

LOJA – ESPECIFICAÇÕES DA INTERFACE
PROFESSOR: ADELARDO ADELINO DANTAS DE MEDEIROS

AS CLASSES Qt

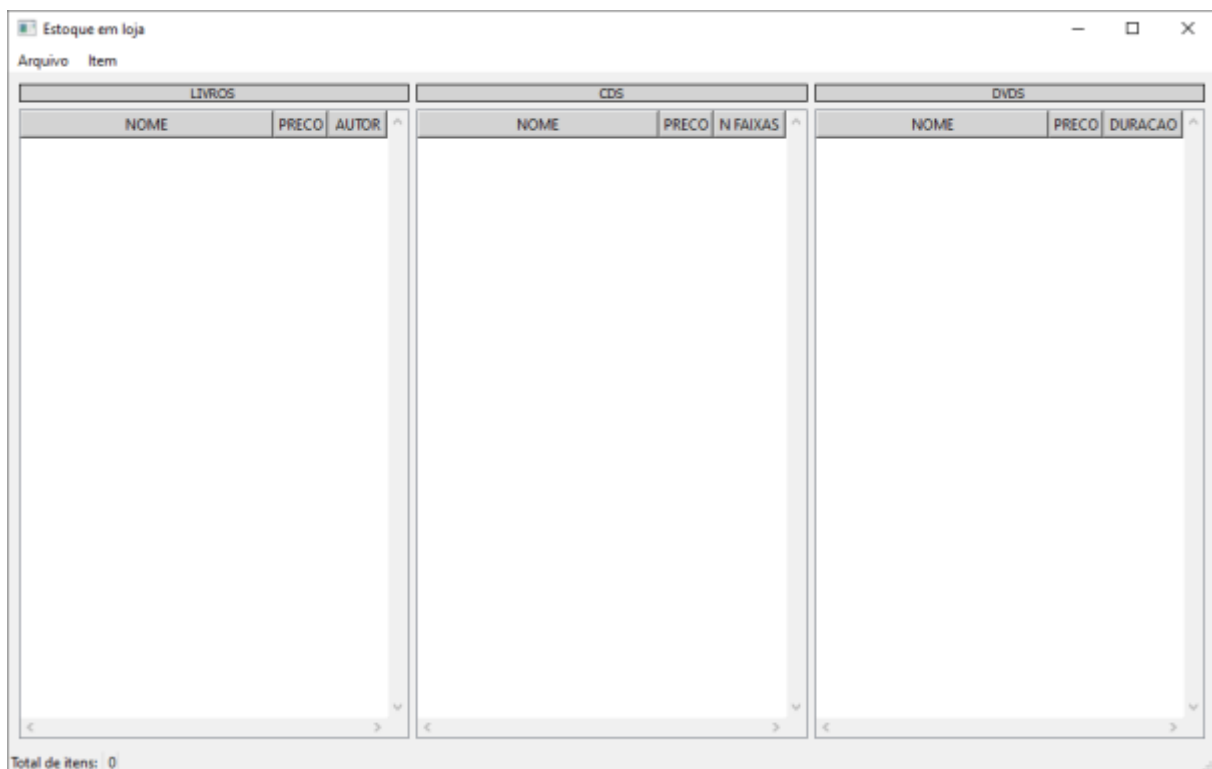
Para representação das 4 janelas que compõem a interface do programa Loja (a principal e as 3 janelas que aparecem quando necessário), devem ser criadas, usando o Qt Creator, 4 classes Qt, com os respectivos arquivos .cpp, .h e .ui:

- **MainLoja**: janela principal do programa, sempre em exibição
- **IncluirLivro**: janela normalmente oculta, exibida quando o usuário inclui um novo livro.
- **IncluirCD**: janela normalmente oculta, exibida quando o usuário inclui um novo CD.
- **IncluirDVD**: janela normalmente oculta, exibida quando o usuário inclui um novo DVD.

A CLASSE **MainLoja**

A janela principal do aplicativo deve se basear na classe **QMainWindow** do Qt, com dimensões 1000x600. No widget central, os objetos principais são três **QTableWidget**: a da esquerda deve apresentar a lista de Livros; a central, a lista de CDs; e a da direita, a lista de DVDs. Cada tabela tem 3 colunas: o nome do produto; o preço (exibido em ponto flutuante, sempre com 2 decimais); e a informação adicional (autor para livros, número de faixas para CDs e duração para DVDs).

Acima de cada tabela, um **QLabel** mostra o tipo de objetos da lista. Cada **QLabel** deve estar alinhado horizontalmente com sua tabela correspondente; após isso, a **QMainWindow** como um todo deve ser alinhada verticalmente. A janela principal deve ter o título “Estoque em loja” (ver figura).



Cada tabela exibe linhas entre as células (propriedade **showGrid** é verdadeira) e tem (inicialmente) 0 linhas e 3 colunas (propriedades **rowCount** e **columnCount** têm valor 0 e 3, respectivamente). As tabelas têm cabeçalho horizontal e não têm cabeçalho vertical (as propriedades **horizontal-**

`HeaderVisible` e `verticalHeaderVisible` devem ser `true` e `false`, respectivamente). As barras de rolagem horizontal e vertical estão sempre visíveis (propriedades `horizontalScrollBarPolicy` e `verticalScrollBarPolicy` devem ser `true`). A navegação por TAB entre itens da tabela deve estar desabilitada (`tabKeyNavigation` `false`). O modo de seleção de itens na tabela deve ser ajustado para só selecionar um item por vez, sem selecionar mais de uma opção com SHIFT ou CTRL (`selectionMode` `SingleSelection`) e para que, ao se selecionar um item, toda a linha seja selecionada (`selectionBehavior` `SelectRows`).

Devem ser criados slots que reajam à seguinte ação nas três tabelas:

- `on_..._cellDoubleClicked(int row, int column)`: uma célula em uma das tabelas recebeu um clique duplo. Ao se dar um clique duplo em uma linha de uma tabela, o objeto correspondente deve ser excluído do estoque da loja.

Na barra de status, são exibidos 2 `QLabel`:

- O primeiro exibe o conteúdo estático (imutável) "Total de itens: ".
- O segundo tem seu valor alterado, quando necessário, para exibir o total de itens no estoque (n° de livros + n° de CDs + n° de DVDs).

OS DADOS DA CLASSE `MainLoja`

Os objetos da classe `MainLoja` têm os seguintes dados membro, todos privados:

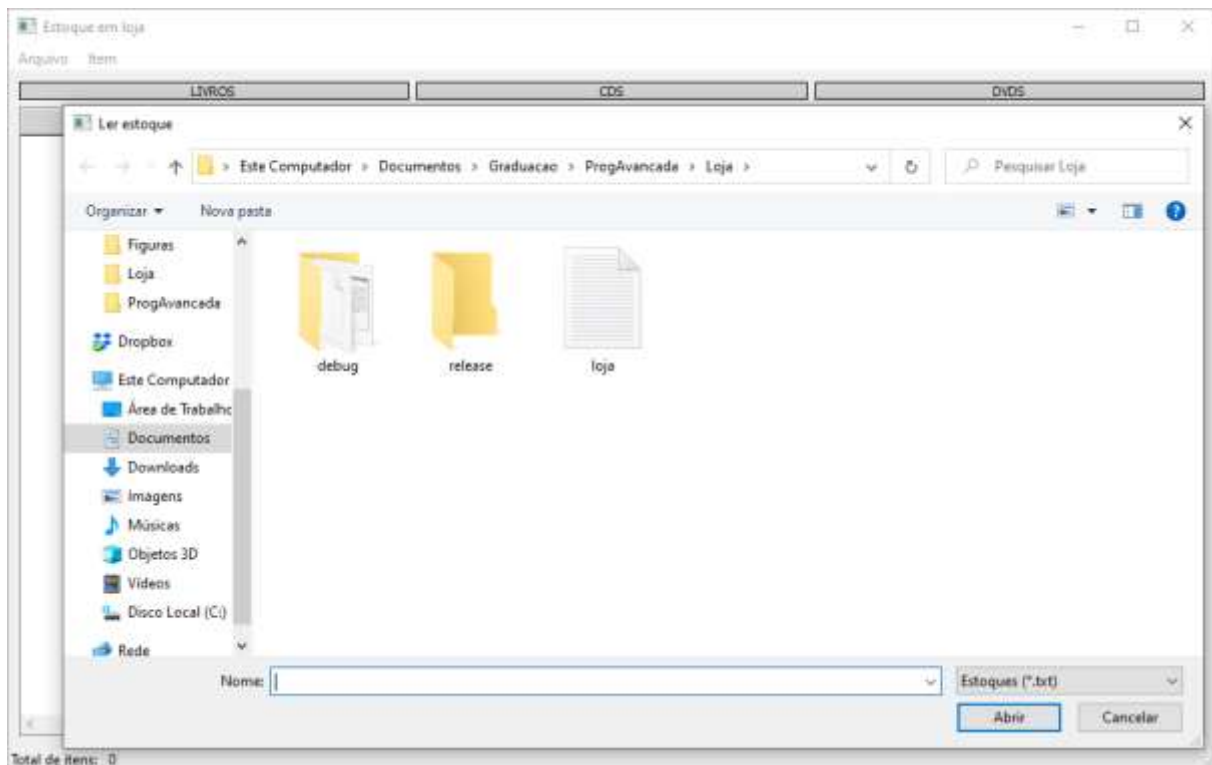
- Um objeto `X` do tipo `Loja`, que é quem armazena efetivamente as informações do estoque.
- Um ponteiro `inclLivro` do tipo `IncluirLivro*`, que aponta para o objeto alocado dinamicamente que representa a janela para incluir livros.
- Um ponteiro `inclCD` do tipo `IncluirCD*`, que aponta para o objeto alocado dinamicamente que representa a janela para incluir CDs.
- Um ponteiro `inclDVD` do tipo `IncluirDVD*`, que aponta para o objeto alocado dinamicamente que representa a janela para incluir DVDs.
- Um ponteiro `total_itens` do tipo `QLabel*`, que aponta para o `QLabel` alocado dinamicamente que deve ser inserido na barra de status para exibir o total de itens do estoque.

OS MENUS DA CLASSE `MainLoja`

- A barra de menus contém dois menus, denominados "Arquivo" e "Item". Os menus seguem a convenção de que as opções que geram a abertura de uma nova janela são seguidas por "...".
- O menu Arquivo tem as opções "Ler..." (lê o estoque a partir de um arquivo) e "Salvar..." (salva o estoque em um arquivo), seguidas de um separador, após o qual está a opção "Sair" (encerra o aplicativo). Para cada uma dessas ações, deve ser criado um slot correspondente:
 - `on_actionLer_triggered()`: gera a abertura de uma janela do tipo `QFileDialog::getOpenFileName`¹ com título "Ler estoque" (ver figura) e lê os dados utilizando o método `ler` da classe `Loja`. Em caso de erro na leitura do arquivo, exibe mensagem de erro abrindo uma `QMessageBox::critical`². Ao final, atualiza na interface as listas de itens.
 - `on_actionSalvar_triggered()`: gera a abertura de uma janela do tipo `QFileDialog::getSaveFileName`¹ com título "Salvar estoque" e salva os dados utilizando o método `salvar` da classe `Loja`. Em caso de erro na escrita do arquivo, exibe mensagem de erro abrindo uma `QMessageBox::critical`².
 - `on_actionSair_triggered()`: chama `QCoreApplication::quit()`².

¹ Tome como modelo o programa "Exemplo 3 – Notepad" da seção sobre Qt.

² Veja, no resumo sobre Qt, mais detalhes sobre todas as funções necessárias.



- O menu Item tem as opções “Incluir Livro...”, “Incluir CD...” e “Incluir DVD...”, que incluem no estoque um novo item do tipo correspondente. Para cada uma dessas ações, deve ser criado um slot correspondente:
 - `on_actionIncluir_livro_triggered()`: `exibe (show2)` o objeto `inclLivro`.
 - `on_actionIncluir_CD_triggered()`: `exibe (show2)` o objeto `inclCD`.
 - `on_actionIncluir_DVD_triggered()`: `exibe (show2)` o objeto `inclDVD`.

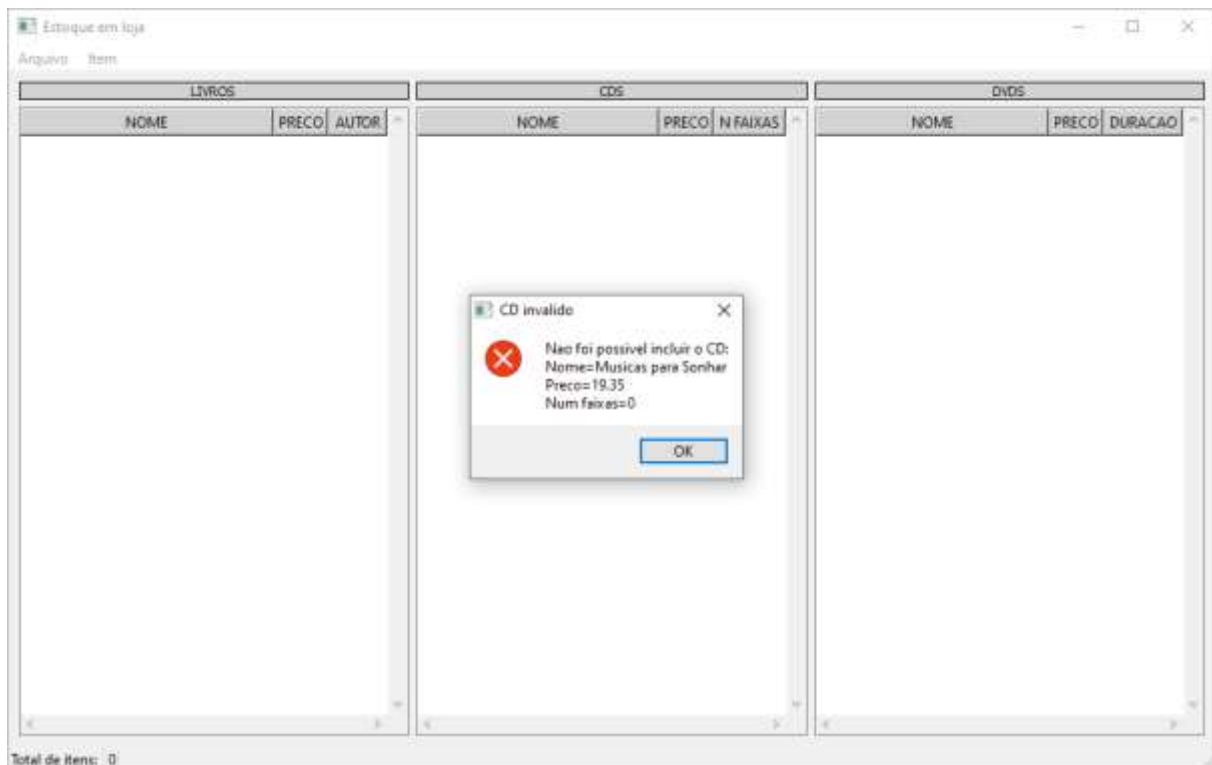
OS SLOTS DA CLASSE `MainLoja`

Além dos slots criados pelo Qt Creator (`on_...`) para reagir a eventos padronizados dos widgets Qt, a classe `MainLoja` tem 3 slots privados adicionais a serem criados manualmente, para incluir objetos no estoque. Cada um desses slots recebe 3 parâmetros de entrada do tipo `QString`:

- `slotIncluirLivro(QString nome, QString preco, QString autor)`
- `slotIncluirCD(QString nome, QString preco, QString numfaixas)`
- `slotIncluirDVD(QString nome, QString preco, QString duracao)`

Esses slots realizam as seguintes operações:

- Convertem os parâmetros `QString` para o tipo apropriado para a classe `Livro`, `CD` ou `DVD`:
 - `QString` para `string C++`: `QString::toStdString()`²
 - `QString` para `float`: `QString::toFloat()`²
 - `float` para `unsigned` (preço para centavos): multiplica por `100.0` e `round()`
- Criam um objeto (`Livro`, `CD` ou `DVD`) e incluem esse objeto no estoque usando a função apropriada da classe `Loja` (`incluirLivro`, `incluirCD` ou `incluirDVD`).
- Em caso de sucesso na criação do objeto e inclusão, reexibem a lista correspondente na interface. Em caso de insucesso, exibem uma mensagem de erro com uma `QMessageBox::critical`², conforme a figura.



INICIALIZAÇÃO – CONSTRUTOR DA CLASSE MainLoja

Ao iniciar o programa, o construtor da classe `MainLoja` deve executar as seguintes operações:

- Chamar os construtores apropriados para inicializar corretamente todos os dados membros.
- Criar dinamicamente (usando `new`) os objetos que representam as 3 janelas de inclusão de itens e armazená-los nos ponteiros `inclLivro`, `inclCD` e `inclDVD`. Os objetos devem ser criados, no sistema parental do Qt, tendo `this` como pai.
- Criar dinamicamente (usando `new`) o `QLabel` que vai exibir o total de itens do estoque e armazená-lo no ponteiro `total_itens`. O `QLabel` deve ser criado, no sistema parental do Qt, tendo `this` como pai.
- Inserir na barra de status (insertWidget¹) os 2 `QLabel` necessários³:
 - O primeiro, que não precisa ter nome, tem o texto estático "Total de itens: ".
 - O segundo é `total_itens`.
- Conectar os sinais `signIncluirLivro`, `signIncluirCD` ou `signIncluirDVD` das classes `IncluirLivro`, `IncluirCD` ou `IncluirDVD` (ver na seção específica) com os slots apropriados da classe `MainLoja` (`slotIncluirLivro`, `slotIncluirCD` ou `slotIncluirDVD`).
- Alterar a cor de fundo (`background`) dos `QLabel` acima das 3 tabelas para `lightgray`, usando `setStyleSheet`².
- Alterar a cor de fundo (`background`) dos cabeçalhos das 3 tabelas para `lightgray`, usando `setStyleSheet`².
- Alterar o comportamento das colunas das 3 tabelas usando `setSectionResizeMode`² para que a 2ª e 3ª colunas se ajustem ao conteúdo (`QHeaderView::ResizeToContents`²) e a 1ª coluna estique para completar o espaço (`QHeaderView::Stretch`²).
- Fixar o texto dos cabeçalhos das 3 tabelas, utilizando `setHorizontalHeaderLabels`².
- Exibir o conteúdo inicial das 3 listas, que deve ser vazio.

³ Tome como modelo o programa "Jogo da Velha" da seção sobre Qt

EXIBINDO UMA LISTA DE ITENS NA TABELA

Para exibir a lista de livros, CDs ou DVDs na `QTableWidget` correspondente (ver figura), devem ser dados os seguintes passos:

- Limpar o conteúdo atual da tabela, exceto os cabeçalhos (`clearContents2`).
- Fixar o novo número de linhas da tabela (`setRowCount2`) de acordo com o número de itens da lista correspondente na Loja (`getNumLivro`, `getNumCD` ou `getNumDVD`).
- Para cada coluna (`j` de 0 a 2) de cada linha (`i` de 0 a `Nº de itens - 1`) da tabela:
 - Criar um novo `QLabel` dinâmico (`prov = new QLabel`).
 - Alterar o conteúdo de `prov` (`setText2`, `setNum2`) para refletir o conteúdo obtido pelas funções de consulta das classes `Livro`, `CD` ou `DVD` (`getNome()`, `getPreco()`, `getAutor()`, etc.) aplicadas ao `i`-ésimo elemento da lista correspondente no estoque, recuperado através das funções de consulta da Loja (`getLivro`, `getCD` ou `getDVD`).
 - Alterar o alinhamento de `prov` (`setAlignment2`): centralizado, alinhado à esquerda, etc.
 - Incluir `prov` na tabela (`setCellWidget(i, j, prov)2`).
- Reexibir o valor apresentado na barra de status.

EXIBINDO TOTAL DE ITENS NA BARRA DE STATUS

- Para exibir o total de itens no estoque (ver figura), é preciso alterar o conteúdo (`setNum1`) do `QLabel` `total_itens`, que está fixado na barra de status. O valor a ser apresentado deve ser a soma do total de livros, CDs e DVDs do estoque, obtidos com os métodos de consulta da classe Loja: `getNumLivro()`, `getNumCD()` e `getNumDVD()`.

LIVROS			CDs			DVDs		
NOME	PRECO	AUTOR	NOME	PRECO	N FAIXAS	NOME	PRECO	DURAÇÃO
Memorial	32.34	Pedro	Soldado	27.86	12	Combate VI	14.38	126
Minha Vida	2.86	Adolfo	Verso Azul	8.52	14	Beroco	8.52	72
Poemas	13.14	João de Sa				Nunca Mais!	11.00	185
						Bu, Tu, Eles	7.00	134

Total de Itens: 9

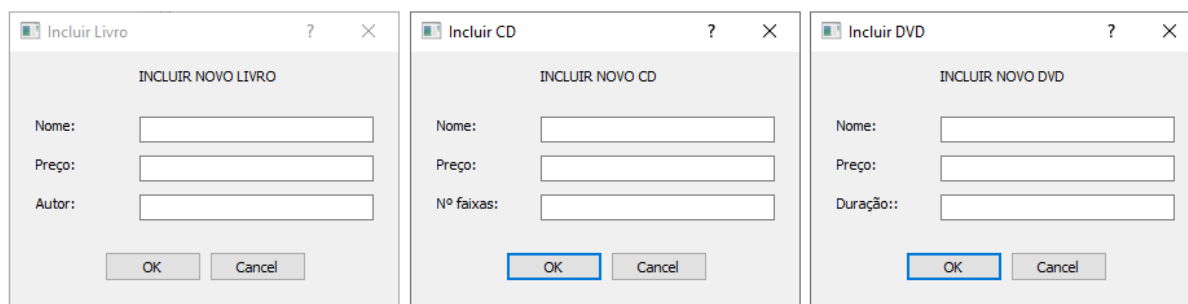
CLICANDO DUPLO EM UM ITEM

Ao dar um clique duplo em um item, ele deve ser removido da lista correspondente⁴. Para isso, um slot `on_..._cellDoubleClicked(row, column)` deve reagir a esse sinal, executando as seguintes operações:

- Desconsidera o parâmetro `column`.
- Verifica se a linha selecionada (`row`) está na faixa válida de índices para a lista do tipo correspondente (livro, CD ou DVD), o que sempre deve ocorrer se não houve erro de programação.
- Exclui o item, usando a função da Loja (`excluirLivro`, `excluirCD` ou `excluirDVD`).
- Em caso de sucesso na exclusão, reexibem a lista correspondente na interface.

A CLASSE `IncluirLivro`⁵

A janela de diálogo para inclusão de novo item deve se basear na classe `QDialog` do Qt, com dimensões 300x200. Na parte inferior, os botões (OK e Cancel) aparecem centralizados. Na parte superior, um `QLabel` mostra o texto "INCLUIR NOVO LIVRO". Na parte central, aparecem 3 linhas de entrada de dados, cada uma delas com um `QLabel` e uma `QLineEdit`. Os `QLabel` estão alinhados verticalmente entre si (mesma coordenada X = 20) e igualmente espaçados (30 pixels de diferença na coordenada Y). As `QLineEdit` estão alinhados verticalmente entre si (mesma coordenada X = 100) e alinhadas horizontalmente com os `QLabel` correspondentes (mesma coordenada Y). A janela de diálogo deve ter o título "Incluir Livro" (ver figura).



OS SINAIS DA CLASSE `IncluirLivro`

A classe `IncluirLivro` deve incluir um sinal `signIncluirLivro`, com 3 parâmetros (nome, preço, autor) do tipo `QString`, a ser emitido quando for necessário sinalizar para a interface principal que um novo livro deve ser incluído.

OS SLOTS DA CLASSE `IncluirLivro`

Deve ser criado um slot `on_buttonBox_accepted()` para quando o botão OK for acionado. Esse slot deve executar as seguintes operações:

- Ler o conteúdo atual das 3 `QLineEdit` (`ui->...->text()`²) em 3 variáveis `QString`.
- Emitir um sinal `signIncluirLivro`, tendo como parâmetro as 3 `QString`.

⁴ Normalmente, em um aplicativo real, antes de excluir deveria ser exibida uma janela de advertência para que o usuário confirmasse a exclusão. Aqui isso não está sendo exigido para não tornar muito grande o projeto, mas fica como tarefa opcional para aqueles que desejarem, utilizando uma janela pop-up `QDialog`.

⁵ A classe `IncluirLivro`, que vai ser detalhada, serve como modelo para as classes `IncluirCD` e `IncluirDVD`, que são praticamente idênticas.