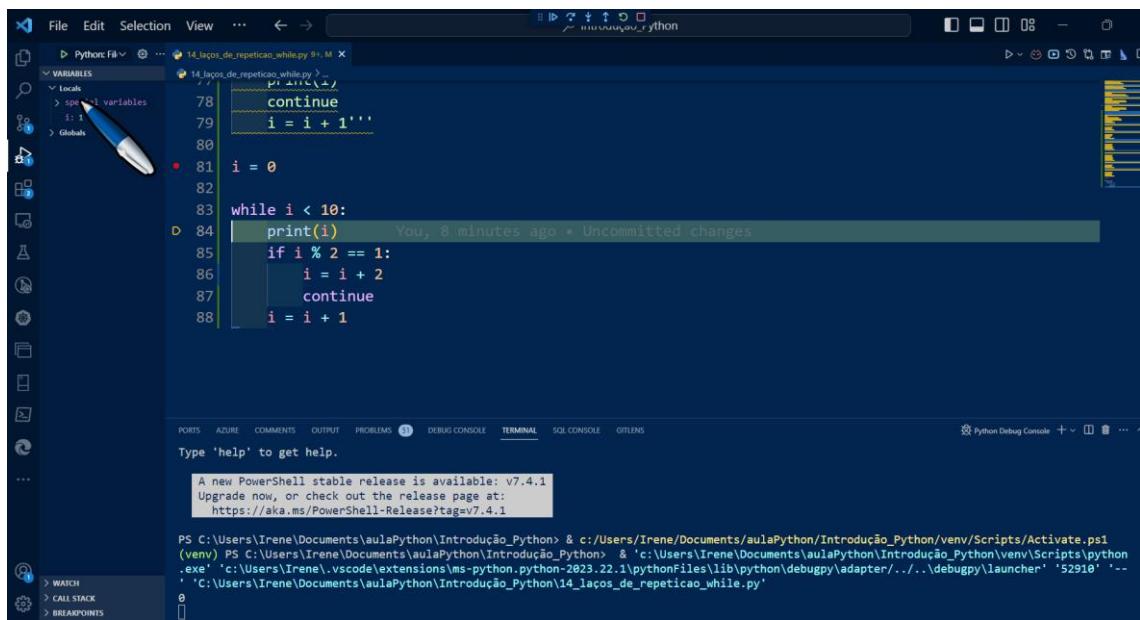


A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_laços_de_repetição_while.py'. A red dot on the left margin indicates a break point at line 81. The code contains a while loop that prints even numbers from 0 to 10. The status bar at the bottom shows 'You, 7 minutes ago * Uncommitted changes'. The terminal tab is active, displaying a PowerShell upgrade message and the command to run the script.

```
14_laços_de_repetição_while.py 9+ M
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:    You, 7 minutes ago * Uncommitted changes
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1

A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
```

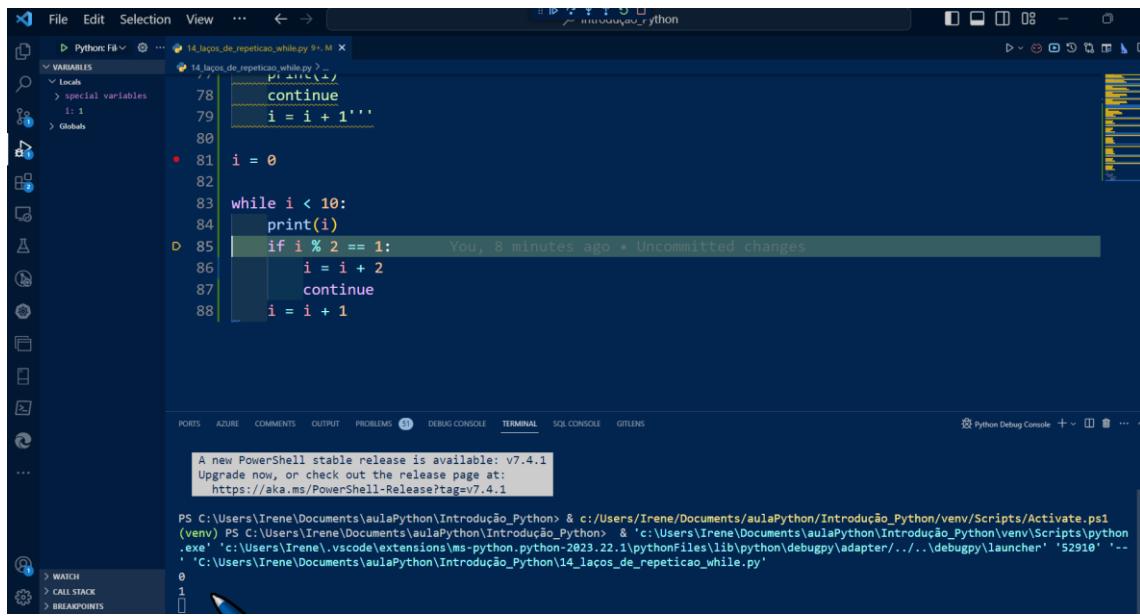


A second screenshot of VS Code showing the same Python file. The break point has moved to line 84. The status bar still shows 'You, 8 minutes ago * Uncommitted changes'. The terminal output remains the same, indicating an uncommitted change and the PowerShell upgrade message.

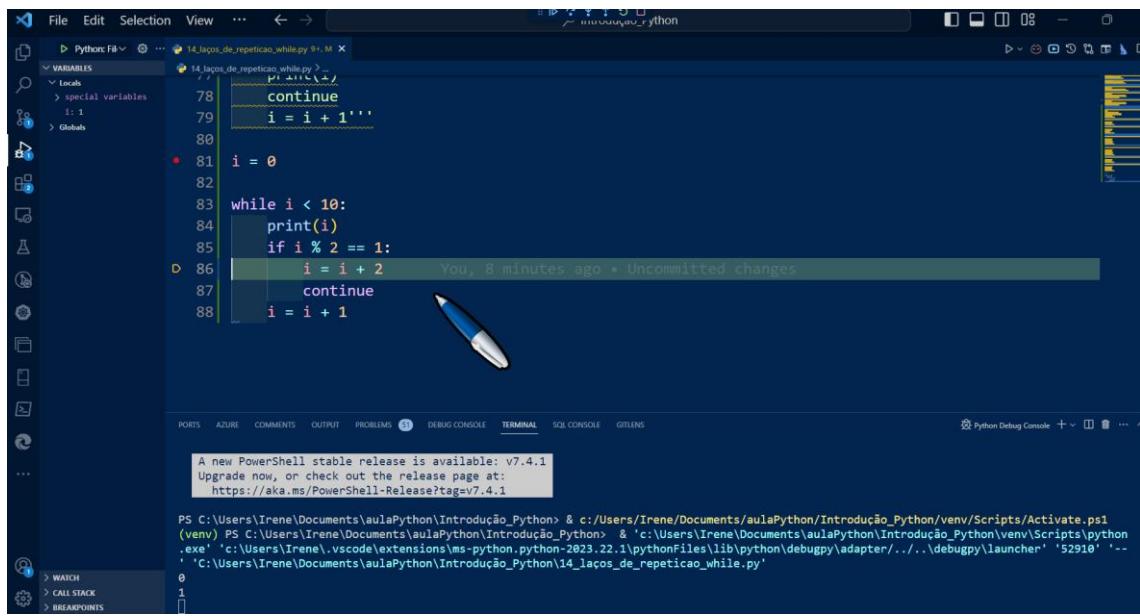
```
14_laços_de_repetição_while.py 9+ M
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:
84     print(i)    You, 8 minutes ago * Uncommitted changes
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1

A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1

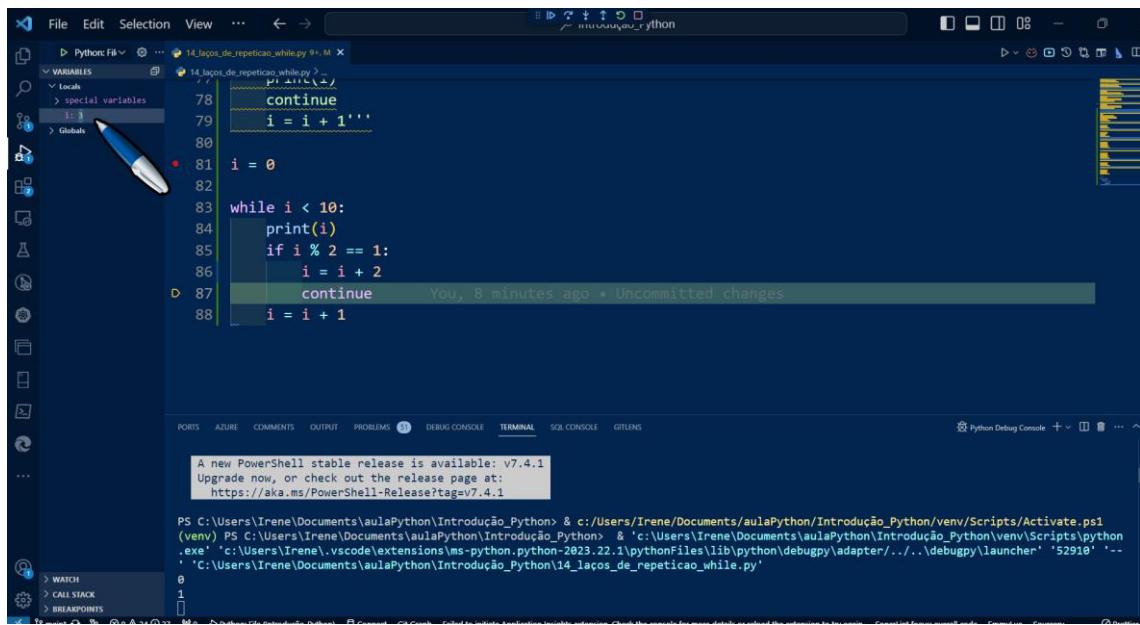
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
```



A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_laços_de_repetição_while.py'. A red dot on the left margin indicates a breakpoint at line 85. The code contains a 'while' loop that prints odd numbers from 1 to 9. A tooltip above the line 85 shows the message: 'You, 8 minutes ago * Uncommitted changes'. The bottom status bar shows the path 'C:\Users\Irene\Documents\aulaPython\Introdução_Python' and the file name '14_laços_de_repetição_while.py'. The terminal tab is active, displaying a PowerShell prompt with upgrade information.



A second screenshot of the VS Code interface, identical to the first one except for the position of the cursor. It is now placed on the line 'i = i + 2' at the start of the second iteration of the loop. A blue pen icon is overlaid on this line, indicating it is the current line of execution.



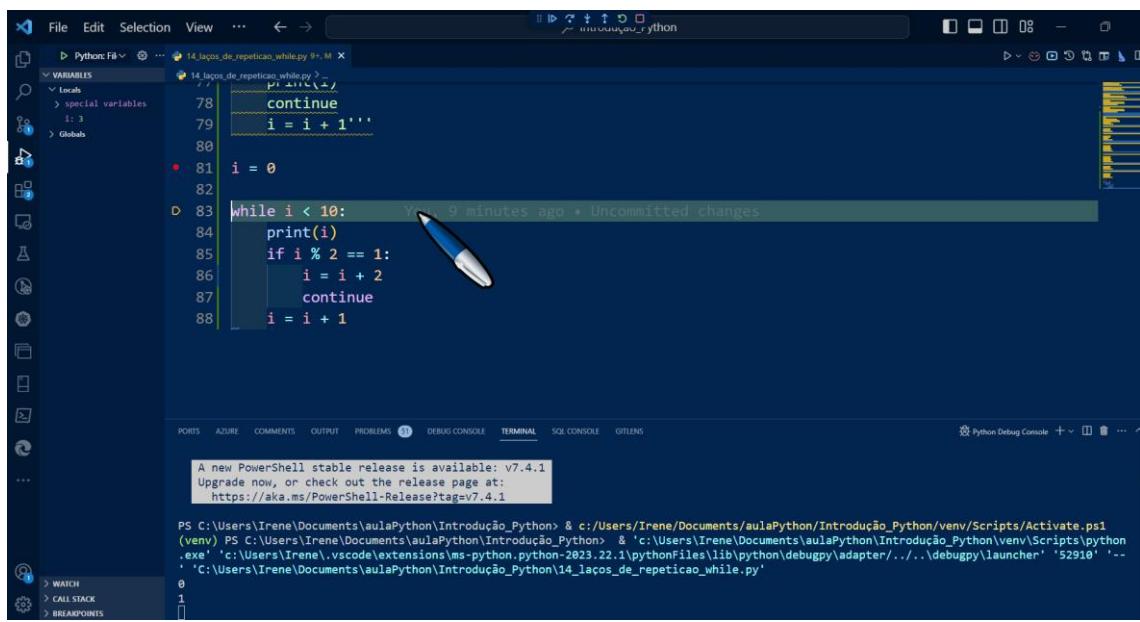
A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python script named '14_laços_de_repetição_while.py'. A blue pen icon is placed over the line of code where the variable 'i' is assigned the value 3. The code uses a while loop to print even numbers from 0 to 10. The terminal below shows the command to run the script and a message about a PowerShell update.

```
14_laços_de_repetição_while.py 9+.M
77     print(i)
78     continue
79     i = i + 1''''
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1

You, 8 minutes ago • Uncommitted changes
```

```
A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython\Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/..../debugpy/launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython\Introdução_Python\14_laços_de_repetição_while.py'
```



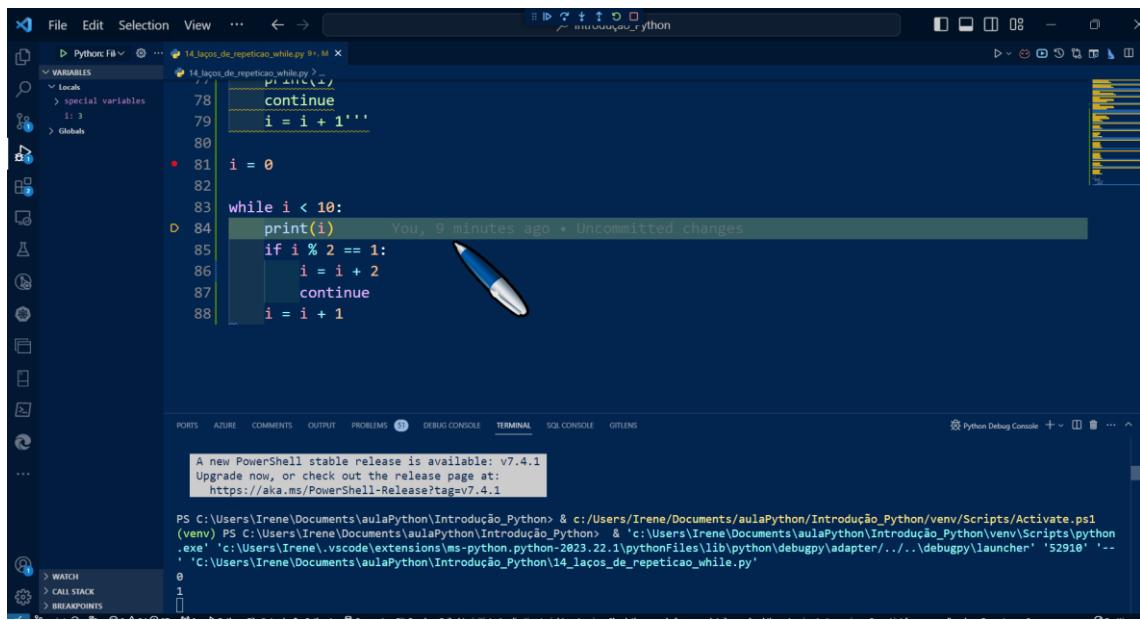
A second screenshot of the VS Code interface, showing the same Python script. This time, the break point is set at line 83, where the condition 'i < 10' is evaluated. The blue pen icon is placed over this line. The terminal output is identical to the first screenshot.

```
14_laços_de_repetição_while.py 9+.M
77     print(i)
78     continue
79     i = i + 1''''
80
81 i = 0
82
83 while i < 10: You, 9 minutes ago • Uncommitted changes
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1

You, 9 minutes ago • Uncommitted changes
```

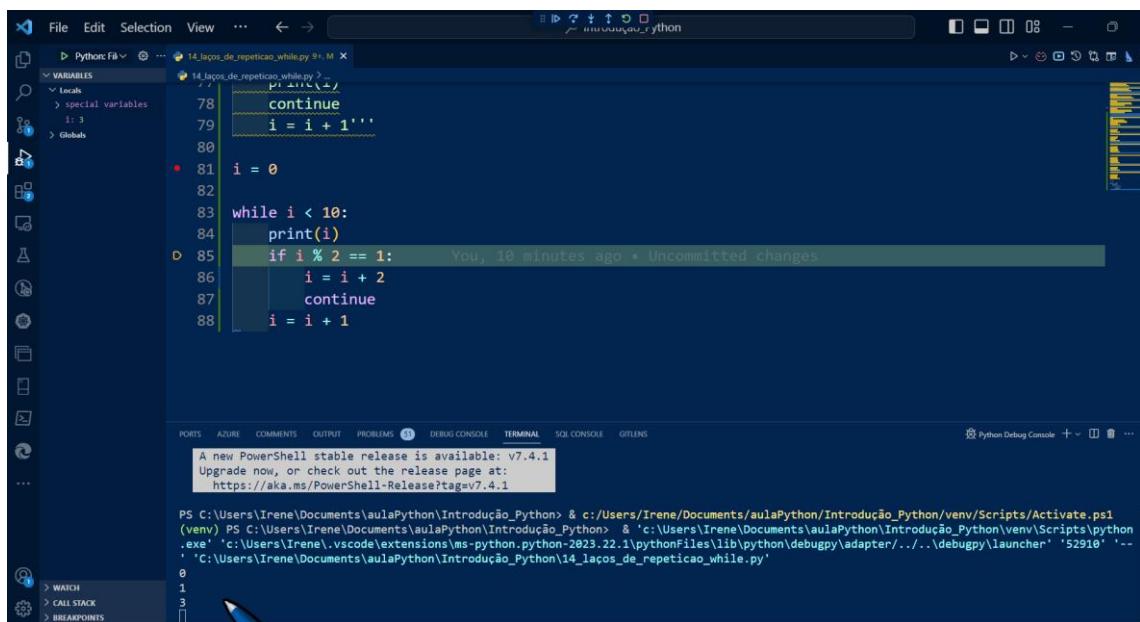
```
A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython\Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/..../debugpy/launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython\Introdução_Python\14_laços_de_repetição_while.py'
```



Screenshot of the Visual Studio Code (VS Code) interface showing a Python file named `14_laços_de_repetição_while.py`. The code contains a while loop that prints even numbers from 0 to 10. A breakpoint is set at line 81, where the variable `i` is initialized to 0. The status bar at the bottom indicates "You, 9 minutes ago * Uncommitted changes".

```
14_laços_de_repetição_while.py
77
    print(i)
    continue
78
    i = i + 1''''
80
• 81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1
```

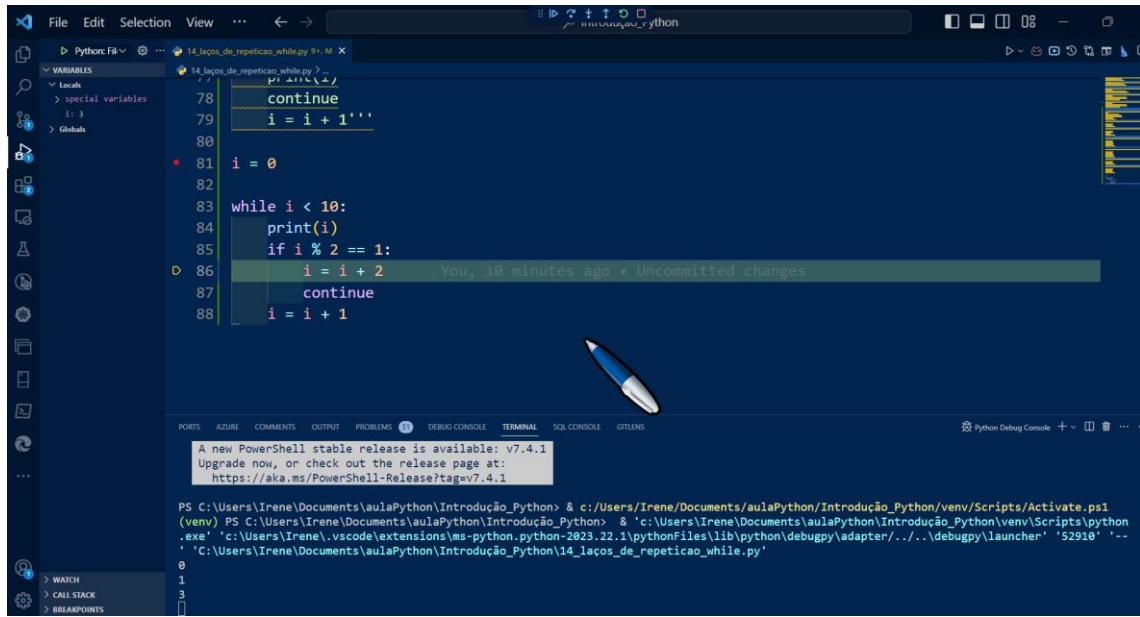


Screenshot of the Visual Studio Code (VS Code) interface showing the execution of the Python script. The terminal window displays the output of the script, which prints even numbers from 0 to 10. The status bar at the bottom indicates "You, 10 minutes ago * Uncommitted changes".

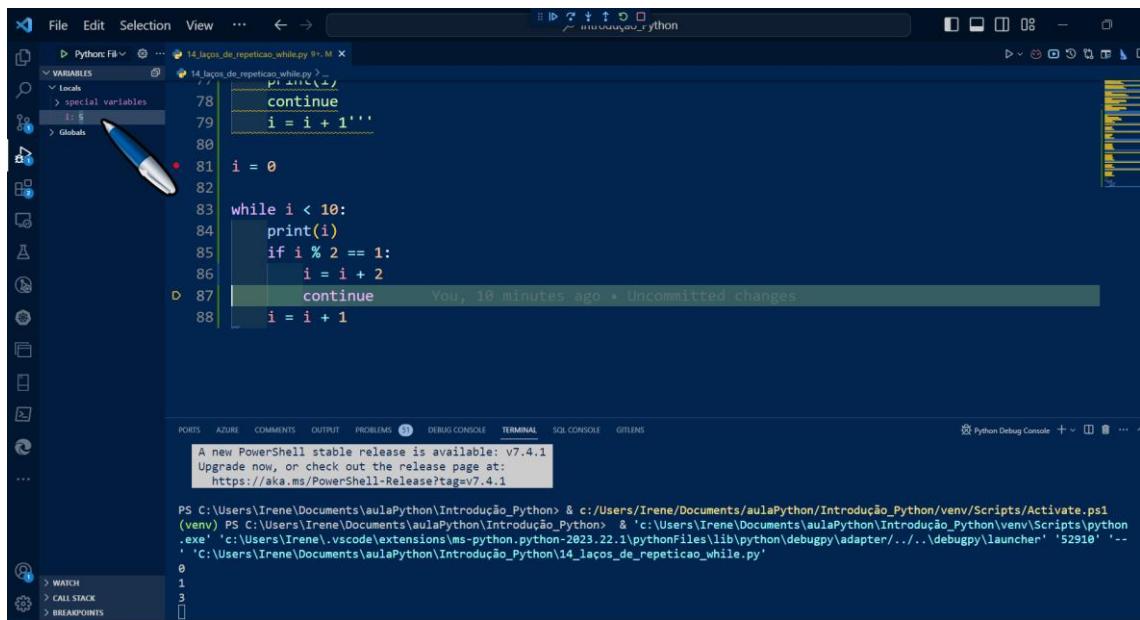
```
A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.4.1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '52910' '--'
'C:\Users\Irene\Documents\aulaPython\Introdução_Python\14_laços_de_repetição_while.py'

0
1
2
3
4
5
6
7
8
9
10
```



A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_laços_de_repetição_while.py'. A red dot on the left margin indicates a breakpoint at line 81. The code contains a while loop that prints odd numbers from 1 to 9. Line 86 contains a 'continue' statement. The status bar at the bottom shows a PowerShell prompt with the command 'PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1 (venv)'.



A second screenshot of the VS Code interface, identical to the first one except for the position of the breakpoint. It is now placed at line 87, just before the 'continue' statement. The rest of the code and the terminal output remain the same.

```
14_lacos_de_repeticao_while.py
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1
```

You, 10 minutes ago * Uncommitted changes

A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
<https://aka.ms/PowerShell-Release?tag=v7.4.1>

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher' '52910' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'

0
1
3

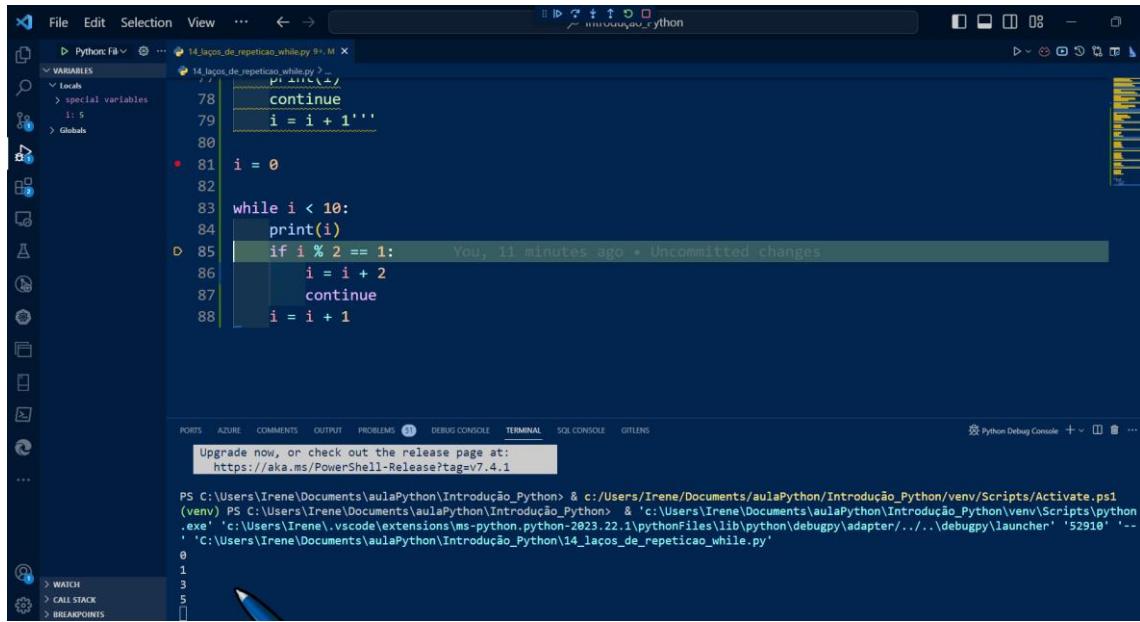
```
14_lacos_de_repeticao_while.py
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1
```

You, 11 minutes ago * Uncommitted changes

A new PowerShell stable release is available: v7.4.1
Upgrade now, or check out the release page at:
<https://aka.ms/PowerShell-Release?tag=v7.4.1>

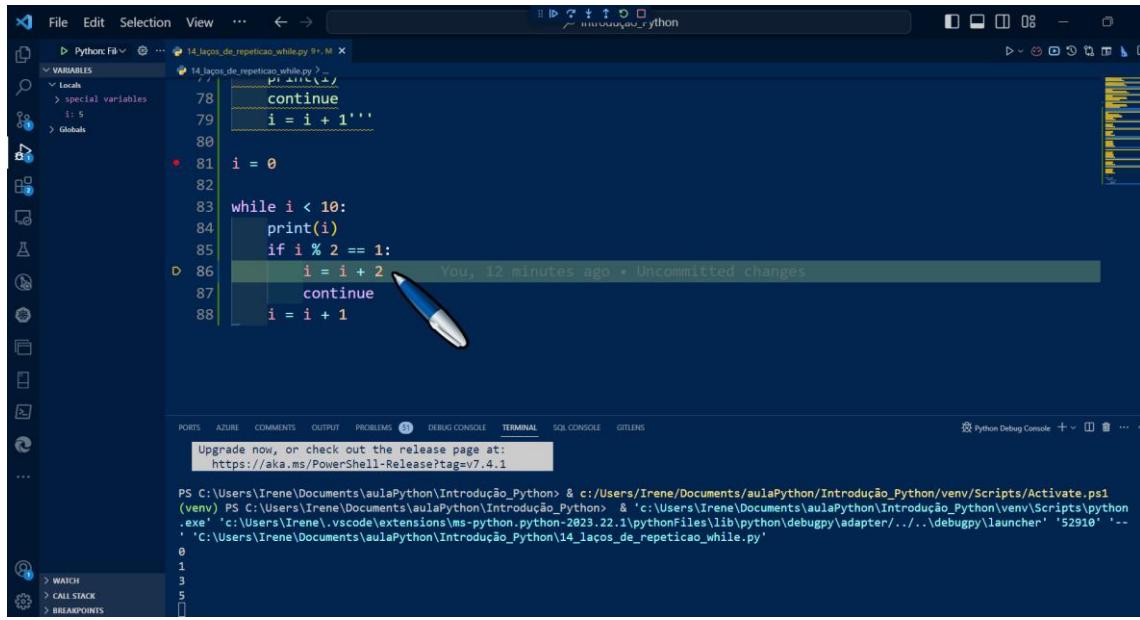
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher' '52910' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_lacos_de_repeticao_while.py'

0
1
3



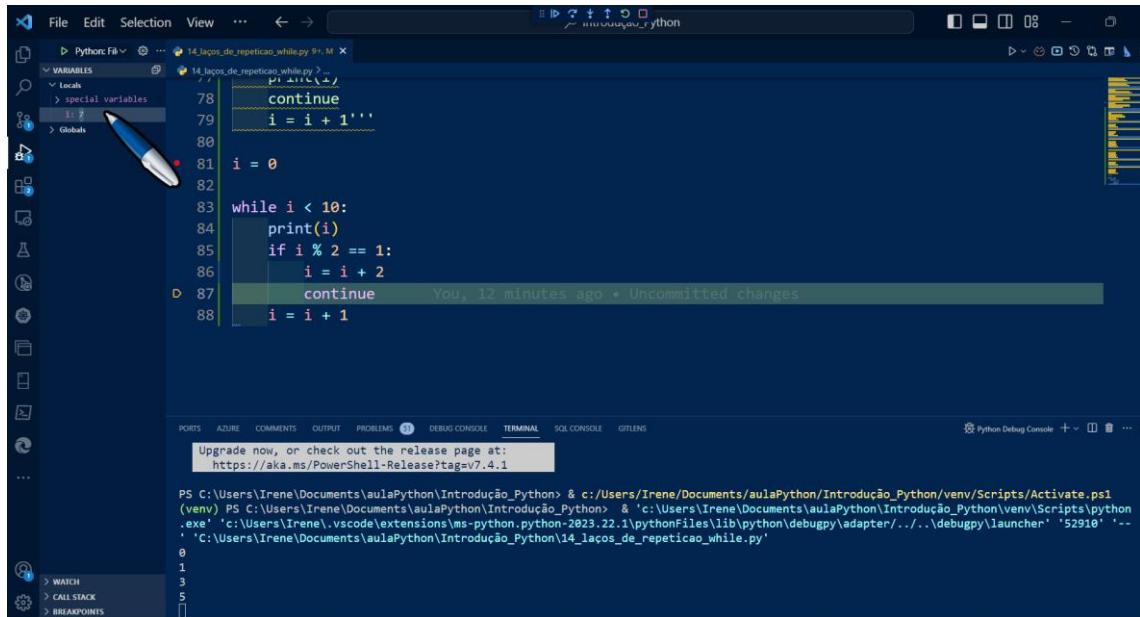
A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_laços_de_repetição_while.py'. A red dot marks a breakpoint at line 86. The code contains a while loop that prints odd numbers from 1 to 9. The terminal below shows the command to run the script and its output.

```
PS C:\Users\Irene\Documents\aulaPython> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/..\..\debugpy\launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
1
3
5
```



A second screenshot of the VS Code interface, showing the same Python script. A blue pen icon is placed over the code area between lines 85 and 86. The terminal output is identical to the first screenshot.

```
PS C:\Users\Irene\Documents\aulaPython> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/..\..\debugpy\launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
1
3
5
```

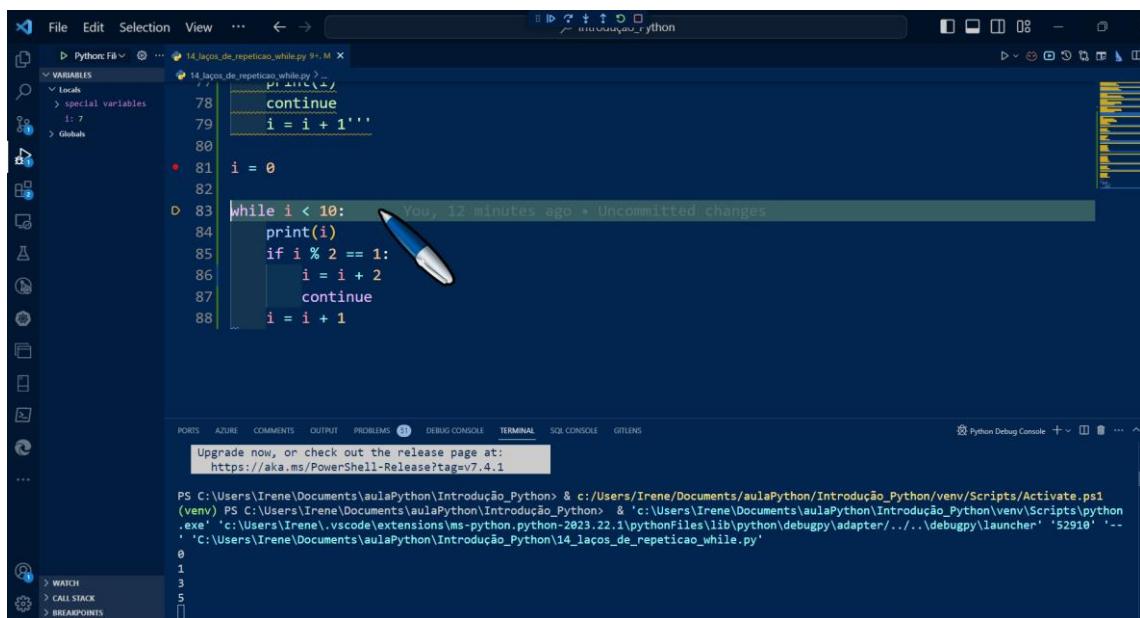


A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_laços_de_repetição_while.py'. A blue pen icon is placed over the code at line 78, which contains the line 'i = i + 1'. The code itself is as follows:

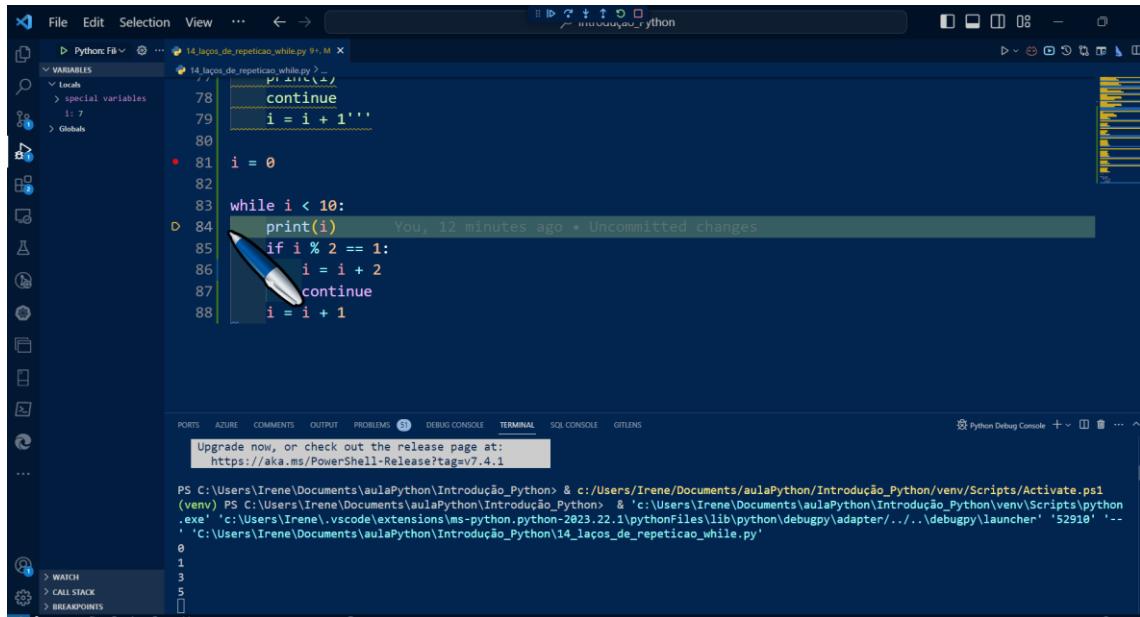
```
i = 7
print(i)
continue
i = i + 1

while i < 10:
    print(i)
    if i % 2 == 1:
        i = i + 2
    continue
    i = i + 1
```

The status bar at the bottom indicates 'You, 12 minutes ago * Uncommitted changes'.

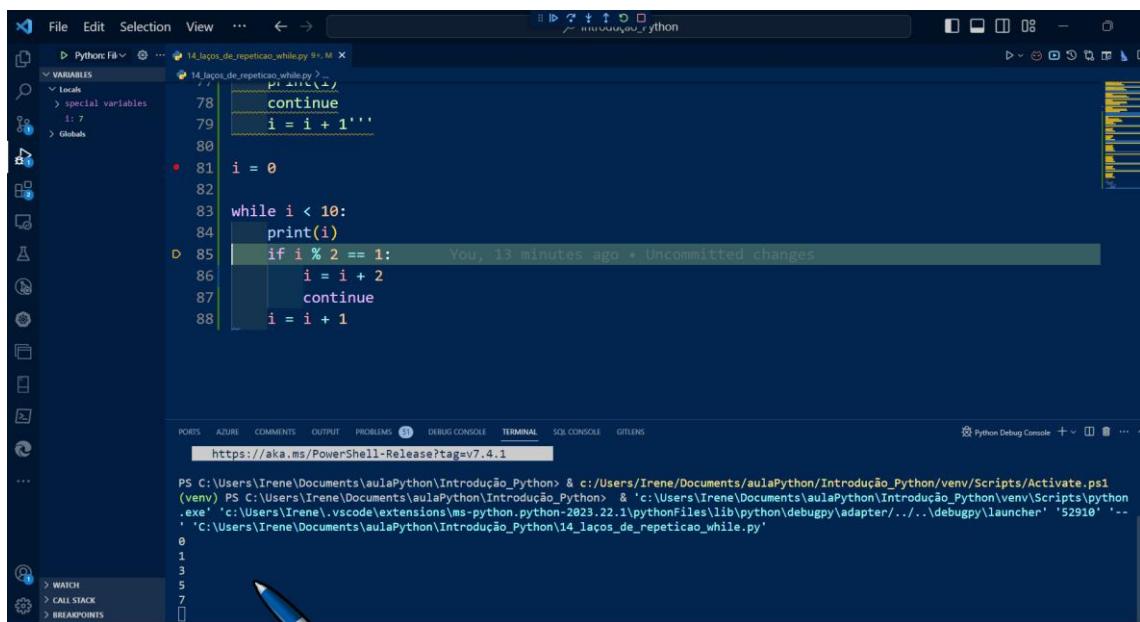


A second screenshot of the VS Code interface, showing the same Python file '14_laços_de_repetição_while.py'. The blue pen icon is now placed over the code at line 83, which contains the line 'while i < 10:'. The code remains the same as in the first screenshot.



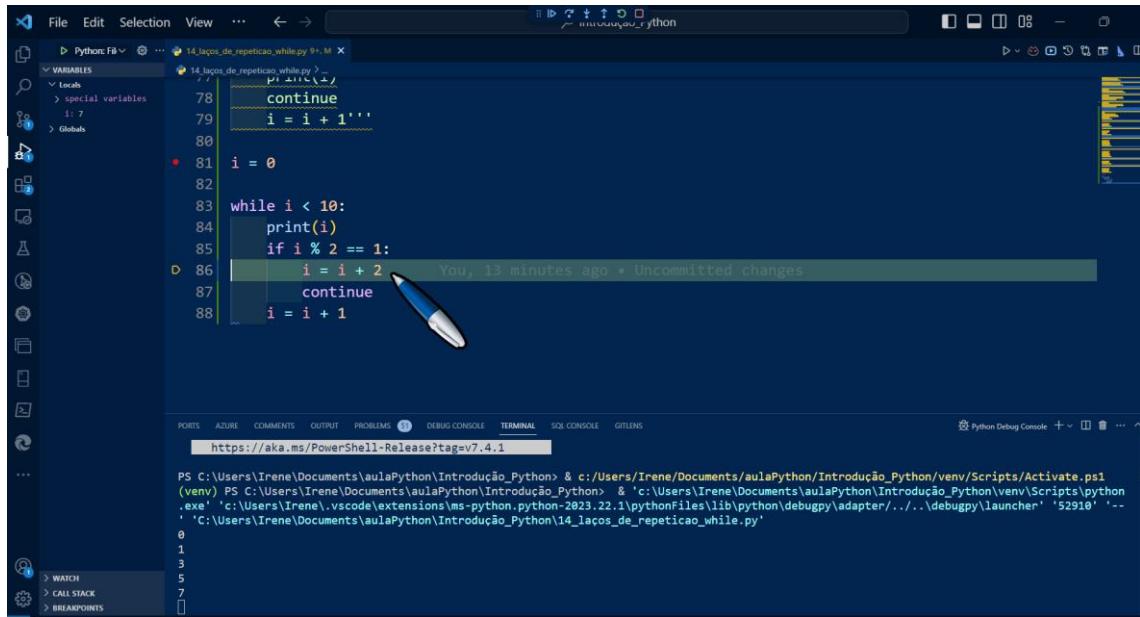
Screenshot of VS Code showing a Python script named `14_laços_de_repetição_while.py`. The code contains a break point at line 81. The terminal shows the command used to run the script.

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python\debugpy\adapter/../debugpy\launcher' '52910' '--'  
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
```



Screenshot of VS Code showing the same Python script. The break point has moved to line 85. The terminal shows the command used to run the script.

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python\debugpy\adapter/../debugpy\launcher' '52910' '--'  
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
```



```
14_laços_de_repetição_while.py
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1
```

You, 13 minutes ago * Uncommitted changes

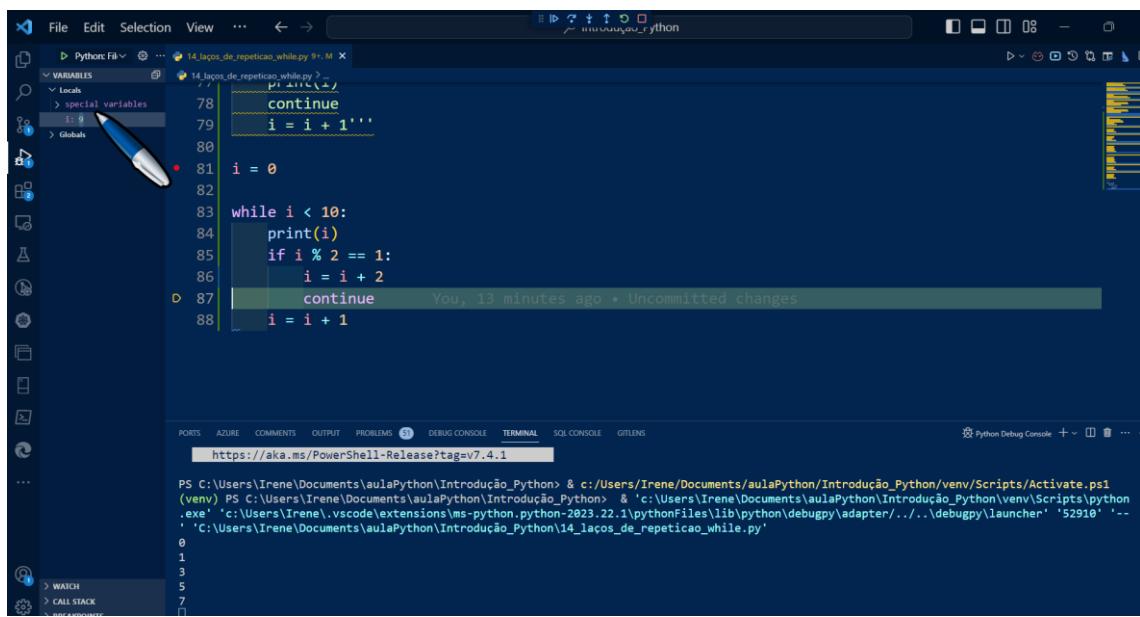
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

https://aka.ms/PowerShell-Release?tag=v7.4.1

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python\debugpy\adapter/../debugpy\launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython\Introdução_Python\14_laços_de_repetição_while.py'
```

0
1
3
5
7

WATCH CALL STACK BREAKPOINTS



```
14_laços_de_repetição_while.py
78     print(i)
79     continue
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87         continue
88     i = i + 1
```

You, 13 minutes ago * Uncommitted changes

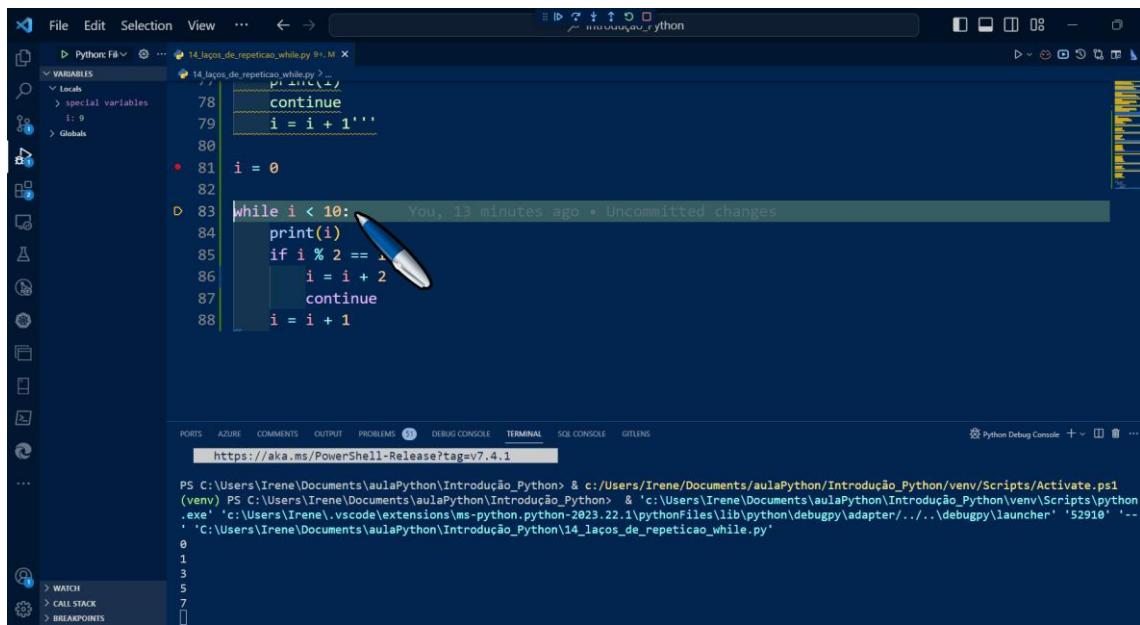
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

https://aka.ms/PowerShell-Release?tag=v7.4.1

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python\debugpy\adapter/../debugpy\launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython\Introdução_Python\14_laços_de_repetição_while.py'
```

0
1
3
5
7

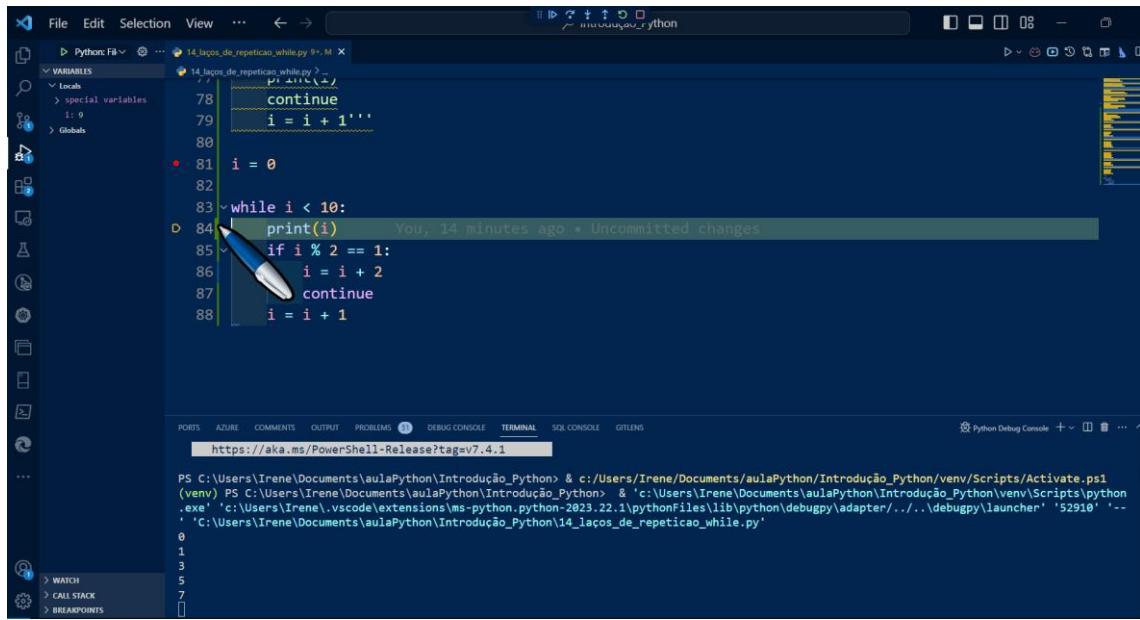
WATCH CALL STACK BREAKPOINTS



A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_laços_de_repetição_while.py'. A blue pen icon is placed over the code at line 81, which contains the assignment 'i = i + 1'. The code is as follows:

```
i = 0
while i < 10:
    print(i)
    if i % 2 == 1:
        i = i + 2
    continue
    i = i + 1
```

The status bar at the bottom indicates 'You, 13 minutes ago * Uncommitted changes'.



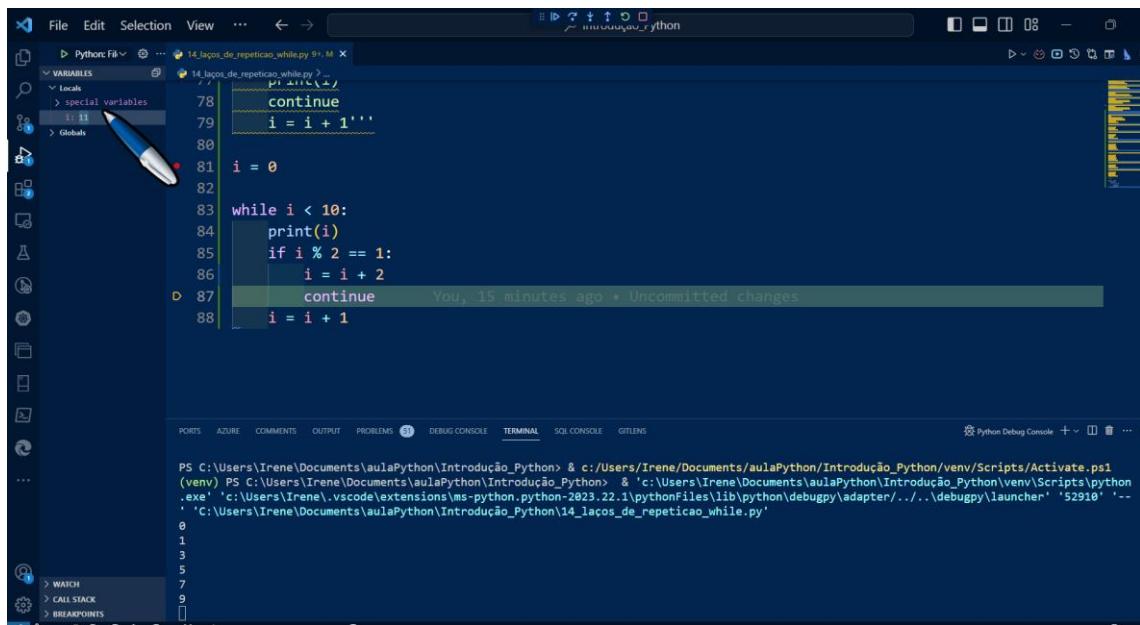
A second screenshot of the VS Code interface, showing the same Python file '14_laços_de_repetição_while.py'. The blue pen icon is now placed over the line 'print(i)' at line 84. The code remains the same as in the first screenshot.

```
File Edit Selection View ... 14_laços_de_repetição_while.py 9+ M
VARIABLES
locals
> special variables
i: 9
> Globals
B
D
E
F
G
H
I
J
K
L
M
N
O
P
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
Python Debug Console + ...
```

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> &c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> &'c:/Users/Irene/Documents\aulaPython\Introdução_Python\venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
1
3
5
7
9
```

```
File Edit Selection View ... 14_laços_de_repetição_while.py 9+ M
VARIABLES
locals
> special variables
i: 9
> Globals
B
D
E
F
G
H
I
J
K
L
M
N
O
P
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
Python Debug Console + ...
```

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> &c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> &'c:/Users/Irene/Documents\aulaPython\Introdução_Python\venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher' '52910' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/14_laços_de_repetição_while.py'
0
1
3
5
7
9
```



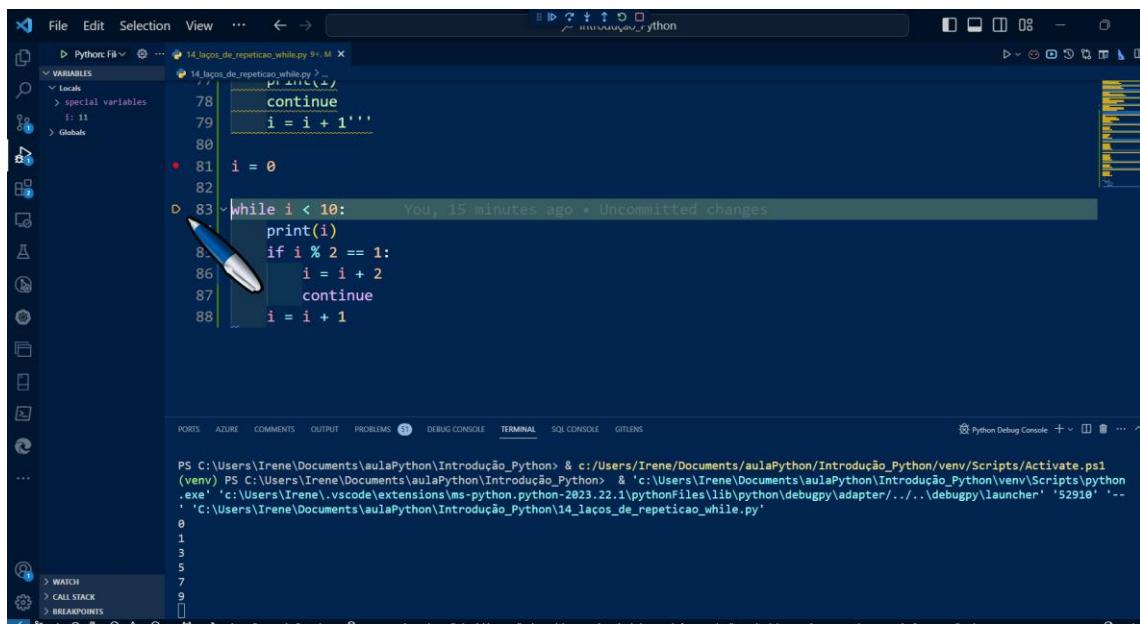
A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '14_lacos_de_repeticao_while.py'. A red dot indicates a breakpoint is set on line 81. The code uses a while loop to print even numbers from 0 to 10. The 'TERMINAL' tab shows the output of running the script, which prints the numbers 0, 1, 3, 5, 7, and 9. The 'WATCH' and 'CALL STACK' sections are visible in the sidebar.

```
14_lacos_de_repeticao_while.py
77 print(i)
78     continue
79     i = i + 1'''"
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1
```

You, 15 minutes ago * Uncommitted changes

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GIT LENS

Python Debug Console



A second screenshot of the VS Code interface, showing the same Python script. This time, the breakpoint is set on line 83. The code remains the same, printing even numbers from 0 to 10. The terminal output is identical to the first screenshot.

```
14_lacos_de_repeticao_while.py
77 print(i)
78     continue
79     i = i + 1'''"
80
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1
```

You, 15 minutes ago * Uncommitted changes

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GIT LENS

Python Debug Console

```
File Edit Selection View ... ← → 🔍 Introdução_Python
Python File ... 14_laços_de_repetição_while.py 91 M
VARIABLES
78     print(i)
79     continue
80     i = i + ...
81 i = 0
82
83 while i < 10:
84     print(i)
85     if i % 2 == 1:
86         i = i + 2
87     continue
88     i = i + 1

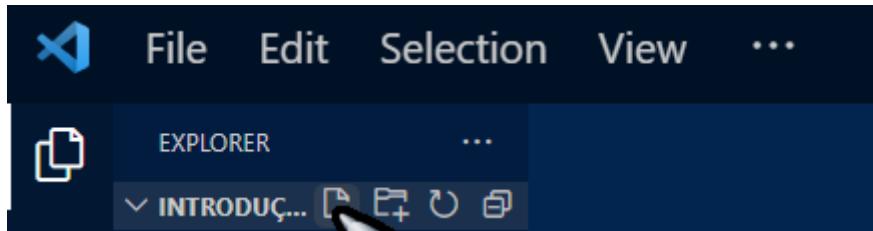
PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GIT LENS
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1\pythonfiles\lib\python\debugpy\adapter/../debugpy\launcher' '52918' '--'
'C:\Users\Irene\Documents\aulaPython\Introdução_Python\14_laços_de_repetição_while.py'
0
1
3
5
7
9

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

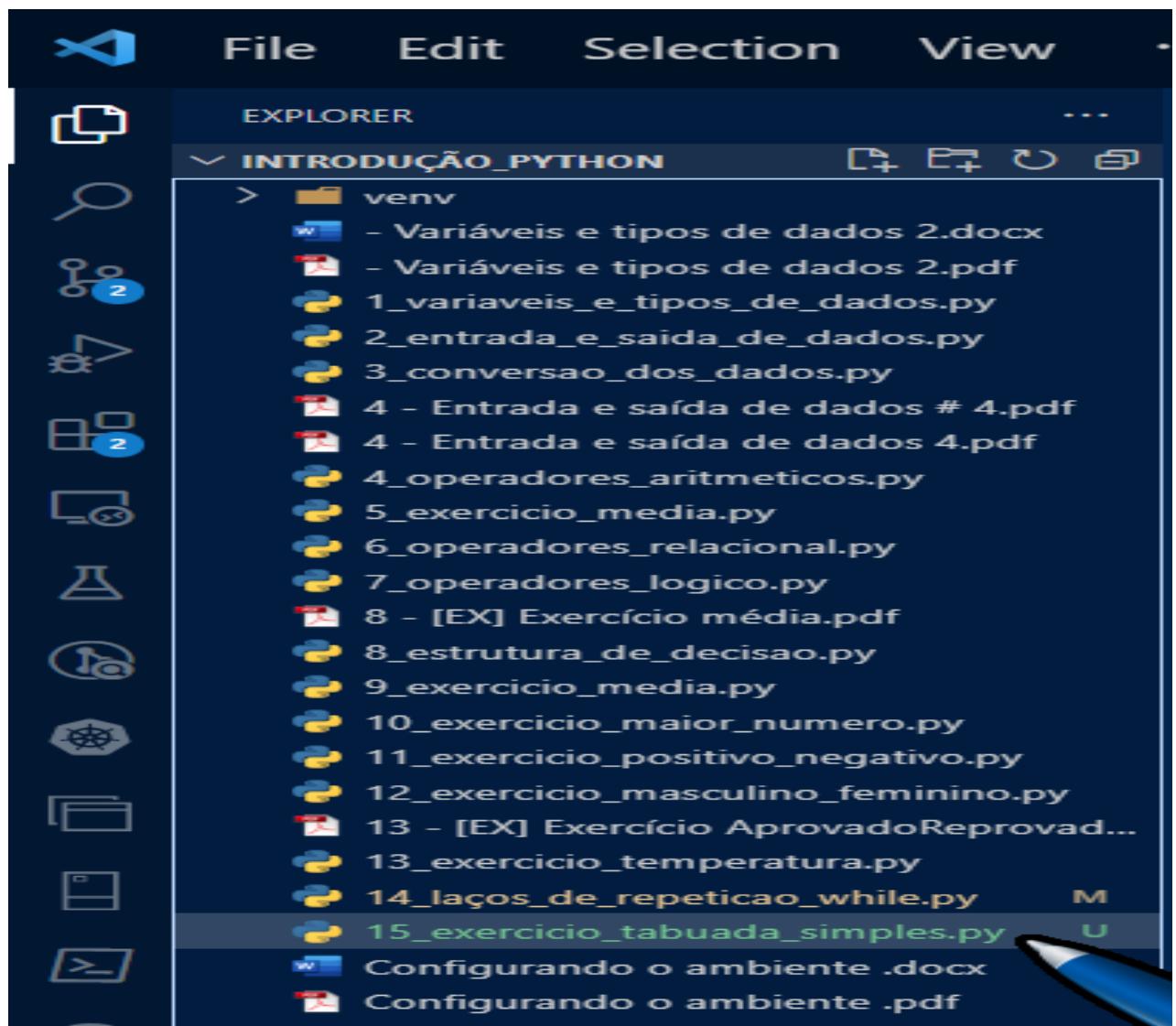
Quando queremos voltar para laço usamos **continue** e quando queremos interromper usamos **break**

21 - [EX] Exercício tabuada.mp

Criar um novo arquivo



Coloque nome do arquivo



The screenshot shows the VS Code interface with the code editor open. The file being edited is '15_exercicio_tabuada_simples.py'. The code contains a single line of Python code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
```

The status bar at the bottom shows the path '(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>' and various status indicators like 'DEBUG CONSOLE', 'TERMINAL', 'SQL CONSOLE', and 'GIT LENS'.

A screenshot of the Visual Studio Code (VS Code) interface. The title bar says "introdução_Python". The Explorer sidebar shows a folder named "INTRODUÇÃO PYTHON" containing various Python files. The main editor area displays a Python script named "15_exercicio_tabuada_simples.py". The code is as follows:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

Vamos debugar o código conforme ensinado acima



A screenshot of the Visual Studio Code interface, similar to the first one but with a red dot placed on the line before the "while" loop, specifically on the line "5 i = 1". This indicates that the developer has selected this line for debugging.

Seleciona a linha

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

Digita o numero 5

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher'
53238' '--' 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The variable 'n1' is highlighted in yellow, and the line 'i = 1' is highlighted in green. A blue pen icon is placed over the code at the start of the 'while' loop. The left sidebar shows a 'VARIABLES' section with 'locals' expanded, showing 'n1: 5'. The bottom terminal window shows the command line and the output of running the script.

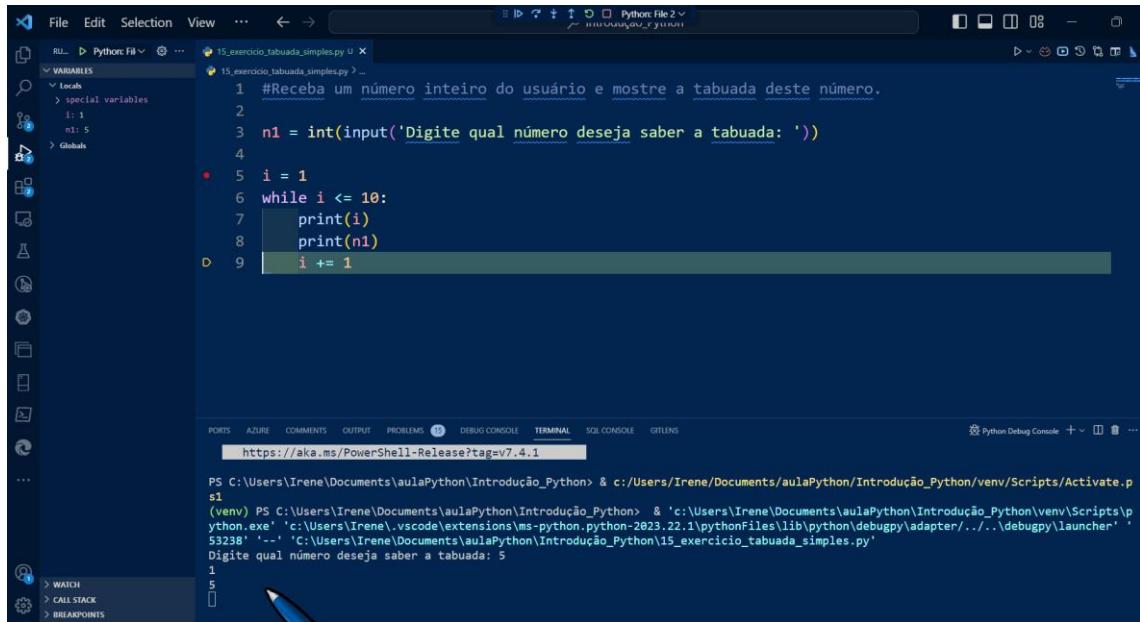
A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simples.py'. The code is identical to the first screenshot. However, a red dot indicates a break point is set on the line 'i = 1'. The blue pen icon is now placed over the line 'while i <= 10:'. The left sidebar and bottom terminal window are also visible.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The screenshot shows the VS Code interface with a Python file named "15_exercicio_tabuada_simples.py" open. A red dot indicates a breakpoint at line 5. The code prints the multiplication table of a user-specified number. The terminal below shows the command to run the script and the output of the program asking for input.

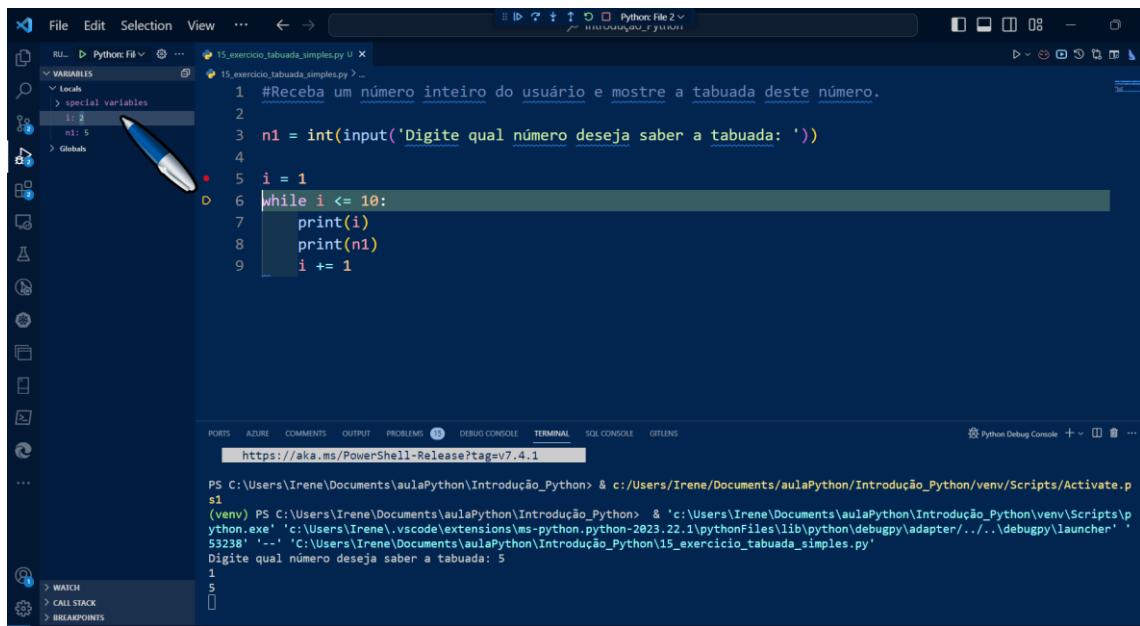
```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

This screenshot is identical to the one above, showing the same Python script with a breakpoint at line 8. The code prints the multiplication table of a user-specified number. The terminal below shows the command to run the script and the output of the program asking for input.



```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The screenshot shows the VS Code interface with a Python file named '15_exercicio_tabuada_simples.py'. The code is a simple multiplication table generator. A red dot at line 5 indicates a breakpoint. The variable 'i' is selected in the code editor. The terminal below shows the command to run the script and its output.



```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

This screenshot is identical to the one above, but the variable 'i' is now highlighted in the code editor, suggesting it is the current focus of the developer's attention.

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 * i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is highlighted in green, indicating it is being run or has been run. A blue pen icon is overlaid on the code editor. The left sidebar shows a 'WATCH' section with variables: 'i: 2' and 'n1: 5'. The bottom right corner features a 'Python Debug Console' tab.

A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simples.py'. The code remains the same, but the output from the terminal is visible at the bottom:

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/..../debugpy/launcher' 53238 '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
```

The blue pen icon is again overlaid on the code editor. The left sidebar shows a 'WATCH' section with variables: 'i: 2' and 'n1: 5'. The bottom right corner features a 'Python Debug Console' tab.

The screenshot shows the Visual Studio Code interface with a Python file named `15_exercicio_tabuada_simples.py` open. The code is a simple program that asks the user for a number and then prints its multiplication table up to 10. A blue pen icon is overlaid on the editor area. The terminal below shows the script being run in a Python environment, with the output of the first five numbers (1, 5, 2, 5) displayed.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' '--debugpy_launcher' 53238' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
```

This screenshot shows the same Python script in VS Code, but the line `while i <= 10:` is highlighted with a green background. The rest of the code and the terminal output are identical to the first screenshot.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter' '--debugpy_launcher' 53238' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
```

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is highlighted in green, indicating it is being run or has been run. The terminal below shows the command line and the output of the program:

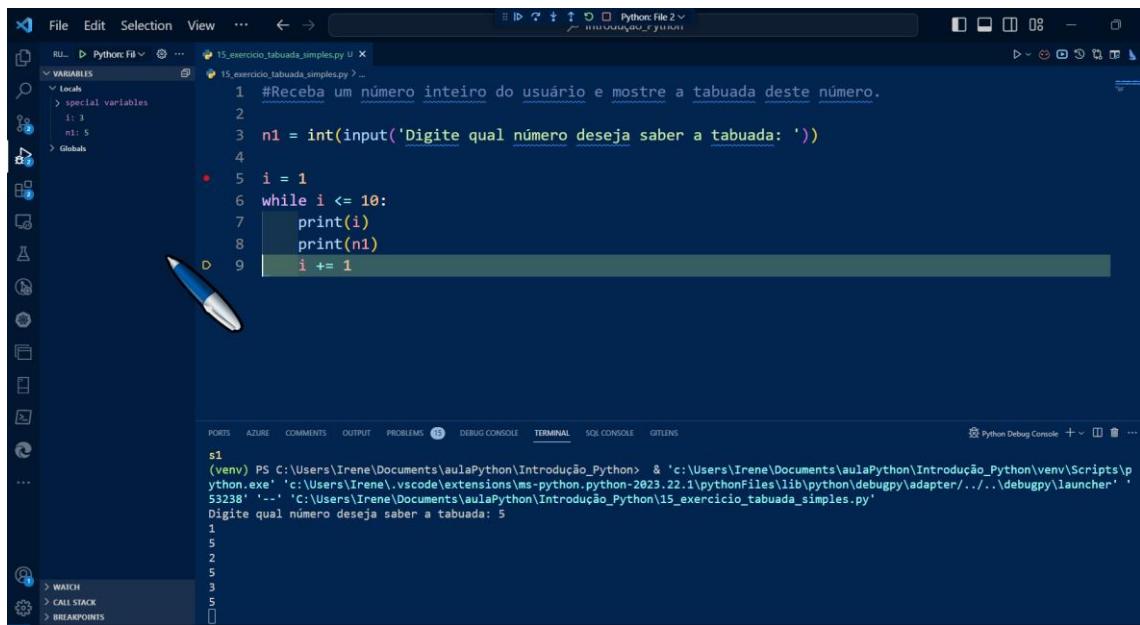
```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.p
s1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\p
ython.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher'
53238' '-l' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
3
```

A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simples.py'. The code is identical to the first screenshot:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is highlighted in green. The terminal shows the command line and the output of the program:

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.p
s1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\p
ython.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher'
53238' '-l' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
3
```



File Edit Selection View ... ⟲ ⟳ 🔍 ⌂ Python File 2 🎨 15_exercicio_tabuada_simples.py

VARIABLES

locals

> special variables

i: 3
n1: 5

> Globals

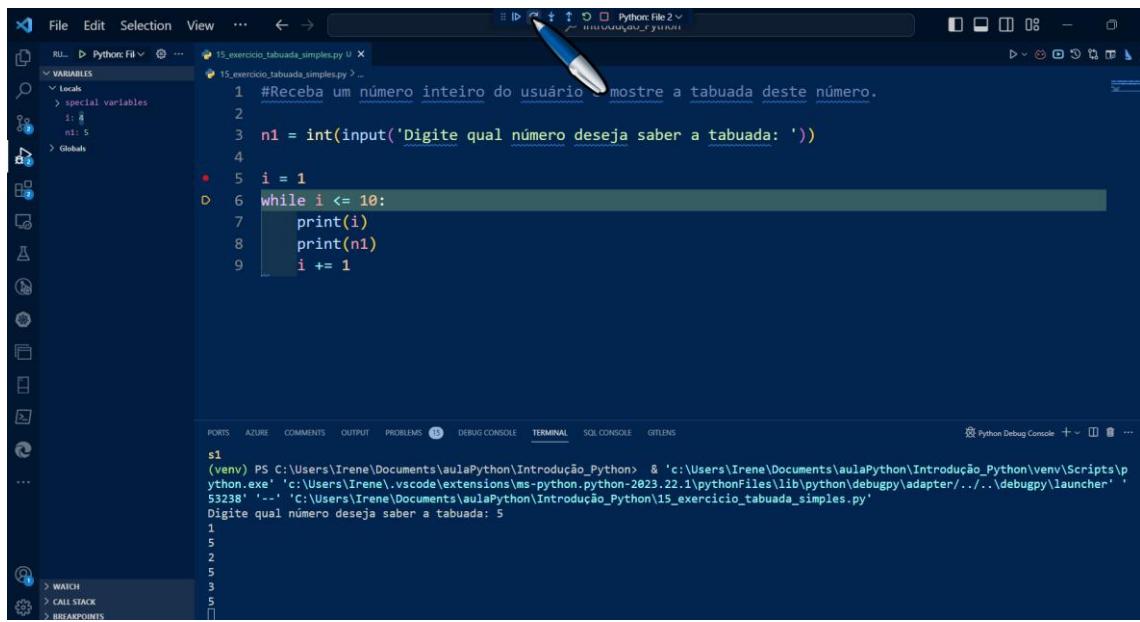
• 5 i = 1
6 while i <= 10:
7 print(i)
8 print(n1)
9 i += 1

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

Python Debug Console + ⚡

```
s1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' 53238 '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
3
5
```

WATCH CALL STACK BREAKPOINTS



File Edit Selection View ... ⟲ ⟳ 🔍 ⌂ Python File 2 🎨 15_exercicio_tabuada_simples.py

VARIABLES

locals

> special variables

i: 8
n1: 5

> Globals

• 5 i = 1
6 while i <= 10:
7 print(i)
8 print(n1)
9 i += 1

PORTS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

Python Debug Console + ⚡

```
s1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' 53238 '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
3
5
```

WATCH CALL STACK BREAKPOINTS

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is highlighted in green, indicating it is being run or has been run. The terminal below shows the output of the script:

```
s1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' 53238 '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
3
5
4
5
5
5
5
```

A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simples.py'. The code is identical to the first screenshot:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is highlighted in green. The terminal shows a different output:

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' 53238 '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
3
5
4
5
5
5
5
```

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The variable 'n1' is highlighted in blue. The 'WATCH' section of the sidebar shows 'n1: 5'. The terminal below shows the output of the program:

```
python.exe 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' 53238 '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual numero deseja saber a tabuada: 5
1
5
2
5
3
5
4
5
5
5
```

A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simples.py'. The variable 'n1' is now highlighted in blue. The 'WATCH' section of the sidebar shows 'n1: 5'. The terminal output is identical to the first screenshot.

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is highlighted in green, indicating it is valid. A blue pen icon is positioned over the first line of code. The bottom right corner of the code editor has a small icon of a pencil.

The bottom half of the screen shows the terminal window with the output of running the script:

```
python.exe 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' 53238 '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
3
5
4
5
5
5
```

A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simples.py'. The code is identical to the one in the first screenshot:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The line 'print(n1)' is highlighted in green. A blue pen icon is positioned over the first line of code. The bottom right corner of the code editor has a small icon of a pencil.

The terminal window at the bottom shows the same output as the first screenshot:

```
python.exe 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' 53238 '--' 'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
1
5
2
5
3
5
4
5
5
5
```

A screenshot of the Visual Studio Code (VS Code) interface. The top bar shows the file path '15_exercício_tabuada_simples.py' and the title 'Python: File 2'. The left sidebar has a 'VARIABLES' section with 'locals' expanded, showing 'i: 5' and 'n1: 5'. The main editor area contains the following Python code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The line 'i += 1' is highlighted with a green background. Below the editor is a terminal window showing the output of the program: 'Digite qual número deseja saber a tabuada: 5' followed by the multiplication table for 5.

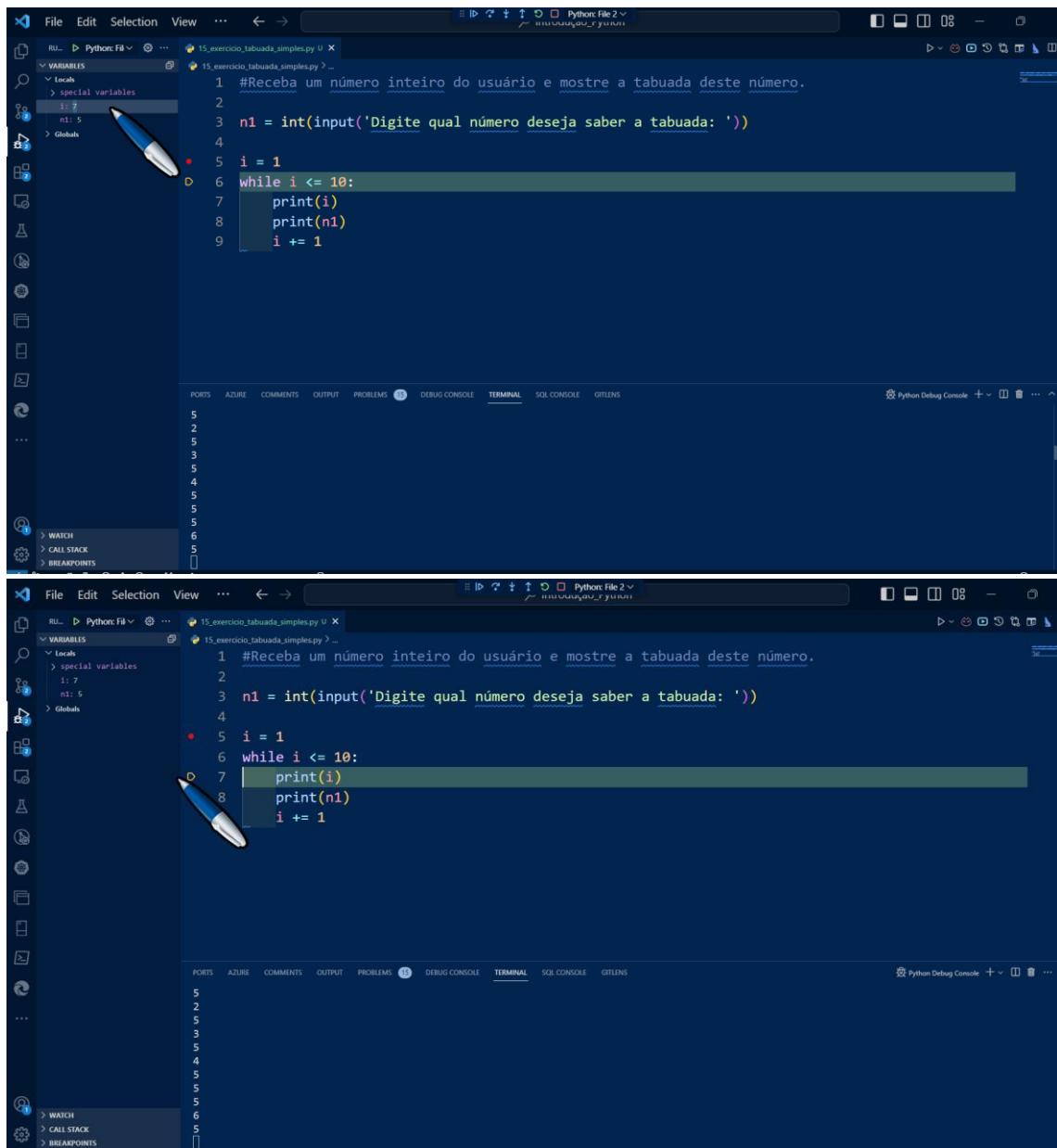
A second screenshot of the VS Code interface, showing the same environment as the first. The terminal window now displays the output for the number 6, showing the multiplication table for 6.

The image consists of three vertically stacked screenshots of the Visual Studio Code (VS Code) interface, showing a Python script named `15_exercicio_tabuada_simples.py`. The code is a simple program that asks the user for a number and then prints its multiplication table from 1 to 10.

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1
```

The first screenshot shows the code with a breakpoint at the start of the loop (`i = 1`). The second screenshot shows the code with a breakpoint at the first iteration of the loop (`print(i)`). The third screenshot shows the code with a breakpoint at the end of the loop (`i += 1`).

The VS Code interface includes a sidebar with icons for file operations, a variables panel showing local variables (`i: 6`, `n1: 5`), and a terminal window below the code editor. The terminal shows the output of the program when run with the input `5`.



```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The screenshot displays two identical instances of the Visual Studio Code (VS Code) interface, both showing the same Python script named `15_exercicio_tabuada_simples.py`. The script is designed to print the multiplication table of a number entered by the user. The code includes a comment at the top, variable declarations for `n1` and `i`, a `while` loop that iterates from 1 to 10, and `print` statements for both the multiplier (`i`) and the product (`n1`). A blue pen icon is placed over the code in both instances. In the top instance, a red dot marks a breakpoint at line 5, where the assignment `i = 1` is located. In the bottom instance, a red dot marks a breakpoint at line 7, where the first `print(i)` statement is located. The status bar at the bottom of each instance shows the terminal tab is active.

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named `15_exercicio_tabuada_simple.py`. The code is as follows:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is highlighted in green, indicating it is valid Python. Below the editor, the terminal window shows the output of the script:

```
2
5
3
5
4
5
5
5
6
5
5
7
```

The status bar at the bottom indicates "Python: File (introdução_Python)".

A screenshot of the Visual Studio Code (VS Code) interface, similar to the first one but with a debugger interface overlaid. The main area shows the same Python file `15_exercicio_tabuada_simple.py` with the same code. The code is highlighted in green.

The left sidebar features a "VARIABLES" tree view with the following contents:

- Locals
 - i: 7
 - n1: 5
- Globals
 - i = 1
 - while i <= 10:
 - print(i)
 - print(n1)
 - i += 1

The status bar at the bottom indicates "Python: File (introdução_Python)".

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simple.py' with the following code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is annotated with a blue pen highlighting the line 'print(i)' in the loop body. The left sidebar shows the 'VARIABLES' section with locals: i: 5 and n1: 5. The bottom status bar shows the output of the program: 5, 3, 5, 4, 5, 5, 5, 6.

A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simple.py' with the following code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is annotated with a blue pen highlighting the line 'print(n1)' in the loop body. The left sidebar shows the 'VARIABLES' section with locals: i: 8 and n1: 5. The bottom status bar shows the output of the program: 5, 3, 5, 4, 5, 5, 5, 6.

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simple.py'. The code is as follows:

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

The variable 'n1' has a value of 5 assigned in the 'locals' section of the Variables sidebar. The terminal below shows the output of the program:

```
5  
5  
5  
5  
5  
5  
5  
5  
5  
5
```

A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simple.py'. The code is identical to the first screenshot:

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

The variable 'n1' still has a value of 5 assigned in the 'locals' section of the Variables sidebar. The terminal below shows the output of the program:

```
5  
4  
5  
5  
5  
6  
5  
5  
5  
5
```

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

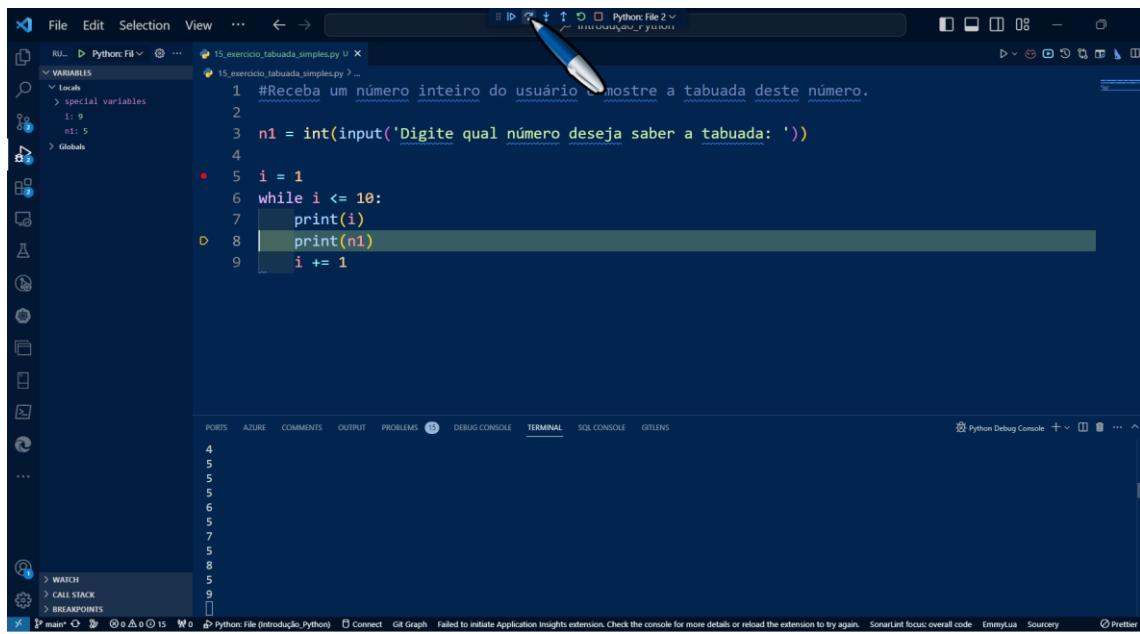
```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is highlighted in green, indicating it is valid Python. Below the code, the terminal window displays the output of the program, which is the multiplication table for the number 5, printed 10 times.

A screenshot of the Visual Studio Code (VS Code) interface, similar to the first one but with a different view. The left sidebar shows a 'VARIABLES' section with a tree view of local variables:

- locals
 - > special variables
 - i: 9
 - n1: 5
- > Globals

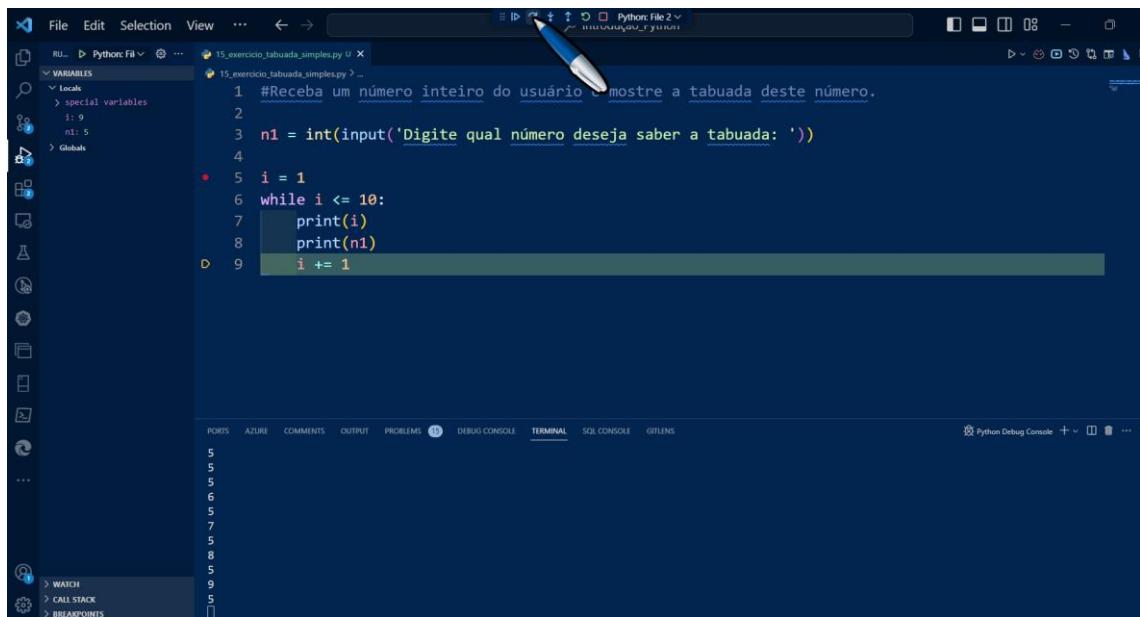
The main code editor window shows the same Python script as before. The terminal window below shows the same output: the multiplication table for 5, repeated 10 times.



A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1
```

The line 'print(n1)' is highlighted with a green background. A red dot at the start of line 5 indicates a breakpoint. The left sidebar shows a 'WATCH' section with variables: 'i: 9' and 'n1: 5'. The bottom status bar shows the path 'main*' and other system information.



A screenshot of the Visual Studio Code (VS Code) interface, showing the execution of the Python script. The terminal window displays the output of the program:

```
5
5
5
6
5
7
5
8
5
9
5
```

The code in the editor remains the same as in the previous screenshot, with the 'print(n1)' line highlighted. The terminal shows the multiplication table for the number 5, from 5x1 to 5x10. The left sidebar shows the 'WATCH' section with variables: 'i: 9' and 'n1: 5'. The bottom status bar shows the path 'main*' and other system information.

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simples.py' with the following code:

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

The variable 'n1' is highlighted in green, and the loop body is highlighted in light blue. The status bar at the bottom shows the output of the script: '5 5 5 5 5 5 5 5 5 5'. The left sidebar includes a 'WATCH' section showing 'i: 10' and 'n1: 5'. The bottom right corner shows a 'Python Debug Console' tab.

A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simples.py' with the following code:

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1
```

The variable 'n1' is highlighted in green, and the loop body is highlighted in light blue. The status bar at the bottom shows the output of the script: '5 5 5 5 5 5 5 5 5 5'. The left sidebar includes a 'WATCH' section showing 'i: 10' and 'n1: 5'. The bottom right corner shows a 'Python Debug Console' tab.

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named '15_exercicio_tabuada_simple.py'. The code is as follows:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The code is annotated with a blue pen, highlighting the line 'print(n1)'.

The left sidebar contains a 'VARIABLES' section showing local variables: 'i: 10' and 'n1: 5'. The bottom right corner of the editor has a small icon of a pen.

The bottom of the screen shows the terminal and other VS Code panels.

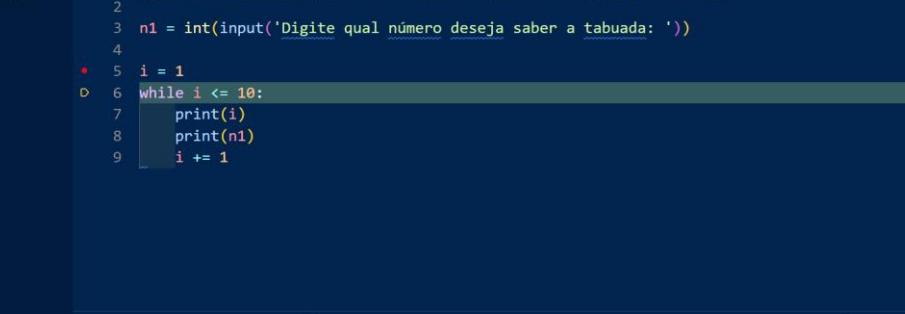
A second screenshot of the VS Code interface, showing the same Python file '15_exercicio_tabuada_simple.py'. The code has been modified:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

The line 'i += 1' is highlighted with a green selection bar. The code is annotated with a blue pen, highlighting the line 'print(i)'.

The left sidebar shows the same 'VARIABLES' section with 'i: 10' and 'n1: 5'. The bottom right corner of the editor has a small icon of a pen.

The bottom of the screen shows the terminal and other VS Code panels.



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The title bar displays "File Edit Selection View ... Python: File 2". The left sidebar includes icons for file operations, variables, locals, and breakpoints. A search bar at the top has the text "15_exercicio_tabuada_simple.py". The main editor area contains the following Python code:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```

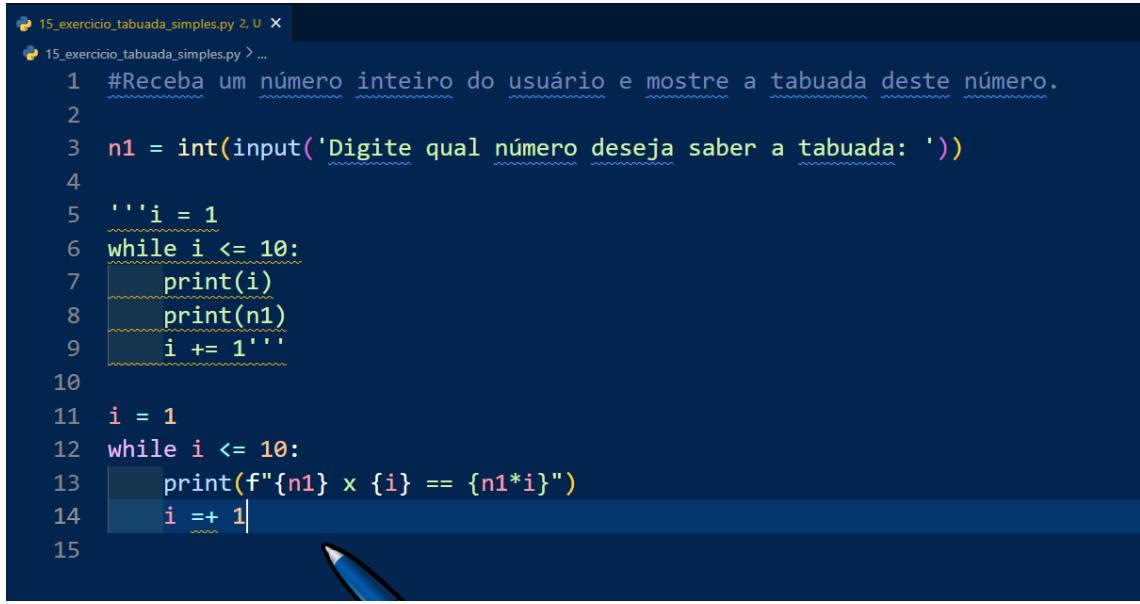
The line "6 while i <= 10:" is highlighted with a blue background. The status bar at the bottom shows tabs for PORTS, AZURE, COMMENTS, OUTPUT, PROBLEMS (with 15 notifications), DEBUG CONSOLE, TERMINAL, SQL CONSOLE, and GITLENS. Below the terminal tab, there is a command palette with the following items: WATCH, CALL STACK, and BREAKPOINTS.

The screenshot shows a Python code editor interface with the following details:

- File Bar:** File, Edit, Selection, View, ..., Python: File, 15_exercicio_tabuada_simples.py
- Variables Panel:** Shows a blue pen icon.
- Code Editor:** Displays the following Python script:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1
```
- Terminal:** Shows the output of the script:

```
5
5
5
5
5
5
5
5
5
5
10
5
```
- Bottom Navigation:** PORTS, AZURE, COMMENTS, OUTPUT, PROBLEMS, DEBUG CONSOLE, TERMINAL, SQL CONSOLE, GITLENS.
- Bottom Status:** Python Debug Console, PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>



```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
11 i = 1
12 while i <= 10:
13     print(f"{n1} x {i} == {n1*i}")
14     i += 1
```

Vamos debugar o código



```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
'''i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1'''  
i = 1  
while i <= 10:  
    print(f"{n1} x {i} == {n1*i}")  
    i += 1
```

Selecionar a linha

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
'''i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1'''  
i = 1  
while i <= 10:  
    print(f"{n1} x {i} == {n1*i}")  
    i += 1
```

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
'''i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1'''  
i = 1  
while i <= 10:  
    print(f"{n1} x {i} == {n1*i}")  
    i += 1
```

Digita o numero 5

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a single file: `15_exercicio_tabuada_simples.py`.
- Code Editor:** Displays the Python code for generating a multiplication table. The variable `n1` is set to 5.
- Variables View:** Shows the variable `n1` with a value of 5.
- Terminal:** Shows the command to activate the virtual environment and run the script, followed by the prompt "Digite qual número deseja saber a tabuada: []".

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
'''i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1'''

i = 1
while i <= 10:
    print(f'{n1} x {i} == {n1*i}')
    i += 1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter'../../debugpy/launcher' '53378' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: [ ]
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a single file: `15_exercicio_tabuada_simples.py`.
- Code Editor:** Displays the Python code for generating a multiplication table. A breakpoint is set at line 11.
- Variables View:** Shows the variable `n1` with a value of 5.
- Terminal:** Shows the command to activate the virtual environment and run the script, followed by the prompt "Digite qual número deseja saber a tabuada: []".

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
'''i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1'''

i = 1
while i <= 10:
    print(f'{n1} x {i} == {n1*i}')
    i += 1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter'../../debugpy/launcher' '53378' '--'
'C:/Users/Irene/Documents/aulaPython/Introdução_Python/15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: [ ]
```

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named `15_exercicio_tabuada_simples.py`. The code prints a multiplication table for a user-specified number. A blue pen points to the line `print(f'{n1} x {i} == {n1*i}'")`. The left sidebar has a 'WATCH' section with variables `i: 1` and `n1: 5`. The bottom right shows a terminal window with the command `PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1 (venv)` and the output `PS C:\Users\Irene\Documents\aulaPython\Introdução_Python & 'c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/python.exe' 'c:/Users/Irene/.vscode/extensions/ms-python.python-2023.22.1/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher' '53378' '--' 'C:/Users/Irene/Documents/aulaPython/Introdução_Python/15_exercicio_tabuada_simples.py'`. The bottom bar includes tabs for PORTS, AZURE, COMMENTS, OUTPUT, PROBLEMS, DEBUG CONSOLE, TERMINAL, SQL CONSOLE, and GITLENS.

A second screenshot of the VS Code interface, showing the same Python file `15_exercicio_tabuada_simples.py`. The blue pen now points to the line `i += 1`. The rest of the code and the terminal output are identical to the first screenshot.

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named `15_exercicio_tabuada_simples.py`. The code is a script to print a multiplication table for a given number. A blue pen icon is overlaid on the screen, pointing to the line `print(f"\{n1\} x {i} == \{n1*i\}")`. The code editor has syntax highlighting and line numbers. On the left, there's a sidebar with a 'WATCH' section showing variables `i: 1` and `n1: 5`. Below the editor is a terminal window displaying the command line and the output of running the script with the input `5`.

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
'''i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1'''
i = 1
while i <= 10:
    print(f"\{n1\} x {i} == \{n1*i\}")
    i += 1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonfiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '53442' '--'
'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
```

A second screenshot of the VS Code interface, showing the same Python script. In this version, the blue pen icon is pointing to the line `print(f"\{n1\} x {i} == \{n1*i\}")`, which is highlighted in green. The rest of the code and the terminal output are identical to the first screenshot.

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
'''i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1'''
i = 1
while i <= 10:
    print(f"\{n1\} x {i} == \{n1*i\}")
    i += 1

PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/Introdução_Python/venv/Scripts/Activate.ps1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonfiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '53442' '--'
'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
```

A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a Python file named `15_exercicio_tabuada_simples.py`. The code prints a multiplication table for a given number. A blue pen icon is placed over the code at line 12. The terminal below shows the command to run the script and the output for the number 5.

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
'''i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1'''

i = 1
while i <= 10:
    print(f'{n1} x {i} == {n1*i}')
    i += 1

Python Debug Console
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' '53442' '--'
'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
5 x 1 == 5
```

A second screenshot of the VS Code interface, showing the same Python script. A blue pen icon is placed over the code at line 12. The terminal shows the same command and output as the first screenshot.

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
'''i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1'''

i = 1
while i <= 10:
    print(f'{n1} x {i} == {n1*i}')
    i += 1

Python Debug Console
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\Documents\aulaPython\Introdução_Python\venv\Scripts\python.exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' '53442' '--'
'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
5 x 1 == 5
```

A screenshot of the Visual Studio Code (VS Code) interface. The title bar says "Python: 15_exercicio_tabuada_simples.py 3.0". The left sidebar shows a "VARIABLES" tree with "locals" expanded, showing "i: 2" and "n1: 5". The main editor area contains a Python script:

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.  
n1 = int(input('Digite qual número deseja saber a tabuada: '))  
  
'''i = 1  
while i <= 10:  
    print(i)  
    print(n1)  
    i += 1'''  
  
i = 1  
while i <= 10:  
    print(f'{n1} x {i} == {n1*i}')  
    i += 1
```

The terminal at the bottom shows the output of running the script:

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & 'c:\Users\Irene\vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../debugpy\launcher' '53442' '--'  
'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'  
Digite qual número deseja saber a tabuada: 5  
5 x 1 == 5
```

A second screenshot of the VS Code interface, identical to the first one except for the terminal output. The terminal now shows the result of the multiplication table for the number 5:

```
5 x 1 == 5  
5 x 2 == 10
```

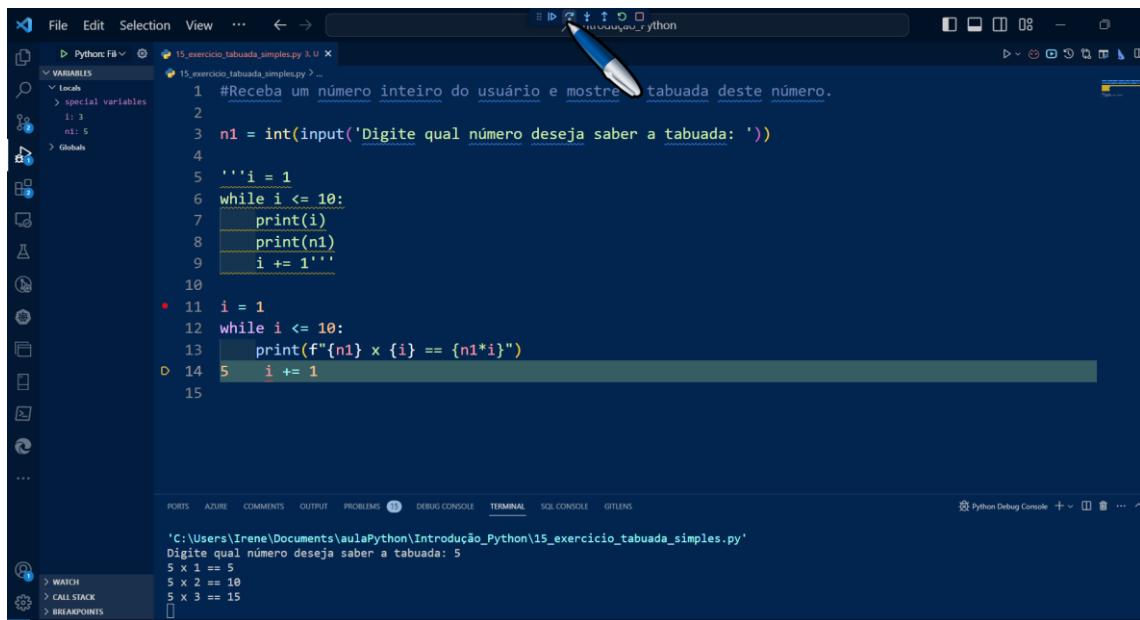
A screenshot of the Visual Studio Code (VS Code) interface. The title bar shows "Python: File > 15_exercicio_tabuada_simples.py 3.0". The left sidebar has a "VARIABLES" section with "locals" expanded, showing "i: 5" and "n1: 5". The main editor area contains a Python script:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 i += 1
15
```

The terminal at the bottom shows the output of running the script with the number 5:

```
exe' 'c:\Users\Irene\.vscode\extensions\ms-python.python-2023.22.1\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '53442' '--'
'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
5 x 1 == 5
5 x 2 == 10
```

A second screenshot of the VS Code interface, identical to the first one except for the highlighted line in the code editor. Line 13, which contains the print statement for the multiplication result, is highlighted in green.



File Edit Selection View ... ← → ⓘ Python: File 15_exercicio_tabuada_simples.py 3.0 X

VARIABLES

locals

> special variables

i: 3
n1: 5

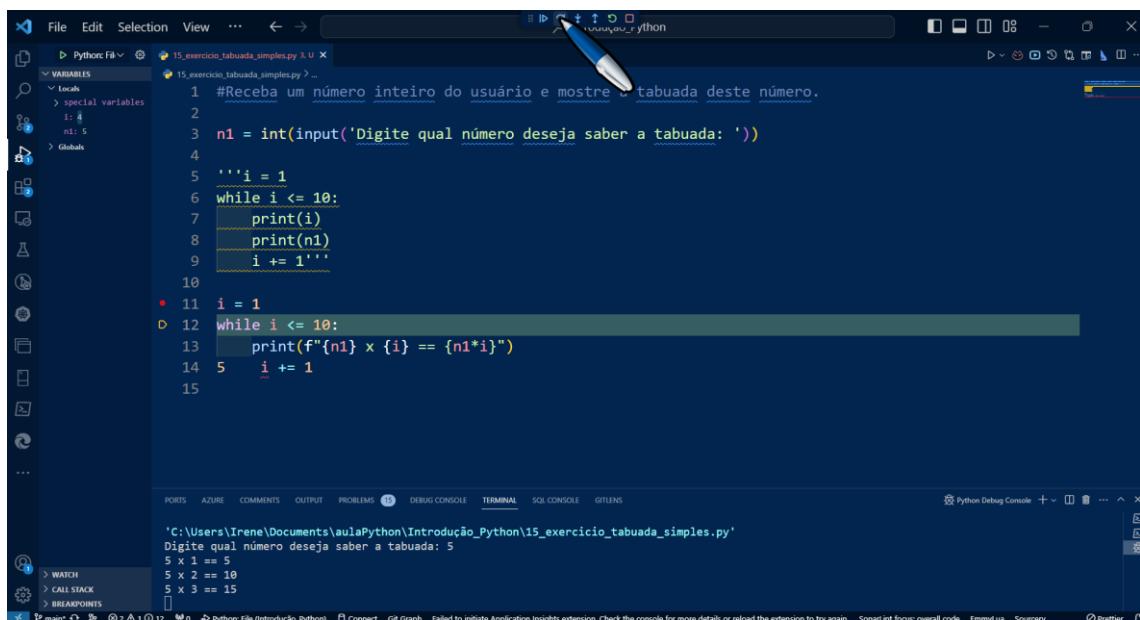
> Globals

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 5   i += 1
15
```

PORIS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
5 x 1 == 5
5 x 2 == 10
5 x 3 == 15

WATCH CALL STACK BREAKPOINTS



File Edit Selection View ... ← → ⓘ Python: File 15_exercicio_tabuada_simples.py 3.0 X

VARIABLES

locals

> special variables

i: 5
n1: 5

> Globals

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 5   i += 1
15
```

PORIS AZURE COMMENTS OUTPUT PROBLEMS DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

'C:\Users\Irene\Documents\aulaPython\Introdução_Python\15_exercicio_tabuada_simples.py'
Digite qual número deseja saber a tabuada: 5
5 x 1 == 5
5 x 2 == 10
5 x 3 == 15

WATCH CALL STACK BREAKPOINTS

A screenshot of Visual Studio Code showing a Python script named `15_exercicio_tabuada_simples.py`. The code is intended to print the multiplication table for a given number. The script uses a while loop to iterate from 1 to 10, printing each number and its product with the input value. The code is annotated with a blue pen highlighting the variable `n1` and the loop condition `i <= 10`.

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1
print(f'{n1} x {i} == {n1*i}')
i += 1
```

A screenshot of Visual Studio Code showing the execution results of the Python script. The terminal window displays the output of the script, which asks for a number and then prints the multiplication table for that number. The input value is 5.

```
Digite qual número deseja saber a tabuada: 5
5 x 1 == 5
5 x 2 == 10
5 x 3 == 15
5 x 4 == 20
```

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 5   i += 1
15
```

Digite qual número deseja saber a tabuada: 5
5 x 1 == 5
5 x 2 == 10
5 x 3 == 15
5 x 4 == 20

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 5   i += 1
15
```

Digite qual número deseja saber a tabuada: 5
5 x 1 == 5
5 x 2 == 10
5 x 3 == 15
5 x 4 == 20

A screenshot of a Python code editor (VS Code) showing a script named `15_exercicio_tabuada_simples.py`. The code prints the multiplication table for the number 5, ranging from 1 to 10. The editor has a dark theme. A blue pen icon is overlaid on the code area. The terminal below shows the output of the program:

```
5 x 1 == 5
5 x 2 == 10
5 x 3 == 15
5 x 4 == 20
5 x 5 == 25
```

A second screenshot of the same Python code editor, showing the same script `15_exercicio_tabuada_simples.py`. In this version, the variable `n1` is set to 6. The code prints the multiplication table for 6, and the terminal shows the output:

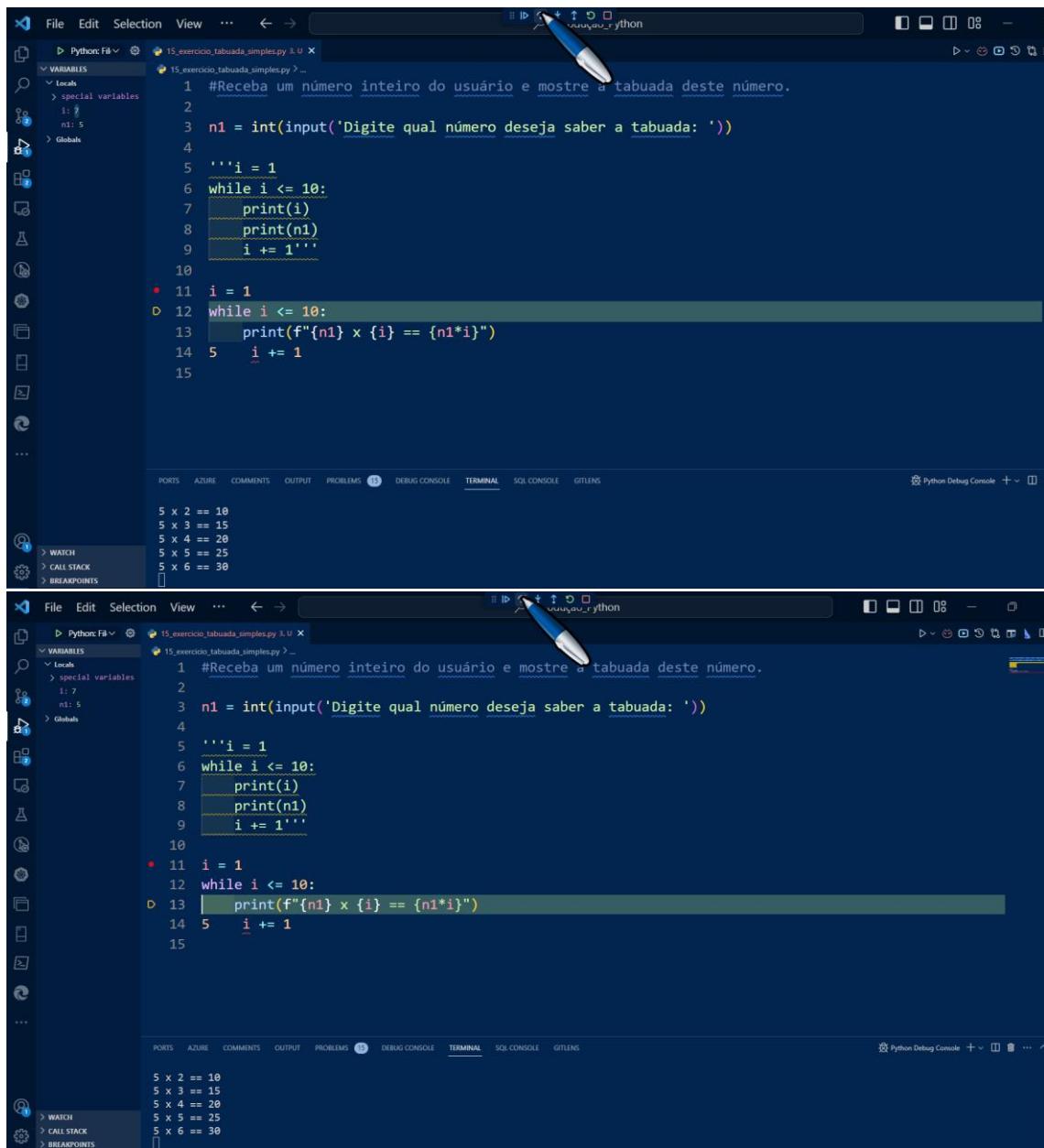
```
6 x 1 == 6
6 x 2 == 12
6 x 3 == 18
6 x 4 == 24
6 x 5 == 30
```

A screenshot of the Visual Studio Code (VS Code) interface. The title bar shows the file path: 15_exercicio_tabuada_simples.py. The code editor displays a Python script named 15_exercicio_tabuada_simples.py. The script is intended to print a multiplication table for a given number n1. It uses a while loop to iterate from 1 to 10, printing each row of the table. The variable n1 is set to 5. The code editor has a dark theme. A blue pen icon is overlaid on the top right of the editor area. The left sidebar contains a 'VARIABLES' section showing local variables i=6 and n1=5. The bottom status bar shows tabs for PORTS, AZURE, COMMENTS, OUTPUT, PROBLEMS, DEBUG CONSOLE, TERMINAL, SQL CONSOLE, and GITLENS. The bottom right corner shows a Python Debug Console tab.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f"{n1} x {i} == {n1*i}")
14     i += 1
15
```

A second screenshot of the Visual Studio Code interface, showing the same Python script 15_exercicio_tabuada_simples.py. The code has been modified to print a multiplication table for n1=6. The variable n1 is now explicitly set to 6. The rest of the script remains the same. The blue pen icon is again overlaid on the top right. The left sidebar shows local variables i=6 and n1=6. The bottom status bar shows tabs for PORTS, AZURE, COMMENTS, OUTPUT, PROBLEMS, DEBUG CONSOLE, TERMINAL, SQL CONSOLE, and GITLENS. The bottom right corner shows a Python Debug Console tab.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f"{n1} x {i} == {n1*i}")
14     i += 1
15
```



```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 i += 1
15
```

5 x 2 == 10
5 x 3 == 15
5 x 4 == 20
5 x 5 == 25
5 x 6 == 30

The screenshot shows the Visual Studio Code interface with a Python file named `15_exercicio_tabuada_simples.py` open. The code is a simple program to print a multiplication table for a given number. A blue pen icon is visible in the top right corner of the editor.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 5   i += 1
15
```

The code is annotated with several numbers from 5 to 15, likely representing the multiplication table for the number 5. The bottom part of the screenshot shows the Python Debug Console output:

```
5 x 3 == 15
5 x 4 == 20
5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
```

This screenshot shows the same Python file `15_exercicio_tabuada_simples.py` in VS Code. The line `12 while i <= 10:` is highlighted in red, indicating it is the current line of execution or a breakpoint.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 5   i += 1
15
```

The bottom Python Debug Console output is identical to the first screenshot:

```
5 x 3 == 15
5 x 4 == 20
5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
```

A screenshot of the Visual Studio Code (VS Code) interface. The title bar shows the file name "15_exercicio_tabuada_simples.py". The code editor displays the following Python script:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14     i += 1
15
```

The terminal below the editor shows the output of the script for the number 5:

```
5 x 3 == 15
5 x 4 == 20
5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
```

A second screenshot of the Visual Studio Code interface, showing the same Python script. The terminal output is different, indicating a bug in the code where the variable 'i' is not properly initialized or updated.

```
5 x 3 == 15
5 x 4 == 20
5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
```

A screenshot of the Visual Studio Code (VS Code) interface. The title bar shows the file path: 15_exercicio_tabuada_simples.py. The main editor area contains the following Python code:

```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1
i = 1
while i <= 10:
    print(f'{n1} x {i} == {n1*i}')
    i += 1
```

The code uses two nested loops to print the multiplication table for a given number. The first loop iterates from 1 to 10, and the second loop inside it prints the result of multiplying the user-specified number by each value from 1 to 10. A blue pen is pointing at the line `i += 1` in the second loop's body.

The left sidebar shows the 'VARIABLES' section with locals: `i: 9` and `n1: 5`. The bottom status bar shows tabs for PORTS, AZURE, COMMENTS, OUTPUT, PROBLEMS, DEBUG CONSOLE, TERMINAL, SQL CONSOLE, and GITLENS. The Python Debug Console tab is active, showing the output of the program:

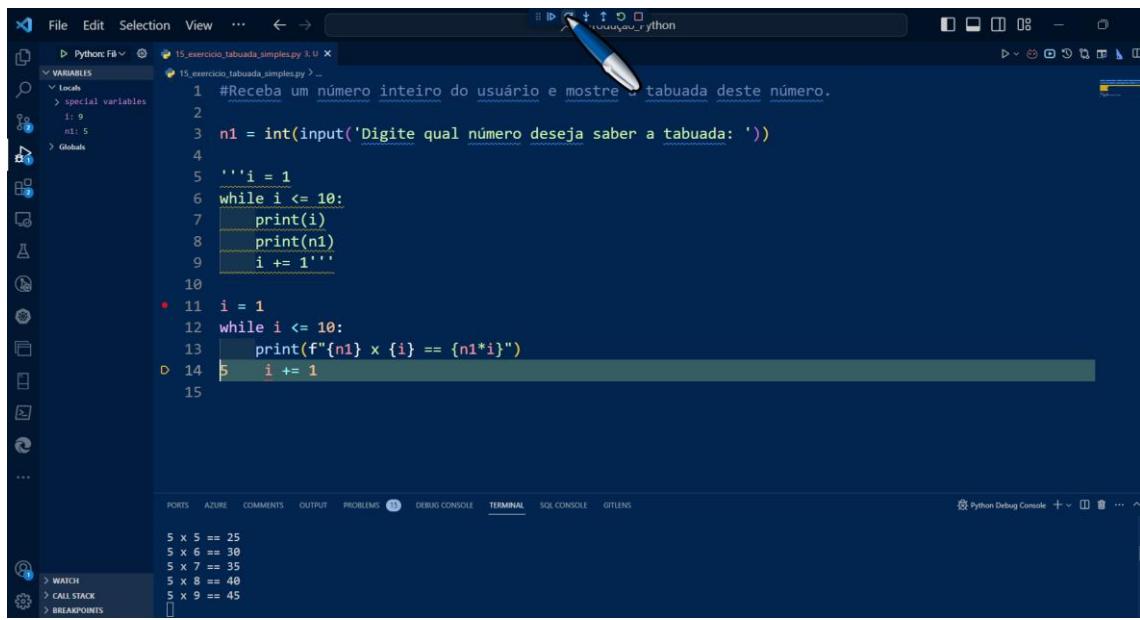
```
5 x 4 == 20
5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
```

A second screenshot of the VS Code interface, showing the same Python script. The title bar again shows the file path: 15_exercicio_tabuada_simples.py. The code is identical to the one in the first screenshot.

The blue pen is now pointing at the line `print(f'{n1} x {i} == {n1*i}'")` in the second loop's body.

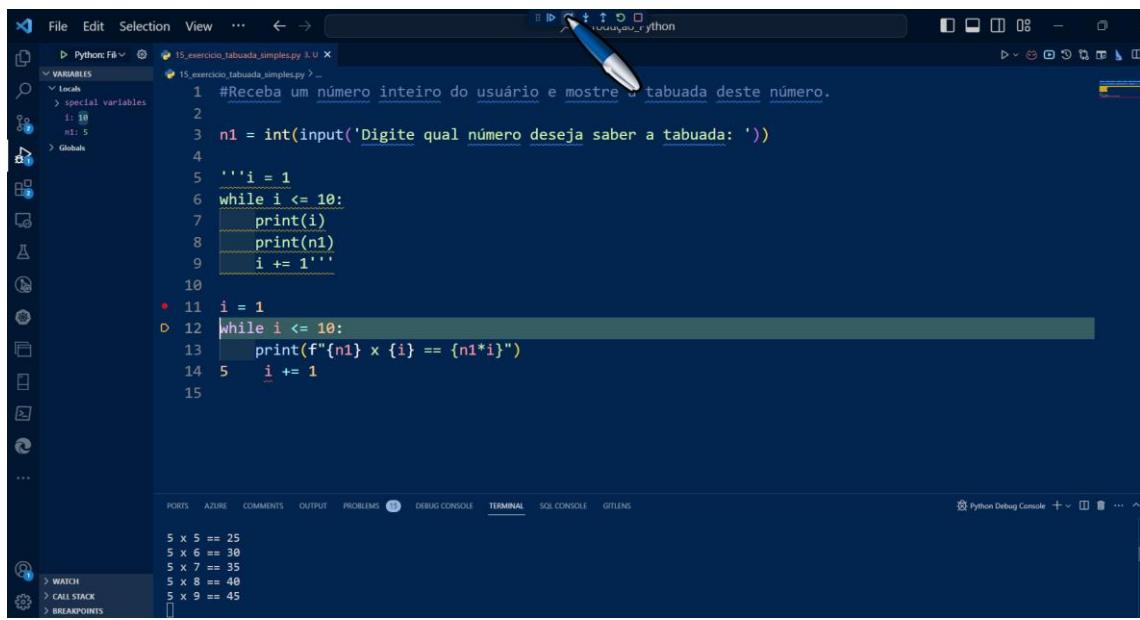
The left sidebar shows the 'VARIABLES' section with locals: `i: 9` and `n1: 5`. The bottom status bar shows tabs for PORTS, AZURE, COMMENTS, OUTPUT, PROBLEMS, DEBUG CONSOLE, TERMINAL, SQL CONSOLE, and GITLENS. The Python Debug Console tab is active, showing the output of the program:

```
5 x 4 == 20
5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
```



```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1
i = 1
while i <= 10:
    print(f'{n1} x {i} == {n1*i}')
i += 1
```

5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
5 x 9 == 45



```
#Receba um número inteiro do usuário e mostre a tabuada deste número.
n1 = int(input('Digite qual número deseja saber a tabuada: '))
i = 1
while i <= 10:
    print(i)
    print(n1)
    i += 1
i = 1
while i <= 10:
    print(f'{n1} x {i} == {n1*i}')
i += 1
```

5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
5 x 9 == 45

A screenshot of the Visual Studio Code (VS Code) interface. The title bar shows the file name "15_exercicio_tabuada_simples.py". The code editor displays a Python script:

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
• 11 i = 1
12 while i <= 10:
13     print(f"{n1} x {i} == {n1*i}")
14     i += 1
15
```

The code uses a while loop to print the multiplication table for the number entered by the user. The output window at the bottom shows the results for n1=5:

```
5 x 5 == 25
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
5 x 9 == 45
```

A second screenshot of the VS Code interface, showing the same Python script. The output window at the bottom shows a different set of results:

```
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
5 x 9 == 45
5 x 10 == 50
```

Screenshot of a Python code editor showing the script `15_exercicio_tabuada_simples.py`. The code prints the multiplication table for the number 5 up to 10. A breakpoint is set at line 12. The output window shows the results of the multiplication table.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
11 i = 1
12 while i <= 10:
13     print(f"{n1} x {i} == {n1*i}")
14 i += 1
15
```

Output:

```
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
5 x 9 == 45
5 x 10 == 50
```

Screenshot of a Python code editor showing the script `15_exercicio_tabuada_simples.py`. The code prints the multiplication table for the number 5 up to 10. A breakpoint is set at line 12. The output window shows the results of the multiplication table.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
11 i = 1
12 while i <= 10:
13     print(f"{n1} x {i} == {n1*i}")
14 i += 1
15
```

Output:

```
5 x 6 == 30
5 x 7 == 35
5 x 8 == 40
5 x 9 == 45
5 x 10 == 50
```

Screenshot of a Python code editor showing the script `15_exercicio_tabuada_simples.py`. The code prints the multiplication table for the number 5 up to 10. A breakpoint is set at line 12. The output window shows the results of the multiplication table.

```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
11 i = 1
12 while i <= 10:
13     print(f"{n1} x {i} == {n1*i}")
14 i += 1
15
```

A screenshot of the Visual Studio Code (VS Code) interface. The title bar says "Introdução_Python". The left sidebar shows icons for file operations like Open, Save, Find, and others. The main editor area contains a Python script named "15_exercício_tabuada_simples.py". The code is as follows:

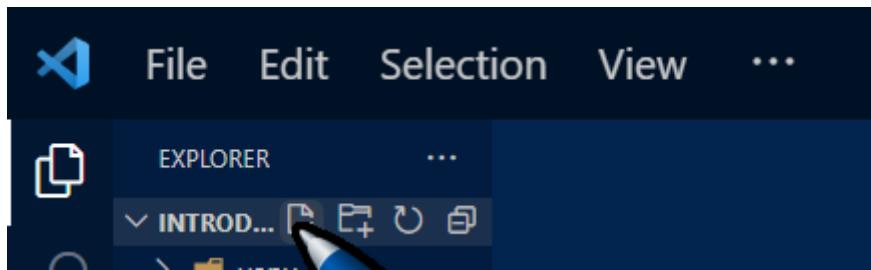
```
1 #Receba um número inteiro do usuário e mostre a tabuada deste número.
2
3 n1 = int(input('Digite qual número deseja saber a tabuada: '))
4
5 '''i = 1
6 while i <= 10:
7     print(i)
8     print(n1)
9     i += 1'''
10
11 i = 1
12 while i <= 10:
13     print(f'{n1} x {i} == {n1*i}')
14 i += 1
15
```

The terminal at the bottom shows the output of the script for the number 5:

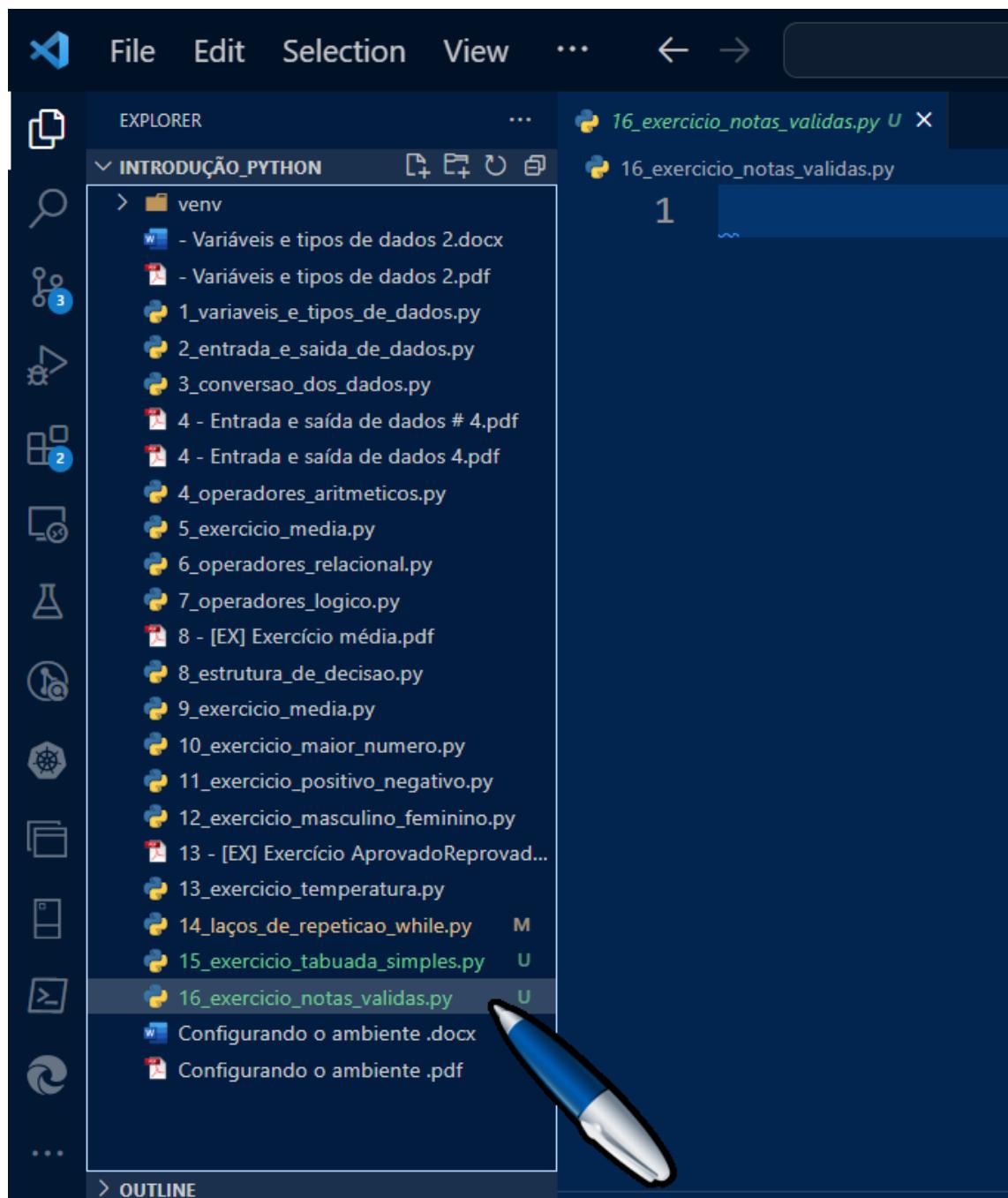
```
5 x 7 == 35
5 x 8 == 40
5 x 9 == 45
5 x 10 == 50
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

22 - [EX] Exercício nota válida

Criar um arquivo



Nome do arquivo





```
16_exercicio_notas_validadas.py U x
16_exercicio_notas_validadas.py > ...
1 #Escreva um programa que receba notas de um aluno (0 - 10), caso
2 #a nota digitada esteja fora deste intervalo peça para o
3 #professor digitar novamente
4
5 while True:
6     nota_aluno = int(input('Digite a nota do aluno: '))
7     if nota_aluno >= 0 and nota_aluno <= 10:
8         print(f'Nota armazenada com sucesso {nota_aluno}')
9         break
10    print('Nota invalida digite novamente')
```

Roda o código



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 16_exercicio_notas_validadas.py
•Digite a nota do aluno: |
```



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 16_exercicio_notas_validadas.py
•Digite a nota do aluno: 70
Nota invalida digite novamente
Digite a nota do aluno: |
```



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 16_exercicio_notas_validadas.py
•Digite a nota do aluno: 70
Nota invalida digite novamente
Digite a nota do aluno: 68
Nota invalida digite novamente
Digite a nota do aluno: |
```



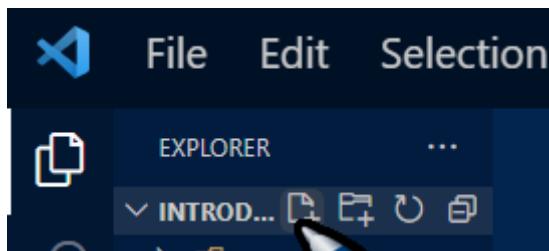
```
Nota invalida digite novamente
Digite a nota do aluno: 68
Nota invalida digite novamente
Digite a nota do aluno: -3
Nota invalida digite novamente
Digite a nota do aluno: |
```

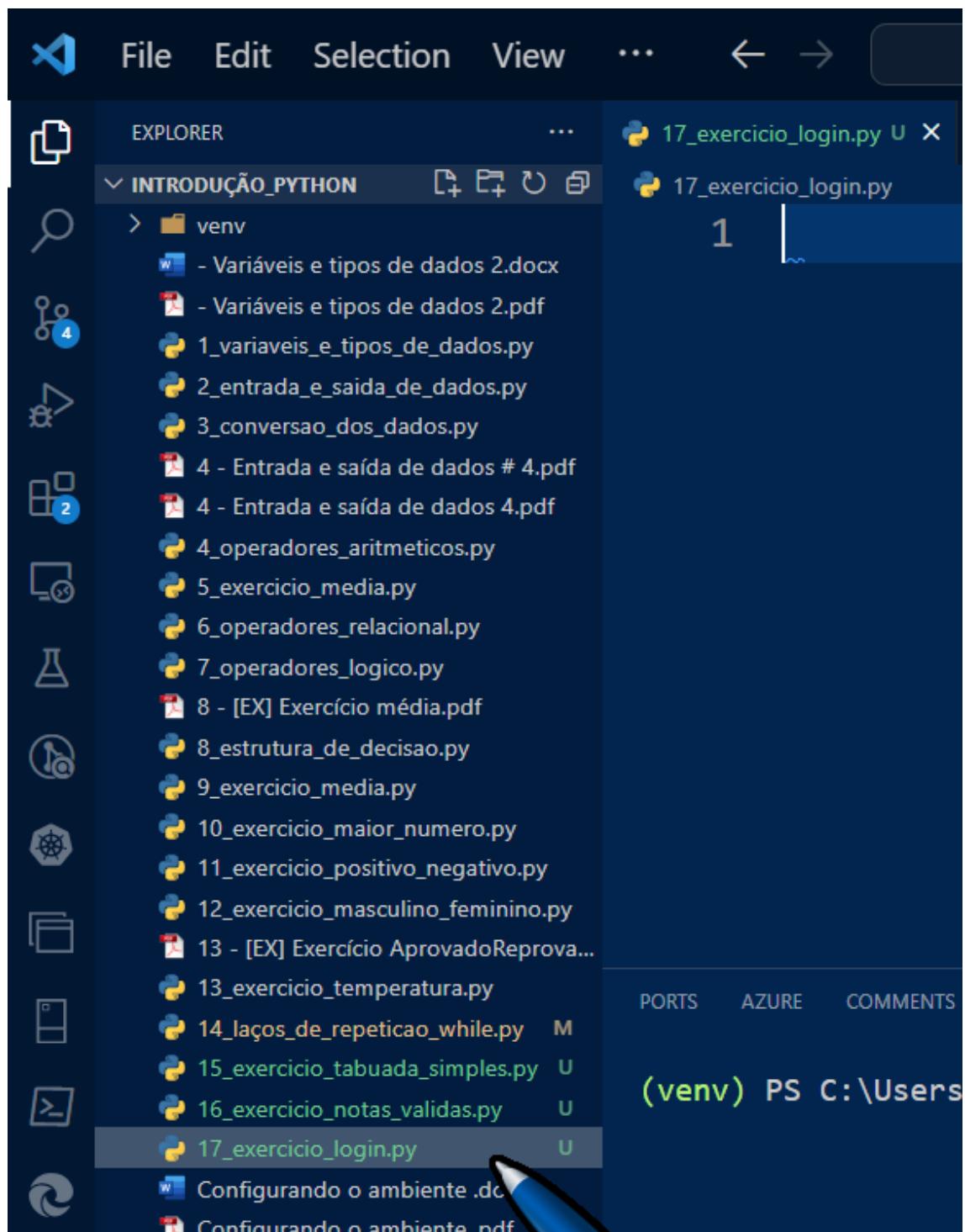
```
| Nota invalida digite novamente  
| Digite a nota do aluno: 68  
| Nota invalida digite novamente  
| Digite a nota do aluno: -3  
| Nota invalida digite novamente  
| Digite a nota do aluno: 6  
| Nota armazenada com sucesso 6  
o (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> |
```



23 - [EX] Exercício login.mp4 laço de repetição while

Novo arquivo







```
17_exercicio_login.py U x
17_exercicio_login.py > ...
1 #Defina um usuário e senha e depois verifique se
2 #login do usuário é válido.
3
4 USUARIO = "irene"
5 SENHA = "minha_senha123"
6
7 while True:
8     usuario_login = input('Didite seu nome de usuário: ')
9     senha_login = input('Digite sua senha: ')
10
11    if usuario_login == USUARIO and senha_login == SENHA:
12        print('Você foi logado no sistema!')
13        break
14    else:
15        print('Usuario ou senha inválido.'
```

Vamos rodar o código



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 17_exercicio_login.py
Didite seu nome de usuário: |
```



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 17_exercicio_login.py
Didite seu nome de usuário: Irene
Digite sua senha:|
```



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 17_exercicio_login.py
Didite seu nome de usuário: Irene
Digite sua senha:irene123
Usuario ou senha inválido.
Didite seu nome de usuário: |
```



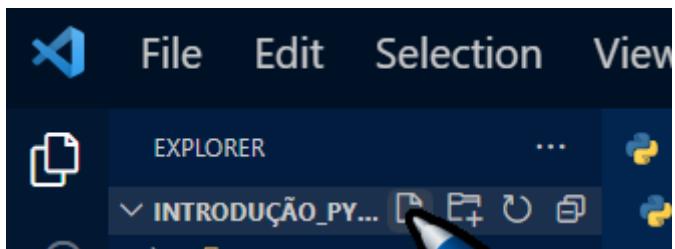
```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 17_exercicio_login.py
Didite seu nome de usuário: Irene
Digite sua senha:irene123
Usuario ou senha inválido.
Didite seu nome de usuário: irene
Digite sua senha:|
```

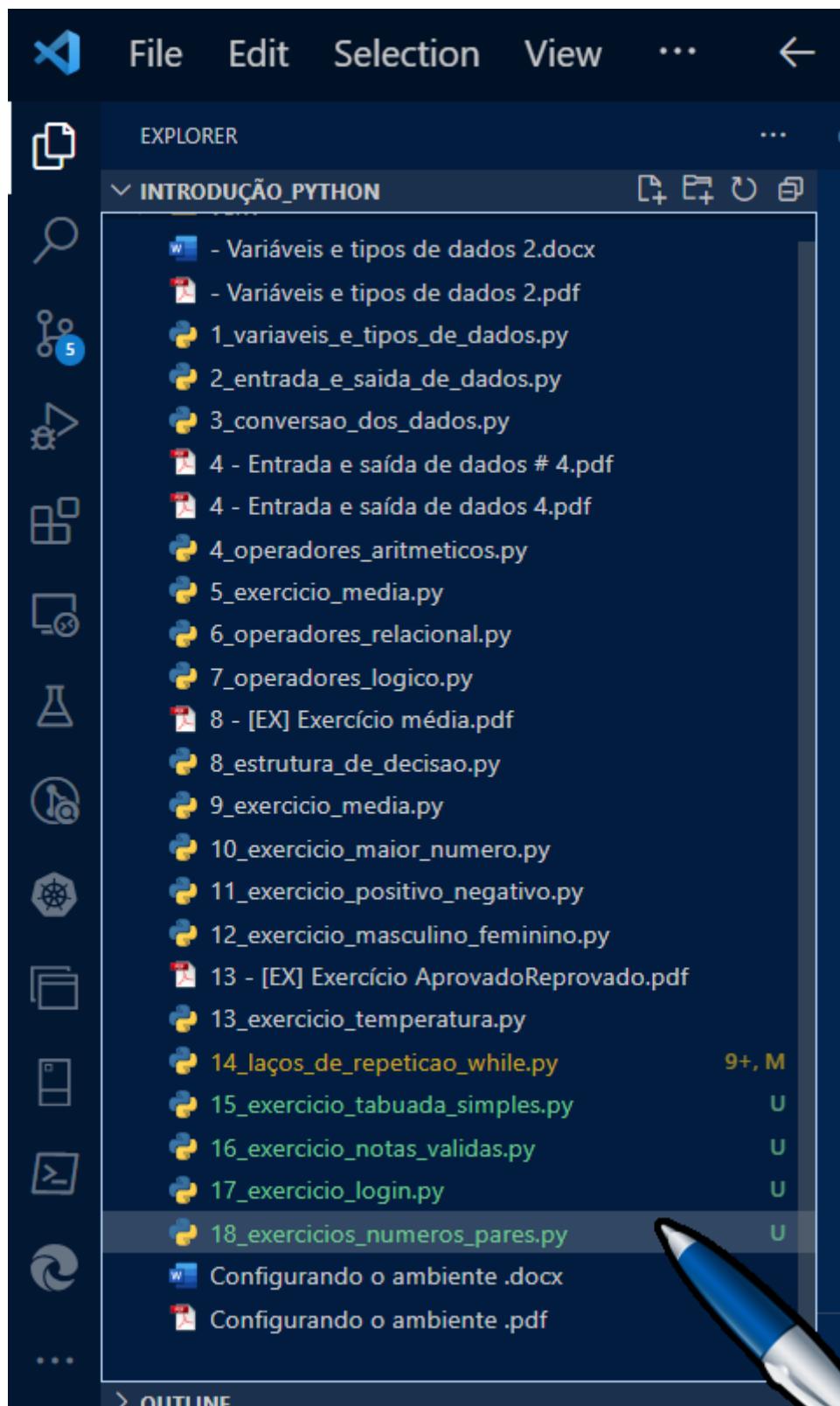


```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 17_exercicio_login.py
Didite seu nome de usuário: Irene
Digite sua senha:irene123
Usuario ou senha inválido.
Didite seu nome de usuário: irene
Digite sua senha:minha_senha123
Você foi logado no sistema!
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> |
```

24 - [EX] Exercício números pares

Crie um arquivo





```
18_exercicios_numeros_pares.py > ...
1 #Receba um número e mostre todos os números pares de 0 até o número digitado.
2
3 n1 = int(input('Digite um número: '))
4
5 i = 1
6 while i <= n1:
7     if i % 2 == 0:
8         print(i)
9     i += 1
10
```

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 18_exercicios_numeros_pares.py
Digite um número: |
```

```
Digite um número: 10
2
4
6
8
10
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> |
```

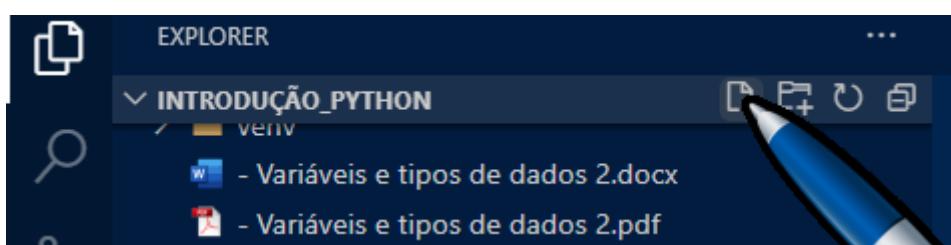
```
986
988
990
992
994
996
998
1000
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> |
```

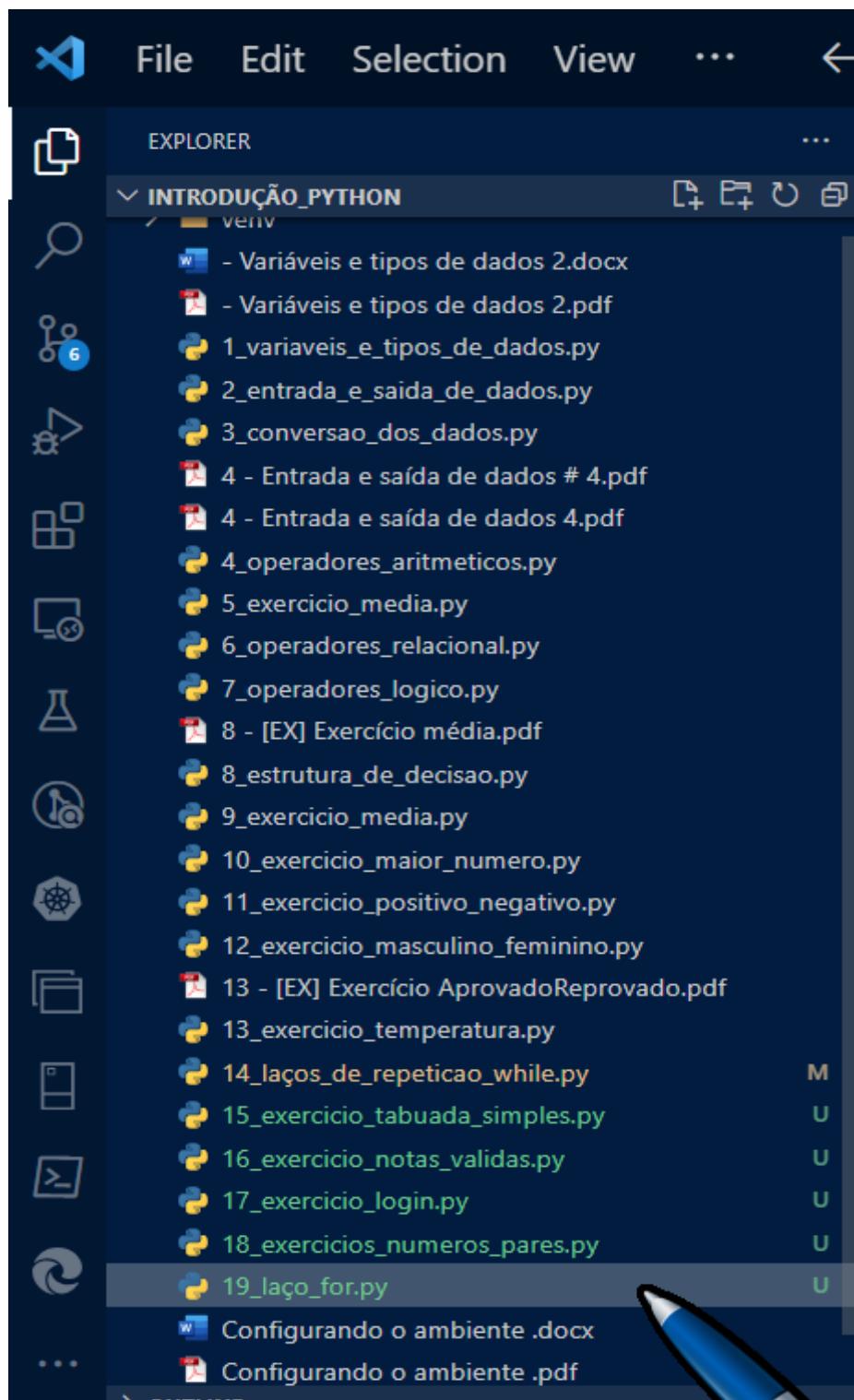
```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 18_exercicios_numeros_pares.py
Digite um número: |
```

ph Failed to initiate Application Insights extension. Check the console for details or reload the extension to try again. SonarLint focus: overall code Sourcery Go Live ▲ 9 Spell ⚙

25 - Laço for #1.mp4

Novo arquivo e nome





2 laços de repetição **While** quantas vezes o código vai usar ou repetir

Laço for forma mais enxuta ex(tabuada melhor for mais simples)

For ➔ para

Podemos debugar para ver passo a passo conforme aprendemos acima ou rodar o código direto.



```
19_Laço_for.py > ...
1 for i in range(0, 5):
2     print(i)
```



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
0
1
2
3
4
```



```
1 '''for i in range(0, 5):
2     print(i)'''
3
4 for i in range(0, 100):
5     print(i)
```



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
```



```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
```

```
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63
```

```
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
:(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> █
```

Step de 2 em dois

```
1  '''for i in range(0, 5):  
2      print(i)'''  
3    
4  '''for i in range(0, 100):  
5      print(i)'''  
6    
7    
8  for i in range(0, 100, 2):  
9      print(i)
```



Vai de 2 em 2 ou como quiser de 4 em 4 de 5 em 5

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
0
2
4
6
8
10
12
14
16
16
18
20
22
24
26
26
28
30
32
34
36
38
40
```

```
68
70
72
74
76
78
80
82
84
86
88
90
92
94
96
98
o (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
1 '''for i in range(0, 5):
2     print(i)'''
3
4 '''for i in range(0, 100):
5     print(i)'''
6
7
8 '''for i in range(0, 100, 2 ):
9     print(i)'''
10
11 for i in range(2, 100, 2 ):
12     print(i)
```

Roda o código

```
76  
78  
80  
82  
84  
86  
88  
90  
92  
94  
96  
98  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> █
```



Ordem crescente

```
5  
4  '''for i in range(0, 100):  
5      print(i)'''  
6  
7  
8  '''for i in range(0, 100, 2 ):  
9      print(i)'''  
10  
11 '''for i in range(2, 100, 2 ):  
12     print(i)'''  
13  
14 for i in range[100, 0, -1 ]:  
15     print(i)
```



```
12  
11  
10  
9  
8  
7  
6  
5  
4  
3  
2  
1  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> █
```



Range função que vai criar intervalo

```
14  for i in range(100, 0, -1 ):  
15      print(i)
```

```
print(list( range(0, 5)))
print(i)
```

```
1
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
[0, 1, 2, 3, 4]
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
python 19_laço_for.py
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
17
18     '''print(list( range(0, 5)))'''
19
20 print(list( range(0, 10)))
21
```

```
POTS AZURE COMMENTS OUTPUT PROBLEMS 17 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
NameError: name 'i' is not defined. Did you mean: 'id'?
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
22
23 print(list( range(100, 0, -1)))
```

```
POTS AZURE COMMENTS OUTPUT PROBLEMS 15 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
JavaScript Debug Terminal + □ ■ ...
python 19_laço_for.py
[100, 99, 98, 97, 96, 95, 94, 93, 92, 91, 90, 89, 88, 87, 86, 85, 84, 83, 82, 81, 80, 79, 78, 77, 76, 75, 74, 73, 72, 71, 70, 69, 68, 67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56, 55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38, 37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

X função ranger

```
21
22
23     '''print(list( range(100, 0, -1)))'''
24 x = [2, 7, 'x', 'z', 45]
25 for i in x:
26     print(i)
```

```
POTS AZURE COMMENTS OUTPUT PROBLEMS 20 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS
(venv) python :\\Users\\Irene\\Documents\\aulaPython\\Introdução_Python>
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr 4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
(venv) PS C:\\Users\\Irene\\Documents\\aulaPython\\Introdução_Python> python 19_laço_for.py
```

```
    22
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
  2
  7
  x
  z
  45
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```



```
22
23     '''print(list( range(100, 0, -1)))'''
24
25     '''x = [2, 7, 'x', 'z', 45]
26     for i in x:
27         print(i)'''
28
29     x = input('Digite seu nome:')
30
31     for i in x:
32         print(i)
```



```
    43
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
  Digite seu nome:irene
• i
  r
  e
  n
  e
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```



```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
  Digite seu nome:Irene França
  I
  r
  e
  n
  e
  F
  r
  a
  n
  ç
  a
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```



26 - Laço for #2.mp4 → esta acima

27 - Laço for #3 e complexidade

For alinhado

```
36  for i in range(0, 3):
37      for j in range(0, 3):
38          print(f"i ={i} j = {j}")
```



Roda código e se quiser pode debuggar conforme aprendido acima sobre debug

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 19_laço_for.py
i =0 j = 0
i =0 j = 1
i =0 j = 2
• i =1 j = 0
i =1 j = 1
i =1 j = 2
i =2 j = 0
i =2 j = 1
i =2 j = 2
◦ (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```



Complexibilidade algoritmo quer dizer quantas vezes o código será rodado.

Código vai demorar a parar

```
41  k = 0
42  for i in range(0, 10000):
43      for j in range(0, 10000):
44          print(f"i ={i} j = {j}")
45          k += 1
46
47  print(k)
```

1
i =7 j = 8799
1
i =7 j = 8800
1
i =7 j = 8801
1
i =7 j = 8802
1
Traceback (most recent call last):
 File "C:\Users\Irene\Documents\aulaPython\Introdução_Python\19_laço_for.py", line 47, in <module>
 print(k)
KeyboardInterrupt
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>



