Desafio BHS

Utilizando um projeto desenvolvido em .net Core 3.1 ou .net 5.0, desenvolva a seguinte solução:

1. Faça a leitura do arquivo Desafio.txt, com base no layout informado abaixo.
2. Cada linha do arquivo tem um tipo, as linhas do tipo 1 são os pedidos de compra, as linhas do tipo 2 são os itens dessa compra.
3. Grave as linhas do tipo 1 na tabela PEDIDOS.
4. Grave as linhas do tipo 2 na tabela ITENS.
5. Crie uma tela onde se possa ver todos os pedidos, que seja possível filtrar para encontrar um determinado pedido, que seja possível ver os itens desse pedido, que seja possível cadastrar um novo pedido, e excluir um pedido alterando o campo STATUS de ‘A’ para ‘E’ e não pode apagar fisicamente o registro, e que não permita atualizar nenhum outro campo ou informação.
6. Ao cadastrar um novo pedido, desenvolver uma funcionalidade que seja possível incluir novos itens, e fechar o pedido. Uma vez fechado, não se altera mais, permitindo somente a exclusão semelhante na regra 5.
7. O arquivo Desafio.txt tem que estar dentro de uma pasta que por sua vez está dentro do projeto ou Solution.
8. Será disponibilizado apenas o arquivo Desafio.txt, e o Script de criação da base de dados e tabelas.

Regras de Avaliação:

* **Tempo é avaliado**: O primeiro a entregar ganha 100 pontos. O segundo: 90 pontos. O terceiro 80 pontos, e a cada entrega subsequente, vai perdendo de 10 em 10 pontos.
* **Qualidade é avaliado**: Para cada bug encontrado, perde-se 5 pontos.
* **Escopo é avaliado**: Para cada regra de escopo definida acima não realizada, perde-se 10 pontos.
* **Capricho é avaliado**: um código bem organizado e bem escrito, identado, com padrões de nomes de variáveis e métodos, avaliado pelos jurados, recebe bônus de 50 pontos. Todos têm chance de ganhar seus 50 pontos.
* **Criatividade é avaliado**: assim como capricho, existem muitas maneiras de trazer essa solução, ser criativo e simples pode ser uma ótima alternativa, também na avaliação dos jurados, esse quesito vale 50 pontos. Todos têm chance.
* **Pró atividade é avaliado**: Não foi solicitado aqui uma aparência bonita, nem funcionalidades extras. Aos que investirem tempo em oferecer algo a mais, pode receber o dobro dos pontos que conquistou do quesito TEMPO. O multiplicador só se aplica se não houver penalidades do quesito ESCOPO.

Contratação:

* Todos têm chances. A régua de contratação é 120 pontos. Portanto, não faça correndo apenas para ser o primeiro, as vezes a correria pode comprometer a qualidade e perder muitos pontos por bugs e falhas de implementação do escopo. Capricho, Criatividade e Pró atividade serão necessários para alcançar essa margem de pontos.
* Se você for o oitavo a entregar, você recebe 20 pontos, mas se for sem bugs, com todo o escopo, caprichado e criativo, chega a 120 pontos se bem avaliado, e se receber o dobro do tempo, ou seja, 40, tendo o total de 140 pontos. Ou seja, todos tem chance de serem contratados.

Premiação

* A melhor pontuação recebe uma Alexa 4ª Geração
* A segunda melhor pontuação recebe uma Alexa 3ª Geração
* A terceira melhor pontuação recebe um Fone JBL
* Todos receberão kits.

Layout do arquivo

|  |  |  |
| --- | --- | --- |
| Campo | Tamanho | Tipo |
| TipoLinha 1 | 1 | Numérico |
| CODIGO | 5 | Numérico |
| DESCRICAO | 40 | Alfanumérico |
| DATAPEDIDO | 12 | Data Formato aaaaMMddHHmmss |
| VALORPEDIDO | 10 | Valor Formato 0.00 |
| STATUS | 1 | Alfanumérico |

|  |  |  |
| --- | --- | --- |
| Campo | Tamanho | Tipo |
| TipoLinha 2 | 1 | Numérico |
| CODIGO | 5 | Numérico |
| CODIGOPEDIDO | 5 | Numérico |
| DESCRICAO | 50 | Alfanumérico |
| VALORITEM | 10 | Valor Formato 0.00 |

O Script abaixo cria as duas tabelas para receber os dados:

CREATE TABLE [dbo].[ITENS](

[CODIGO] [int] NOT NULL,

[CODIGOPEDIDO] [int] NOT NULL,

[DESCRICAO] [varchar](50) NOT NULL,

[VALORITEM] [numeric](13, 2) NOT NULL,

CONSTRAINT [PK\_ITENS] PRIMARY KEY CLUSTERED

(

[CODIGO] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

CREATE TABLE [dbo].[PEDIDOS](

[CODIGO] [int] NOT NULL,

[DESCRICAO] [varchar](40) NOT NULL,

[DATAPEDIDO] [datetime] NOT NULL,

[VALORPEDIDO] [numeric](13, 2) NOT NULL,

[STATUS] [char](1) NOT NULL,

CONSTRAINT [PK\_PEDIDOS] PRIMARY KEY CLUSTERED

(

[CODIGO] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[ITENS] WITH CHECK ADD CONSTRAINT [FK\_ITENS\_PEDIDOS] FOREIGN KEY([CODIGOPEDIDO])

REFERENCES [dbo].[PEDIDOS] ([CODIGO])

GO

ALTER TABLE [dbo].[ITENS] CHECK CONSTRAINT [FK\_ITENS\_PEDIDOS]

GO