 

Programa IT Academy – Processo Seletivo – Edição #16

## 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 desejar e como resultado o programa deverá exibir:
   1. 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”).

1. [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:
   1. Localizar todos os registros referentes a este produto, independentemente de terem sido comercializados ou não em 2020;
   2. Dentre todos os registos encontrados, identificar o Preço Máximo ao Consumidor (alíquota 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.
2. [Comparativo da LISTA DE CONCESSÃO DE CRÉDITO TRIBUTÁRIO (PIS/COFINS)] Com base somente nos produtos que foram comercializados em 2020, o programa deverá:
   1. 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.
   2. 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 | 45,18% | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |
| Positiva | 33,49% | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |

TOTAL

100,00%

## Observações:

1. Sugere-se o desenvolvimento de um programa **na linguagem de sua preferência**, com uma **interface também de sua preferência podendo ser gráfica ou textual/console**, com um **menu com as opções enumeradas nos requisitos**;
2. Juntamente a este enunciado foi fornecido um arquivo no formato CSV contendo nomes, valores em decimais, bem como o respectivo dicionário de dados;
3. Você deve escrever o código que lê o arquivo e armazena os dados lidos em memória (do jeito que você quiser).
4. Não é necessário gravar dados em nenhum formato, nem usar sistemas de banco de dados.
5. O programa deverá lidar com dados de entrada inválidos e informar uma mensagem adequada caso ocorram.
6. Para facilitar, não é necessário lidar com a acentuação de palavras.
7. Na escrita do relatório apresente comentários sobre como você realizou os testes. Não esqueça de incluir uma autoavaliação.

-x-

**Explicação da Solução:**

Para solucionar o problema, foram criadas quatro Classes principais, cada uma visando abordar um ponto específico do problema. Além destas, duas classes auxiliares, uma das quais não faz parte do programa final. A pasta com a solução foi dividida em duas sub-pastas; a primeira (testes) foi criada com o intuito de fazer a leitura e análise do arquivo – como estavam distribuidas as informações na tabela, quais colunas continham quais informações e etc; a segunda, que contém o programa final, é dividido em um Main (classe principal, que chama o método execute da Classe App, que por sua vez contem três chamadas para as outras classes – cada uma de acordo com um dos pontos do exercício). Para ler o arquivo, utilizei a Classe Scanner do java.util e a Classe File do java.io, depois iterei sobre o arquivo, quebrando cada linha lida no Scanner em arranjos unidimencionais. A localização do arquivo e o fato de ele ser separado nos pontos e vírugula (;) estão definidos em um arquivo externo as pastas denominado “CsvConfig.java”, que exporta o caminho e o separador para todas as outras classes. As três classes das etapas dos exercícios foram resolvidas de modos muito similares, possuindo um método principal chamado pelo App e classes internas que executam funções – melhores explicadas no código – específicas para apresentar cada informação na tela. Ainda, todas percorrem o arquivo