



ESCOLA
POLITÉCNICA



Programa IT Academy – Processo Seletivo – Edição #16

Etapa 2 feito por:

João Pedro de Moura Medeiros

Etapa 2

Enunciado

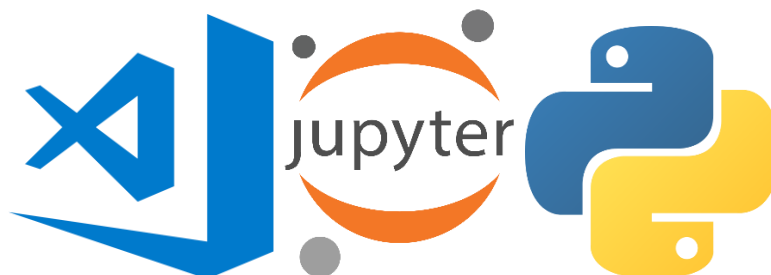
Nesta etapa, você vai escrever um programa de computador. Para isso deve ser feita a leitura do arquivo .csv enviado junto com este enunciado. Neste arquivo você encontra dados sobre medicamentos disponíveis no Brasil. Você deve implementar as seguintes funcionalidades:

1. [Consultar medicamentos pelo nome] Permitir que o usuário informe o nome do medicamento (ou parte do nome do medicamento) que deseja e como resultado o programa deverá exibir:
 - a. Uma lista com os medicamentos encontrados e suas informações (Nome, Produto, Apresentação e valor PF Sem Impostos);
Atenção: somente devem aparecer no resultado os registros de produtos que foram comercializados em 2020 (observar a coluna de dados "COMERCIALIZAÇÃO 2020").
2. [Buscar pelo código de barras] O programa deverá solicitar ao usuário o número correspondente ao código de barras de um produto (coluna de dados "EAN 1", por exemplo '525516020019503') e então:
 - a. Localizar todos os registros referentes a este produto, independentemente de terem sido comercializados ou não em 2020;
 - b. Dentre todos os registros encontrados, identificar o Preço Máximo ao Consumidor (aliquota de 0%, coluna de dados "PMC 0%") mais alto e o mais baixo. Exibir na tela o mais alto, o mais baixo e a diferença entre eles.
3. [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)] Com base somente nos produtos que foram comercializados em 2020, o programa deverá:
 - a. Consultar a coluna de dados "LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)" para determinar o percentual de produtos classificados como "Negativa", "Neutra" ou "Positiva" para esta coluna.
 - b. Mostrar os respectivos valores percentuais da seguinte maneira (dados fictícios):
[* repare que a quantidade de asteriscos é proporcional ao respectivo percentual, por exemplo, neste caso são 21 asteriscos para a classificação Negativa.]

| CLASSIFICACAO | PERCENTUAL | GRAFICO |
|---------------|------------|---------|
| Negativa | 21,33% | ***** |
| Neutra | 45,10% | ***** |
| Positiva | 33,49% | ***** |
| | | |
| TOTAL | 100,00% | |

Programa

O programa de computador foi feito no Visual Studio Code (VS Code), com a ferramenta auxiliar Jupyter-Notebook para a compilação do código na linguagem Python.



Programa

```

1  '''AQUI SERÁ O CÓDIGO QUE IRÁ EXECUTAR TODO O PROGRAMA, JUNTANDO TODOS OS CÓDIGOS EM UM SÓ'''
2
3  # Chama a função menu
4  menu()
5
6  i = int
7  while i: # Foi criado um while() para ficar rodando o programa
8      try: # Inserido um try e um except para analisarem se o usuário inseriu algum dado inválido
9          i = int(input("Digite o número de alguma funcionalidade:"))
10         if(i == 1):
11             print("Funcionalidade 1 [Consultar medicamentos pelo nome]")
12             func1() # Executará o código da funcionalidade
13         elif(i == 2):
14             print("Funcionalidade 2 [Buscar pelo código de barras]")
15             func2() # Executará o código da funcionalidade
16         elif(i == 3):
17             print("Funcionalidade 3 [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]")
18             func3() # Executará o código da funcionalidade
19
20     # No enunciado não consta, mas para melhor uso do programa pelo usuário, foi criado esse elif para se ele quiser encerrar o programa
21     elif(i == 4):
22         print("Funcionalidade 4 [Terminar programa]")
23         print("O programa se encerrou")
24         break
25     else:
26         print("Erro de numeração, digite apenas os números das funcionalidade de |1-4|")
27 except ValueError:
28     print("Digite apenas números de 1-4!!!!")

```

Testes de interação com o programa:

- Teste 1: O usuário quer acessar a funcionalidade 1 [Consultar medicamentos pelo nome], então ele digita 1 e executa a funcionalidade.

```
|
|                                     MENU
|
|          DIGITE OS NÚMERO DAS FUNCIONALIDADES ABAIXO PARA ELAS EXECUTAREM
|
| Funcionalidade 1 [Consultar medicamentos pelo nome]
| Funcionalidade 2 [Buscar pelo código de barras]
| Funcionalidade 3 [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]
| Funcionalidade 4 [Terminar programa]
|
|
|
|
| Funcionalidade 1 [Consultar medicamentos pelo nome]
```

- Teste 2: O usuário quer acessar a funcionalidade 3 [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/CONFIS)], então ele digita 3 e executa a funcionalidade.

3|

Digite o número de alguma funcionalidade:

```
|
|
| MENU
|
| DIGITE OS NÚMERO DAS FUNCIONALIDADES ABAIXO PARA ELAS EXECUTAREM
|
| Funcionalidade 1 [Consultar medicamentos pelo nome]
| Funcionalidade 2 [Buscar pelo código de barras]
| Funcionalidade 3 [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]
| Funcionalidade 4 [Terminar programa]
|
|
| Funcionalidade 3 [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]
```

- Teste 3: O usuário quer sair do programa, então ele digita o número da funcionalidade 4 [Terminar programa] e executa a funcionalidade.
Observação: No enunciado não consta, mas para melhor uso do programa pelo usuário, foi criado esse elif para se ele quiser encerrar o programa

Digite o número de alguma funcionalidade:

```
|
|
| MENU
|
| DIGITE OS NÚMERO DAS FUNCIONALIDADES ABAIXO PARA ELAS EXECUTAREM
|
| Funcionalidade 1 [Consultar medicamentos pelo nome]
| Funcionalidade 2 [Buscar pelo código de barras]
| Funcionalidade 3 [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]
| Funcionalidade 4 [Terminar programa]
|
|
| Funcionalidade 4 [Terminar programa]
| O programa se encerrou
```

- Teste 4: O usuário inseriu um número errado.

Digite o número de alguma funcionalidade:

```
|
|
| MENU
|
| DIGITE OS NÚMERO DAS FUNCIONALIDADES ABAIXO PARA ELAS EXECUTAREM
|
| Funcionalidade 1 [Consultar medicamentos pelo nome]
| Funcionalidade 2 [Buscar pelo código de barras]
| Funcionalidade 3 [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]
| Funcionalidade 4 [Terminar programa]
|
|
| Erro de numeração, digite apenas os números das funcionalidade de [1-4]
```

Leitura do arquivo .csv

A leitura do arquivo .csv foi feita através de um método da biblioteca Pandas, do Python, para isso foi necessário importar a biblioteca Pandas.

Importação da biblioteca Pandas:

Importação das bibliotecas

```
1 '''AQUI SERA FEITA A IMPORTAÇÃO DAS BIBLIOTECAS'''
2
3 # Importação da biblioteca 'pandas' para ser feita a leitura, análise e manipulação de dados do arquivo .csv
4 import pandas as pd
✓ 0.2s
```

Leitura do arquivo .csv com o método read_csv() do pandas:

Leitura do arquivo .csv

```
1 '''AQUI SERA FEITA A LEITURA DO ARQUIVO .CSV'''
2
3 # A variável "csv" receberá a leitura de um arquivo .csv feito pelo método "read_csv()" do pandas, com um separador de dados "sep"
4 # Encoding foi implementado pois o csv não reconhecia palavras acentuadas e o caractere "ç"
5 csv = pd.read_csv("../TA_PRECO_MEDICAMENTO.csv", sep=";", encoding = "ISO-8859-1")
6
```

Execução e resultado de leitura do arquivo csv:

```
7 # TESTE
8 display(csv) # Display printará todos os registros do arquivo .csv
✓ 1.7s
```

| | SUBSTÂNCIA | CNPJ | LABORATÓRIO | CÓDIGO GIREM | REGISTRO | EAN 1 | EAN 2 | EAN 3 | PRODUTO | APRESENTAÇÃO | PMC 18% ALC | PMC 20% | RESTRIÇÃO HOSPITALAR | CAP | CONFAZ 81 |
|-------|---|--------------------|---|--------------|-------------|-------------|-------|-------|--------------------------------|---|-------------|---------|----------------------|-----|-----------|
| 0 | SALICILATO DE FENILÁCIDO SALICÍLICO ÓXIDO DE... | 33.379.884/0001-96 | LABORATORIO SIMOES LTDA. | 5,20501E+14 | 57600510011 | 7,89621E+12 | - | - | TALCO ALVIO | TALQUEIRA C/ 100 G | 8,5 | 9,7 | Não | Não | Não |
| 1 | DIENOGESTEVALERATO DE ESTRADIOL | 56.990.534/0001-67 | SCHERING DO BRASIL QUÍMICA E FARMACÉUTICA LTDA | 5,30917E+14 | 1,002E+12 | 7,89612E+12 | - | - | CHARMELIE | COM REV A + COM REV B + COM REV C + COM REV D ... | 163,25 | 186,12 | Não | Não | Não |
| 2 | HIDROCLOROTIAZIDA-MALEATO DE ENALAPRIL | 45.987.013/0001-34 | ORGANON FARMACÉUTICA LTDA. | 5,255E+14 | 1,0029E+12 | 7,89734E+12 | - | - | CO-RENITEC | 20 MG + 12,5 MG COM CT BL AL/AL X 30 | 73,78 | 75,62 | Não | Não | Não |
| 3 | HIDROCLOROTIAZIDA-MALEATO DE ENALAPRIL | 45.987.013/0001-34 | ORGANON FARMACÉUTICA LTDA. | 5,255E+14 | 1,0029E+12 | 7,89734E+12 | - | - | CO-RENITEC | 10 MG + 25 MG COM CT BL AL/AL X 30 | 48,45 | 49,67 | Não | Não | Não |
| 4 | MONTELUCASTE | 45.987.013/0001-34 | ORGANON FARMACÉUTICA LTDA. | 5,25502E+14 | 1,0029E+12 | 7,89734E+12 | - | - | SINGULAIR | 5 MG COM MAST CT 3 BL AL PLAS INC X 10 | 75,32 | 77,2 | Não | Não | Não |
| - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| 26305 | CLORIDRATO DE SEVELÂMER | 13.078.518/0001-90 | FUNDAÇÃO BAIANA DE PESQ. CIENTÍFICA E DESENV. ... | 5,44018E+14 | 1,9883E+12 | 7,8986E+12 | - | - | CLORIDRATO DE SEVELAMER | 800 MG COM REV CT FR PLAS OPC X 180 | NaN | NaN | Sim | Sim | Sim |
| 26306 | INSULINA HUMANA | 13.078.518/0001-90 | FUNDAÇÃO BAIANA DE PESQ. CIENTÍFICA E DESENV. ... | 5,44018E+14 | 1,9883E+12 | 7,8986E+12 | - | - | BAHIAFARMA INSULINA HUMANA NPH | 100 UI/ML SUS INJ CT FA VD INC X 10 ML | NaN | NaN | Não | Sim | Sim |

Figura 1 Foto de uma parte da tabela do arquivo .csv

Funcionalidades – Explicações e lógica usada

Os códigos das funcionalidades 1 a 3 estão dentro de funções criadas, para serem depois executadas na aba do programa.

```
def func1():
```

```
    if(i == 1):  
        print("Funcionalidade 1 [Consultar medicamentos pelo nome]")  
        func1() # Executará o código da funcionalidade
```

1. [Consultar medicamentos pelo nome]

Permitir que o usuário informe o medicamento desejado:

Foi implementada a função input() para o usuário poder digitar o medicamento desejado.

```
nomeMedicamento = str(input("Digite o nome do medicamento:")) # Variável criada para armazenar o nome do medicamento indicado pelo usuário  
nomeMedicamentoFormt = nomeMedicamento.upper() # Método upper() transformará nome em maiúsculo, caso o usuário digite em minúsculo  
print("Nome digitado:", nomeMedicamentoFormt)
```

Filtrar apenas registros de produtos que foram comercializados em 2020:

Para pegar as informações apenas daquele ano desejado, como pedido no enunciado, foi adicionado o método chamado loc(), o qual extrai do Data Frame os dados daquele ano com as linhas e suas respectivas colunas.

```
comercializados2020 = csv[csv["COMERCIALIZAÇÃO 2020"] == "Sim"] # Variável criada para armazenar dados de produtos que foram comercializados em 2020
```

Filtrar o medicamento solicitado pelo usuário:

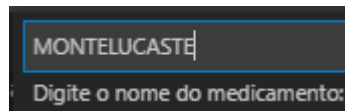
Realizada a filtragem, basta apenas extrair as informações do medicamento informado pelo usuário, com base no enunciado, transformando em uma lista de informações e printando na tela.

```
tabMedicamento = comercializados2020.loc[comercializados2020["SUBSTÂNCIA"] == nomeMedicamentoFormt] # Variável criada para armazenar os dados do nome do medicamento e suas informações,  
# indicado pelo usuário  
novoDF = tabMedicamento.loc[:,["SUBSTÂNCIA", "PRODUTO", "APRESENTAÇÃO", "PF Sem Impostos"]] # Variável criada para filtrar apenas as colunas desejadas, como descrito no enunciado  
listaTab = novoDF.values.tolist() # Aqui será feito o envio dos valores do data frame "novoDF" para uma lista  
  
cont = 0  
for lis in listaTab: # For criado para printar as listas, de forma mais organizada  
    cont += 1  
    print(f"Lista {cont}: ", lis)
```

Testes da funcionalidade 1. [Consultar medicamentos pelo nome]:

- Teste 1: O usuário executou a funcionalidade 1 [Consultar medicamentos pelo nome].

Pede-se para informar o medicamento, como exemplo foi digitado o medicamento MONTELUCASTE:



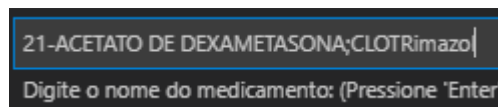
MONTELUCASTE
Digite o nome do medicamento:

Printa na tela as informações:

```
Nome digitado: MONTELUCASTE
Lista 1: ['MONTELUCASTE', 'SINGULAIR', '5 MG COM MAST CT 3 BL AL PLAS INC X 10', '44,67']
Lista 2: ['MONTELUCASTE', 'SINGULAIR', '10 MG COM REV CT 3 BL AL PLAS INC X 10', '44,67']
Lista 3: ['MONTELUCASTE', 'SINGULAIR', '4 MG COM MAST CT BL AL PLAS INC X 10', '30,03']
Lista 4: ['MONTELUCASTE', 'SINGULAIR', '4 MG COM MAST CT 3 BL AL PLAS INC X 10', '44,67']
Lista 5: ['MONTELUCASTE', 'SINGULAIR', '5 MG COM MAST CT BL AL PLAS INC X 10', '30,03']
Lista 6: ['MONTELUCASTE', 'SINGULAIR', '10 MG COM REV CT BL AL PLAS INC X 10', '30,03']
```

- Teste 2: O usuário executou a funcionalidade 1 [Consultar medicamentos pelo nome]

Pede-se para digitar o medicamento, como exemplo foi digitado o nome 21-ACETATO DE DEXAMETASONA;CLOTRImazol:



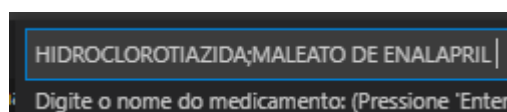
21-ACETATO DE DEXAMETASONA;CLOTRImazo
Digite o nome do medicamento: (Pressione 'Enter')

Printa na tela as informações:

```
Nome digitado: 21-ACETATO DE DEXAMETASONA;CLOTRIMAZOL
Lista 1: ['21-ACETATO DE DEXAMETASONA;CLOTRIMAZOL', 'BAYCUTEN N', '10 MG/G + 0,443 MG/G CREM DERM CT BG AL X 40 G', '21,07']
```

- Teste 3: O usuário executou a funcionalidade 1 [Consultar medicamentos pelo nome].

Pede-se para digitar o medicamento, como exemplo foi digitado o medicamento HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL:



HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL
Digite o nome do medicamento: (Pressione 'Enter')

Printa na tela as informações:

```
Nome digitado: HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL
Lista 1: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'MALEATO DE ENALAPRIL + HIDROCLOROTIAZIDA', '20 MG + 12,5 MG COM CT BL AL AL X 30', '47,72']
Lista 2: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'CO-PRESSOTEC', '20 MG + 12,50 MG COM CT ENV AL POLIET X 30\xa0', '38,01']
Lista 3: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'CO-PRESSOTEC', '10 MG + 25 MG COM CT ENV AL POLIET X 30\xa0', '24,41']
Lista 4: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'MALEATO DE ENALAPRIL + HIDROCLOROTIAZIDA', '20 MG + 12,50 MG COM CT ENV AL POLIET X 30\xa0', '28,45']
Lista 5: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'MALEATO DE ENALAPRIL + HIDROCLOROTIAZIDA', '10 MG + 25 MG COM CT ENV AL POLIET X 30\xa0', '18,7']
Lista 6: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'VASOPRIL PLUS', '20 MG + 12,5 MG COM CT STR AL X 30\xa0', '49']
Lista 7: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'VASOPRIL PLUS', '10 MG + 25 MG COM CT STR AL X 30\xa0\xa0', '30,82']
Lista 8: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'VASOPRIL PLUS', '10 MG + 25 MG COM CT STR AL X 60', '55,07']
Lista 9: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'VASOPRIL PLUS', '20 MG + 12,5 MG COM CT STR AL X 60', '83,78']
Lista 10: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'MALEATO DE ENALAPRIL + HIDROCLOROTIAZIDA', '20 MG + 12,5 MG COM CT STR AL X 30\xa0', '28,45']
Lista 11: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'MALEATO DE ENALAPRIL + HIDROCLOROTIAZIDA', '10 MG + 25 MG COM CT STR AL X 30\xa0\xa0', '18,69']
Lista 12: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'MALENA HCT', '20 MG + 12,5 MG COM CT BL AL AL X 30', '18,85']
Lista 13: ['HIDROCLOROTIAZIDA;MALEATO DE ENALAPRIL', 'MALEATO DE ENALAPRIL + HIDROCLOROTIAZIDA', '20 MG + 12,5 MG COM CT BL AL AL X 30', '28,44']
```

2. [Buscar pelo código de barras]

Permitir que o usuário informe o código de barras e localize todos os registros correspondente a ele:

Foi criado um input() para receber o número do código de barras indicado pelo usuário, filtrando todos os registros relacionados com o número digitado.

```
numCod = int(input("Insira o número correspondente ao código de barras de um produto:"))

# Letra A
registros = csv.loc[csv['EAN 1'] == str(numCod)] # Variável criada para armazenar o código do medicamento indicado pelo usuário
display(registros.head())
```

Identificar o PMC (mais alto), PMC (mais baixo) e sua diferença:

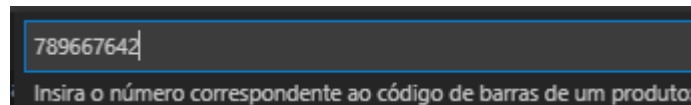
Para identificar as informações pedidas, foi filtrada a coluna “PMC 0%” retirando os valores nulos presentes para não haver conflito, logo após foi transformado em uma lista para facilitar a manipulação dos dados e assim foi feita a filtração do PMC (mais alto) e PMC (mais baixo), colocando os dois em uma função que calculará sua diferença.

Se fez um for para trocar a "," pelo "." para conseguir fazer o cálculo da diferença.

```
16 # Letra B
17 pmc = registros["PMC 0%"] # Filtrar a tabela "PMC 0%" para análise e manipulação dos dados
18 pmcAtualizado = pmc.dropna() # Retirar os valores nulos para a facilidade de manipulação seguinte
19 listaPMC = pmcAtualizado.tolist() # Transformando tabela em uma lista para facilitar a manipulação de dados
20
21 listaFloat = []
22 for i in listaPMC: # For criado para substituir a "," por "." para fazer a manipulação de dados seguinte
23     newI = i.replace(",", ".")
24     listaFloat.append(float(newI))
25
26 def calcDiferenca(val1, val2): # Método criado para calcular a diferença entre o PMC(mais alto) e PMC(mais baixo)
27     diferenca = val1 - val2
28     return diferenca
29
30 maisAlto = listaFloat[-1] # Armazenando o PMC(mais alto) em uma variável
31 maisBaixo = listaFloat[0] # Armazenando o PMC(mais baixo) em uma variável
32
33 print("PMC(mais alto):", maisAlto)
34 print("PMC(mais baixo):", maisBaixo)
35 print("Diferença entre os PMC(mais alto) e PMC(mais baixo):", calcDiferenca(maisAlto, maisBaixo))
36
```

Testes da funcionalidade 2. [Buscar pelo código de barras]:

- Teste 1: O usuário executou a funcionalidade 2 [Buscar pelo código de barras]
Pede-se para digitar o nome do código de barras, foi digitado o código: 789667642



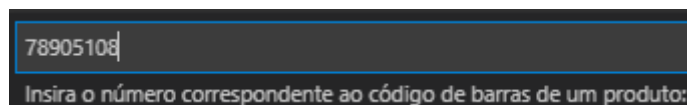
Registros correspondentes ao código de barra:

| | SUBSTÂNCIA | CNPJ | LABORATÓRIO | CÓDIGO GGREM | REGISTRO | EAN 1 | EAN 2 | EAN 3 | PRODUTO | APRESENTAÇÃO | ... | PMC 18% ALC | PMC 20% | RESTRIÇÃO HOSPITALAR | CAP | CONFAZ 87 | ICMS 0% |
|------|--------------------------|--------------------|---|--------------|------------|-----------|-------|-------|----------|---|-----|-------------|---------|----------------------|-----|-----------|---------|
| 5963 | MICOFENOLATO DE MOFETILA | 44.734.671/0001-51 | CRISTÁLIA PRODUTOS QUÍMICOS FARMACÉUTICOS LTDA. | 5,06716E+14 | 1,0298E+12 | 789667642 | - | - | MOFECELL | 500 MG COM REV CT BL AL PLAS TRANS X 10 | ... | 223,83 | 229,43 | Não | Sim | Sim | Não |
| 5964 | MICOFENOLATO DE MOFETILA | 44.734.671/0001-51 | CRISTÁLIA PRODUTOS QUÍMICOS FARMACÉUTICOS LTDA. | 5,06716E+14 | 1,0298E+12 | 789667642 | - | - | MOFECELL | 500 MG COM REV CT BL AL PLAS TRANS X 20 | ... | 447,7 | 458,9 | Não | Sim | Sim | Não |
| 5965 | MICOFENOLATO DE MOFETILA | 44.734.671/0001-51 | CRISTÁLIA PRODUTOS QUÍMICOS FARMACÉUTICOS LTDA. | 5,06716E+14 | 1,0298E+12 | 789667642 | - | - | MOFECELL | 500 MG COM REV CT BL AL PLAS TRANS X 30 | ... | 671,55 | 688,33 | Não | Sim | Sim | Não |
| 5966 | MICOFENOLATO DE MOFETILA | 44.734.671/0001-51 | CRISTÁLIA PRODUTOS QUÍMICOS FARMACÉUTICOS LTDA. | 5,06716E+14 | 1,0298E+12 | 789667642 | - | - | MOFECELL | 500 MG COM REV CT BL AL PLAS TRANS X 40 | ... | 895,41 | 917,79 | Não | Sim | Sim | Não |
| 5967 | MICOFENOLATO DE MOFETILA | 44.734.671/0001-51 | CRISTÁLIA PRODUTOS QUÍMICOS FARMACÉUTICOS LTDA. | 5,06716E+14 | 1,0298E+12 | 789667642 | - | - | MOFECELL | 500 MG COM REV CT BL AL PLAS TRANS X 50 | ... | 1119,27 | 1147,26 | Não | Sim | Sim | Não |

Printa na tela o PMC (mais alto), PMC (mais baixo) e sua diferença:

```
PMC(mais alto): 917.8
PMC(mais baixo): 183.55
Diferença entre os PMC(mais alto) e PMC(mais baixo): 734.25
```

- Teste 2: O usuário executou a funcionalidade 2 [Buscar pelo código de barras]
Pede-se para digitar o nome do código de barras, foi digitado o código: 789667642



Registros correspondentes ao código de barra:

| | SUBSTÂNCIA | CNPJ | LABORATÓRIO | CÓDIGO GGREM | REGISTRO | EAN 1 | EAN 2 | EAN 3 | PRODUTO | APRESENTAÇÃO | ... | PMC 18% ALC | PMC 20% |
|-------|---|--------------------|----------------------|--------------|-----------|----------|-------|-------|------------------|---|-----|-------------|---------|
| 12175 | CLORETO DE BENZALCÔNIO-ÓXIDO DE ZINCO;CÂNFORA | 84.683.382/0001-95 | MINÂNCORA & CIA LTDA | 5,257E+14 | 1,069E+12 | 78905108 | - | - | POMADA MINANCORA | 20% + 0,5% + 5% POM DERM PT PLAS OPC X 30 G | ... | 12,52 | 14,28 |

Printa na tela o PMC (mais alto), PMC (mais baixo) e sua diferença:

```
PMC(mais alto): 11.16
PMC(mais baixo): 11.16
Diferença entre os PMC(mais alto) e PMC(mais baixo): 0.0
```

- Teste 3: O usuário executou a funcionalidade 2 [Buscar pelo código de barras] Pede-se para digitar o nome do código de barras, não foi digitado um código e sim uma palavra, logo o try e except irá fazer o caminho:

```
Del|
Insira o número correspondente ao código de barras de um produto:
```

Printa a mensagem:

```
Dado de entrada inválido, por favor execute novamente a funcionalidade e insira o dado correto
```

- 3) [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]

Filtrar apenas registros de produtos que foram comercializados em 2020:

Para pegar as informações apenas daquele ano desejado, como pedido no enunciado, foi adicionado se extrai do Data Frame os dados daquele ano com as linhas e suas respectivas colunas.

```
prod2020 = csv[csv["COMERCIALIZAÇÃO 2020"] == "Sim"] # Filtrar itens somente comercializados em 2020
```

Determinar o percentual:

Para determinar o percentual da coluna “LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)”, a coluna foi filtrada para se adquirir os dados dela, após foi consultado a quantidade total de dados dela e os dados agrupados em “Positiva”, “Negativa” e “Neutra” para ser feito o cálculo de %.

```
# Aqui será feita a filtragem da coluna ["LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)"]
listConcessao = prod2020["LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)"]

valorTotal = listConcessao.count() # Teste para ver a quantidade total
# TESTE
# display(valorTotal)
# 13383

qntPosNegNeu = listConcessao.value_counts() # Teste para ver a quantidade "Positiva", "Negativa" e "Neutra"
# TESTE
display(qntPosNegNeu)
'''Positiva    8861
Negativa      4474
Neutra         48'''

lista = qntPosNegNeu.tolist() # Transformando em lista para manipular os dados sendo a ordem [Positiva, Negativa, Neutra]

listaCalc = []
for i in lista: # For criada para fazer o percentual dos produtos classificados como "Negativa", "Neutra" ou "Positiva"
    num = (i*100)/valorTotal
    num = round(num, 2)
    listaCalc.append(num)
# TESTE
# print(listaCalc)
# Lista com os percentuais sendo [66.21==Positiva, 33.43==Negativa, 0.36==Neutra]
percPOS = f"{listaCalc[0]}%"
percNEG = f"{listaCalc[1]}%"
percNEU = f"{listaCalc[2]}%"
```

Mostrar asteriscos com a mesma proporção dos valores percentuais:

Foi pensado em criar um for passando a multiplicação de uma string “*” e o percentual, para conseguir o resultado desejado, armazenando em uma lista e passando os dados em uma tabela criada.

```
35 listaAsteriscos = []
36 for i in listaCalc: # For criada para fazer a quantidade de asteriscos proporcional ao respectivo percentual
37     numInteiro = int(i)
38     asteriscos = "*" * numInteiro
39     listaAsteriscos.append(asteriscos)
40 # TESTE
41 # print(listaAsteriscos)
42 # ['*****', '*****', '']
43 asteriscosPOS = listaAsteriscos[0]
44 asteriscosNEG = listaAsteriscos[1]
45 asteriscosNEU = listaAsteriscos[2]
46
47 # Criação de uma tabela para mostrar os valores percentuais como solicitado
48 data = [{"CLASSIFICAÇÃO": ["NEGATIVA", "NEUTRA", "POSITIVA", "TOTAL"],
49     "PERCENTUAL": [percNEG, percNEU, percPOS, "100,00%"],
50     "GRAFICO": [asteriscosNEG, asteriscosNEU, asteriscosPOS, ""]} ]
51 df = pd.DataFrame(data)
52 print(df)
```

Testes da funcionalidade 3. [LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]:

- Teste 1: O usuário executou a funcionalidade 3. [LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)]

Printa na tela as informações:



Autoavaliação

Após ter terminado o programa, me senti bem, consegui programar e executar com facilidade todas as funcionalidades, fiz elas funcionarem com eficiência. Tive muita ajuda dos métodos, inclusive dos da biblioteca Pandas, os quais facilitaram minha programação nas funcionalidades e na leitura do arquivo .csv, pois ajudaram na leitura, análise e manipulação de dados do arquivo .csv.

Não apresentei dificuldades em nenhuma parte do processo seletivo, tive completamente clareza no que fazer em todos os objetivos pedidos no enunciado, sendo todos executados com eficiência.