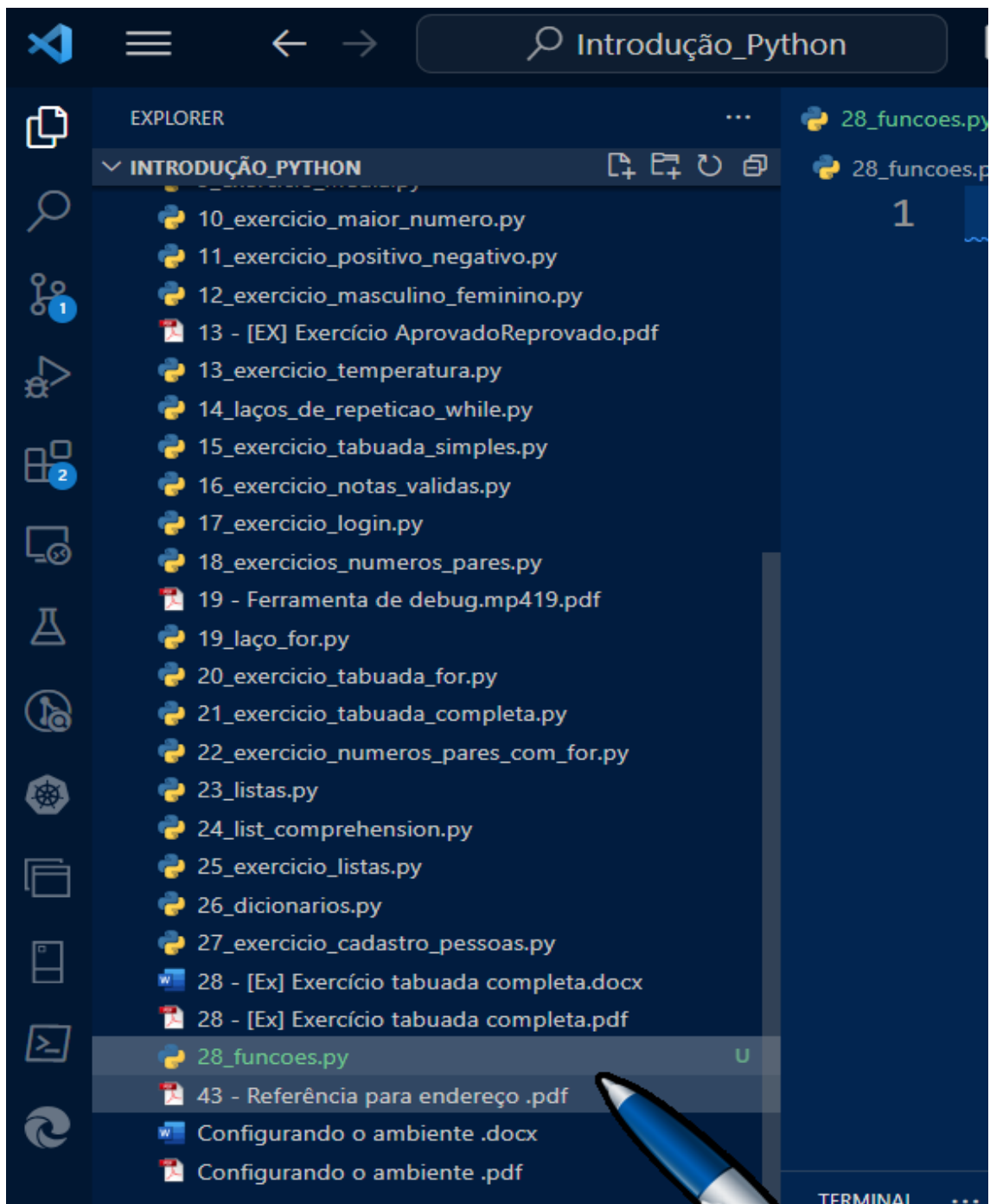
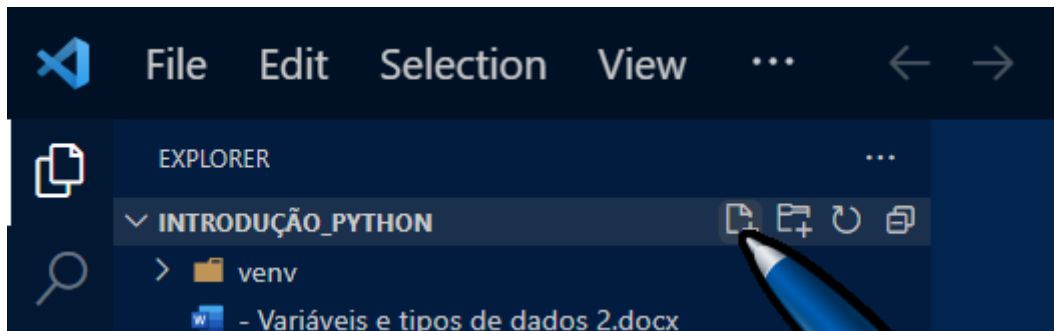



51 - Funções #1

Criar arquivo e dar um nome a ele



```
28_funcoes.py > ...
1  #Funções ->estrutura da linguagem Python que pode ou,
2  #não receber um valor,processar alguma função, executar
3  #uma determinada ação.Dividir para conquistar.
4
5  def minha_funcao():
6      soma = 0
7      for i in range(0, 101):
8          soma = soma + i
9      print(soma)
10
11 def soma():
12     print('Estou somando')
13
14 soma()
15 minha_funcao()
16 minha_funcao()
17 minha_funcao()
18
19
```




PORTS AZURE COMMENTS OUTPUT PROBLEMS 31 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 28_funcoes.py
Estou somando
5050
5050
5050
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> 
```

52 - Funções #2 Parâmetros, args e kwargs

```
19 #Funções ->estrutura da linguagem Python que pode ou,
20 #não receber um valor,processar alguma função, executar
21 #uma determinada ação.Dividir para conquistar.
22 #parametros de entrada *args.
23
24 def soma_numeros(*args):
25     soma = 0
26     for i in args:
27         soma = soma + i
28     print(soma)
29
30 soma_numeros(1, 2, 3, 4, 5, 6, 7, 8)
31
32
33
```



PORTS AZURE COMMENTS OUTPUT PROBLEMS 46 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS JavaScript Debug Terminal

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 28_funcoes.py
36
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> 
```

```
28_funcoes.py > ...
36
37 #Funções -> estrutura da linguagem Python que pode ou,
38 #não receber um valor, processar alguma função, executar
39 #uma determinada ação. Dividir para conquistar.
40 #parametros de entrada *args dar , empacota tudo dentro de tupla
41 #parametros de entrada **kwargs nomes para parametro empacota tudo dentro do dicionario.
42
43 def soma_numeros(**kwargs):
44     x = kwargs.get('teste4')
45     if x:
46         print('foi passado')
47     else:
48         print('não foi passado')
49
50 soma_numeros(teste1 = 1, teste2 = 2, teste3 = 3, teste4 = 4)
51
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 59 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 28_funcoes.py
não foi passado
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 28_funcoes.py
foi passado
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

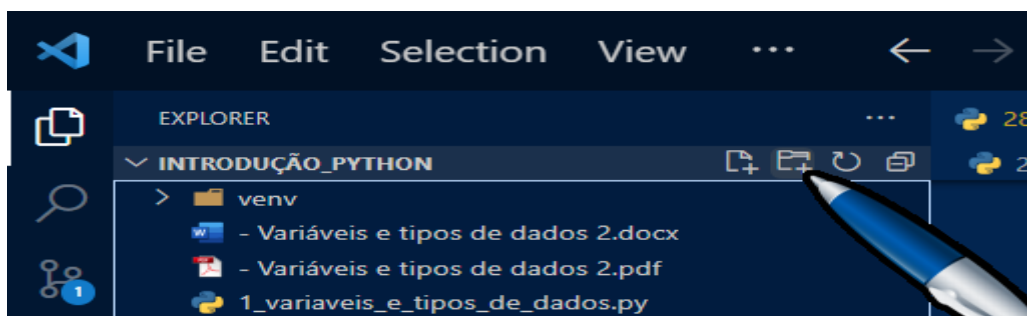
53 - Funções #3 Retornando

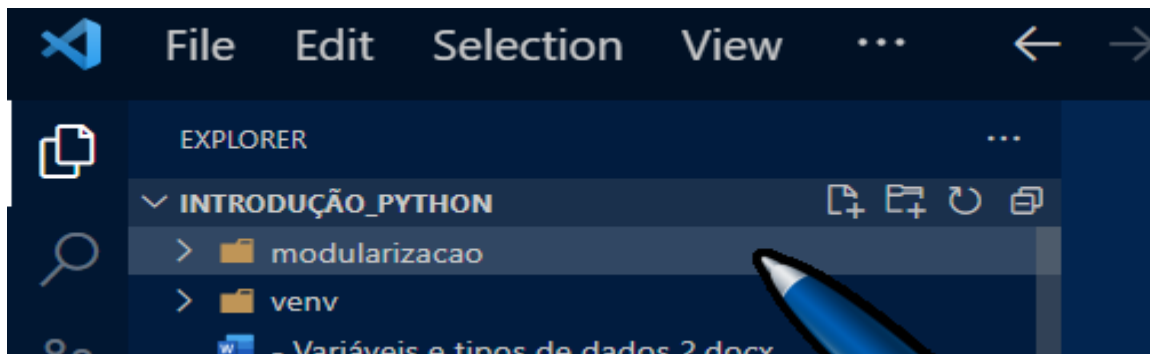
```
52 # Retornar valores da função
53 #Código abaixo do return não será executado nunca
54
55 def soma_valores(n1, n2):
56     soma = n1 + n2
57     if soma > 5:
58         return soma
59     print('TO AQUI')
60
61 soma = soma_valores(5, 2)
62 print(soma)
63
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 111 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 28_funcoes.py
7
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

54 – Modularização Vamos criar uma nova pasta e colocar nome

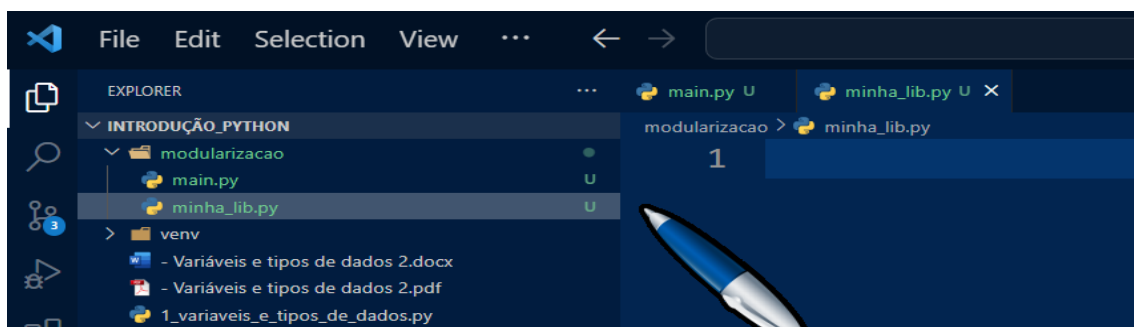
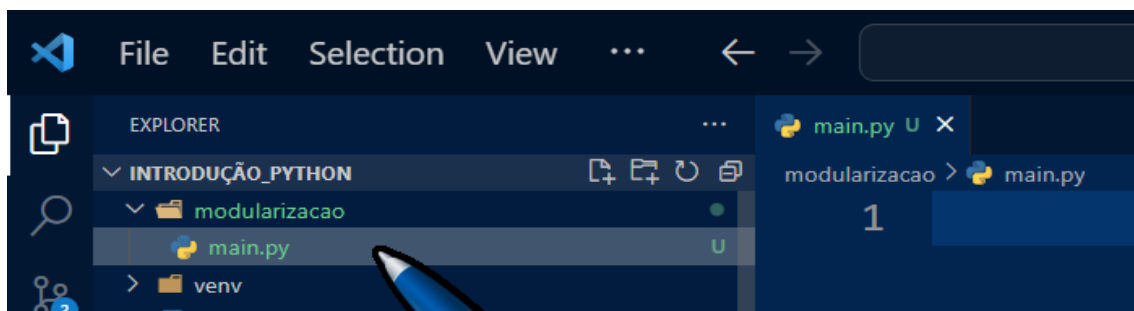
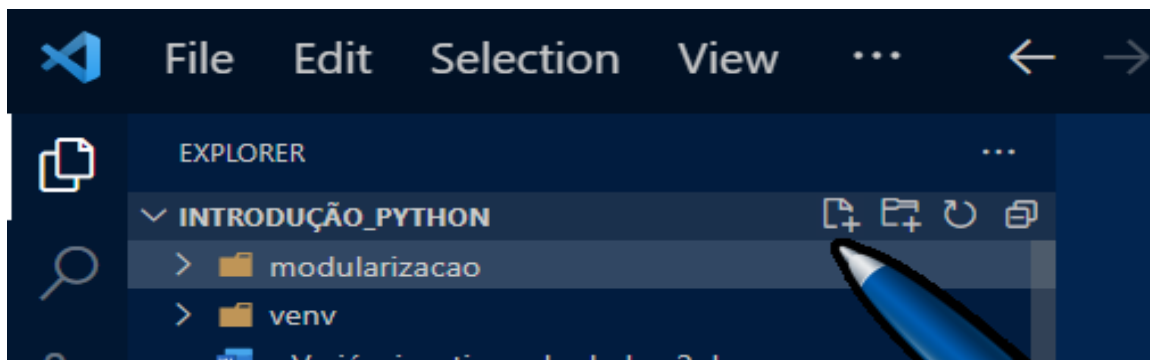




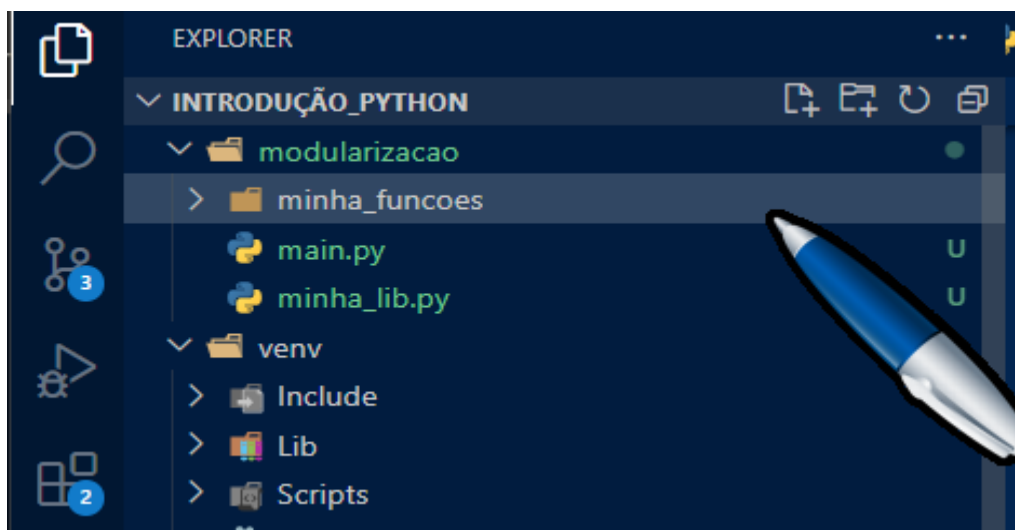
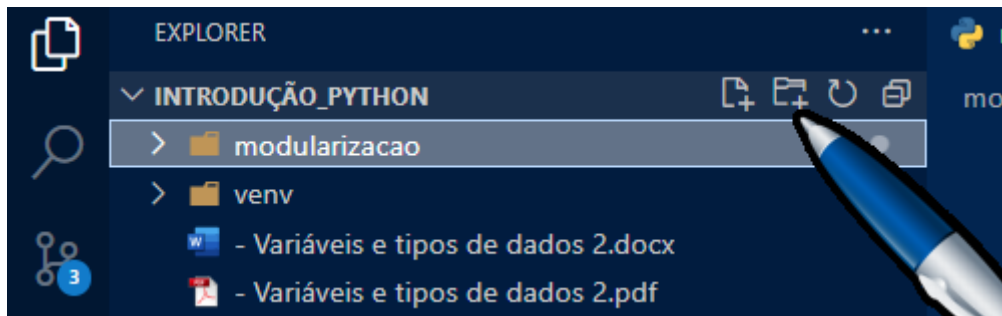
Entra dentro da pasta modularização cd modularização

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

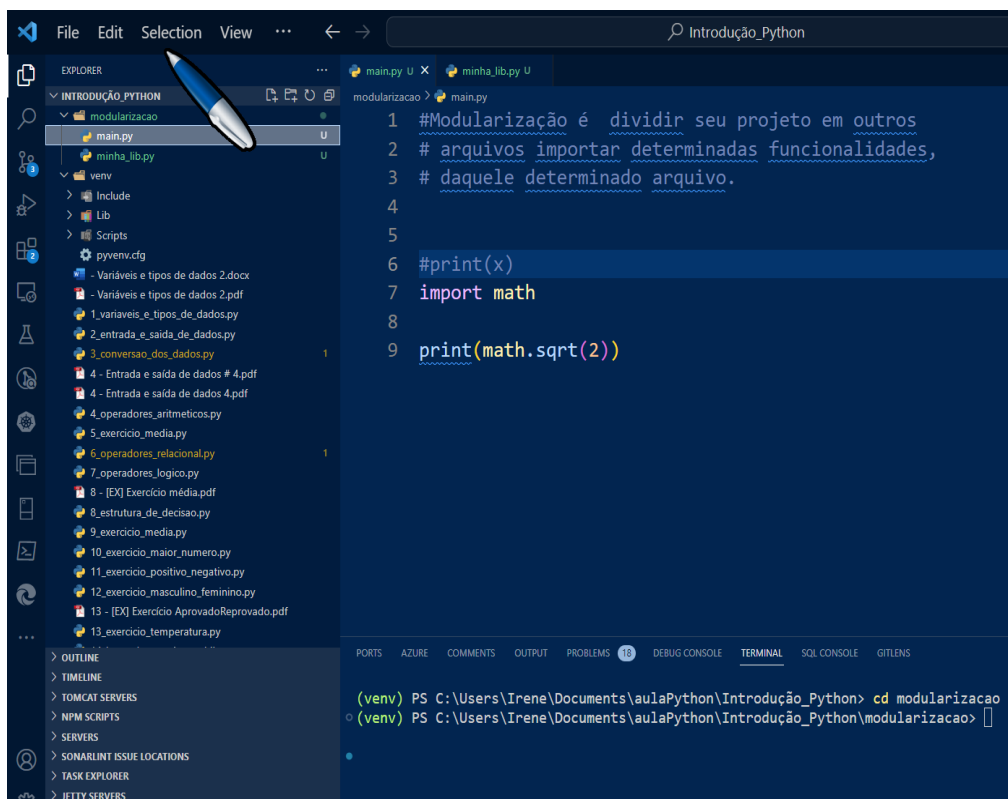
Dentro da pasta modularização crie um arquivo main.py e minha_lib.py



Dentro da pasta Modularização crie outra pasta minhas_funcoes

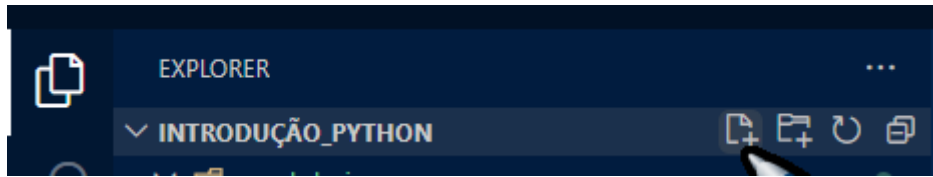


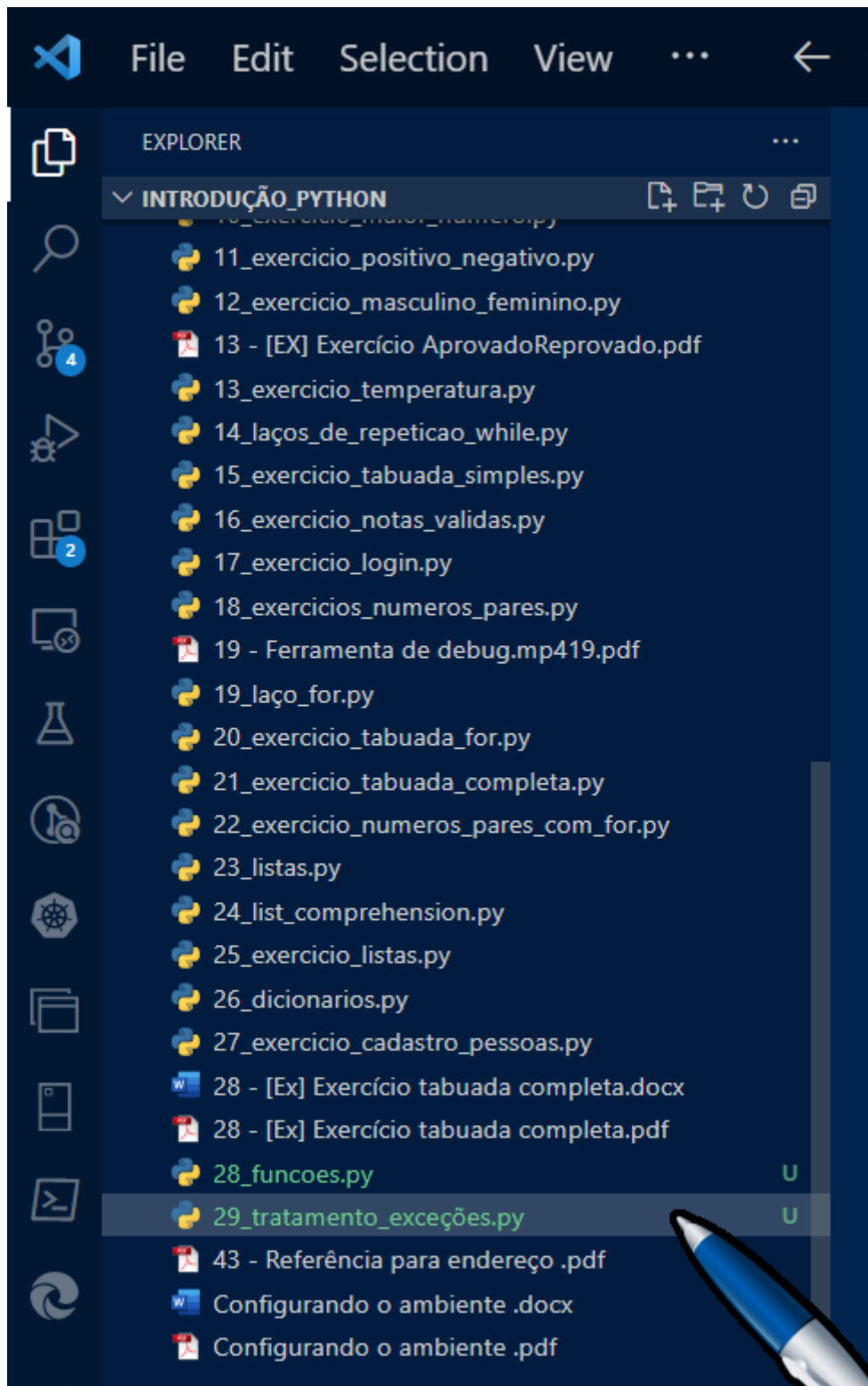
Deletei pasta minhas_funcoes



55 - Tratamento de exceções #1

Criar arquivo e dar nome





Sai da pasta modularização.

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

```
29_tratamento_exceções.py > ...
1  #Tratamento de exceções #1 tente fazer alguma coisa ,
2  # se não,se conseguir ok, senão tente outra,
3  #try = tente except = se não consegui finally = finalizado
4
5  n1 = int(input('Digite um número:'))
6  n2 = int(input('Digite um número:'))
7
8  try:
9      print(n1/n2)
10 except:
11     print('Não consegui')
12 finally:
13     print('Finalizado!')
14
--
PORTS  AZURE  COMMENTS  OUTPUT  PROBLEMS  27  DEBUG CONSOLE  TERMINAL  SQL CONSOLE  GITLENS

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 29_tratamento_exceções.py
Digite um número:2
Digite um número:0
Não consegui
Finalizado!
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> 
```

56 - Tratamento de exceções #2

```
17 #Tratamento de exceções #1 tente fazer alguma coisa ,
18 # se não,se conseguir ok, senão tente outra,
19 #try = tente except = se não consegui finally = finalizado
20 # vamos tratar cada exceção de forma diferente
21
22 try:
23     x = int(input('Digite um número: '))
24     print(5/x)
25 except ValueError:
26     print('Digite um número!')
27 except ZeroDivisionError:
28     print('Não digite 0')
29
PORTS  AZURE  COMMENTS  OUTPUT  PROBLEMS  46  DEBUG CONSOLE  TERMINAL  SQL CONSOLE  GITLENS

• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 29_tratamento_exceções.py
Digite um número: 0
Não digite 0
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 29_tratamento_exceções.py
Digite um número: Irene
Digite um número!
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 29_tratamento_exceções.py
```

57 - Tratamento de exceções #3


```
30 #Tratamento de exceções #1 tente fazer alguma coisa ,
31 # se não,se conseguir ok, senão tente outra,
32 #try = tente except = se não consegui finally = finalizado
33 # vamos tratar cada exceção de forma diferente
34 # Vamos capturar a exceção que foi dada.
35
36 try:
37     x = int(input('Digite um número: '))
38     print(5/x)
39 except Exception as e:
40     print('Erro interno do sistema')
```

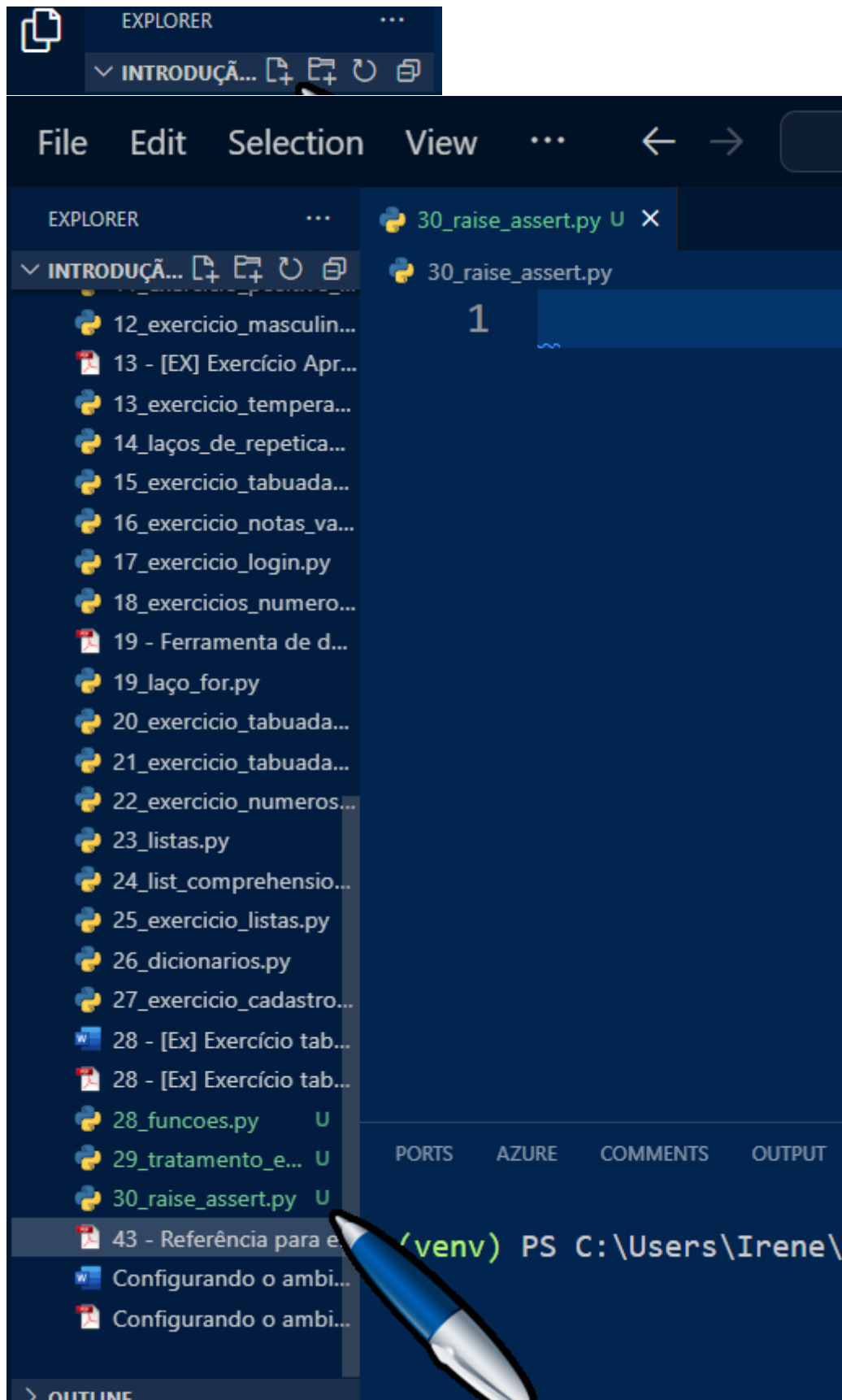


PORTS AZURE COMMENTS OUTPUT PROBLEMS 72 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 29_tratamento_exceções.py
  Digite um número: 0
  Erro interno do sistema
○ (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> |
```

58 - Raise e assert

Criar arquivo e de nome



```
30_raise_assert.py
1 # Raise e assert, comandos para levantamento de exceções
2 # ValueError
3
4 raise ValueError("Você errou")
5
6
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 10 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 30_raise_assert.py
Traceback (most recent call last):
  File "C:\Users\Irene\Documents\aulaPython\Introdução_Python\30_raise_assert.py", line 3, in <module>
    raise ValueError("Você errou")
ValueError: Você errou
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
30_raise_assert.py 1, U X
30_raise_assert.py
1 # Raise e assert, comandos para levantamento de exceções
2 # ValueError, ZeroDivisionError
3
4 #raise ValueError("Você errou")
5 raise ZeroDivisionError("Você errou")
6
7
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 13 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 30_raise_assert.py
Traceback (most recent call last):
  File "C:\Users\Irene\Documents\aulaPython\Introdução_Python\30_raise_assert.py", line 3, in <module>
    raise ValueError("Você errou")
ValueError: Você errou
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 30_raise_assert.py
Traceback (most recent call last):
  File "C:\Users\Irene\Documents\aulaPython\Introdução_Python\30_raise_assert.py", line 5, in <module>
    raise ZeroDivisionError("Você errou")
ZeroDivisionError: Você errou
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
30_raise_assert.py
1 # Raise e assert, comandos para levantamento de exceções
2 # ValueError, ZeroDivisionError, OSError
3
4 #raise ValueError("Você errou")
5 #raise ZeroDivisionError("Você errou")
6 raise OSError("Você errou")
7
8
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 15 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 30_raise_assert.py
Traceback (most recent call last):
  File "C:\Users\Irene\Documents\aulaPython\Introdução_Python\30_raise_assert.py", line 6, in <module>
    raise OSError("Você errou")
OSError: Você errou
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
30_raise_assert.py
1 # Raise e assert, comandos para levantamento de exceções
2 # ValueError, ZeroDivisionError, OSError, KeyError
3
4 #raise ValueError("Você errou")
5 #raise ZeroDivisionError("Você errou")
6 #raise OSError("Você errou")
7 raise KeyError("Você errou")
8
9
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 17 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 30_raise_assert.py
Traceback (most recent call last):
  File "C:\Users\Irene\Documents\aulaPython\Introdução_Python\30_raise_assert.py", line 7, in <module>
    raise KeyError("Você errou")
KeyError: 'Você errou'
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
9 # Raise e assert, comandos para levantamento de exceções
10 # ValueError, ZeroDivisionError, OSError, KeyError
11
12 def soma(n1, n2):
13     if n1 < 0 or n2 < 0:
14         raise ValueError("n1 e n2 não podem ser negativos")
15     return n1 + n2
16
17 print(soma(2, 2))
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 23 DEBUG CONSOLE TERMINAL SQL CONSOLE GIT LENS

• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 30_raise_assert.py
4
○ (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> █

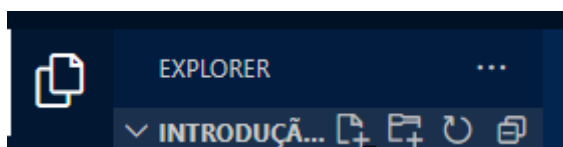
```
19
20 # Raise e assert, comandos para levantamento de exceções
21 # ValueError, ZeroDivisionError, OSError, KeyError
22 # assert
23
24 x = -2
25 assert x > 0, "Você errou"
26 print(x)
```

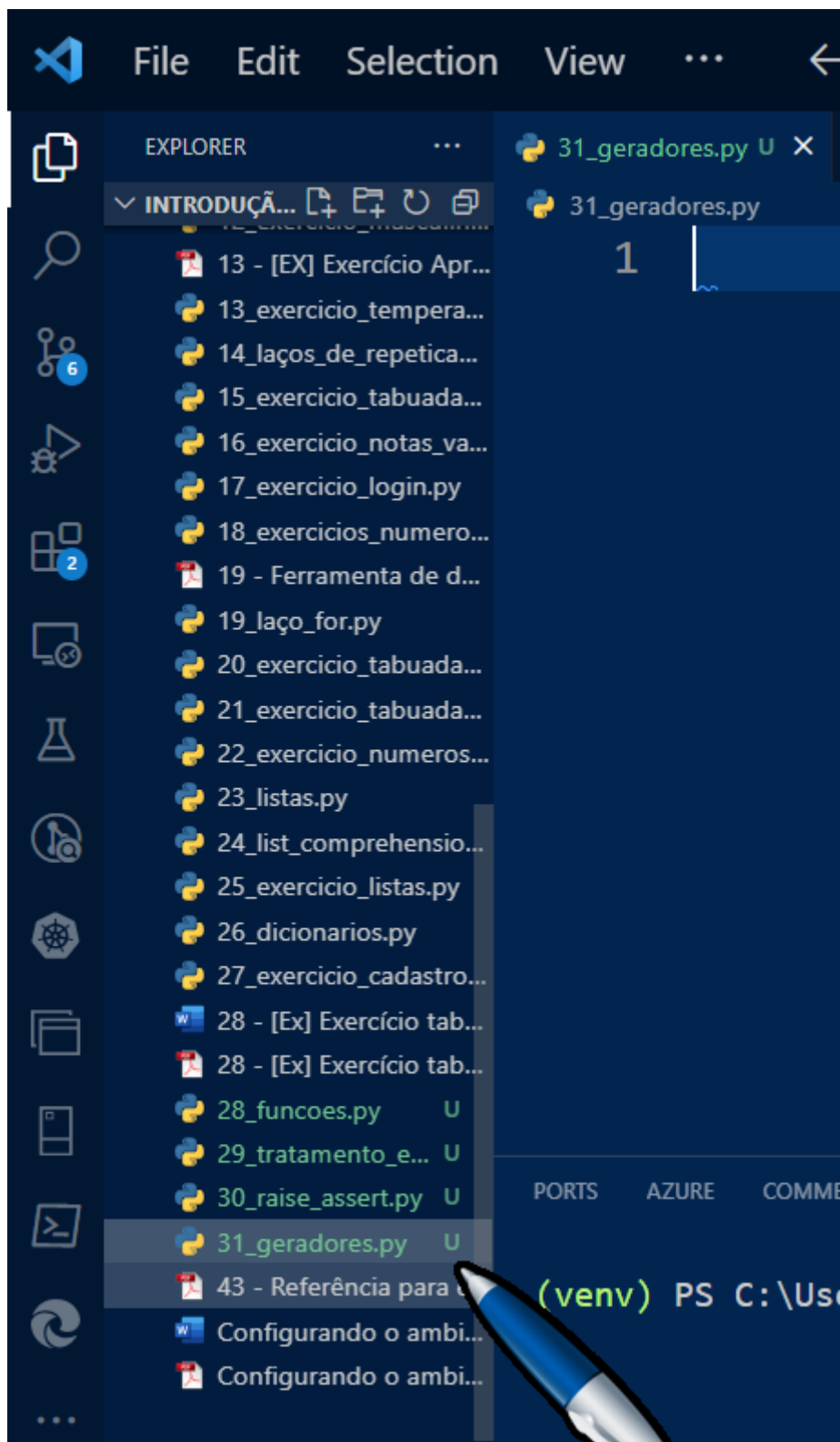
PORTS AZURE COMMENTS OUTPUT PROBLEMS 28 DEBUG CONSOLE TERMINAL SQL CONSOLE GIT LENS

• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 30_raise_assert.py
○ Traceback (most recent call last):
 File "C:\Users\Irene\Documents\aulaPython\Introdução_Python\30_raise_assert.py", line 25, in <module>
 assert x > 0, "Você errou"
 ^^^^^
AssertionError: Você errou
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> █

59 – Geradores

Criar um arquivo e de nome





Instale a Biblioteca, pympler = pip install pympler

31_geradores.py

```
1 #Geradores para evitar gastar memoria ram
2 #Instale a Biblioteca, pympler = pip install pympler
3
4 from pympler.asizeof import asizeof
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 18 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> pip install pympler
Collecting pympler
  Downloading Pympler-1.0.1-py3-none-any.whl (164 kB)
    164.8/164.8 kB 2.5 MB/s eta 0:00:00
Installing collected packages: pympler
Successfully installed pympler-1.0.1

[notice] A new release of pip available: 22.3.1 -> 23.3.2
[notice] To update, run: python.exe -m pip install --upgrade pip
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> 
```



31_geradores.py > ...

```
1 #Geradores para evitar gastar memoria ram
2 #Instale a Biblioteca, pympler = pip install pympler
3 import re
4 from pympler.asizeof import asizeof
5
6 def dobro(lista):
7     for i in lista:
8         yield i*2
9
10 def dobro2(lista):
11     lista_2 = []
12     for i in lista:
13         lista_2.append(i)
14     return lista_2
15
16 y = dobro2(range(0, 10000))
17 x = dobro(range(0, 10000))
18
19 print(asmizeof(y))
20 print(asmizeof(x))
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 24 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 31_geradores.py
405176
496
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> █
```



```
22
23 #Geradores para evitar gastar memoria ram
24 #Instale a Biblioteca, pympler = pip install pympler
25 #yield função geradora
26
27 from pympler.asizeof import asizeof
28
29 def dobro(lista):
30     for i in lista:
31         yield i*2
32
33 x = dobro(range(0, 100))
34
35 for i in x:
36     print(i)
```

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

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python 31_geradores.py

0
2
4
6
8
10
12
14

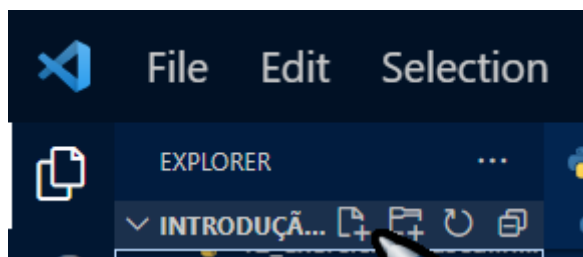
```
22
23 #Geradores para evitar gastar memoria ram
24 #Instale a Biblioteca, pympler = pip install pympler
25 #yield função geradora
26
27 from pympler.asizeof import asizeof
28
29 def dobro(lista):
30     for i in lista:
31         yield i*2
32
33 x = dobro(range(0, 100))
34
35 for i in x:
36     print(i)
```

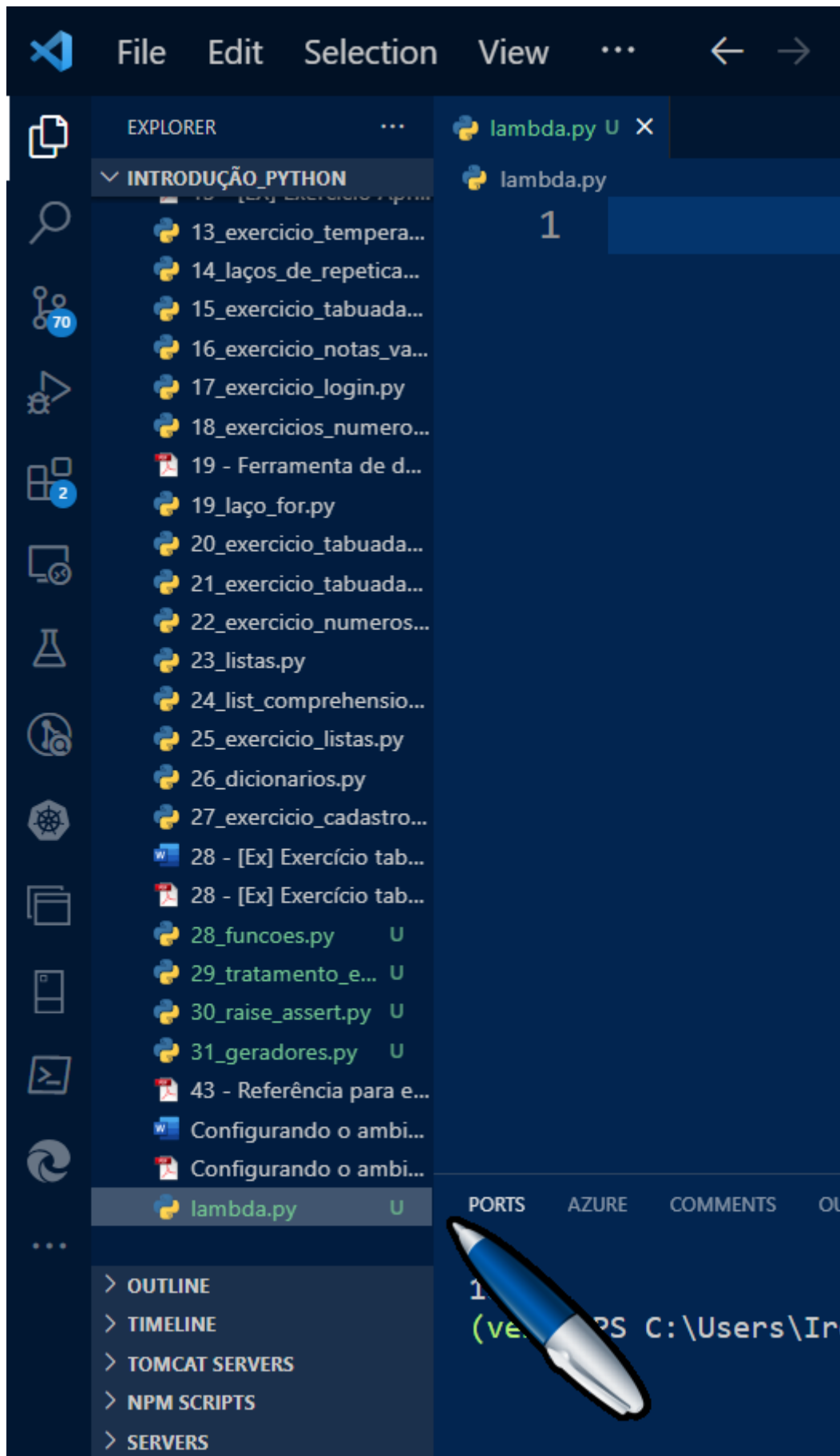
PORTS AZURE COMMENTS OUTPUT PROBLEMS 38 CONSOLE **TERMINAL** SQL CONSOLE GITLENS

184
186
188
190
192
194
196
198
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>

60 - Lambda - programação funcional

Criar arquivo e nome





```

1  #Lambda - programação funcional sempre retorna algo
2
3  def teste():
4      return lambda *idade: print(idade)
5
6  x = teste()
7
8  x('Irene', 'Agatha')
9
10

```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 10 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python lambda.py
('Irene', 'Agatha')
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>

```

61 - Filter - programação funcional

```

9
10 # Filter - programação funcional ,
11 x = [{'nome': 'Irene', 'idade': 54}, {'nome': 'Thalita', 'idade': 29}]
12
13 x = list(filter(lambda x: x['nome'] == 'Irene', x))
14
15 print(x)

```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 11 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python lambda.py
[{'nome': 'Irene', 'idade': 54}]
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>

```

62 - MAP programação funcional

```

17 # MAP programação funcional mudança em uma determinada lista,
18
19 x = [{'nome': 'Irene', 'idade': 20}, {'nome': 'Agatha', 'idade': 22}]
20
21 x = list(map(lambda x: {'nome': x['nome'], 'idade': 'menor que 30 anos' if x['idade'] < 30 else(x, x)}))
22
23 print(x)

```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 12 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS JavaScript Debug Terminal


```

(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python lambda.py
[{'nome': 'Irene', 'idade': 'menor que 30 anos'}, {'nome': 'Agatha', 'idade': 'menor que 30 anos'}]
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>

```

63 - Arquivos #1


```
283 arquivo = open('pessoas.txt', 'r')
284
285 resultados = arquivo.readlines()
286 x = []
287 for i in resultados:
288     x.append(i.split())
289
290 print(x[1][1])
291
292
293
294
```



PORTS AZURE COMMENTS OUTPUT PROBLEMS 143 DEBUG CONSOLE **TERMINAL** SQL CONSOLE GITLENS

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

```
282
283 arquivo = open('pessoas.txt', 'r')
284
285 resultados = arquivo.readlines()
286 x = []
287 for i in resultados:
288     x.append(i.split())
289
290 print(x[1][1])
291 arquivo.close()
292
293
294
```



You, 1 second ago • Uncommitted changes

PORTS AZURE COMMENTS OUTPUT PROBLEMS 143 DEBUG CONSOLE **TERMINAL** SQL CONSOLE GITLENS

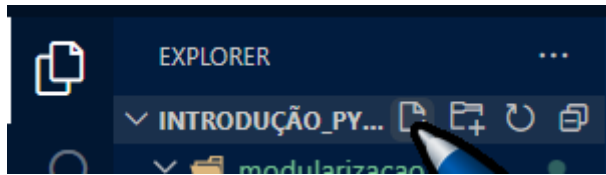
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> |

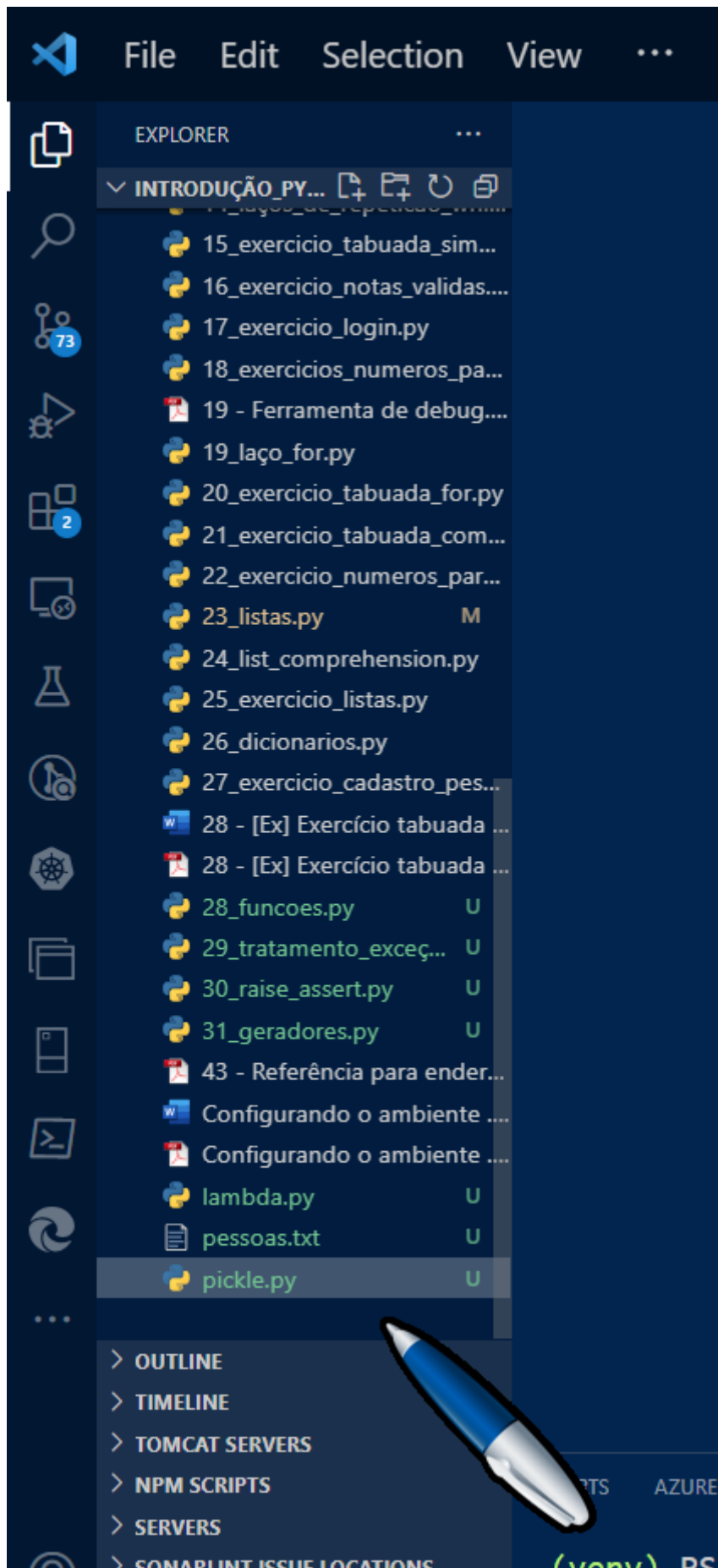
64 - Arquivos #2

```
with open('pessoas.txt', 'r') as arq:
    x = arq.read()
    print(x)
```

65 - Arquivos #3 Arquivos binário

Criar arquivo e nome





```
1 #import pickle,seriarizar objeto pegar algo da memoria,
2 #tornar persistente.
3
4 import pickle
5
6 '''x = 1
7 # print(type(x))
8 print(pickle.dumps(x))'''
9 #import pickle,seriarizar objeto pegar algo da memoria,
10 #tornar persistente.
11 x = [1,2,3,4]
12 #print(type(x))
13 print(pickle.dumps(x))
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 17 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

b'\x80\x04K\x01.'

- (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py
- b'\x80\x04\x95\r\x00\x00\x00\x00\x00\x00]\x94(K\x01K\x02K\x03K\x04e.'
- (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> █

```
14
15 #import pickle,seriarizar objeto pegar algo da memoria,
16 #tornar persistente.
17 x = [1,2,3,4]
18
19 string = pickle.dumps(x)
20 print(pickle.loads(string))
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 25 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS JavaScript Debug Terminal + ▮

- (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py
- [1, 2, 3, 4]
- (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> █
-


```
21 #import pickle,seriarizar objeto pegar algo da memoria,  
22 #tornar persistente.  
23 x = {'nome': 'irene','idade':20}  
24  
25 string = pickle.dumps(x)  
26 print(pickle.loads(string))
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 34 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py  
{'nome': 'irene', 'idade': 20}  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
21 #import pickle,seriarizar objeto pegar algo da memoria,  
22 #tornar persistente.  
23 x = {'nome': 'irene','idade':20}  
  
25 string = pickle.dumps(x)  
26 print(pickle.loads(string)['nome'])
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 34 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py  
{'nome': 'irene', 'idade': 20}  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py  
irene  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```
28 #import pickle,seriarizar objeto pegar algo da memoria,  
29 #tornar persistente.  
30  
31 x = [1,2,3,4,]  
32  
33 arq = open('arquivo.pk1', 'wb')  
34 pickle.dump(x, arq)  
35  
36 arq = open('arquivo.pk1', 'rb')  
37 retornou = pickle.load(arq)  
38  
39 print(retornou)
```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 34 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS JavaScript Debug Terminal + - [] ...

```
PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> & c:/Users/Irene/Documents/aulaPython/venv/Scripts/Activate.ps1  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py  
[1, 2, 3, 4]  
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>
```

```

41 import pickle
42 #import pickle,seriarizar objeto pegar algo da memoria,
43 #tornar persistente.
44
45 class Pessoa:
46     nome = 'Irene'
47     idade = 20
48
49
50 arq = open('arquivo.pk1', 'wb')
51 pickle.dump(Pessoa, arq)
52
53 arq = open('arquivo.pk1', 'rb')
54 retornou = pickle.load(arq)
55

```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 56 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```

• (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py
<class '__main__.Pessoa'>
○ (venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>

```

```

42 #import pickle,seriarizar objeto pegar algo da memoria,
43 #tornar persistente.
44
45 class Pessoa:
46     nome = 'Irene'
47     idade = 20
48
49
50 arq = open('arquivo.pk1', 'wb')
51 pickle.dump(Pessoa, arq)
52
53 arq = open('arquivo.pk1', 'rb')
54 retornou = pickle.load(arq)
55
56 print(retornou.idade)
57
58

```

PORTS AZURE COMMENTS OUTPUT PROBLEMS 57 DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```

<class '__main__.Pessoa'>
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py
Irene
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python> python pickle1.py
20
(venv) PS C:\Users\Irene\Documents\aulaPython\Introdução_Python>

```

