

	<b>Exercício Escolar 1</b>	Semestre: <b>2023.2</b>	
	Curso: Sistemas de Informação	Disciplina: Programação 1	
	Prof: Raphael Dourado	<b>Nota: ____/10</b>	
	Prazo de entrega: 20/12/2023		
Orientações gerais			
<ul style="list-style-type: none"> <li>O projeto pode ser feito em equipes de no máximo 3 pessoas</li> <li>No dia da entrega um dos membros da equipe, a ser definido por sorteio, deve apresentar sua solução para o professor e turma.</li> </ul>			



## Exercício Escolar 1

### Projeto: Sistema de BI para dados de vendas.

**Contexto:** Uma rede de lojas varejistas possui um sistema de PDV (Ponto de Venda) para realizar as vendas, porém este sistema não oferece relatórios gerenciais. A empresa contratou a sua equipe para desenvolver um sistema que se integre ao sistema de PDV e ofereça esta nova funcionalidade. Porém, por questões de segurança, a equipe de TI da empresa que desenvolve o PDV não autorizou o acesso direto de seu sistema ao banco de dados, de forma que o acordo foi que você irá receber os dados através de arquivos CSV, onde cada arquivo conterá os dados de um período de tempo (dia, semana, etc.). Sua tarefa é gerar os relatórios a partir destes CSVs.

Os arquivos CSV que você irá receber possuem o formato abaixo. Por uma limitação do sistema PDV, as linhas não estarão ordenadas por código de pedido:

```
CodPedido;DtPedido;CodFilial;CodProduto;Qtd;VlUnit
1;1701823242;11;12;2;65.54
3;1701823287;5;141;9;134.6
2;1701823262;15;160;2;21.6
1;1701823242;11;161;4;34.50
3;1701823287;5;165;2;69.0
1;1701823242;11;165;1;34.50
2;1701823262;15;165;1;34.50
1;1701823242;11;198;1;34.50
2;1701823262;15;234;1;34.50
3;1701823287;5;456;6;31.0
```

onde:

- CodPedido** é o código do pedido. Há a garantia que um mesmo código de pedido não se repete em dois arquivos CSV diferentes

- **DtPedido** é um inteiro longo que representa a data do pedido no formato EpochTime (<https://www.epochconverter.com>). **Não há garantia** de que os pedidos estarão ordenados no arquivo por data, como pode ser observado no exemplo acima, onde estão ordenados por filial.
- **CodFilial** é o código da filial onde o pedido foi feito
- **CodProduto** é o código do produto. Pode ocorrer de haver dois itens no mesmo pedido com o mesmo código de produto.
- **Qtd** é a quantidade vendida do produto
- **VIUnit** é o valor unitário do produto

Seu sistema deve ser capaz de:

1. [6.0 pontos]

- a. [4.5 pontos] Realizar a carga dos arquivos CSV de pedidos e manter um outro arquivo CSV (rel\_pedidos.csv) com os dados consolidados, no formato abaixo:

```
CodPedido;DtPedido;CodFilial;QtdTotalItens;TotalPedido
1;1701823242;11;8;338.08
2;1701823262;15;4;112.20
3;1701823287;5;17;1535.40
```

Onde **QtdTotalItens** é o total de itens vendidos (soma das quantidades) e **TotalPedido** é o valor total do pedido.

- b. [1.5 pontos] Manter os pedidos **ordenados por data** no relatório.

2. [3.0 pontos] Gerar relatório de faturamento por filial. Este relatório deve ser atualizado sempre que uma nova carga for realizada. O relatório deve ser um CSV (rel\_filiais.csv) seguindo o formato abaixo:

```
CodFilial;TotalItensVendidos;FatTotal
11;8;338.08
15;4;112.20
5;17;1535.40
```

Onde **TotalItensVendidos** é a soma de todos os itens vendidos em todos os pedidos daquela filial e **FatTotal** é a soma do valor total de todos os pedidos feitos naquela filial.

**Requisitos e regras adicionais:**

- [1.0 ponto] O programa deve estar devidamente modularizado em funções, evitando a duplicação de código e o uso de variáveis globais.
- Menu do programa: seu programa deve ter apenas duas opções: 1) Realizar carga, onde o usuário irá digitar o nome do arquivo CSV para ser utilizado na carga e 2) Encerra. Ao realizar uma carga, tanto o relatório de pedidos quanto o de filiais devem ser atualizados.
- Os arquivos de carga CSV contêm um **número indeterminado de linhas** (podendo exceder 1 milhão). Logo, as estruturas de dados devem ser alocadas dinamicamente. A capacidade inicial deve ser 50, e caso necessário mais espaço, uma nova estrutura com o triplo do tamanho deve ser alocada. Não é permitido alocar estruturas com tamanhos exorbitantes. Também não é permitido o uso de listas dinâmicas neste projeto.

Bom trabalho!