

Microatividade 1:

```
array.sort.py x
G: > Meu Drive > Pessoal > Full Stack > N5 - Python > array.sort.py > ...
1 array = [46, 30, 11, 83, 74, 98, 20, 32, 10, 79, 27, 37, 18, 49, 5]
2 print("Array original:", array)
3 array.sort()
4 print("Array ordenado em ordem crescente:", array)
5 array.sort(key=None, reverse=True)
6 print("Array ordenado em ordem decrescente:", array)
7
8 array_strings = ["Gabriel Santos", "15/03/2010", "123.456.789-10", "MG-12.345.678"]
9 print("Array original:", array_strings)
10 array_strings.sort()
11 print("Array ordenado em ordem crescente:", array_strings)
12 array_strings.sort(key=None, reverse=True)
13 print("Array ordenado em ordem decrescente:", array_strings)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + ▢ ☒ ⋮ ↶

```
PS C:\Users\guipb> & C:/Users/guipb/AppData/Local/Microsoft/WindowsApps/python3.12.exe "g:/Meu Drive/Pessoal/Full Stack/N5 - Python/array.sort.py"
Array original: [46, 30, 11, 83, 74, 98, 20, 32, 10, 79, 27, 37, 18, 49, 5]
Array ordenado em ordem crescente: [5, 10, 11, 18, 20, 27, 30, 32, 37, 46, 49, 74, 79, 83, 98]
Array ordenado em ordem decrescente: [98, 83, 79, 74, 49, 46, 37, 32, 30, 27, 20, 18, 11, 10, 5]
Array original: ['Gabriel Santos', '15/03/2010', '123.456.789-10', 'MG-12.345.678']
Array ordenado em ordem crescente: ['123.456.789-10', '15/03/2010', 'Gabriel Santos', 'MG-12.345.678']
Array ordenado em ordem decrescente: ['MG-12.345.678', 'Gabriel Santos', '15/03/2010', '123.456.789-10']
PS C:\Users\guipb>
```

Microatividade 2:

```
bubble.sort.py X
G: > Meu Drive > Pessoal > Full Stack > N5 - Python > bubble.sort.py > ...
1  def bubbleSort(array):
2      for i in range(len(array)):
3          for j in range(0, len(array) - i - 1):
4              if array[j] > array[j + 1]:
5                  temp = array[j]
6                  array[j] = array[j + 1]
7                  array[j + 1] = temp
8
9  numbers = [64, 34, 25, 12, 22, 11, 90, 88, 76, 55, 43, 21, 9, 4, 3]
10
11  bubbleSort(numbers)
12
13  print("Array ordenado:", numbers)

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
Python + v [ ] [ ] ...

PS C:\Users\guipb> & C:/Users/guipb/AppData/Local/Microsoft/WindowsApps/python3.12.exe "g:/Meu Drive/Pessoal/Full Stack/N5 - Python/bubble.sort.py"
Array ordenado: [3, 4, 9, 11, 12, 21, 22, 25, 34, 43, 55, 64, 76, 88, 90]
PS C:\Users\guipb>
```

Microatividade 3:

```
selection.sort.py X
G: > Meu Drive > Pessoal > Full Stack > N5 - Python > selection.sort.py > ...
1  array = [87, 41, 5, 71, 66, 17, 80, 42, 23, 88, 78, 53, 35, 19, 40]
2  print("Array original:", array)
3
4  for i in range(len(array)):
5      min_index = i
6      for j in range(i + 1, len(array)):
7          if array[min_index] > array[j]:
8              min_index = j
9
10     array[i], array[min_index] = array[min_index], array[i]
11
12  print("Array ordenado:", array)

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
Python + v [ ] [ ] ...

PS C:\Users\guipb> & C:/Users/guipb/AppData/Local/Microsoft/WindowsApps/python3.12.exe "g:/Meu Drive/Pessoal/Full Stack/N5 - Python/selection.sort.py"
Array original: [87, 41, 5, 71, 66, 17, 80, 42, 23, 88, 78, 53, 35, 19, 40]
Array ordenado: [5, 17, 19, 23, 35, 40, 41, 42, 53, 66, 71, 78, 80, 87, 88]
PS C:\Users\guipb>
```

Microatividade 4:

```
ler.txt.py X
C: > Users > guipb > Downloads > ler.txt.py > ...
1  arquivo = open('loremipsum.txt', 'r')
2  conteudo = arquivo.read()
3
4  print("Conteúdo do arquivo:")
5  print(conteudo)
6
7  arquivo.seek(0)
8  primeira_linha = arquivo.readline()
9  print("\nPrimeira linha do arquivo:")
10 print(primeira_linha.strip())
11
12
13 arquivo.seek(0)
14 tres_primeiros_caracteres = arquivo.read(3)
15 print("\nPrimeiros 3 caracteres do arquivo:")
16 print(tres_primeiros_caracteres)
17
18
19 with open('loremipsum.txt', 'r') as arquivo_com_with:
20     conteudo_com_with = arquivo_com_with.read()
21
22 print("\nConteúdo do arquivo com 'with':")
23 print(conteudo_com_with)
```

```
PS G:\Meu Drive\Pessoal\Full Stack\N5 - Python> & C:/Users/guipb/AppData/Local/Microsoft/WindowsApps/python3.12.exe "g:/Meu Drive/Pessoal/Full Stack/N5 - Python/ler.txt.py"
```

Conteúdo do arquivo:

Lorem ipsum dolor sit amet, consectetur adipiscing elit,

sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi

ut aliquip ex ea commodo consequat. Duis aute irure dolor in

reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia

deserunt mollit anim id est laborum.

Primeira linha do arquivo:

Lorem ipsum dolor sit amet, consectetur adipiscing elit,

Primeiros 3 caracteres do arquivo:

Lor

Conteúdo do arquivo com 'with':

Lorem ipsum dolor sit amet, consectetur adipiscing elit,

sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi

ut aliquip ex ea commodo consequat. Duis aute irure dolor in

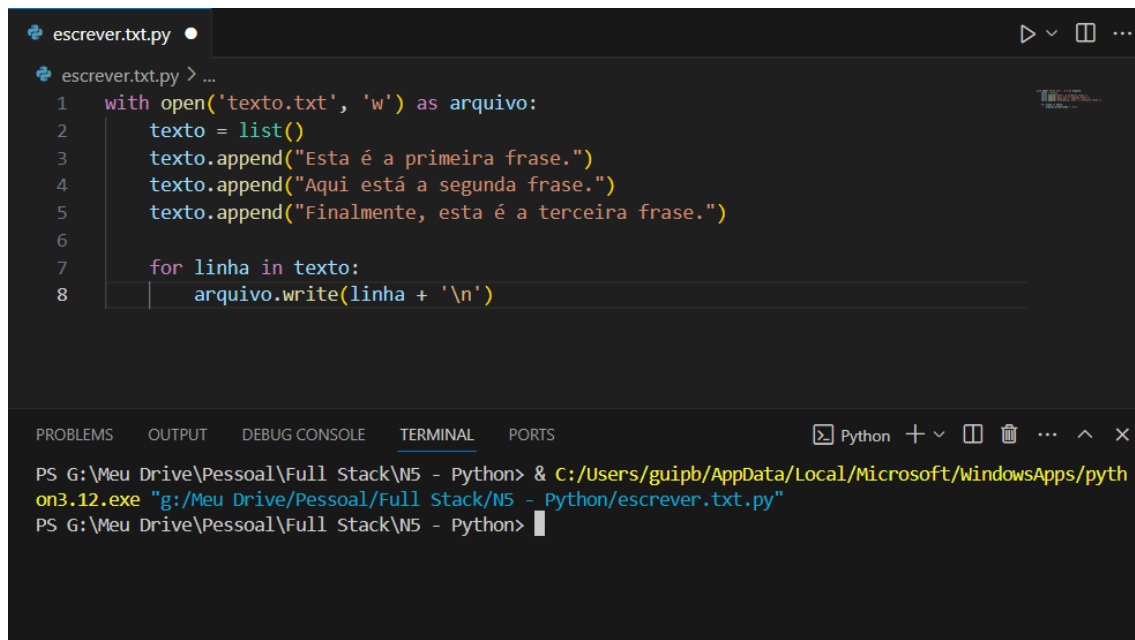
reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia

deserunt mollit anim id est laborum.

```
PS G:\Meu Drive\Pessoal\Full Stack\N5 - Python>
```

Microatividade 5:

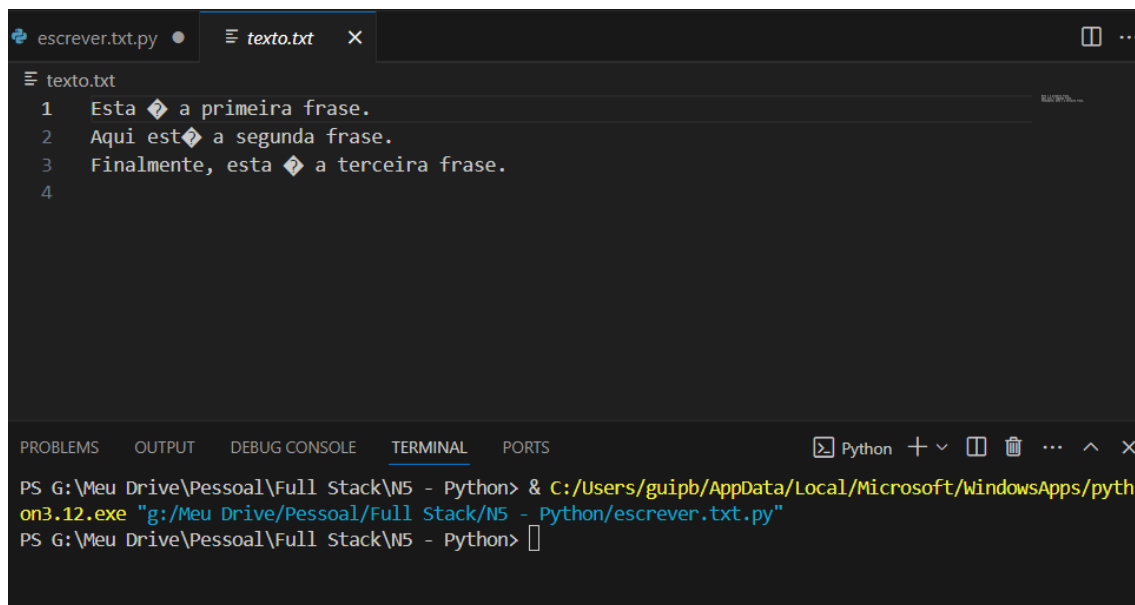


The screenshot shows a Python IDE with a file named `escrever.txt.py` open. The code in the editor is as follows:

```
1 with open('texto.txt', 'w') as arquivo:
2     texto = list()
3     texto.append("Esta é a primeira frase.")
4     texto.append("Aqui está a segunda frase.")
5     texto.append("Finalmente, esta é a terceira frase.")
6
7     for linha in texto:
8         arquivo.write(linha + '\n')
```

Below the editor, the **TERMINAL** tab is active, showing the command to run the script:

```
PS G:\Meu Drive\Pessoal\Full Stack\N5 - Python> & C:/Users/guipb/AppData/Local/Microsoft/WindowsApps/python3.12.exe "g:/Meu Drive/Pessoal/Full Stack/N5 - Python/escrever.txt.py"
PS G:\Meu Drive\Pessoal\Full Stack\N5 - Python>
```



The screenshot shows a text editor with a file named `texto.txt` open. The content of the file is as follows:

```
1 Esta é a primeira frase.
2 Aqui está a segunda frase.
3 Finalmente, esta é a terceira frase.
4
```

Below the editor, the **TERMINAL** tab is active, showing the command to run the script:

```
PS G:\Meu Drive\Pessoal\Full Stack\N5 - Python> & C:/Users/guipb/AppData/Local/Microsoft/WindowsApps/python3.12.exe "g:/Meu Drive/Pessoal/Full Stack/N5 - Python/escrever.txt.py"
PS G:\Meu Drive\Pessoal\Full Stack\N5 - Python>
```

Missão Prática:

```
kdd.py > main
1  import time
2
3  def bubble_sort(lista):
4      n = len(lista)
5      for i in range(n):
6          for j in range(0, n-i-1):
7              if lista[j] > lista[j+1]:
8                  lista[j], lista[j+1] = lista[j+1], lista[j]
9      return lista
10
11 def selection_sort(lista):
12     n = len(lista)
13     for i in range(n):
14         min_idx = i
15         for j in range(i+1, n):
16             if lista[j] < lista[min_idx]:
17                 min_idx = j
18         lista[i], lista[min_idx] = lista[min_idx], lista[i]
19     return lista
20
21 def ler_arquivo(nome_arquivo):
22     palavras = []
23     with open(nome_arquivo, 'r') as file:
24         for linha in file:
25             palavras.extend(linha.split())
26     return palavras
27
28 def main():
29     arquivo_txt = "arquivo.txt"
30     palavras = ler_arquivo(arquivo_txt)
31
32     palavras_bubble = palavras[:]
33     inicio_bubble = time.time()
34     bubble_sort(palavras_bubble)
35     fim_bubble = time.time()
36     print(f"Bubble Sort: {palavras_bubble[:10]}... (Tempo: {fim_bubble - inicio_bubble:.6f} segundos)")
37
```

```
38     palavras_selection = palavras[:]
39     inicio_selection = time.time()
40     selection_sort(palavras_selection)
41     fim_selection = time.time()
42     print(f"Selection Sort: {palavras_selection[:10]}... (Tempo: {fim_selection - inicio_selection:.6f} segundos)")
43
44     palavras_sort = palavras[:]
45     inicio_sort = time.time()
46     palavras_sort.sort()
47     fim_sort = time.time()
48     print(f"Método nativo sort(): {palavras_sort[:10]}... (Tempo: {fim_sort - inicio_sort:.6f} segundos)")
49
50     with open('palavras_ordenadas.txt', 'w') as file:
51         file.write(' '.join(palavras_sort))
52
53 if __name__ == "__main__":
54     main()
```

```
PROBLEMS  CONTROLLER  DEBUG CONSOLE  TERMINAL  FILES
Método nativo sort(): ['"a', '"nÃo', '"o', '(PcD)', 'A', 'AlÃm', 'As', 'As', 'Como', 'Como']... (Tempo: 0.000000 segundos)
PS G:\Meu Drive\Pessoal\Full Stack\W5 - Python> & C:/Users/guipb/AppData/Local/Microsoft/WindowsApps/python3.12.exe "g:/Meu Drive/Pessoal/Full Stack\W5 - Python/kdd.py"
Bubble Sort: ['"a', '"nÃo', '"o', '(PcD)', 'A', 'AlÃm', 'As', 'As', 'Como', 'Como']... (Tempo: 0.027282 segundos)
Selection Sort: ['"a', '"nÃo', '"o', '(PcD)', 'A', 'AlÃm', 'As', 'As', 'Como', 'Como']... (Tempo: 0.014175 segundos)
Método nativo sort(): ['"a', '"nÃo', '"o', '(PcD)', 'A', 'AlÃm', 'As', 'As', 'Como', 'Como']... (Tempo: 0.000000 segundos)
PS G:\Meu Drive\Pessoal\Full Stack\W5 - Python>
```

```
palavras_ordenadas.txt
1 idade capacidades capacidades. capacitar cargos cargos cargos carreira carreira carreira carreira carreira. ca

Método nativo sort(): ['a', 'nãeo', 'o', '(PcD)', 'A', 'AlÃem', 'As', 'As', 'Como', 'Como']... (Tempo: 0.000000 segundos)
PS G:\Meu Drive\Pessoal\Full Stack\WS - Python> & c:/Users/guipb/AppData/Local/Microsoft/WindowsApps/python3.12.exe "g:/Meu Drive/Pessoal/Full Stack\WS - Python/kdd.py"
Bubble Sort: ['a', 'nãeo', 'o', '(PcD)', 'A', 'AlÃem', 'As', 'As', 'Como', 'Como']... (Tempo: 0.027282 segundos)
Selection Sort: ['a', 'nãeo', 'o', '(PcD)', 'A', 'AlÃem', 'As', 'As', 'Como', 'Como']... (Tempo: 0.014175 segundos)
Método nativo sort(): ['a', 'nãeo', 'o', '(PcD)', 'A', 'AlÃem', 'As', 'As', 'Como', 'Como']... (Tempo: 0.000000 segundos)
PS G:\Meu Drive\Pessoal\Full Stack\WS - Python>
```