

A screenshot of a file explorer window. The top bar shows 'EXPLORER' and a search icon. The address bar displays 'D:'. The main area shows a list of files and folders. Files include 'atividade casa 02-03.py', 'atividade casa 08-03.py', 'atividade dia 15-03.py', 'atividade dia 16-03.py', 'atividade dia 23-02.py', 'atividade fatorial 16-02.py', 'atividade idade 09-02.py', 'atividade imc 16-03.py', 'atividade lista_fatorial 16-...', 'atividade numero primo 2...', 'atividade_idade_peso 16-0...', 'exemplo lista e for.py', 'raiz_quadrada.py', and 'testes.py'. There are two expanded folders: 'lista 1' containing files 'atividade 1 lista.py' through 'atividade 8 lista.py', and 'lista 2' containing files 'atividade 1.py' through 'atividade 7.py'. At the bottom, there are collapsed folders 'WindowsApps' and 'WpSystem', and an 'OUTLINE' pane on the left.

atividade 7.py • atividade 1.py ✕

visual code > lista 2 > atividade 1.py > ...

```

1  # 1) Escreva um programa que leia a idade de 10 pessoas e armazene-as em uma lista.
2  # Calcule e mostre:
3  # a) a menor idade
4  # b) a média das idades
5  # c) a quantidade de pessoas que tem idade entre 20 e 30 anos (inclusive)
6  # d) a quantidade de pessoas com idade maior que a média
7
8  idade=[]
9  total=0
10 idademaior=0
11 for i in range(10):
12     idade.append(int(input("coloque sua idade: ")))
13     total+=idade[i]
14     if idade[i] >= 20 and idade[i] <= 30:
15         idademaior+=1
16 media=total/10
17 pessoasmedia=0
18 for i in range (10):
19     if idade[i] > media:
20         pessoasmedia+=1
21 print(min(idade),"é a menor idade")
22 print("a media de idade é de", media)
23 print("a quantidade de pessoas com idade maior que 20 e menor que 30 é de", idademaior)
24 print("a quantidade de pessoas com idade maior que a media é", pessoasmedia)

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell

Copyright (C) Microsoft Corporation. Todos os direitos reservados.

2: Python

FileEditSelectionViewGoRunTerminalHelp

atividade 1.py - d: - Visual Studio Code

EXPLORER

▼ D:

atividade casa 02-03.py

atividade casa 08-03.py

atividade dia 15-03.py

atividade dia 16-03.py

atividade dia 23-02.py

atividade fatorial 16-02.py

atividade idade 09-02.py

atividade imc 16-03.py

atividade lista_fatorial 16-...

atividade numero primo 2...

atividade_idade_peso 16-0...

exemplo lista e for.py

raiz_quadrada.py

testes.py

▼ lista 1

atividade 1 lista.py

atividade 2 lista.py

atividade 3 lista.py

atividade 4 lista.py

atividade 5 lista.py

atividade 6 lista.py

atividade 7 lista.py

atividade 8 lista.py

▼ lista 2

atividade 1.py

atividade 2.py

atividade 3.py

atividade 4.py

atividade 5.py

atividade 6.py

atividade 7.py

> WindowsApps

> WpSystem

> OUTLINE

atividade 7.py

atividade 1.py X

visual code > lista 2 > atividade 1.py > ...

```
1  # 1) Escreva um programa que leia a idade de 10 pessoas e armazene-as em uma lista.
2  # Calcule e mostre:
3  # a) a menor idade
4  # b) a média das idades
5  # c) a quantidade de pessoas que tem idade entre 20 e 30 anos (inclusive)
6  # d) a quantidade de pessoas com idade maior que a média
7
8  idade=[]
9  total=0
10 idademaior=0
11 for i in range(10):
12     idade.append(int(input("coloque sua idade: ")))
13     total+=idade[i]
14     if idade[i] >= 20 and idade[i] <= 30:
15         idademaior+=1
16 media=total/10
17 pessoasmedia=0
18 for i in range (10):
19     if idade[i] > media:
20         pessoasmedia+=1
21 print(min(idade),"é a menor idade")
22 print("a media de idade é de", media)
23 print("a quantidade de pessoas com idade maior que 20 e menor que 30 é de", idademaior)
24 print("a quantidade de pessoas com idade maior que a media é", pessoasmedia)
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

2: Python

coloque sua idade: 10
coloque sua idade: 20
coloque sua idade: 30
coloque sua idade: 40
coloque sua idade: 50
coloque sua idade: 60
coloque sua idade: 70
coloque sua idade: 80
coloque sua idade: 90
coloque sua idade: 100
10 é a menor idade
a media de idade é de 55.0
a quantidade de pessoas com idade maior que 20 e menor que 30 é de 2
a quantidade de pessoas com idade maior que a media é 5
PS D:\>

Python 3.9.1 64-bit

0 0

Ln 24, Col 77

Spaces: 4

UTF-8

CRLF

Python

13:56

18/03/2021

File

Edit

Selection

View

Go

Run

Terminal

Help

atividade 2.py - d: - Visual Studio Code

EXPLORER

...

▼ D:

atividade 16-02 numero_p...

atividade casa 02-03.py

atividade casa 08-03.py

atividade dia 15-03.py

atividade dia 16-03.py

atividade dia 23-02.py

atividade fatorial 16-02.py

atividade idade 09-02.py

atividade imc 16-03.py

atividade lista_fatorial 16-...

atividade numero primo 2...

atividade_idade_peso 16-0...

exemplo lista e for.py

raiz_quadrada.py

testes.py

▼ lista 1

atividade 1 lista.py

atividade 2 lista.py

atividade 3 lista.py

atividade 4 lista.py

atividade 5 lista.py

atividade 6 lista.py

atividade 7 lista.py

atividade 8 lista.py

▼ lista 2

atividade 1.py

atividade 2.py

atividade 3.py

atividade 4.py

atividade 5.py

atividade 6.py

atividade 7.py

> WindowsApps

> OUTLINE

visual code > lista 2 > atividade 2.py > ...

1

2) Faça um programa que preencha uma lista com 10 cores diferentes. Depois permita fazer

2

uma pesquisa se uma determinada cor existe armazenada na lista, se existir deve ser

3

impresso na tela a cor e em qual posição (índice) esta cor está armazenada. A pesquisa

4

deve ser feita até que seja digitado FIM na cor a ser pesquisada na lista.

5

6

cores=[]

7

for i in range(10):

8

cores.append(str(input("Coloque uma cor: ")))

9

cont=""

10

while cont.upper() != "FIM":

11

buscar=(str(input("qual a cor desejada? ")))

12

pos=0

13

for i in range(10):

14

if buscar == cores[i]:

15

buscar=cores[i]

16

pos=i

17

print("a cor é:",buscar,"a posição na lista é:",pos)

18

cont=(str(input("deseja continuar? n/ para finalizar digite FIM: ")))

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

1: Python

+

^

×

Windows PowerShell

Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Python 3.9.1 64-bit

0 0

Ln 18, Col 74

Spaces: 4

UTF-8

CRLF

Python

14:03

18/03/2021

FileEditSelectionViewGoRunTerminalHelp

atividade 2.py - d: - Visual Studio Code

EXPLORER

▼ D:

- atividade 16-02 numero_p...
- atividade casa 02-03.py
- atividade casa 08-03.py
- atividade dia 15-03.py
- atividade dia 16-03.py
- atividade dia 23-02.py
- atividade fatorial 16-02.py
- atividade idade 09-02.py
- atividade imc 16-03.py
- atividade lista_fatorial 16-...
- atividade numero primo 2...
- atividade_idade_peso 16-0...
- exemplo lista e for.py
- raiz_quadrada.py
- testes.py

▼ lista 1

- atividade 1 lista.py
- atividade 2 lista.py
- atividade 3 lista.py
- atividade 4 lista.py
- atividade 5 lista.py
- atividade 6 lista.py
- atividade 7 lista.py
- atividade 8 lista.py

▼ lista 2

- atividade 1.py
- atividade 2.py
- atividade 3.py
- atividade 4.py
- atividade 5.py
- atividade 6.py
- atividade 7.py

> WindowsApps

> OUTLINE

atividade 7.py

atividade 2.py

atividade dia 15-03.py

visual code > lista 2 > atividade 2.py > ...

```
1 # 2) Faça um programa que preencha uma lista com 10 cores diferentes. Depois permita fazer
2 # uma pesquisa se uma determinada cor existe armazenada na lista, se existir deve ser
3 # impresso na tela a cor e em qual posição (índice) esta cor está armazenada. A pesquisa
4 # deve ser feita até que seja digitado FIM na cor a ser pesquisada na lista.
5
6 cores=[]
7 for i in range(10):
8     cores.append(str(input("Coloque uma cor: ")))
9 cont=""
10 while cont.upper() != "FIM":
11     buscar=(str(input("qual a cor desejada? ")))
12     pos=0
13     for i in range(10):
14         if buscar == cores[i]:
15             buscar=cores[i]
16             pos=i
17     print("a cor é:",buscar,"a posição na lista é:",pos)
18     cont=(str(input("deseja continuar? n/ para finalizar digite FIM: ")))
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

1: Python

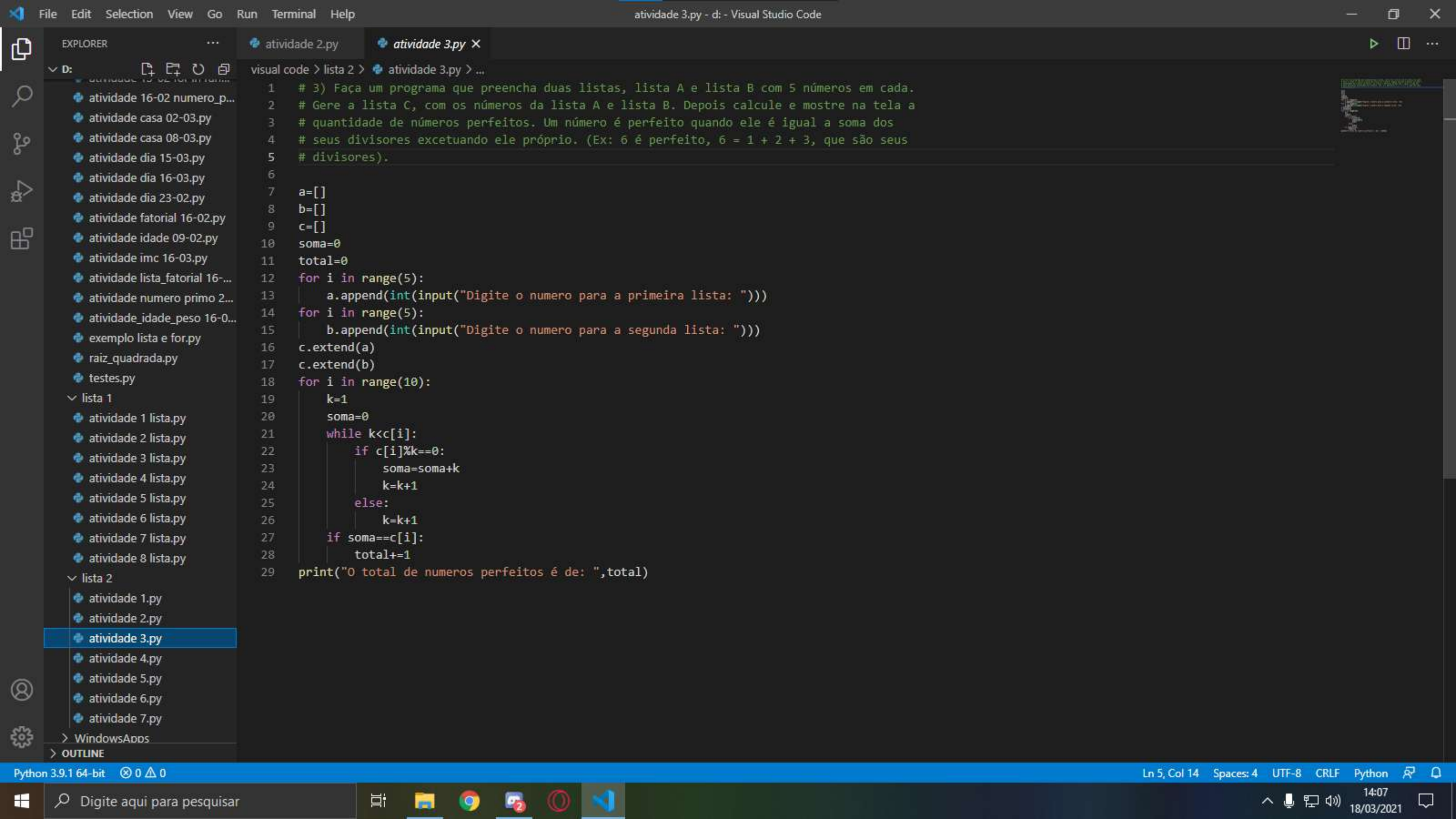
Coloque uma cor: amarelo
Coloque uma cor: vermelho
Coloque uma cor: verde
Coloque uma cor: marrom
Coloque uma cor: cinza
Coloque uma cor: roxo
Coloque uma cor: azul
Coloque uma cor: branco
Coloque uma cor: preto
Coloque uma cor: rosa
qual a cor desejada? branco
a cor é: branco a posição na lista é: 7
deseja continuar? n/ para finalizar digite FIM: sim
qual a cor desejada? verde
a cor é: verde a posição na lista é: 2
deseja continuar? n/ para finalizar digite FIM: fim
PS D:\>

Python 3.9.1 64-bit 0 0 0

Ln 18, Col 74 Spaces: 4 UTF-8 CRLF Python

Digite aqui para pesquisar

14:05 18/03/2021



```
1  # 3) Faça um programa que preencha duas listas, lista A e lista B com 5 números em cada.
2  # Gere a lista C, com os números da lista A e lista B. Depois calcule e mostre na tela a
3  # quantidade de números perfeitos. Um número é perfeito quando ele é igual a soma dos
4  # seus divisores excetuando ele próprio. (Ex: 6 é perfeito, 6 = 1 + 2 + 3, que são seus
5  # divisores).
6
7  a=[]
8  b=[]
9  c=[]
10 soma=0
11 total=0
12 for i in range(5):
13     a.append(int(input("Digite o numero para a primeira lista: ")))
14 for i in range(5):
15     b.append(int(input("Digite o numero para a segunda lista: ")))
16 c.extend(a)
17 c.extend(b)
18 for i in range(10):
19     k=1
20     soma=0
21     while k<c[i]:
22         if c[i]%k==0:
23             soma=soma+k
24             k=k+1
25         else:
26             k=k+1
27     if soma==c[i]:
28         total+=1
29 print("O total de numeros perfeitos é de: ",total)
```

FileEditSelectionViewGoRunTerminalHelp

atividade 3.py - d: - Visual Studio Code

EXPLORER

▼ D:
atividade 16-02 numero_p...
atividade casa 02-03.py
atividade casa 08-03.py
atividade dia 15-03.py
atividade dia 16-03.py
atividade dia 23-02.py
atividade fatorial 16-02.py
atividade idade 09-02.py
atividade imc 16-03.py
atividade lista_fatorial 16-...
atividade numero primo 2...
atividade_idade_peso 16-0...
exemplo lista e for.py
raiz_quadrada.py
testes.py
▼ lista 1
atividade 1 lista.py
atividade 2 lista.py
atividade 3 lista.py
atividade 4 lista.py
atividade 5 lista.py
atividade 6 lista.py
atividade 7 lista.py
atividade 8 lista.py
▼ lista 2
atividade 1.py
atividade 2.py
atividade 3.py
atividade 4.py
atividade 5.py
atividade 6.py
atividade 7.py
> WindowsApps
> OUTLINE

atividade 2.py

atividade 3.py X

visual code > lista 2 > atividade 3.py > ...

```
1 # 3) Faça um programa que preencha duas listas, lista A e lista B com 5 números em cada.  
2 # Gere a lista C, com os números da lista A e lista B. Depois calcule e mostre na tela a  
3 # quantidade de números perfeitos. Um número é perfeito quando ele é igual a soma dos  
4 # seus divisores excetuando ele próprio. (Ex: 6 é perfeito, 6 = 1 + 2 + 3, que são seus  
5 # divisores).  
6  
7 a=[]  
8 b=[]  
9 c=[]  
10 soma=0  
11 total=0  
12 for i in range(5):  
13     a.append(int(input("Digite o numero para a primeira lista: ")))  
14 for i in range(5):  
15     b.append(int(input("Digite o numero para a segunda lista: ")))  
16 c.extend(a)  
17 c.extend(b)  
18 for i in range(10):  
19     k=1  
20     soma=0  
21     while k<c[i]:  
22         if c[i]%k==0:  
23             soma=soma+k  
24             k=k+1  
25         else:  
26             k=k+1
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

1: Python

Digite o numero para a primeira lista: 1
Digite o numero para a primeira lista: 2
Digite o numero para a primeira lista: 3
Digite o numero para a primeira lista: 4
Digite o numero para a primeira lista: 28
Digite o numero para a segunda lista: 5
Digite o numero para a segunda lista: 6
Digite o numero para a segunda lista: 7
Digite o numero para a segunda lista: 8
Digite o numero para a segunda lista: 9
0 total de numeros perfeitos é de: 2
PS D:\>

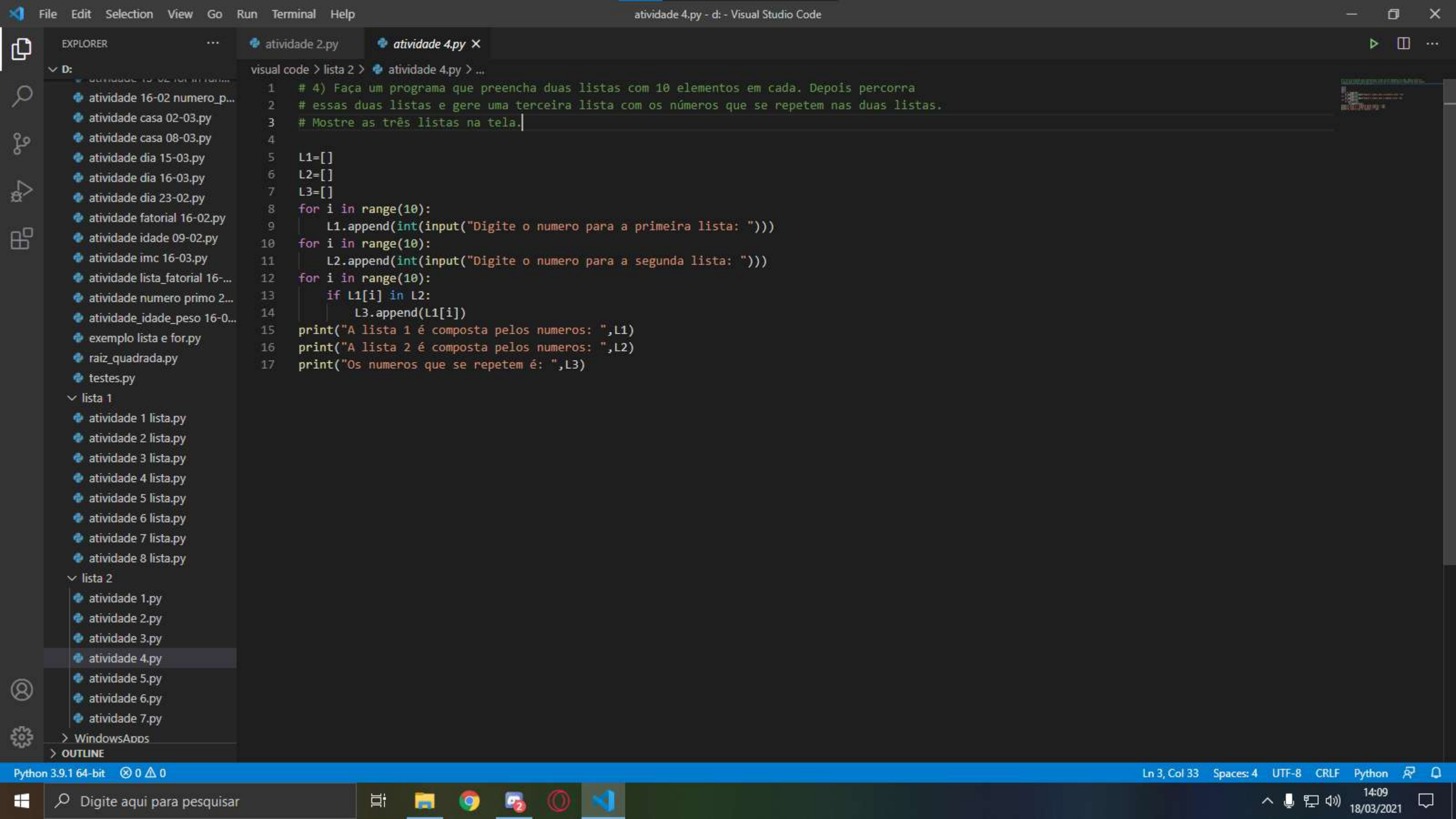
Python 3.9.1 64-bit

0 0

Ln 5, Col 14 Spaces: 4 UTF-8 CRLF Python

Digite aqui para pesquisar

14:07 18/03/2021



EXPLORER

D:

- atividade 16-02 numero_p...
- atividade casa 02-03.py
- atividade casa 08-03.py
- atividade dia 15-03.py
- atividade dia 16-03.py
- atividade dia 23-02.py
- atividade fatorial 16-02.py
- atividade idade 09-02.py
- atividade imc 16-03.py
- atividade lista_fatorial 16-...
- atividade numero primo 2...
- atividade_idade_peso 16-0...
- exemplo lista e for.py
- raiz_quadrada.py
- testes.py

lista 1

- atividade 1 lista.py
- atividade 2 lista.py
- atividade 3 lista.py
- atividade 4 lista.py
- atividade 5 lista.py
- atividade 6 lista.py
- atividade 7 lista.py
- atividade 8 lista.py

lista 2

- atividade 1.py
- atividade 2.py
- atividade 3.py
- atividade 4.py
- atividade 5.py
- atividade 6.py
- atividade 7.py

> WindowsApps

> OUTLINE

atividade 2.py

atividade 4.py X

visual code > lista 2 > atividade 4.py > ...

```
1 # 4) Faça um programa que preencha duas listas com 10 elementos em cada. Depois percorra
2 # essas duas listas e gere uma terceira lista com os números que se repetem nas duas listas.
3 # Mostre as três listas na tela.
4
5 L1=[]
6 L2=[]
7 L3=[]
8 for i in range(10):
9     L1.append(int(input("Digite o numero para a primeira lista: ")))
10 for i in range(10):
11     L2.append(int(input("Digite o numero para a segunda lista: ")))
12 for i in range(10):
13     if L1[i] in L2:
14         L3.append(L1[i])
15 print("A lista 1 é composta pelos numeros: ",L1)
16 print("A lista 2 é composta pelos numeros: ",L2)
17 print("Os numeros que se repetem é: ",L3)
```

FileEditSelectionViewGoRunTerminalHelp

atividade 4.py - d: - Visual Studio Code

EXPLORER

▼ D:

atividade 16-02 numero_p...

atividade casa 02-03.py

atividade casa 08-03.py

atividade dia 15-03.py

atividade dia 16-03.py

atividade dia 23-02.py

atividade fatorial 16-02.py

atividade idade 09-02.py

atividade imc 16-03.py

atividade lista_fatorial 16-...

atividade numero primo 2...

atividade_idade_peso 16-0...

exemplo lista e for.py

raiz_quadrada.py

testes.py

▼ lista 1

atividade 1 lista.py

atividade 2 lista.py

atividade 3 lista.py

atividade 4 lista.py

atividade 5 lista.py

atividade 6 lista.py

atividade 7 lista.py

atividade 8 lista.py

▼ lista 2

atividade 1.py

atividade 2.py

atividade 3.py

atividade 4.py

atividade 5.py

atividade 6.py

atividade 7.py

> WindowsApps

> OUTLINE

atividade 2.py

atividade 4.py X

visual code > lista 2 > atividade 4.py > ...

1# 4) Faça um programa que preencha duas listas com 10 elementos em cada. Depois percorra

2# essas duas listas e gere uma terceira lista com os números que se repetem nas duas listas.

3# Mostre as três listas na tela.

4

5L1=[]

6L2=[]

7L3=[]

8for i in range(10):

9| L1.append(int(input("Digite o numero para a primeira lista: ")))

10for i in range(10):

11| L2.append(int(input("Digite o numero para a segunda lista: ")))

12for i in range(10):

13| if L1[i] in L2:

14| | L3.append(L1[i])

15print("A lista 1 é composta pelos numeros: ",L1)

16print("A lista 2 é composta pelos numeros: " ,L2)

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

1: Python

Digite o numero para a primeira lista: 1

Digite o numero para a primeira lista: 2

Digite o numero para a primeira lista: 3

Digite o numero para a primeira lista: 4

Digite o numero para a primeira lista: 5

Digite o numero para a primeira lista: 6

Digite o numero para a primeira lista: 7

Digite o numero para a primeira lista: 8

Digite o numero para a primeira lista: 9

Digite o numero para a primeira lista: 10

Digite o numero para a segunda lista: 5

Digite o numero para a segunda lista: 6

Digite o numero para a segunda lista: 7

Digite o numero para a segunda lista: 8

Digite o numero para a segunda lista: 9

Digite o numero para a segunda lista: 10

Digite o numero para a segunda lista: 11

Digite o numero para a segunda lista: 12

Digite o numero para a segunda lista: 13

Digite o numero para a segunda lista: 14

A lista 1 é composta pelos numeros: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

A lista 2 é composta pelos numeros: [5, 6, 7, 8, 9, 10, 11, 12, 13, 14]

Os numeros que se repetem é: [5, 6, 7, 8, 9, 10]

PS D:\>

Python 3.9.1 64-bit

0 0

Ln 17, Col 42

Spaces: 4

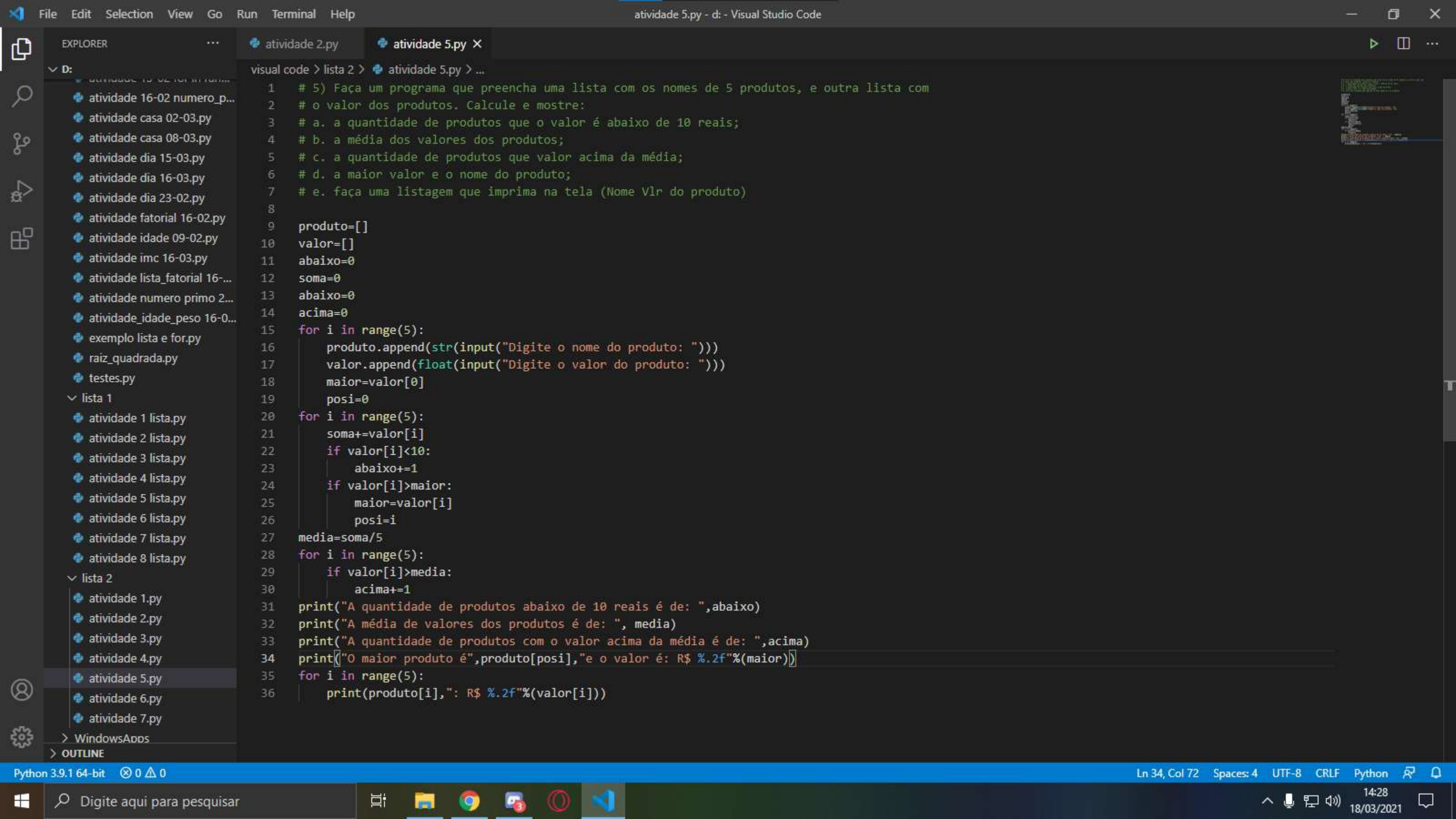
UTF-8

CRLF

Python

14:10

18/03/2021



visual code > lista 2 > atividade 5.py > ...

```
1  # 5) Faça um programa que preencha uma lista com os nomes de 5 produtos, e outra lista com
2  # o valor dos produtos. Calcule e mostre:
3  # a. a quantidade de produtos que o valor é abaixo de 10 reais;
4  # b. a média dos valores dos produtos;
5  # c. a quantidade de produtos que valor acima da média;
6  # d. a maior valor e o nome do produto;
7  # e. faça uma listagem que imprima na tela (Nome Vlr do produto)
8
9  produto=[]
10 valor=[]
11 abaixo=0
12 soma=0
13 abaixo=0
14 acima=0
15 for i in range(5):
16     produto.append(str(input("Digite o nome do produto: ")))
17     valor.append(float(input("Digite o valor do produto: ")))
18     maior=valor[0]
19     posi=0
20 for i in range(5):
21     soma+=valor[i]
22     if valor[i]<10:
23         abaixo+=1
24     if valor[i]>maior:
25         maior=valor[i]
26         posi=i
27 media=soma/5
28 for i in range(5):
29     if valor[i]>media:
30         acima+=1
31 print("A quantidade de produtos abaixo de 10 reais é de: ",abaixo)
32 print("A média de valores dos produtos é de: ", media)
33 print("A quantidade de produtos com o valor acima da média é de: ",acima)
34 print("O maior produto é",produto[posi],"e o valor é: R$ %.2f"%(maior))
35 for i in range(5):
36     print(produto[i],": R$ %.2f"%(valor[i]))
```


EXPLORER

atividade 16-02 numero p...

atividade casa 08-03.py

- atividade dia 16-03.py
- atividade dia 22-03

atividade idade 09-02.py

atividade lista fatorial

atividade idade peso 16-0...

- raiz quadrada py

▼ lista 1

- atividade 2 lista.py

- atividade 3 lista.py
- atividade 4 lista.py

- atividade 5 lista.py
- atividade 6 lista.py

- atividade 7 lista.py
- atividade 8 lista.py

7 p311

- atividade 2.py
- atividade 3.py

atividade 4.py

atividade 6.py

OUTLINE

3.5.1.04 DR

atividade 2.py

atividade 5.py X

visual code > lista 2 > atividade 5.py > ...

```

1  # 5) Faça um programa que preencha uma lista com os nomes de 5 produtos, e outra lista com
2  # o valor dos produtos. Calcule e mostre:
3  # a. a quantidade de produtos que o valor é abaixo de 10 reais;
4  # b. a média dos valores dos produtos;
5  # c. a quantidade de produtos que valor acima da média;
6  # d. a maior valor e o nome do produto;
7  # e. faça uma listagem que imprima na tela (Nome Vlr do produto)
8
9  produto=[]
10 valor=[]
11 abaixo=0
12 soma=0
13 abaixo=0
14 acima=0
15 for i in range(5):
16     produto.append(str(input("Digite o nome do produto: ")))
17     valor.append(float(input("Digite o valor do produto: ")))
18     maior=valor[0]
19     posi=0

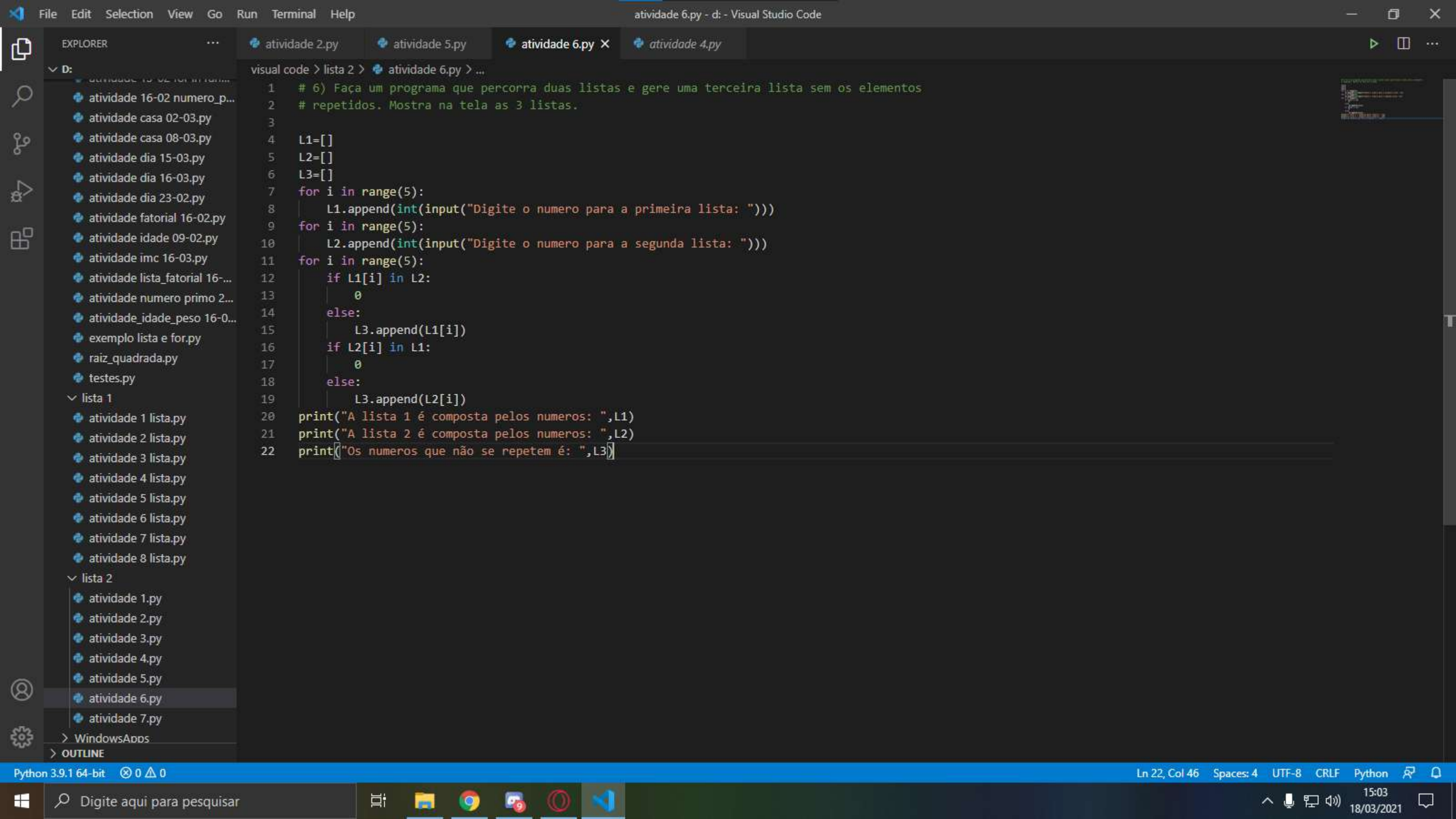
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Digite o nome do produto: coca cola lata
Digite o valor do produto: 4.50
Digite o nome do produto: macarrão
Digite o valor do produto: 5
Digite o nome do produto: bolacha
Digite o valor do produto: 3
Digite o nome do produto: leite
Digite o valor do produto: 3
Digite o nome do produto: salgadinho
Digite o valor do produto: 5
A quantidade de produtos abaixo de 10 reais é de: 5
A média de valores dos produtos é de: 4.1
A quantidade de produtos com o valor acima da média é de: 3
O maior produto é macarrão e o valor é: R$ 5.00
coca cola lata : R$ 4.50
macarrão : R$ 5.00
bolacha : R$ 3.00
leite : R$ 3.00
salgadinho : R$ 5.00
PS D:\>
```

1: Python

+



EXPLORER

▼ D:

- atividade 16-02 numero_p...
- atividade casa 02-03.py
- atividade casa 08-03.py
- atividade dia 15-03.py
- atividade dia 16-03.py
- atividade dia 23-02.py
- atividade fatorial 16-02.py
- atividade idade 09-02.py
- atividade imc 16-03.py
- atividade lista_fatorial 16-...
- atividade numero primo 2...
- atividade_idade_peso 16-0...
- exemplo lista e for.py
- raiz_quadrada.py
- testes.py
- ▼ lista 1
 - atividade 1 lista.py
 - atividade 2 lista.py
 - atividade 3 lista.py
 - atividade 4 lista.py
 - atividade 5 lista.py
 - atividade 6 lista.py
 - atividade 7 lista.py
 - atividade 8 lista.py
- ▼ lista 2
 - atividade 1.py
 - atividade 2.py
 - atividade 3.py
 - atividade 4.py
 - atividade 5.py
 - atividade 6.py
 - atividade 7.py
- > WindowsApps
- > OUTLINE

atividade 2.py

atividade 5.py

atividade 6.py X

atividade 4.py

visual code > lista 2 > atividade 6.py > ...

```
1 # 6) Faça um programa que percorra duas listas e gere uma terceira lista sem os elementos
2 # repetidos. Mostra na tela as 3 listas.
3
4 L1=[]
5 L2=[]
6 L3=[]
7 for i in range(5):
8     L1.append(int(input("Digite o numero para a primeira lista: ")))
9 for i in range(5):
10    L2.append(int(input("Digite o numero para a segunda lista: ")))
11 for i in range(5):
12     if L1[i] in L2:
13         0
14     else:
15         L3.append(L1[i])
16     if L2[i] in L1:
17         0
18     else:
19         L3.append(L2[i])
20 print("A lista 1 é composta pelos numeros: ",L1)
21 print("A lista 2 é composta pelos numeros: ",L2)
22 print("Os numeros que não se repetem é: ",L3)
```



FileEditSelectionViewGoRunTerminalHelp

atividade 6.py - d: - Visual Studio Code

EXPLORER

▼ D:
atividade 16-02 numero_p...
atividade casa 02-03.py
atividade casa 08-03.py
atividade dia 15-03.py
atividade dia 16-03.py
atividade dia 23-02.py
atividade fatorial 16-02.py
atividade idade 09-02.py
atividade imc 16-03.py
atividade lista_fatorial 16-...
atividade numero primo 2...
atividade_idade_peso 16-0...
exemplo lista e for.py
raiz_quadrada.py
testes.py
▼ lista 1
atividade 1 lista.py
atividade 2 lista.py
atividade 3 lista.py
atividade 4 lista.py
atividade 5 lista.py
atividade 6 lista.py
atividade 7 lista.py
atividade 8 lista.py
▼ lista 2
atividade 1.py
atividade 2.py
atividade 3.py
atividade 4.py
atividade 5.py
atividade 6.py
atividade 7.py
> WindowsApps
> OUTLINE

atividade 2.py

atividade 5.py

atividade 6.py X

atividade 4.py

visual code > lista 2 > atividade 6.py > ...
1 # 6) Faça um programa que percorra duas listas e gere uma terceira lista sem os elementos
2 # repetidos. Mostra na tela as 3 listas.
3
4 L1=[]
5 L2=[]
6 L3=[]
7 for i in range(5):
8 | L1.append(int(input("Digite o numero para a primeira lista: ")))
9 for i in range(5):
10 | L2.append(int(input("Digite o numero para a segunda lista: ")))
11 for i in range(5):
12 | if L1[i] in L2:
13 | | 0
14 | else:
15 | | L3.append(L1[i])
16 | if L2[i] in L1:
17 | | 0
18 | else:
19 | | L3.append(L2[i])
20 print("A lista 1 é composta pelos numeros: ",L1)
21 print("A lista 2 é composta pelos numeros: ",L2)
22 print("Os numeros que não se repetem é: ",L3)

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

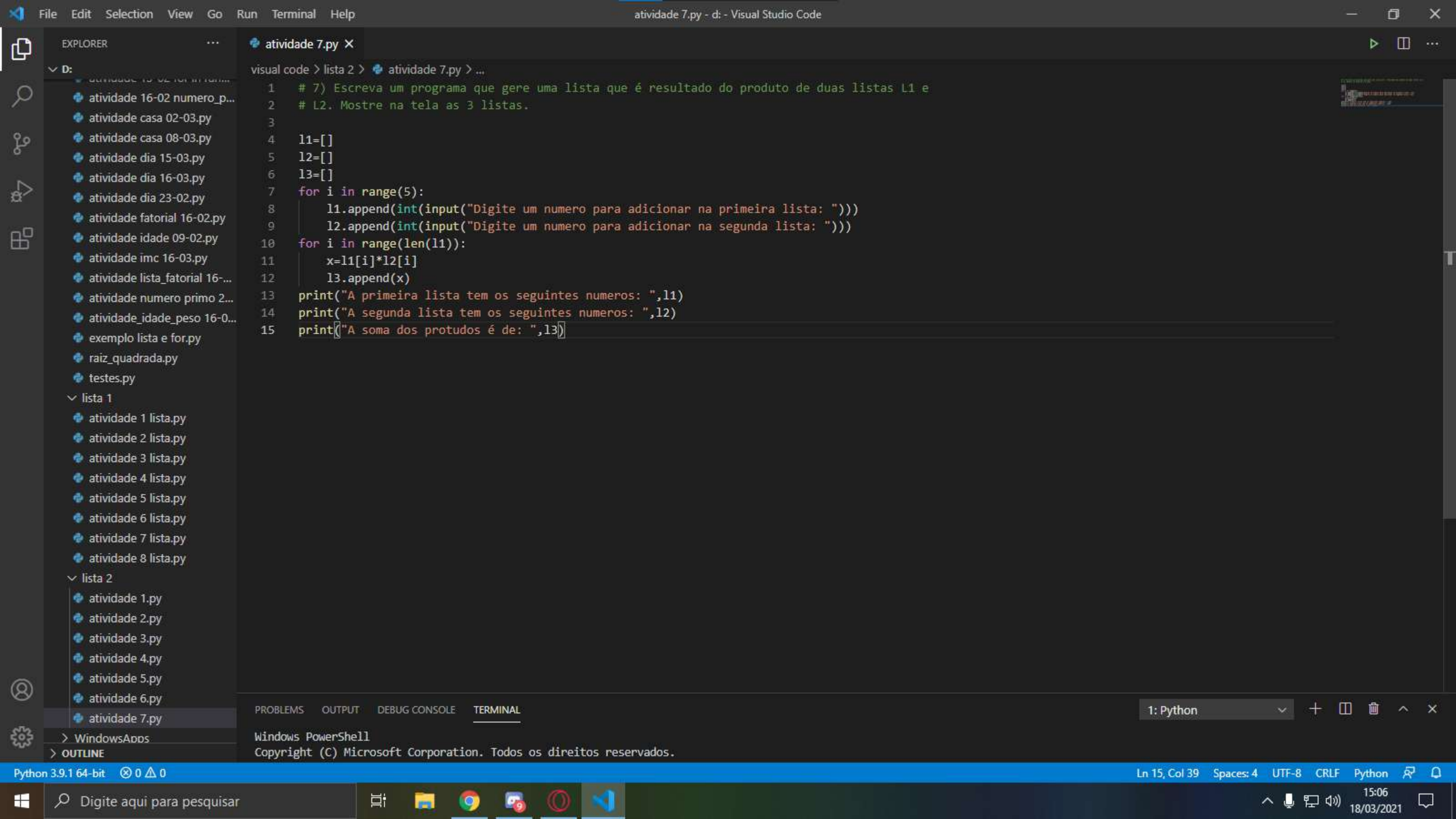
1: Python
Digite o numero para a primeira lista: 1
Digite o numero para a primeira lista: 2
Digite o numero para a primeira lista: 3
Digite o numero para a primeira lista: 4
Digite o numero para a primeira lista: 5
Digite o numero para a segunda lista: 3
Digite o numero para a segunda lista: 4
Digite o numero para a segunda lista: 5
Digite o numero para a segunda lista: 6
Digite o numero para a segunda lista: 7
A lista 1 é composta pelos numeros: [1, 2, 3, 4, 5]
A lista 2 é composta pelos numeros: [3, 4, 5, 6, 7]
Os numeros que não se repetem é: [1, 2, 6, 7]
PS D:\>

Python 3.9.1 64-bit

0 0

Ln 22, Col 46 Spaces: 4 UTF-8 CRLF Python

15:04 18/03/2021



EXPLORER

atividade 16-02 numero_p...

atividade casa 08-03.py

atividade dia 16-03.py

- atividade lateral 10-20py
- atividade idada 00-02 ny

atividade lista_fatorial

atividade_idade_peso 16-0...

- `testes.py`

atividade 1 lista.py

atividade 3 lista.py

- atividade 5 lista.py
- atividade 6 lista.py


atividade 8 lista.py

atividade 1.py

- atividade 3.py
- atividade 4.py

atividade 6.py

> WindowsApps

 Digite aqui para pesquisar

atividade 7.py X

visual code > lista 2 > atividade 7.py > ...

```
1 # 7) Escreva um programa que gere uma lista que é resultado do produto de duas listas L1 e
2 # L2. Mostre na tela as 3 listas.
3
4 l1=[]
5 l2=[]
6 l3=[]
7 for i in range(5):
8     l1.append(int(input("Digite um numero para adicionar na primeira lista: ")))
9     l2.append(int(input("Digite um numero para adicionar na segunda lista: ")))
10 for i in range(len(l1)):
11     x=l1[i]*l2[i]
12     l3.append(x)
13 print("A primeira lista tem os seguintes numeros: ",l1)
14 print("A segunda lista tem os seguintes numeros: ",l2)
15 print("A soma dos produtos é de: ",l3)
```

1. **Identify the problem.** The first step in the problem-solving process is to identify the problem. This involves recognizing the symptoms of the problem and determining the underlying cause.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: Python

```
Digite um numero para adicionar na primeira lista: 1
Digite um numero para adicionar na segunda lista: 2
Digite um numero para adicionar na primeira lista: 3
Digite um numero para adicionar na segunda lista: 4
Digite um numero para adicionar na primeira lista: 5
Digite um numero para adicionar na segunda lista: 6
Digite um numero para adicionar na primeira lista: 7
Digite um numero para adicionar na segunda lista: 8
Digite um numero para adicionar na primeira lista: 9
Digite um numero para adicionar na segunda lista: 10
A primeira lista tem os seguintes numeros: [1, 3, 5, 7, 9]
A segunda lista tem os seguintes numeros: [2, 4, 6, 8, 10]
A soma dos protudos é de: [2, 12, 30, 56, 90]
PS D:\>
```