

Escola Técnica Estadual “São Paulo” - ETESP

Linguagem C# - Visual Studio 2019

Em nosso próximo projeto continuaremos utilizando o acesso desconectado, porém, sem a utilização do assistente. Faremos a programação de todo o código necessário para Incluir, Alterar e Excluir dados. Veremos também a utilização de “tabelas relacionadas”.


Então, vamos lá!!

Etapa 01: Criação das tabelas: Você poderá incluí-las no Banco de Dados já existente ou, se preferir, criar um novo banco de dados.

I- Tabela de Fornecedores (nome → Fornecedores)

Campo	Tipo	Observação
CodFornec	Número	Campo Chave
RazaoSocial	Texto Curto	Tamanho 50; Requerido

OBS: Cadastre três fornecedores:



CodFornec	RazaoSocial	Clique para Adicionar
1	Fornecedor 01	
2	Fornecedor 02	
3	Fornecedor 03	
*		

II - Tabela de Produtos (nome → Produtos). OBS: Haverá um “relacionamento” com a tabela “Fornecedores”

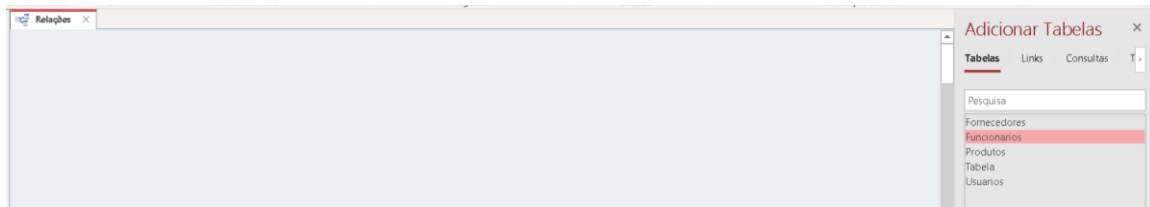
Campo	Tipo	Observação
CodProd	Número	Campo Chave
Descricao	Texto Curto	Tamanho 50; Requerido
CodFornec (*)	Número	Relacionado com o campo “CodFornec” da Tabela de Fornecedores / Requerido
QtdeEstoque	Número	Regra de validação ≥ 0
QtdeMinima	Número	Regra de validação ≥ 0
QtdeMaxima	Número	Regra de validação ≥ 0
ValorUnitario	Moeda	Regra de validação > 0

(*) Nota: Após criar a estrutura da tabela acima e antes de cadastrar os produtos, precisaremos criar o “relacionamento” entre as tabelas: Fornecedores e Produtos. Clique no menu “Ferramentas de Bancos de Dados” → Relações

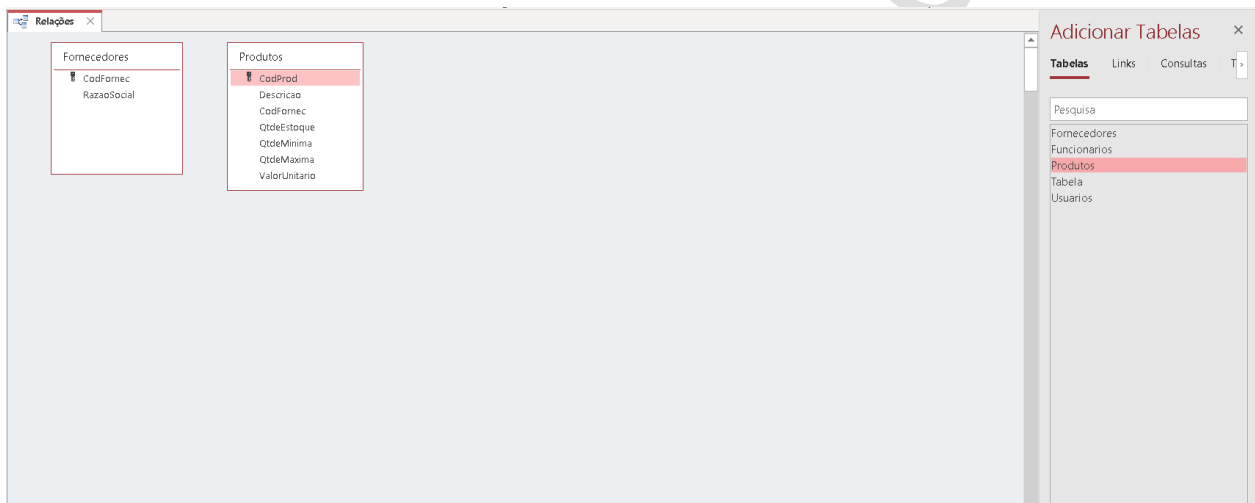
Prof. Roberto de Castro



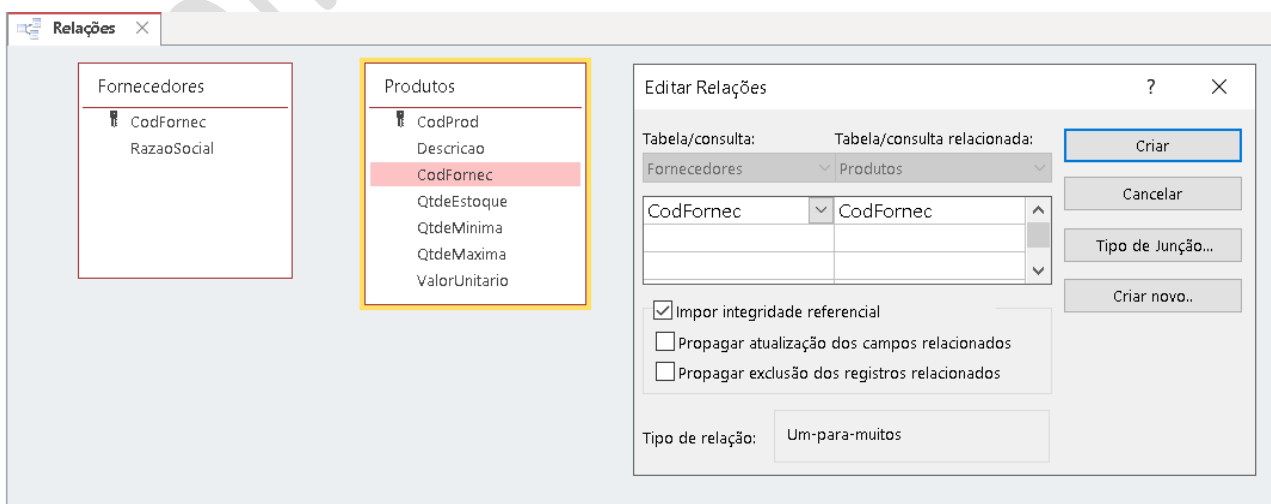
Será exibida uma janela “em branco” com a opção “Adicionar Tabelas”.



Vamos incluir as duas tabelas: Fornecedores e Produtos. No item “Adicionar Tabelas” dê dois cliques na tabela “Fornecedores” e dois cliques na tabela “Produtos”:

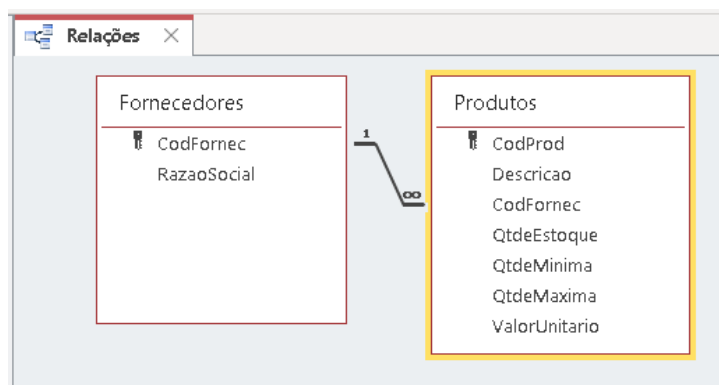


Selecione (dê um clique) o campo “CodFornec” da tabela “Produtos” e “arraste” para o campo “CodFornec” da tabela “Fornecedores”. Será exibida a caixa “Editar Relações”:



Prof. Roberto de Castro

Observe o tipo de relacionamento: Um-para-muitos, que significa: um (e apenas um) código de fornecedor cadastrado na tabela de Fornecedores poderá ser cadastrado muitas vezes na tabela de Produtos. Vamos assinalar a caixa “Impor integridade referencial” (para garantir que um fornecedor na tabela de fornecedores NÃO poderá ser excluído, caso existam produtos cadastrados com o código dele. Clique em Criar e observe o “diagrama”:



Agora podemos cadastrar três produtos (tomando o cuidado de digitar o código de fornecedor existente).

CodProd	Descricao	CodFornec	QtdeEstoque	QtdeMinima	QtdeMaxima	ValorUnitario	Clique para Adicionar
1	Produto A	1	10	5	100	R\$100,00	
2	Produto B	2	20	10	200	R\$200,00	
3	Produto C	3	30	15	300	R\$300,00	
*		0	0	0	0	R\$0,00	

Etapa 01 concluída!

Vamos para a Etapa 02... Guarde muito bem este Banco de Dados pois será utilizado nas próximas atividades.

Etapa 02: Criação do projeto no Visual Studio (Exe037A BD03)

Abra o Visual Studio, crie o projeto “Windows Form” (.Net Framework)

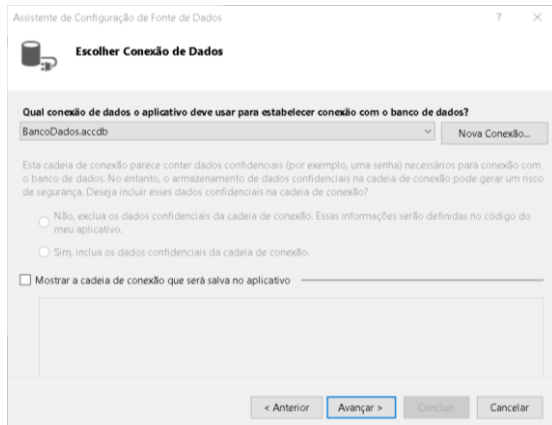
Inicialmente faremos a configuração/criação da fonte de dados.. **Clique no link “Adicionar Nova Fonte de Dados”.**

Os detalhes das próximas etapas já foram tratados anteriormente, porém, para “agilizar” o desenvolvimento deste projeto, mostrarei apenas os “prints” das telas:

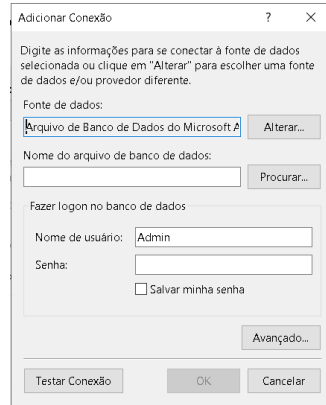
Clique em “Avançar”

Clique em “Avançar”

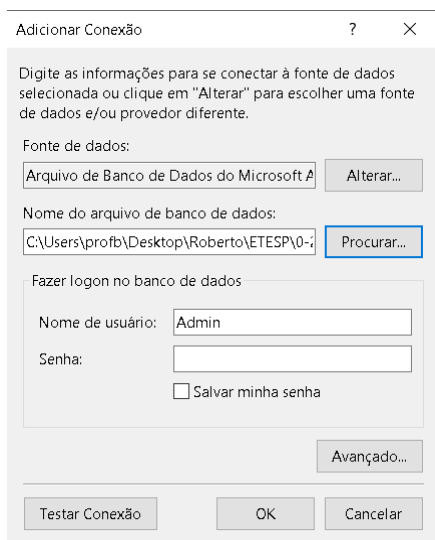
Prof. Roberto de Castro



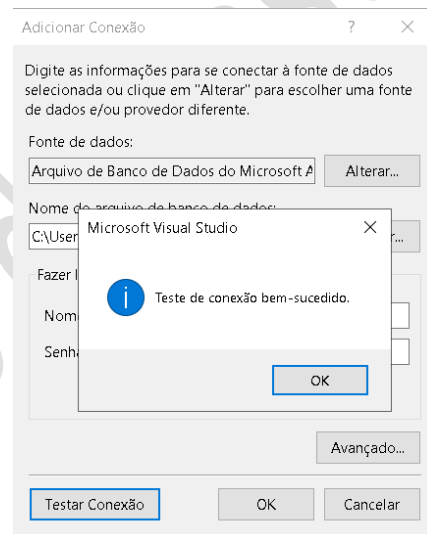
Clique em Nova Conexão



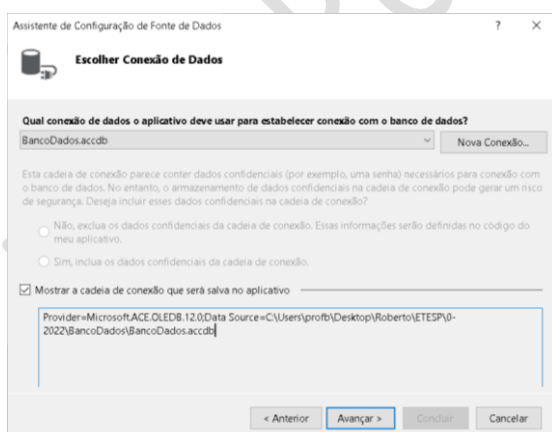
Clique em "Procurar" e indique o local e o nome do Banco de Dados



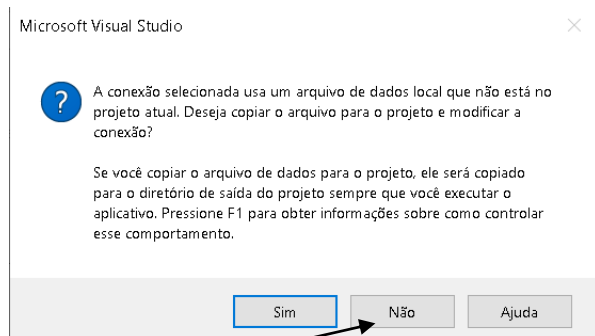
Clique em "Testar Conexão"



Clique em "OK" na "caixa de diálogo" e novamente em "OK" na janela "Adicionar Conexão"

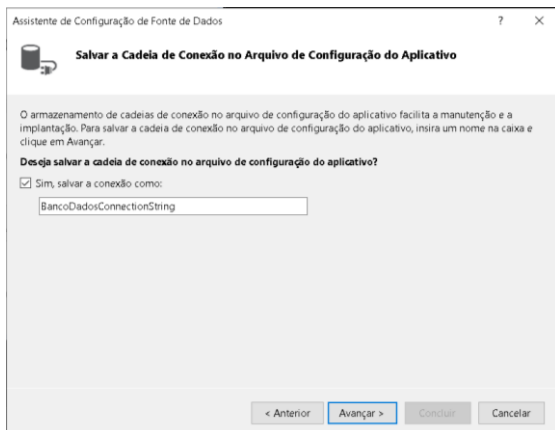


Clique em "Avançar"



Clique em "Não" para não gerar uma cópia do Banco de Dados

Prof. Roberto de Castro

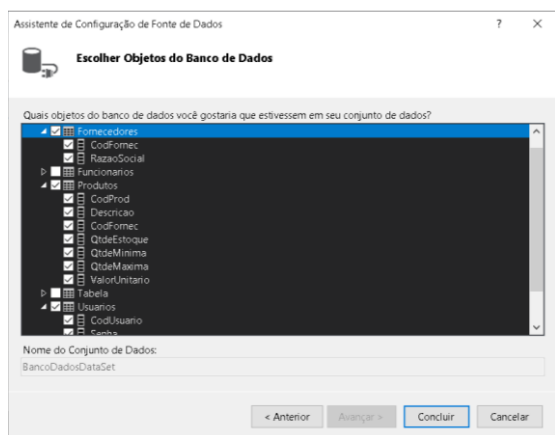


Clique em “Avançar”

Muita atenção na próxima etapa....

Faremos a conexão com a fonte de dados e já informaremos todas as tabelas que serão necessárias para o desenvolvimento do projeto:

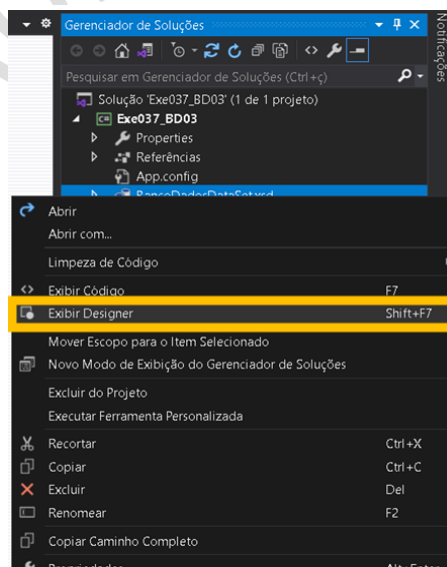
[Tabela de Fornecedores e Tabela de Produtos!](#)

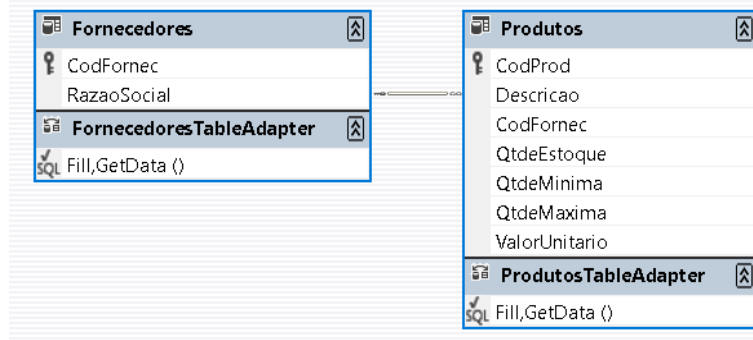


Clique nas tabelas Fornecedores e Produtos (marque todos os campos destas tabelas).

Clique em “Concluir”

Vamos “exibir” o diagrama gerado para o nosso DataSet. No “Gerenciador de Soluções” (Solution Manager), clique com o botão direito do mouse em “BancoDadosDataSet” e selecione a opção “Exibir designer”:





Perceba, no diagrama acima, a “seta” indicando o relacionando entre as tabelas: Fornecedores e Produtos!!!

O primeiro formulário será utilizado para fazer o cadastro dos Produtos (Incluir, Excluir, Alterar). Altere o nome do formulário para FrmProdutos.

INTERFACE GRÁFICA

A interface gráfica, intitulada "Cadastro de Prontos", é dividida em três seções principais:

- Dados Cadastrais:** Contém campos de entrada para "Codigo do Produto:" (TxtCodProd), "Descrição do Produto:" (TxtDescProd), "Nome do Fornecedor:" (CboRazaoSocial), "Quantidade Estoque:" (TxtQtdeEstoque), "Quantidade Mínima:" (TxtQtdeMinima), "Quantidade Máxima:" (TxtQtdeMaxima) e "Valor Unitário R\$:" (TxtValorUnitario).
- Operações:** Um painel com botões para "NOVO", "GRAVAR", "EDITAR", "EXCLUIR" e "LISTAR".
- Navegação:** Um painel com botões de navegação: "<", "<", ">", ">".

Abaixo da interface, há uma série de botões de controle:

- BtnNovo
- BtnGravar
- BtnEditar
- BtnExcluir
- BtnListar
- BtnFechar
- BtnUltimo
- BtnProximo
- BtnAnterior
- BtnPrimeiro

Considerações sobre a interface gráfica:

1. Inicialmente, as caixas de texto (TextBox) estarão “travadas” (propriedade Enabled = false). O objetivo é garantir uma segurança nas informações. Se o usuário necessitar/desejar alterar alguma informação, deverá clicar na opção “Editar” e as “caixas de texto” ficarão disponíveis para alteração.

Prof. Roberto de Castro

2. Observe que incluímos um “ComboBox”, que estará vinculado à tabela de fornecedores e exibirá uma lista com todos os fornecedores cadastrados. Ao selecionar um nome de fornecedor na lista, o próprio “ComboBox” enviará o “código do fornecedor” para a “tabela de produtos”. (veremos a configuração logo abaixo).
3. As caixas de texto estarão vinculadas à “tabela de produtos” (ver configuração da propriedade DataBindings).

PRINCIPAIS PROPRIEDADES QUE DEVERÃO SER ALTERADAS NOS CONTROLES

CONTROLE	PROPRIEDADE	CONTEÚDO
Form	Name	FrmProdutos
	MaximizeBox	False
	StartPosition	CenterScreen
	Text	Cadastro de Produtos
TextBox “TxtCodProd”	Name	TxtCodProd
	DataBindings (*)	Ver procedimento abaixo
	MaxLength	5
	Enabled	False
	TabIndex	0
TextBox “TxtDescProd”	Name	TxtDescProd
	DataBindings (*)	Ver procedimento abaixo
	MaxLength	50
	Enabled	False
	TabIndex	1
ComboBox “CboRazaoSocial”	Name	CboRazaoSocial
	DataBindings (**)	Ver procedimento abaixo
	Enabled	False
	TabIndex	2
TextBox “TxtQtdeEstoque”	Name	TxtQtdeEstoque
	DataBindings (*)	Ver procedimento abaixo
	MaxLength	5
	Enabled	False
	TabIndex	3
TextBox “TxtQtdeMinima”	Name	TxtQtdeMinima
	DataBindings (*)	Ver procedimento abaixo

Prof. Roberto de Castro

	MaxLength	5
	Enabled	False
	TabIndex	4
TextBox "TxtQtdeMaxima"	Name	TxtQtdeMaxima
	DataBindings (*)	Ver procedimento abaixo
	MaxLength	5
	Enabled	False
	TabIndex	5
TextBox "TxtValorUnitario"	Name	TxtValorUnitario
	DataBindings (*)	Ver procedimento abaixo
	MaxLength	10
	Enabled	False
	TabIndex	6
Button "BtnNovo"	Name	BtnNovo
	Text	&Novo
Button "BtnGravar"	Name	BtnGravar
	Enabled	False
	Text	&Novo
Button "BtnEditar"	Name	BtnEditar
	Text	&Editar
Button "BtnExcluir"	Name	BtnExcluir
	Text	E&xcluir
Button "BtnListar"	Name	Btnlistar
	Text	&Listar
Button "BtnFechar"	Name	BtnFechar
	Text	&Fechar
Button "BtnPrimeiro"	Name	BtnPrimeiro
	Text	<
Button "BtnAnterior"	Name	BtnAnterior
	Text	<
Button "BtnProximo"	Name	BtnProximo
	Text	>

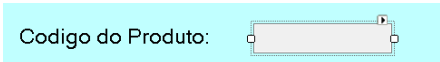
Prof. Roberto de Castro

Button "BtnUltimo"	Name	BtnUltimo
	Text	>

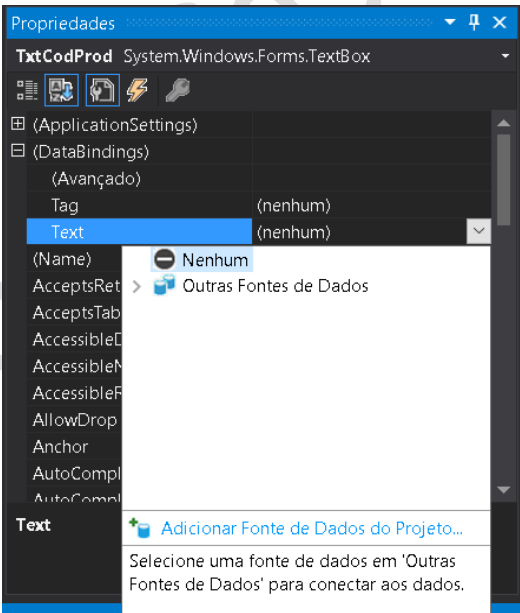
(*) Configuração da propriedade "DataBindings" dos "TextBox". A propriedade "DataBindings" irá vincular o "TextBox" a um campo de uma tabela, por exemplo: TxtCodProd será vinculado ao campo "CodProd" da tabela "Produtos"; TxtDescProd será vinculado ao campo "Descricao" da tabela de produtos e assim sucessivamente.

Vamos configurar a propriedade DataBindings do TxtCodProd:

1. Selecione (dê um clique) no TextBox.
2. Localize a propriedade "Databindings" na caixa de propriedades e clique no sinal de "+" (expanda a opção).
3. Em "Text" clique na seta para baixo. **IMPORTANTE:** Como este TextBox é o primeiro que está sendo configurado, selecione a opção "Outras Fontes de Dados" (Other Data Source). Veja a figura:

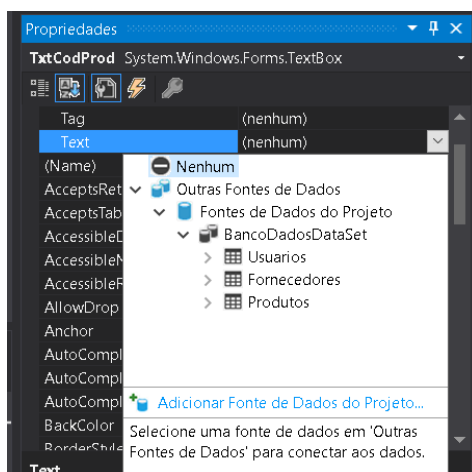


Caixa de texto "TxtCodProd" selecionada

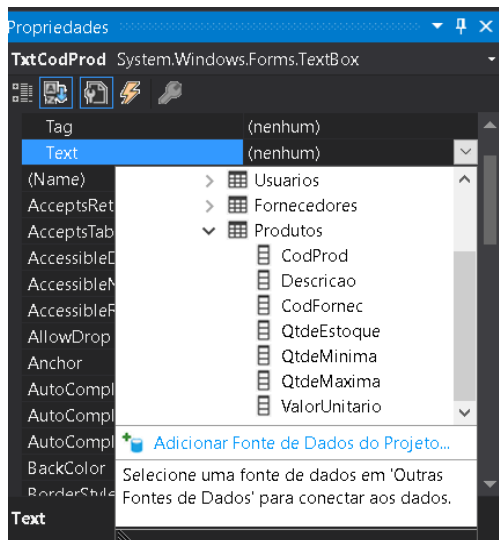


Propriedade "DataBindings" → Text → Outras Fontes de Dados

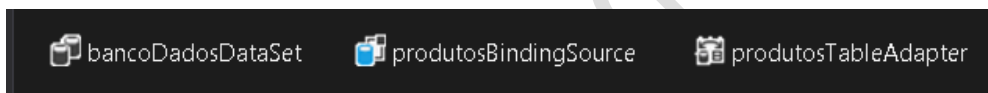
4. Expanda (clique) as opções "Outras fontes de dados" → "Fontes de dados do Projeto" e → "BancoDadosDataSet". Nesta etapa serão exibidas as tabelas Fornecedores e Produtos:



5. Expanda (clique) na tabela de “Produtos”. Serão exibidos os campos desta tabela:



6. Selecione (clique) o campo da tabela que deseja vincular ao TextBox. Neste caso específico, clique em “CodProd”.
7. A partir da criação deste “primeiro” vínculo, o sistema irá incluir no projeto:



8. Continuaremos a vincular os proximos TextBox (a configuração do ComboBox está em outro tópico).
9. Selecione (dê um clique) o TextBox “TxtDescProd”.
10. Localize a propriedade “DataBindings” na caixa de propriedades e clique no sinal de “+” (expanda a opção). Em “Text” clique na seta para baixo.

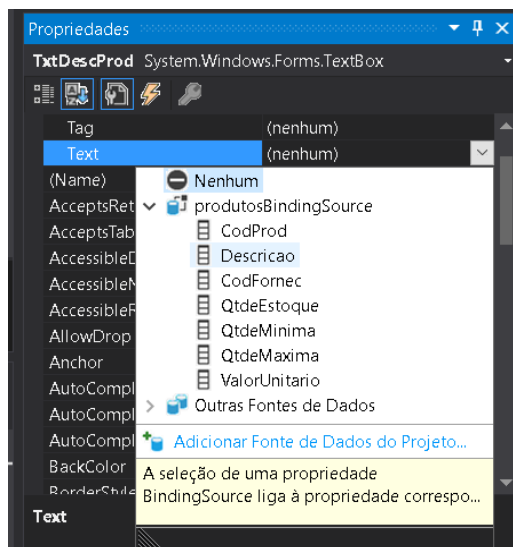
IMPORTANTE: Como já vinculamos previamente outro TextBox, o cenário muda um pouco:

Descrição do Produto:

Caixa de Texto “TxtDescProd” selecionada

Expanda a opção “produtosBindingSource”
NÃO CLIQUE EM “Outras Fontes de Dados”

11. Expanda a opção “produtosBindingSource”. Serão exibidos os campos da tabela de “Produtos”.



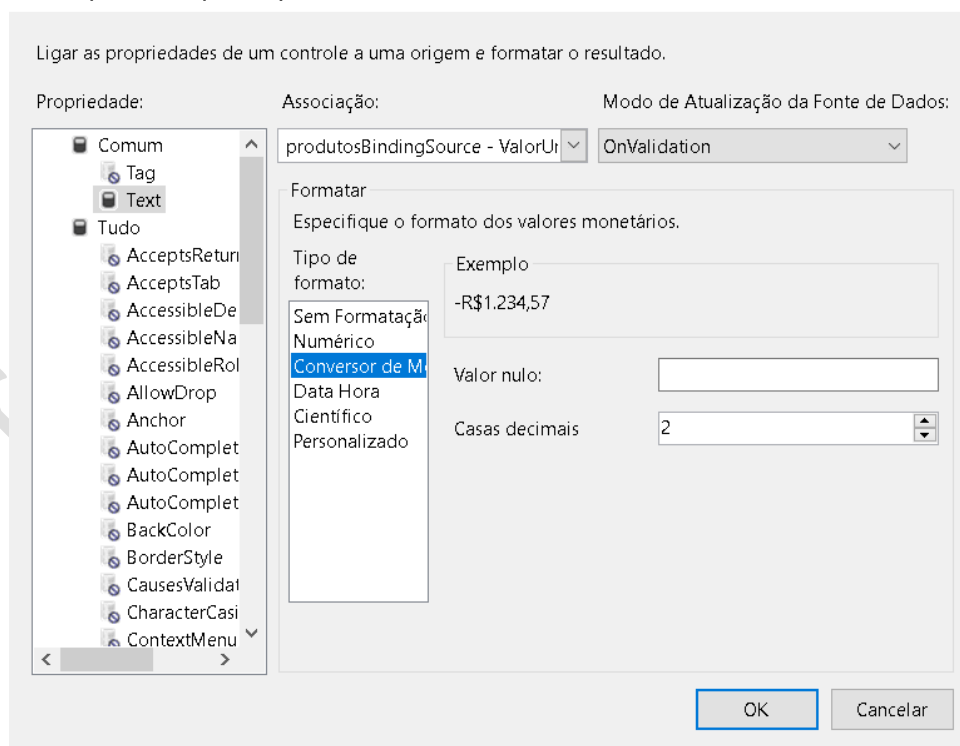
12. Selecione (clique) o campo da tabela que deseja vincular ao TextBox. Neste caso específico, clique em “Descricao”.

13. Continue o processo para os demais “TextBox” (TxtQtdeEstoque, TxtQtdeMinima, TxtQtdeMaxima e TxtValorUnitario). A configuração do ComboBox será em outro momento!!!

14. Para o TextBox “TxtValorUnitario”, configuraremos, também, a propriedade para exibir a informação em “Valor Monetário”. Selecione (dê um clique) o TextBox “TxtValorUnitario”. Expanda a propriedade “DataBindings”. Em “Avançado” clique nos “...” e selecione a opção “Conversor de Moeda”:

Formatação e Associação Avançada

? X



Prof. Roberto de Castro

(**) Vamos agora para a configuração do último vínculo: O “ComboBox”. Selecione o ComboBox “CboRazaoSocial” (dê um clique) e clique na “seta” que está sobre este item:



Selecione (marque) a opção “Usar Itens de Associação de Dados”.

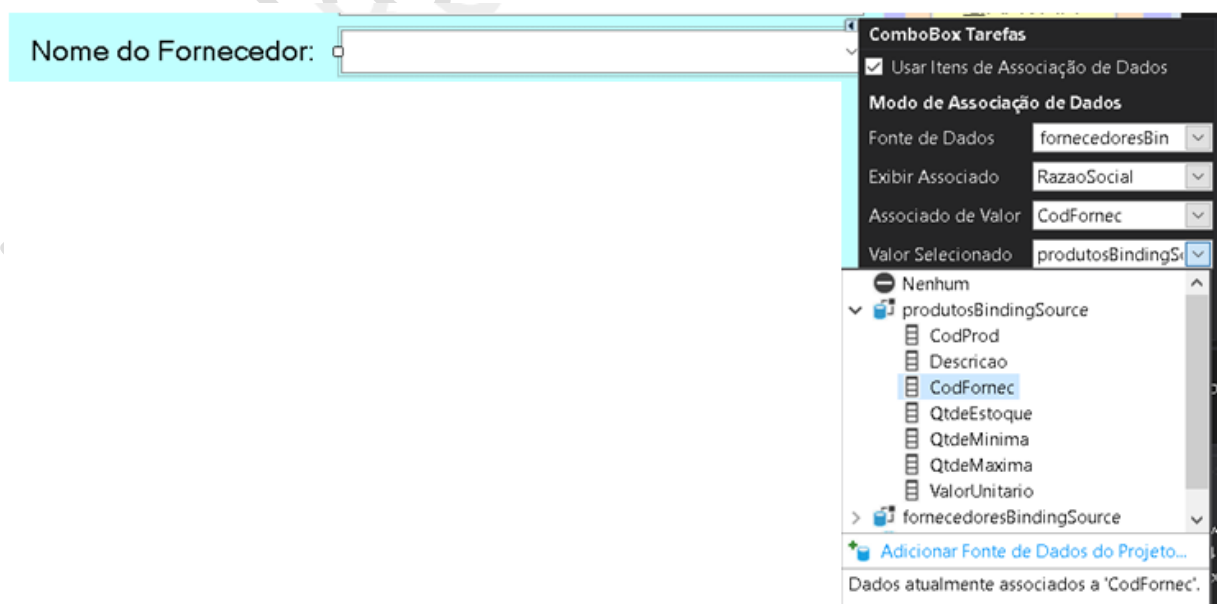


Na opção “Fonte de Dados”, selecione “Outras fontes de dados” (pois desejamos a tabela de Fornecedores). Vá expandindo as opções até localizar e selecionar a tabela de Fornecedores.

Na opção “Exibir Associado” (Display Member), selecione RazaoSocial.

Na opção “Associado de Valor” (Value Member), selecione CodFornec

E, por último, em “Valor Selecionado” (Select Value) devemos assinalar o campo “CodFornec” que está na tabela de produtos:



Prof. Roberto de Castro

Pronto, todos os controles (TextBox e ComboBox) devidamente vinculados às “Fontes de Dados”.

Execute o projeto...

CADASTRO DE PRODUTOS

Dados Cadastrais

Código do Produto: 1

Descrição do Produto: Produto A

Nome do Fornecedor: Fornecedor 03

Quantidade Estoque: 10

Quantidade Mínima: 5

Quantidade Máxima: 100

Valor Unitário R\$: R\$100,00

Operações

NOVO

GRAVAR

EDITAR

EXCLUIR

LISTAR

Navegação

|< < > >|

FECHAR

Perceba, no ComboBox, que o sistema exibe o nome do fornecedor (através dos vínculos que fizemos, o próprio sistema realizou uma busca na tabela de fornecedores e trouxe o fornecedor relacionado).

Vamos agora programar os botões de navegação...

```
private void BtnProximo_Click(object sender, EventArgs e)
{
    this.produtosBindingSource.MoveNext();
    //OBS: O this é opcional, mas agiliza o processo de digitação.
}
```

```
private void BtnAnterior_Click(object sender, EventArgs e)
{
    this.produtosBindingSource.MovePrevious();
}
```

```
private void BtnUltimo_Click(object sender, EventArgs e)
{
    this.produtosBindingSource.MoveLast();
}
```

```
private void BtnPrimeiro_Click(object sender, EventArgs e)
{
    this.produtosBindingSource.MoveFirst();
}
```

Agora, testar novamente o projeto, para verificar o resultado da programação desenvolvida até o momento.

Nas próximas etapas faremos a programação dos botões: Novo, Gravar, Editar, Excluir e Listar.

Basicamente, o botão “Novo” deverá executar os procedimentos abaixo:

1. Travar o GroupBox de navegação.
2. Habilitar o botão “Gravar” (pois está com a opção Enabled = false).
3. Travar todos os botões (incluindo o próprio botão “Novo”). A única opção que ficará disponível para o usuário, será o botão “Gravar”. Isto é necessário pois, após um comando de “AddNew” (veremos abaixo), o Banco de Dados ficará aguardando uma instrução de “gravação”, por esta razão, iremos “travar” todas as opções de manipulação ao banco e disponibilizar somente a opção de “gravação”.
4. Destravar as caixas de texto e o comboBox, permitindo ao usuário a digitação de dados para fazer o cadastramento de novo produto.

Programação botão “Novo”:

```
private void BtnNovo_Click(object sender, EventArgs e)
{
    Destruar();
    TxtCodProd.Focus();

    produtosBindingSource.AddNew();
}
```

//Procedimento para destravar controles

```
private void Destruar()
{
    //Destrua as caixas de texto
    foreach (Control Controle in GrpDados.Controls)
    {
        if (Controle is TextBox)
        {
            Controle.Enabled = true;
        }
    }

    CboRazaoSocial.Enabled = true;

    //Travar botões
    GrpNavegacao.Enabled = false;
    BtnNovo.Enabled = false;
    BtnGravar.Enabled = true;
    BtnEditar.Enabled = false;
    BtnExcluir.Enabled = false;
    BtnListar.Enabled = false;
}
```

O botão “Gravar” deverá executar os procedimentos abaixo:

1. Fazer os testes necessários para validar a entrada de dados (TextBox).
2. Fazer a atualização da Tabela de Produtos na base de dados.
3. Destravar o GroupBox de navegação.
4. Desabilitar o botão “Gravar” (Enabled = False).
5. Destravar todos os botões (incluindo o próprio botão “Novo”). A única opção que ficará indisponível para o usuário, será o botão “Gravar”.
6. Travar as caixas de texto e o comboBox.

Programação botão “Gravar”:

```
private void BtnGravar_Click(object sender, EventArgs e)
{
    //Falta implementar rotina para testar os campos de entrada!!!

    try
    {
        produtosBindingSource.EndEdit();

        //Atualiza fisicamente o BD
        produtosTableAdapter.Update(bancoDadosDataSet.Produtos);
        MessageBox.Show("Gravação efetuada com sucesso!!", "Mensagem");

        produtosBindingSource.MoveFirst();
    }
    catch
    {
        MessageBox.Show("Erro Gravação!!", "ATENÇÃO");
    }

    Travar();
}

//Procedimento para travar controles
private void Travar()
{
    //Travar as caixas de texto
    foreach (Control Controle in GrpDados.Controls)
    {
        if (Controle is TextBox)
        {
            Controle.Enabled = false;
        }
    }

    CboRazaoSocial.Enabled = false;
```

```
//Destravar botões
GrpNavegacao.Enabled = true;
BtnNovo.Enabled = true;
BtnGravar.Enabled = false;
BtnEditar.Enabled = true;
BtnExcluir.Enabled = true;
BtnListar.Enabled = true;
}
```

Agora, faremos a programação do botão “Editar”.

A opção de “Editar” irá permitir ao usuário a alteração nos dados do produto, exceto o “código do produto” (pois é o campo chave e, em regra, campo chave NÃO DEVERÁ SER ALTERADO!!).

A programação do “Editar” seguirá os mesmos procedimentos do botão “Novo” (sem a declaração do AddNew):

1. Travar o GroupBox de navegação.
2. Habilitar o botão “Gravar” (pois está com a opção Enabled = false).
3. Travar todos os botões (incluindo o próprio botão “Editar”). A única opção que ficará disponível para o usuário, será o botão “Gravar”.
4. Destravar as caixas de texto e o comboBox, permitindo ao usuário a digitação de dados para fazer o cadastramento de novo produto. (ATENÇÃO: o textbox do código do produto permanecerá travado para alteração!!!)

Programação do “Editar”:

```
private void BtnEditar_Click(object sender, EventArgs e)
{
    Destravar();
    TxtCodProd.Enabled = false;
}
```

Agora, praticamente finalizando a programação deste formulário de produtos, implementaremos a opção de excluir.

```
private void BtnExcluir_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Confirma exclusão do produto??", "ATENÇÃO", MessageBoxButtons.YesNo,
        MessageBoxIcon.Question) == DialogResult.Yes)
    {
        produtosBindingSource.RemoveCurrent();
        produtosTableAdapter.Update(bancoDadosDataSet.Produtos);
    }
}
```

Execute o projeto novamente para testes de inclusão, alteração, exclusão...

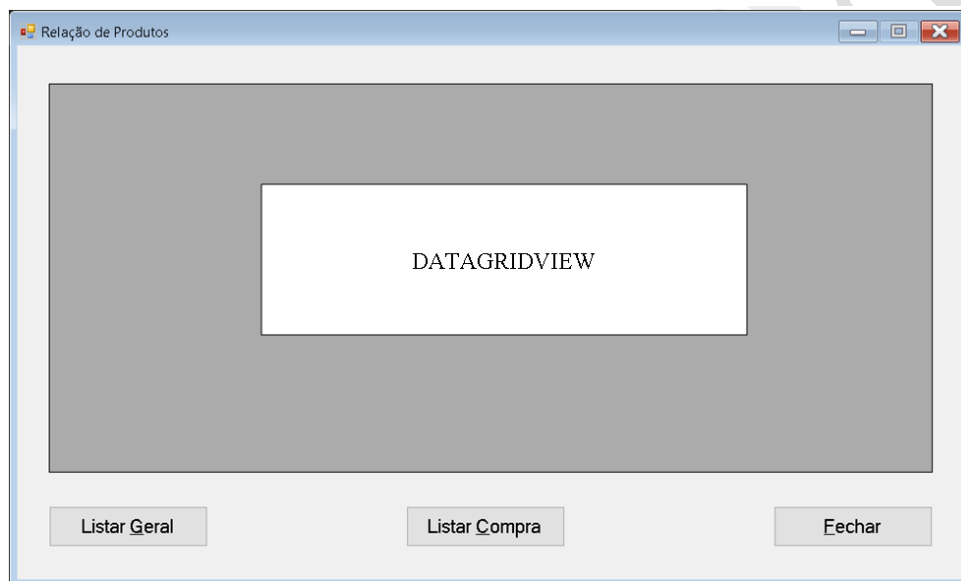
Prof. Roberto de Castro

A opção “Listar” irá exibir os produtos cadastrados, utilizando o controle “DataGridView”. Vamos inserir um novo formulário no projeto. No menu da IDE do Visual Studio clique em Projeto (Project) → Add Windows Forms (Adicionar Formulário). Digite o nome do formulário: FrmListar e clique em Adicionar.

PRINCIPAIS PROPRIEDADES QUE DEVERÃO SER ALTERADAS NO FORMULÁRIO “FrmListar”

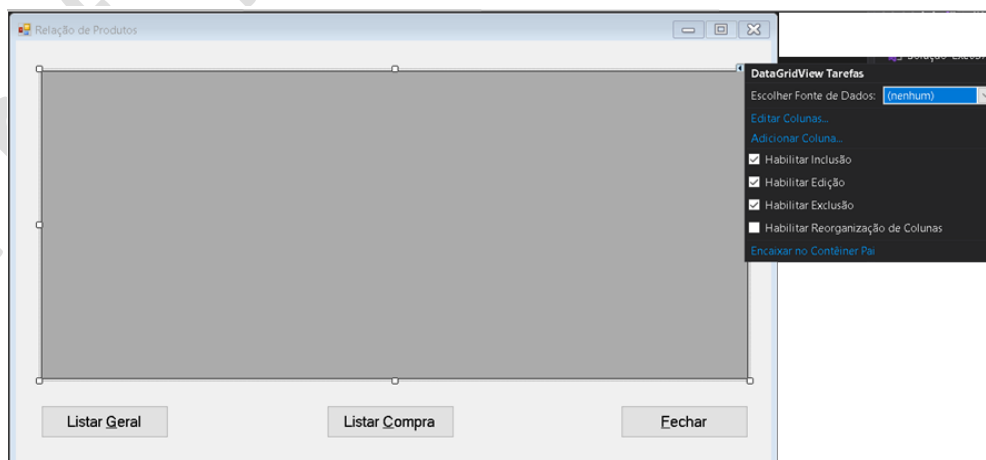
CONTROLE	PROPRIEDADE	CONTEÚDO
Form	Name	FrmListar
	MaximizeBox	False
	StartPosition	CenterScreen
	Text	Relação de Produtos

Vamos inserir o controle “DataGridView” e três “Buttons” (Botões). Veja a interface gráfica:



Neste momento podemos incluir a programação do Botão Fechar, que será: **this.Close();**

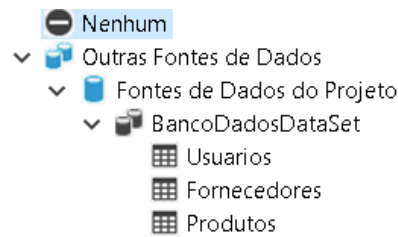
O DataGridView deverá ser vinculado a uma base de dados. Selecione (dê um clique) o DataGridView e clique na “seta” que está no topo deste controle (uma caixa de opções será exibida):



Inicialmente desmarque as caixas de seleção: Habilitar Inclusão, Habilitar Edição e Habilitar Exclusão (estas opções são realizadas no formulário cadastro). O DataGridView será utilizado somente para “consulta”!

Prof. Roberto de Castro

Na sequência, expanda a caixa de opções “Escolher fonte de Dados” e selecione a opção “outras fontes de dados”. Vá expandindo até visualizar as tabelas:



[Adicionar Fonte de Dados do Projeto...](#)

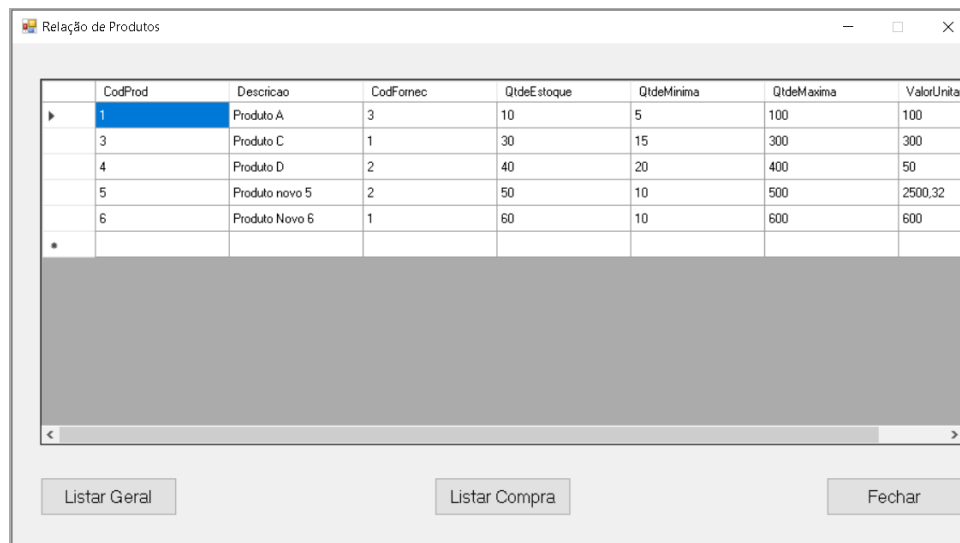
Clique na tabela “Produtos”. O DataGridView será “formatado” e “instâncias” do DataSet serão criadas:

Podemos fazer um teste, porém, antes precisaremos configurar o botão “Listar” que está no formulário de “Cadastro de Produtos” para “chamar” este formulário de listagem. Inclua a programação abaixo:

```
private void BtnListar_Click(object sender, EventArgs e)
{
    FrmListar MeuListar = new FrmListar();
    MeuListar.ShowDialog();
}
```

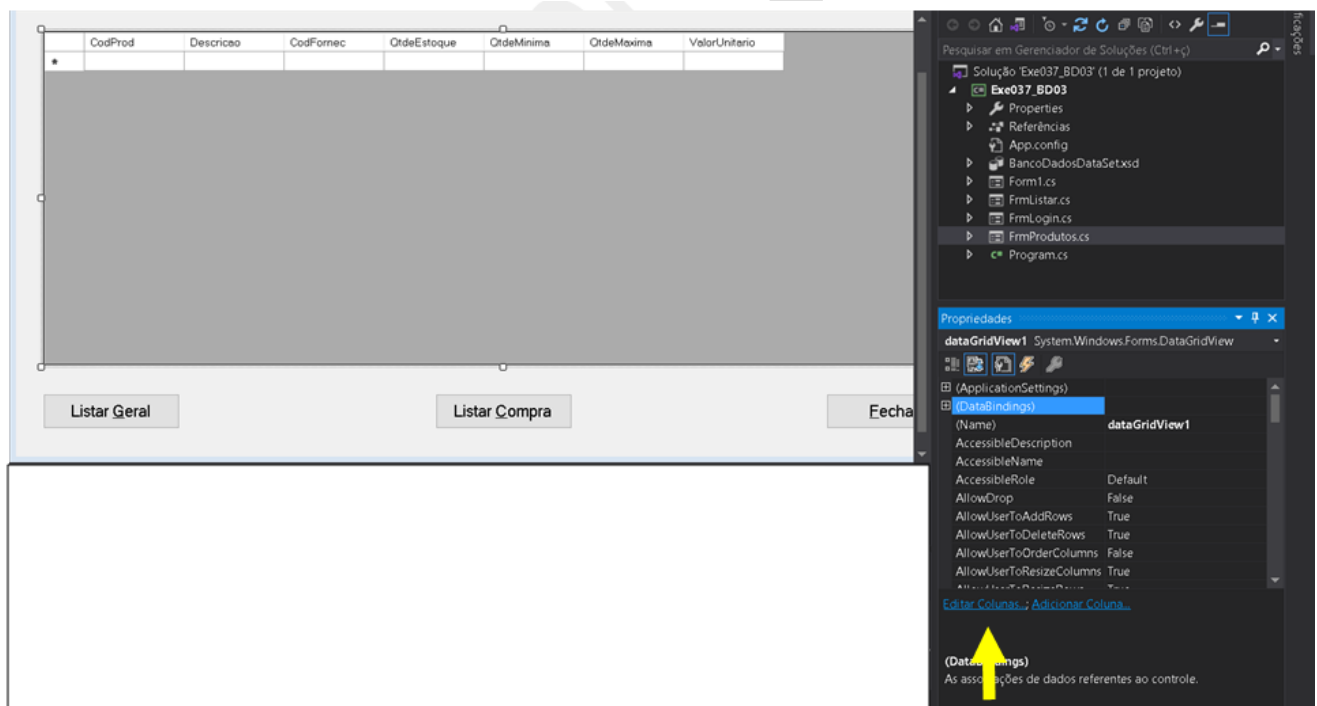
OBS: A opção “ShowDialog” exibe o formulário de listagem e “trava” o formulário de “Cadastro de Produtos”.

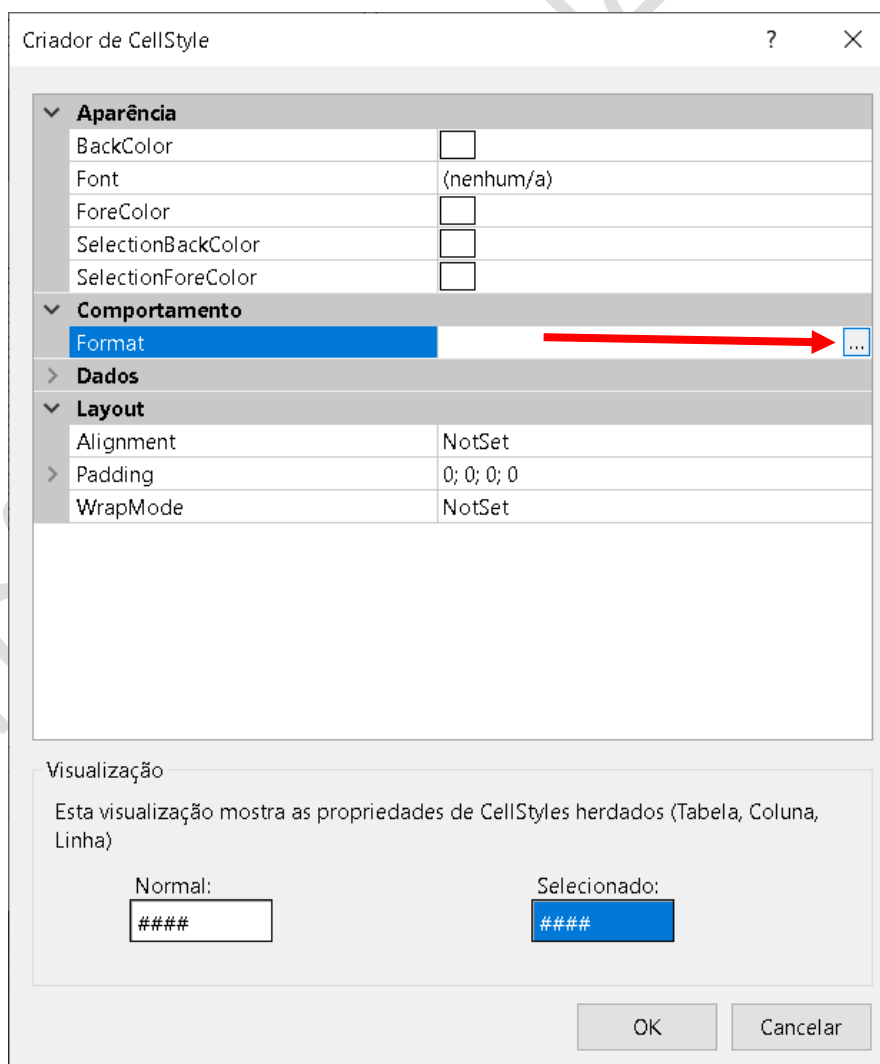
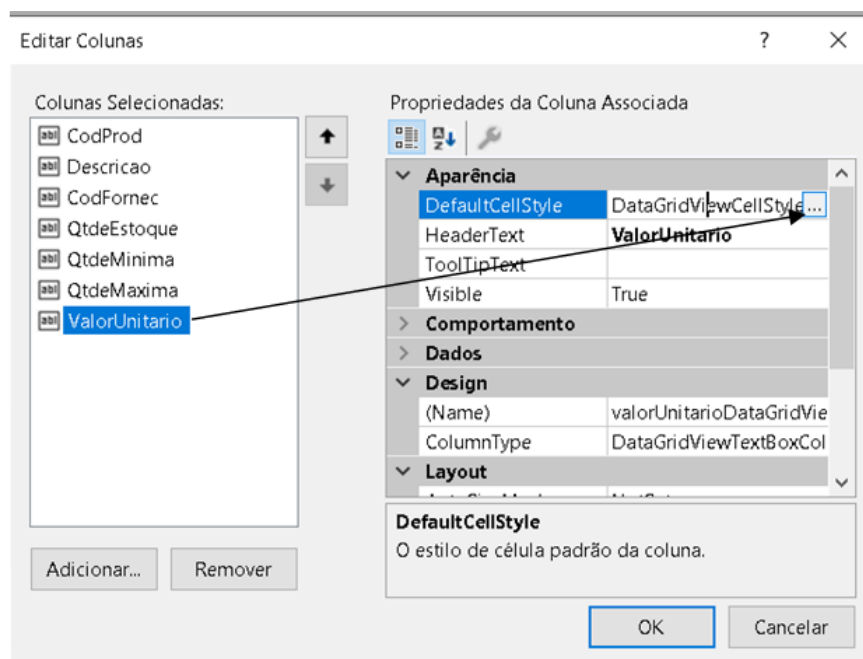
Teste o projeto.... No formulário “Cadastro de Produtos” clique a opção “Listar” e veja o resultado:

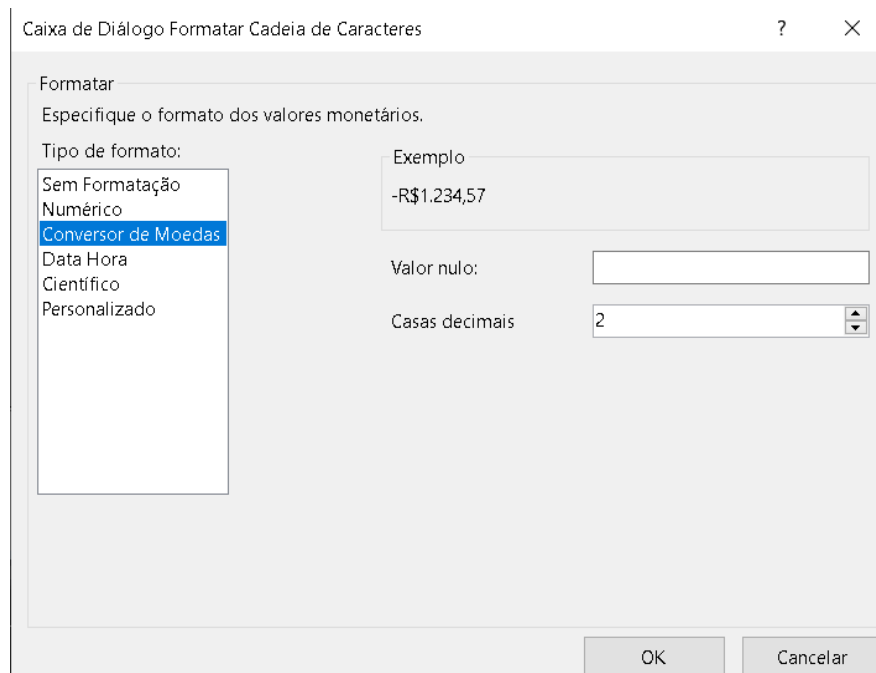


Finalize o teste. Se desejar poderá “ajustar” o tamanho do formulário de listagem. Poderá também fazer algumas configurações nas colunas do DataGridView, por exemplo: exibir o “valor unitário” no formato “moeda”.

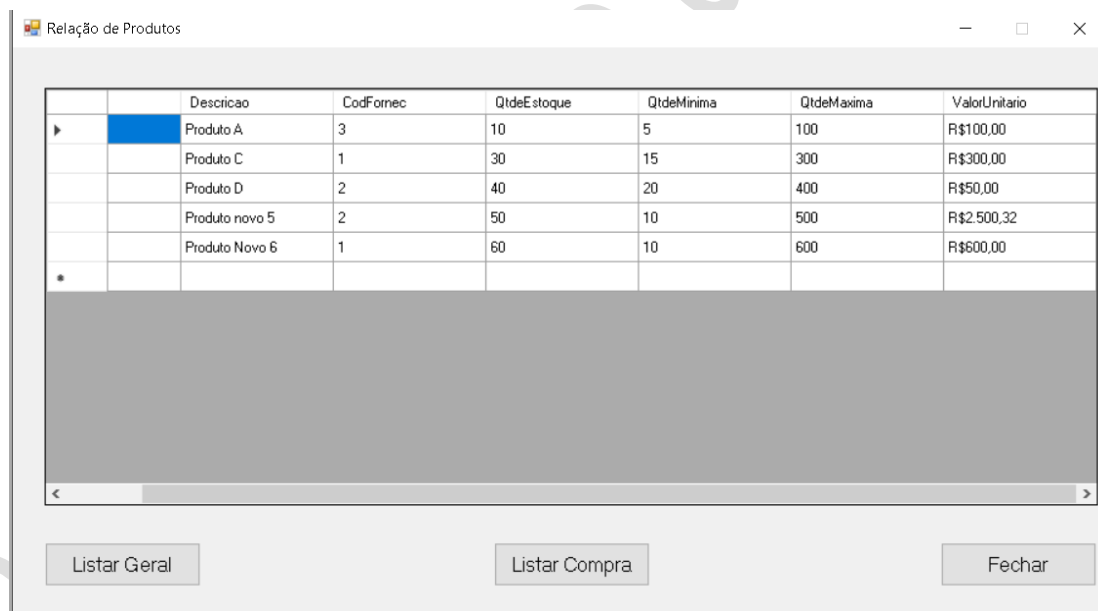
Selecione (dê um clique) o DataGridView. Na “Caixa de Propriedades” clique na opção “Editar Colunas” (veja o destaque em amarelo abaixo):







Selecione “Conversor de Moedas” e clique em “OK” até “fechar” todas as opções. Teste o projeto novamente...



Vamos continuar com a programação...

A programação da opção “Listar Geral” é a mesma que está no Form_Load deste formulário (só copiar):

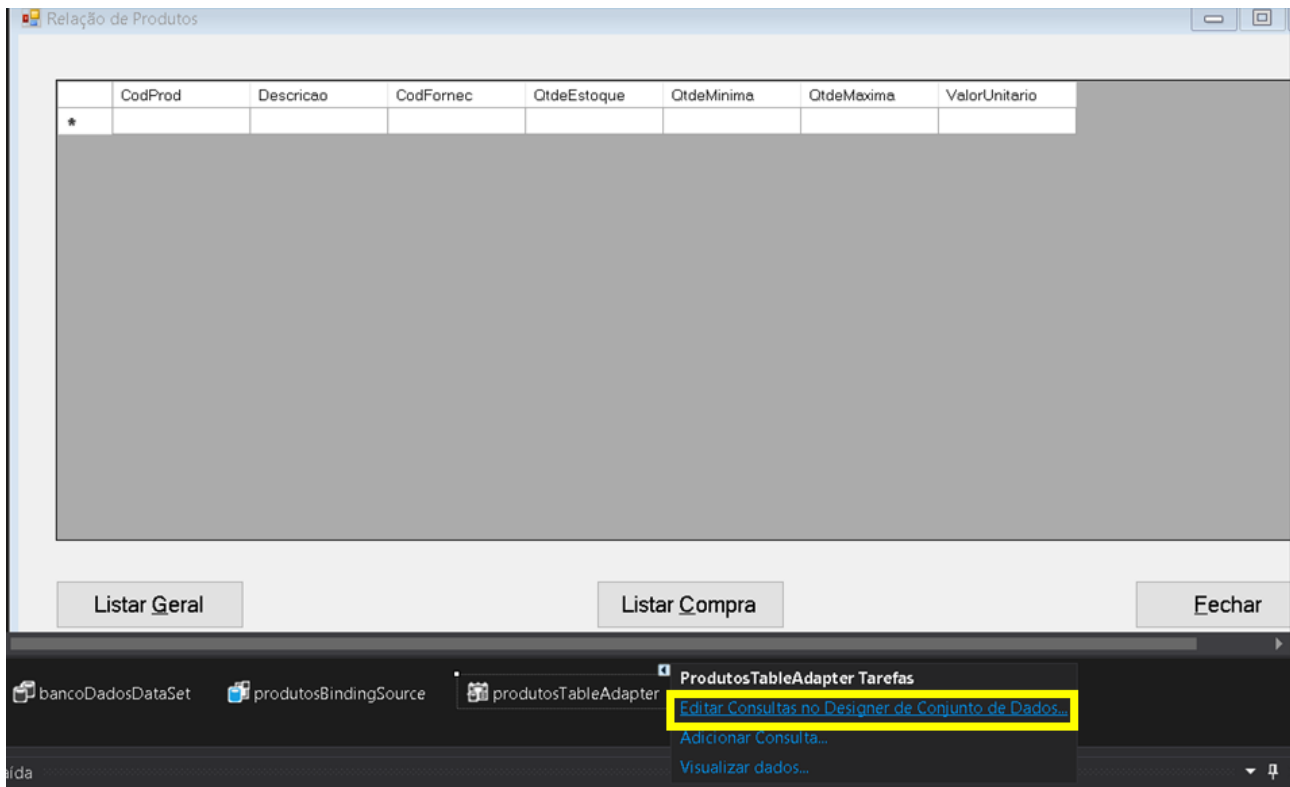
```
private void BtnListaGeral_Click(object sender, EventArgs e)
{
    this.produtosTableAdapter.Fill(this.bancoDadosDataSet.Produtos);
}
```

Prof. Roberto de Castro

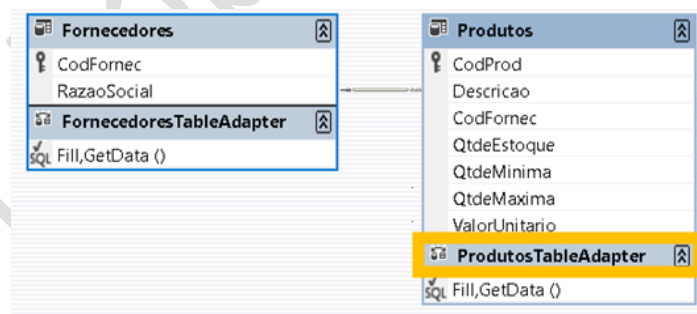
Para a programação da opção “Listar Comprar”, precisaremos incluir uma consulta “nova” no Adapter, pois os produtos que serão exibidos deverão atender a condição de QtdeEstoque < QtdeMinima.

Vamos criar a instrução “SQL” para gerar o resultado desejado.

Selecione (dê um clique) o “produtosTableAdapter” que está no formulário “FrmListar” e clique na opção “Editar Consultas no Designer de Conjunto de Dados...”:

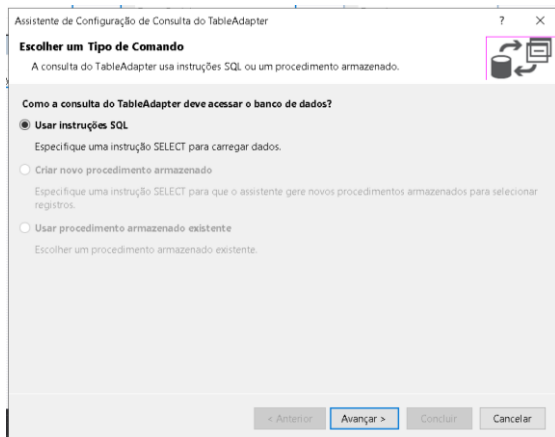


Será exibido o “esquema”:

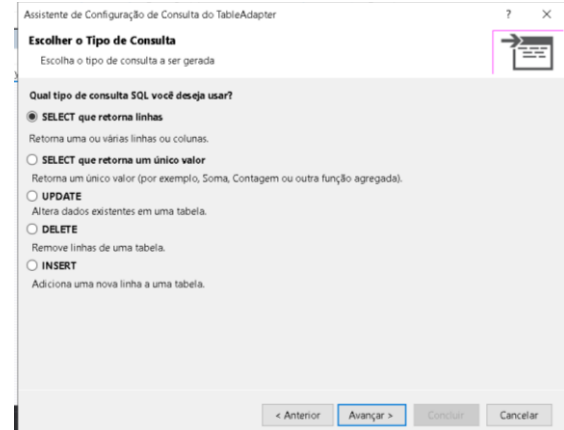


Em “ProdutosTableAdapter”, pressione com o “botão direito” do mouse e selecione a opção: Adicionar → Consulta.

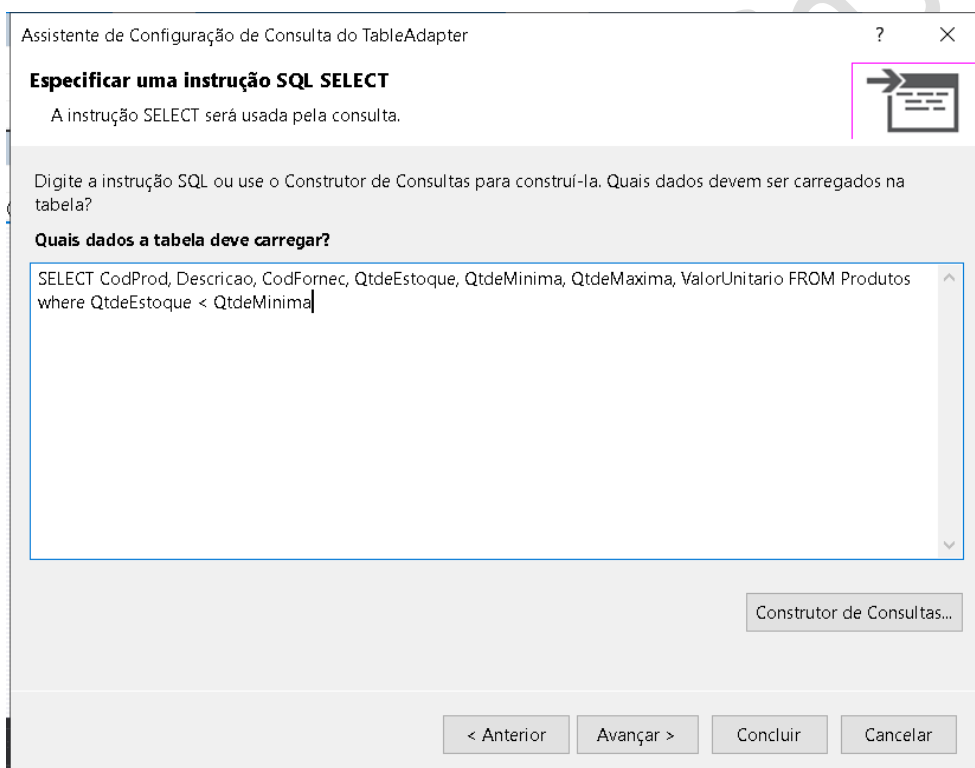
Prof. Roberto de Castro



Clique em “Avançar”



Mantenha selecionada a opção “SELECT que retorna linhas” e clique em Avançar

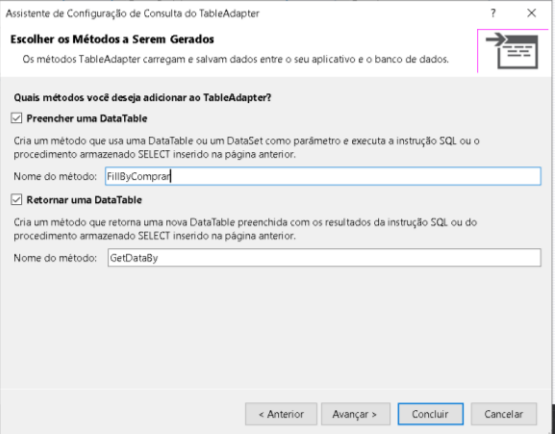


Digite o comando SQL

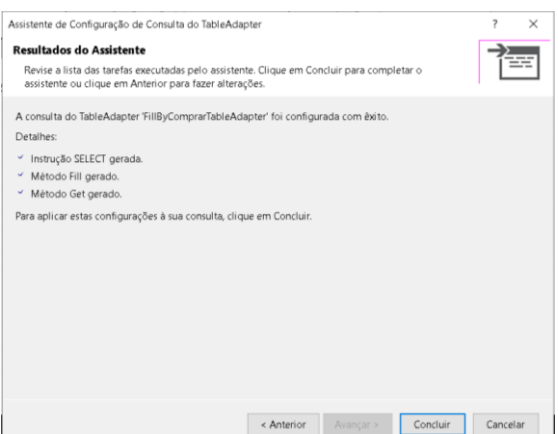
SELECT CodProd, Descricao, CodFornec, QtdeEstoque, QtdeMinima, QtdeMaxima, ValorUnitario FROM
Produtos where QtdeEstoque < QtdeMinima

Muita atenção ao digitar os nomes dos campos e da tabela!!!

Prof. Roberto de Castro



Digite o nome desta “consulta” será FillByComprar
e Clique em Avançar



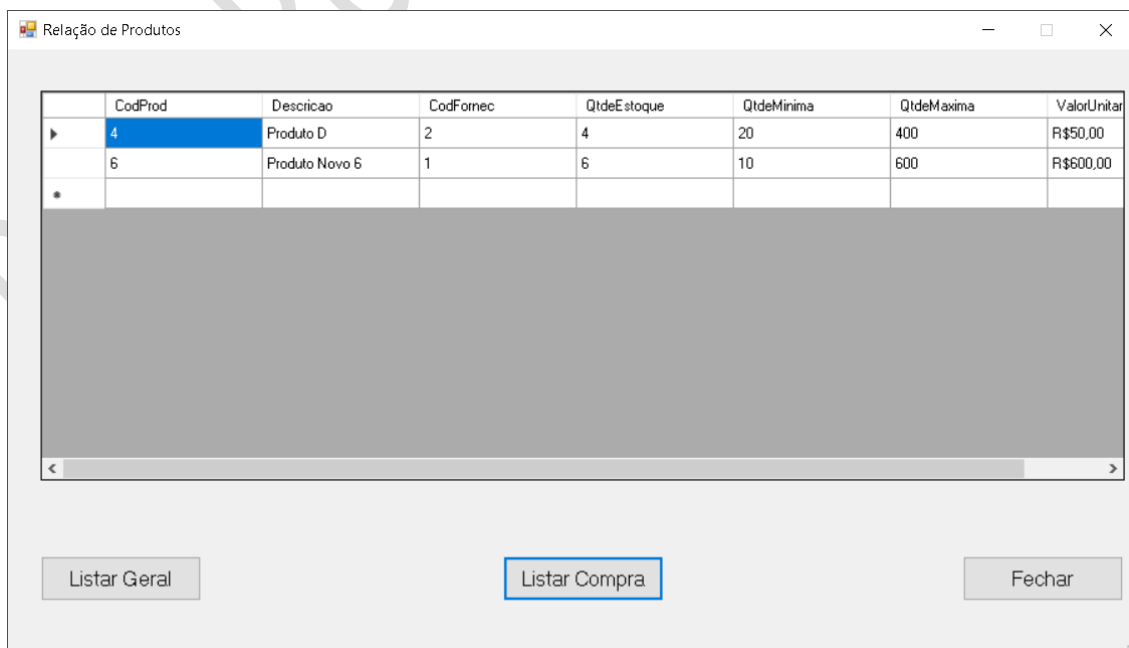
Clique em “Concluir”

E agora, para finalizar a programação do formulário “FrmListar”, inclua o código abaixo no botão “Listar Compra”:

```
private void BtnListaCompra_Click(object sender, EventArgs e)
{
    this.produtosTableAdapter.FillByComprar(this.bancoDadosDataSet.Produtos);

    //Verifica se existem produtos a serem exibidos (estoque < minimo)
    if (produtosBindingSource.Count == 0)
    {
        MessageBox.Show("Não existem produtos abaixo do estoque!!!", "ATENÇÃO");
    }
}
```

Caso não existam produtos com Estoque < Mínimo, você poderá alterar alguns dados no cadastro (Estoque < Mínimo) para poder visualizar a listagem:



	CodProd	Descrição	CodFornec	QtdeEstoque	QtdeMinima	QtdeMaxima	ValorUnitar
▶	4	Produto D	2	4	20	400	R\$50,00
	6	Produto Novo 6	1	6	10	600	R\$600,00
*							

< >

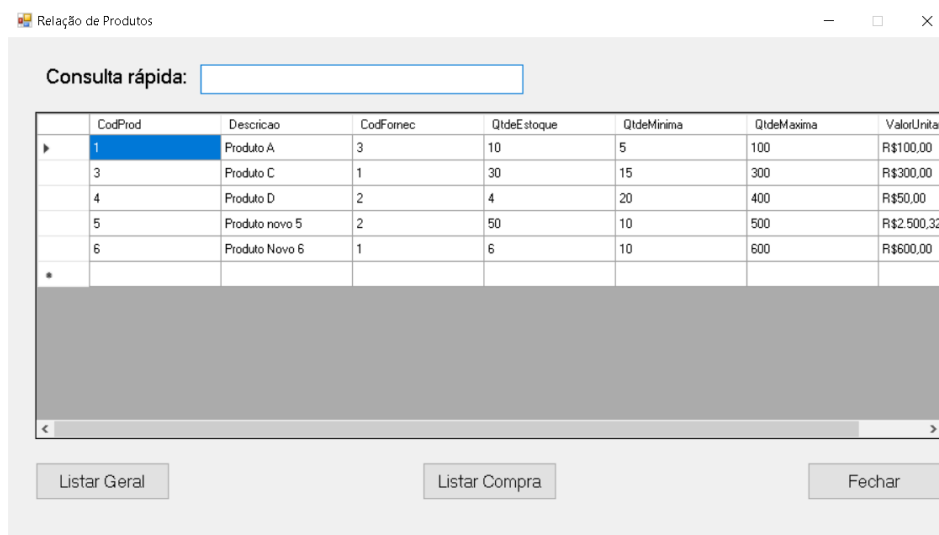
Listar Geral Listar Compra Fechar

Prof. Roberto de Castro

Antes de finalizarmos este “módulo de listagem”, incluiremos uma opção de “consulta rápida” com o objetivo de ilustrar o recurso de “consulta parcial” (consulta realizada com a Descrição parcial do produto).

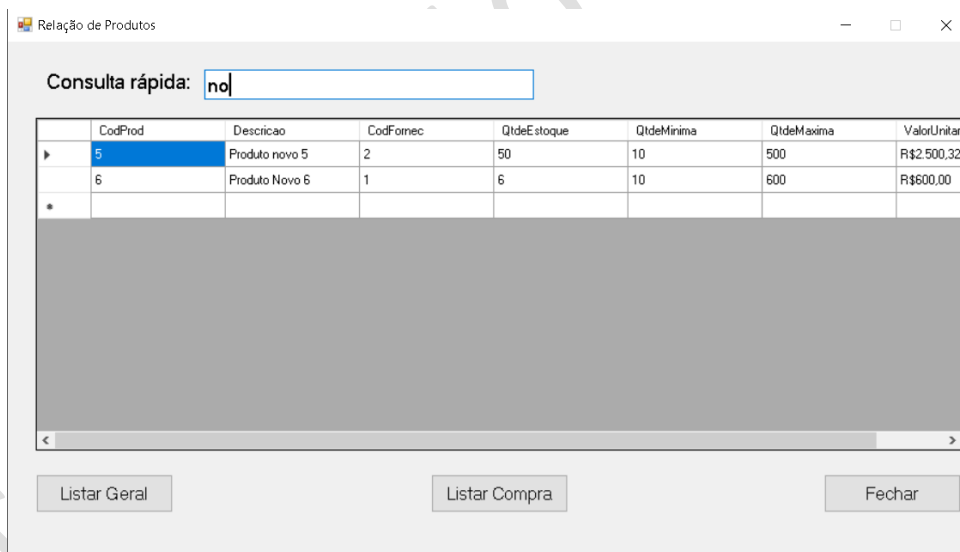
Observe as imagens abaixo:

- 1) Situação inicial (logo após a carga do formulário) - Lista completa dos produtos



CodProd	Descrição	CodFornec	QtdeEstoque	QtdeMinima	QtdeMaxima	ValorUnitar
1	Produto A	3	10	5	100	R\$100,00
3	Produto C	1	30	15	300	R\$300,00
4	Produto D	2	4	20	400	R\$50,00
5	Produto novo 5	2	50	10	500	R\$2.500,32
6	Produto Novo 6	1	6	10	600	R\$600,00
*						

- 2) A listagem dos produtos irá se “atualizando” na medida em que o usuário for digitando o nome de um produto (descrição). A programação será incluída no evento “Change” do TextBox



CodProd	Descrição	CodFornec	QtdeEstoque	QtdeMinima	QtdeMaxima	ValorUnitar
5	Produto novo 5	2	50	10	500	R\$2.500,32
6	Produto Novo 6	1	6	10	600	R\$600,00
*						

Principais etapas para a inclusão desta consulta rápida:

1. Incluir uma nova consulta no “ProdutosTableAdapter”, a consulta será gravada com o nome de FillByConsultaRapida e terá a seguinte instrução SQL:

```
SELECT CodProd, Descricao, CodFornec, QtdeEstoque, QtdeMinima, QtdeMaxima, ValorUnitario FROM  
Produtos where Descricao like ?
```

IMPORTANTE: O símbolo de “?” indica uma variável e que o valor será fornecido pelo projeto.

2. Incluir a programação abaixo, no evento “TextChanged” do TextBox “TxtDescricao”. OBS: O evento TextChaged ocorrerá sempre em que o TextBox for alterado:

```
private void TxtDescricao_TextChanged(object sender, EventArgs e)
{
    this.produtosTableAdapter.FillByConsultaRapida(bancoDadosDataSet.Produtos, "%" +
        TxtDescricao.Text + "%");
}
```

Execute o projeto. Faça o teste e observe o resultado. A lista de produtos será atualizada a cada tecla que você digitar no TxtDescricao. O resultado da pesquisa funcionará como um “filtro dinâmico”, **mas isto poderá ter um custo na “performance” de execução de todo o projeto!!!** Este recurso de atualizar uma lista dinamicamente poderá degradar e muito o desempenho do aplicativo, pois o evento será executado a todo o momento. Trata-se de um recurso interessante para conhecimento, mas na prática, deverá ser considerada a viabilidade ou não desta opção. Um recurso intermediário seria incluir a programação em um botão “Pesquisar”, que seria executado após a digitação da “descrição do produto”. Pense nisto!!

E chegamos ao final deste projeto...

Proposta de Atividade/Exercício: A partir do projeto desenvolvido, implemente as solicitações abaixo:

1. O botão “Gravar/Alterar” do formulário “Cadastro” deverá validar as entradas de dados (TextBox) para que estejam em conformidade com os requisitos informados no Banco de Dados: Nome requerido (obrigatório); QtdeEstoque > 0; QtdeMinima < QtdeMaxima; ValorUnitário > 0....
2. Implementar formulário para efetuar as operações de Inclusão / Alteração / Exclusão e listagem dos Fornecedores.