

Premium

Pacote de Fontes

Manual do Desenvolvedor

Conteúdo	3
Fontes.....	3
LinhaBase Developing Tools for .Net	3
Requisitos do Sistema	4
Restrições	4
O Código de Barras.....	5
A simbologia QR Code	5
Compactação de Dados	5
Parâmetros	5
Correção de Erros (Error Correction)	5
Máscara	5
Instalando o Pacote de Fontes de Código de Barras.....	6
Instalação	6
Licenças	7
Autenticação	7
O Arquivo Assembly .Net.....	8
Nome do arquivo	8
NameSpace	8
Função	8
As fontes de Código de Barras	9
Dados Técnicos das Fontes de Código de Barras	10
Granulação da Imagem do Código de Barras	10
Criando o Código de Barras	11
Criando Código de Barras em .Net	11
Exemplo em Visual Basic.Net.....	11
Criando Código de Barras com Crystal Reports para Visual Studio .Net.....	13
Criando Código de Barras com Crystal Reports para Visual Studio .Net com Dataset executando um query com valor variável em modo de execução.....	23
Criando Código de Barras com Report Viewer para Visual Studio .Net	28
Criando Código de Barras com Report Viewer para Visual Studio .Net com Dataset executando um query com valor variável em modo de execução	39
Ajuste do Tamanho do Campo do Código de Barras	44
Apêndices.....	46
Erros	46
Informações.....	46
Suporte	46

Conteúdo

Fontes

- **LBQR1b.ttf** – Fonte TTF QR Code Tamanho 1.
- **LBQR2b.ttf** – Fonte TTF QR Code Tamanho 2.
- **LBQR3b.ttf** – Fonte TTF QR Code Tamanho 3.
- **LBQR4b.ttf** – Fonte TTF QR Code Tamanho 4.

LinhaBase Developing Tools for .Net

- **linhabaseqrcode.dll** – Assembly .Net responsável pela codificação do dado em QR Code.
- **Manual.pdf** – Manual do produto. *
- **Licenca.pdf** – Licença de uso do produto formato Acrobat. *

* Necessário ter instalado no computador o Adobe Acrobat Reader.

Requisitos do Sistema

- **Sistema operacional:** Windows XP, Windows Vista, Windows 7 com Microsoft Internet Explorer 5.5 ou superior, Windows 8, Windows 8.1, Windows 10.
- Microsoft .Net Framework 2.0 ou superior.
- **Computador:** Pentium ou superior.
- **Impressora:** Para impressão do código de barras.
- **Espaço em disco:** 10MB disponíveis.
- **Memória:** 128 MB de memória RAM.
- Internet para realização da autenticação de uso.

Restrições

- Fontes de Código de Barras não são indicadas para uso Web ou similares, onde em alguns casos as fontes TTF também devem ser instaladas nos computadores clientes para seu funcionamento.
- As dimensões do código de barras tendem a reduzir ao transformar a impressão em PDF, o que faz com que o código de barras original deva ser maior para uma boa impressão.

O Código de Barras

A simbologia QR Code

O QR Code (Quick Response) é um código de barras bi-dimensional (2D) composto por padrões negros quadrados em um fundo branco. O código de barras é capaz de armazenar mais informação do que um código de barras convencional. Este código de barras foi desenvolvido no Japão e é um dos mais populares código de barras bi-dimensionais. Outra razão para esta popularidade que o QR Code é adotado por muitos aplicativos móveis ou smartphones para a ligação de objetos do mundo físico para uma URL da web.

Compactação de Dados

O QR Code é capaz de codificar uma grande quantidade de dados usando métodos de compactação diferentes. Cada um dos métodos de compactação é utilizado para um tipo específico de dados. Por exemplo, o método numérico é otimizado para números. O codificador da LinhaBase verifica automaticamente o dado a ser codificado e detecta o método de compactação mais apropriado.

O LinhaBase QR Mapping é capaz de utilizar as seguinte compactação de dados:

- Numeric: Compactação otimizada para números.
- Alphanumeric: Compactação otimizada para números e letras.
- Binary: Compactação otimizada para dados binários de 8 bits.
- Kanji: Compactação otimizada para dados Kanji.

Parâmetros

Correção de Erros (Error Correction)

O QR Code usa a técnica de correção de erros Reed-Solomon. Isto permite que o código de barras possa estar parcialmente danificado, sem causar qualquer perda de dados. Há quatro diferentes níveis de correção de erro que podem ser escolhidos. Quanto maior o nível, o mais o código de barras será capaz de resistir a danos. No entanto, a desvantagem é que a técnica de correção de dados insere caracteres de correção e faz com que o código de barras tenha sua capacidade de dados reduzida.

São suportados os seguintes níveis de correção:

- Nível L: 7% do dado podem ser recuperado.
- Nível M: 15% do dado podem ser recuperado.
- Nível Q: 25% do dado podem ser recuperado.
- Nível H: 30% do dado podem ser recuperado.

Máscara

A confiabilidade do código QR pode ser melhorada por um método chamado Máscara. A Máscara regulariza a distribuição dos padrões do código de barras (quadrados pretos). Diferentes tipos de padrões de máscaras são suportados e podem ser:

- | | |
|--|---|
| - Máscara 0.
- Máscara 1.
- Máscara 2.
- Máscara 3.
- Máscara 4. | - Máscara 5.
- Máscara 6.
- Máscara 7.
- Máscara Automática. |
|--|---|

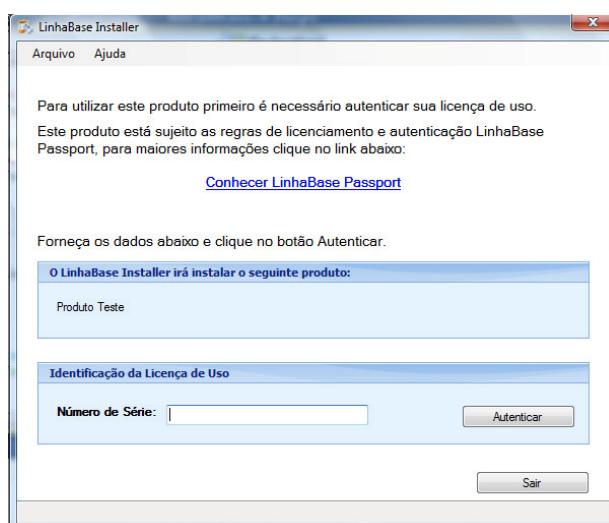
Instalando o Pacote de Fontes de Código de Barras

Instalação

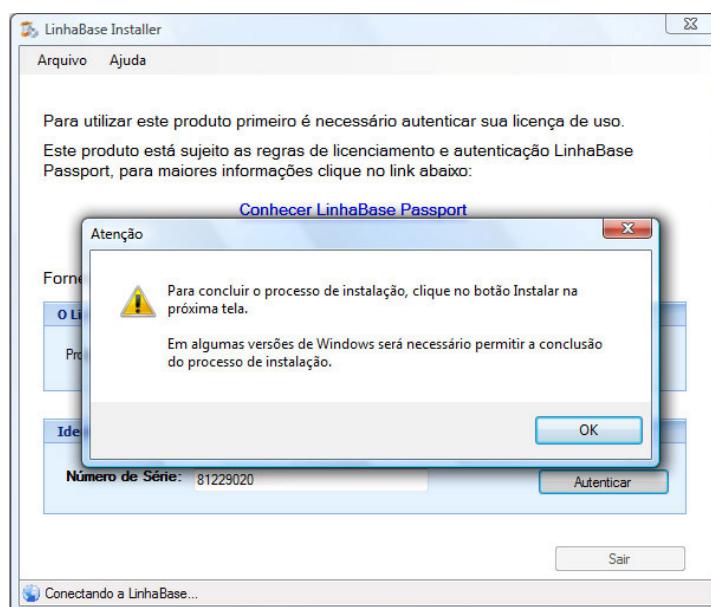
O Pacote de Fontes de Código de Barras LinhaBase precisa ser autenticado para ter a sua instalação concluída.

Para instalar o Pacote de Fontes faça o seguinte:

- 1 – Execute a instalação do LinhaBase Installer.
- 2 – Execute o LinhaBase Installer de seu produto (cada produto LinhaBase tem seu Installer próprio, certifique que está executando o Installer correto).
- 3 – No LinhaBase Installer verifique o nome do produto e forneça o Número de Série e clique no botão Autenticar.



- 4 – Tendo autenticado o produto clique no botão Ok para iniciar a etapa final da instalação do produto.



5 – Siga as instruções para concluir o processo de instalação.

6 – Após ter terminado o processo de instalação clique no botão Sair no LinhaBase Installer.

Licenças

Licença corresponde ao número de computadores que irão utilizar um produto LinhaBase. Este produto é fornecido com as seguintes formas de licenciamento:

- 1 Licença de Uso;
- 5 Licenças de Uso;
- Licenciamento Ilimitado.

O licenciamento de 1 computador (1 licença) e de 5 computadores (5 licenças) é chamado de licenciamento de uso interno e este tipo de licenciamento utiliza a tecnologia de autenticação de licenças LinhaBase Passport. O licenciamento ilimitado permite utilizar o produto LinhaBase em qualquer quantidade de computadores.

Autenticação

Os licenciamentos do tipo 1 licença e 5 licenças de uso utilizam o processo de autenticação de licenças, que visa garantir sua autenticidade e se baseia na construção de chaves digitais.

A autenticação é extremamente fácil de realizar e deve ser feita toda vez que ocorrer a instalação do produto.

O Arquivo Assembly .Net

Nome do arquivo

linhabaseqrcode.dll

NameSpace

LinhaBaseQrMapping.Map

Função

A função de codificação QR é dada abaixo:

```
Function QrEncode (DataToEncode As String,  
                  ErrCorrection As String,  
                  MaskPattern As Integer) As String
```

QrEncode

Tipo: String

Descrição: Função de mapeamento para codificação do código de barras. Esta função retornará uma saída em branco (vazia) caso o valor a ser codificado (DataToEncode) contiver uma sequência de caracteres inválidos ou superior as especificações do código de barras.

DataToEncode

Tipo: String

Descrição: Valor a ser convertido em Código de Barras. É importante que este valor não contenha caracteres em acento.

ErrCorrection

Tipo: String

Descrição: Parâmetro de correção de erros. Este valor pode ser:

"L", "M", "Q" ou "H".

O valor deve ser fornecido entre aspas.

MaskPattern

Tipo: Integer

Descrição: Parâmetro de para ajuste da máscara. Este valor pode ser:

0, 1, 2, 3, 4, 5, 6, 7, 8(para máscara automática).

Obs: Os arquivos Linhabaseqrcode.dll, CipherLite.dll e QRCode.dll devem estar posicionados no diretório da aplicação (pasta onde se encontra o arquivo .exe do software desenvolvido).

As fontes de Código de Barras

O Pacote de Fontes de Código de Barras é fornecido com 4 modelos de fontes que produzem códigos de barras com tamanhos diferentes e que podem ser escolhidos de forma que melhor atender as necessidades de produção do código de barras.

Exemplos:



Fonte LBQrCode1b Tamanho 11



Fonte LBQrCode2b Tamanho 11



Fonte LBQrCode3b Tamanho 11



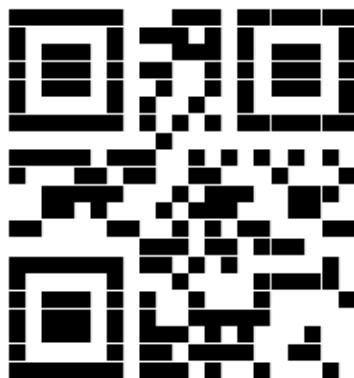
Fonte LBQrCode4b Tamanho 11

Dados Técnicos das Fontes de Código de Barras

Nome do Arquivo	Nome Interno
LBQR1b.ttf	LBQRCode1b
LBQR2b.ttf	LBQRCode2b
LBQR3b.ttf	LBQRCode3b
LBQR4b.ttf	LBQRCode4b

Granulação da Imagem do Código de Barras

A imagem do código de barras pode em alguns casos aparecer granulada ou com falhas entre os padrões do código de barras, como apresentada na imagem abaixo:



Este efeito geralmente ocorre na apresentação do código de barras no monitor do computador e pode ser resolvido aumentando ou diminuindo o tamanho do código de barras.

Na impressão, o código de barras que no monitor apresenta granulação é geralmente impresso corretamente quando utilizado impressoras de boa qualidade como impressoras laser. Impressoras de baixa resolução não são indicadas para produção de código de barras 2D.

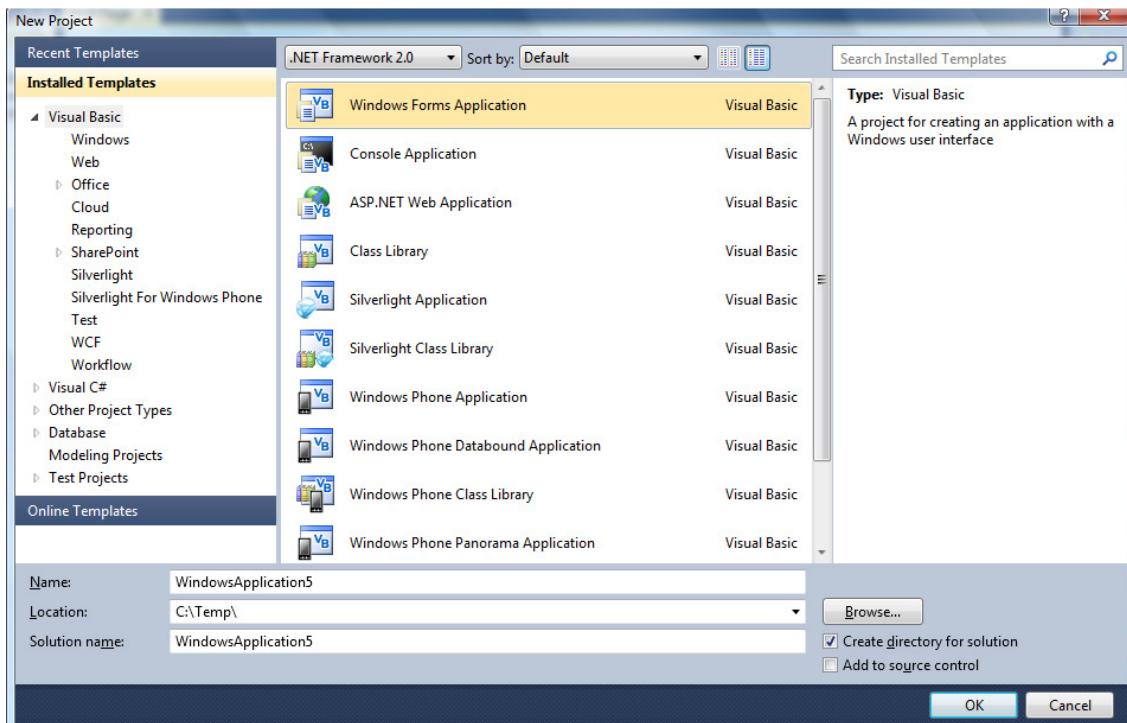
Criando o Código de Barras

Criando Código de Barras em .Net

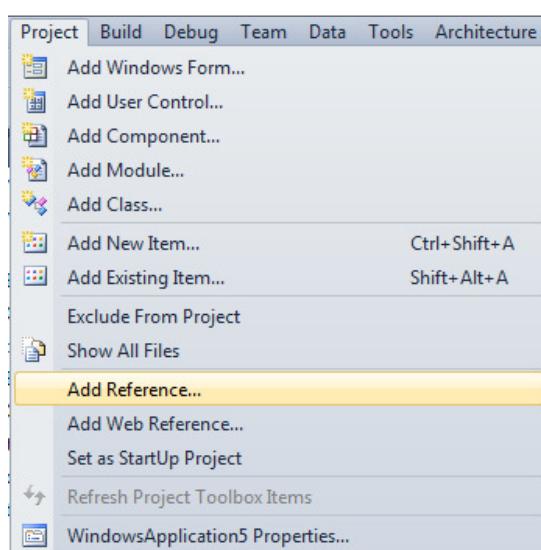
Exemplo em Visual Basic.Net

Abaixo fornecemos um exemplo de impressão de código de barras no Visual Basic.Net.

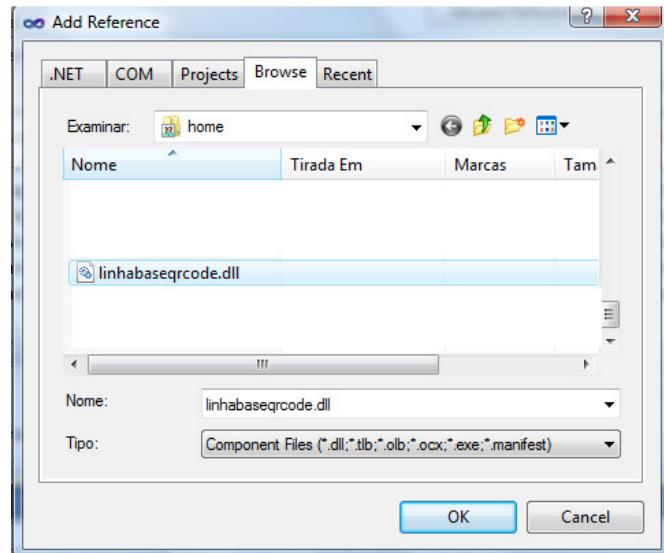
- Abra o Visual Studio.Net e crie uma nova aplicação Windows.



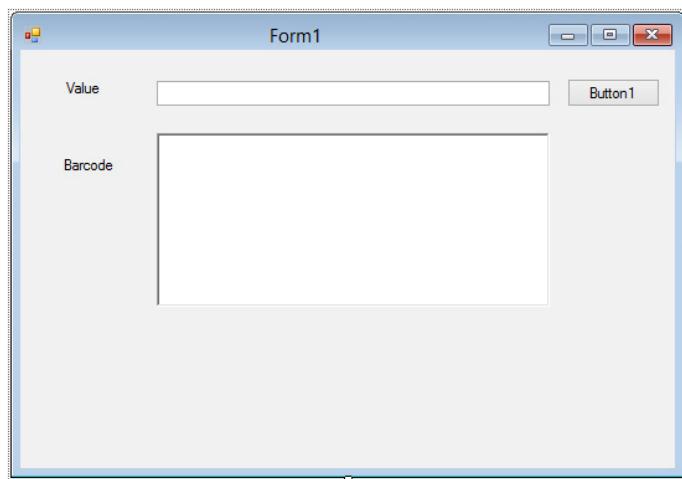
- Clique em Project – Add Reference para adicionar referência ao arquivo LinhabaseqrCode.dll.



- Localize o arquivo Linhabaseqrcode.dll.



- Crie um Form como na figura abaixo.



OBS: A caixa de texto do código de barras é do tipo RichTextBox.

- Insira o código:

```
Imports LinhaBaseQrMapping.Map

Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        Dim Qr As New LinhaBaseQrMapping.Map

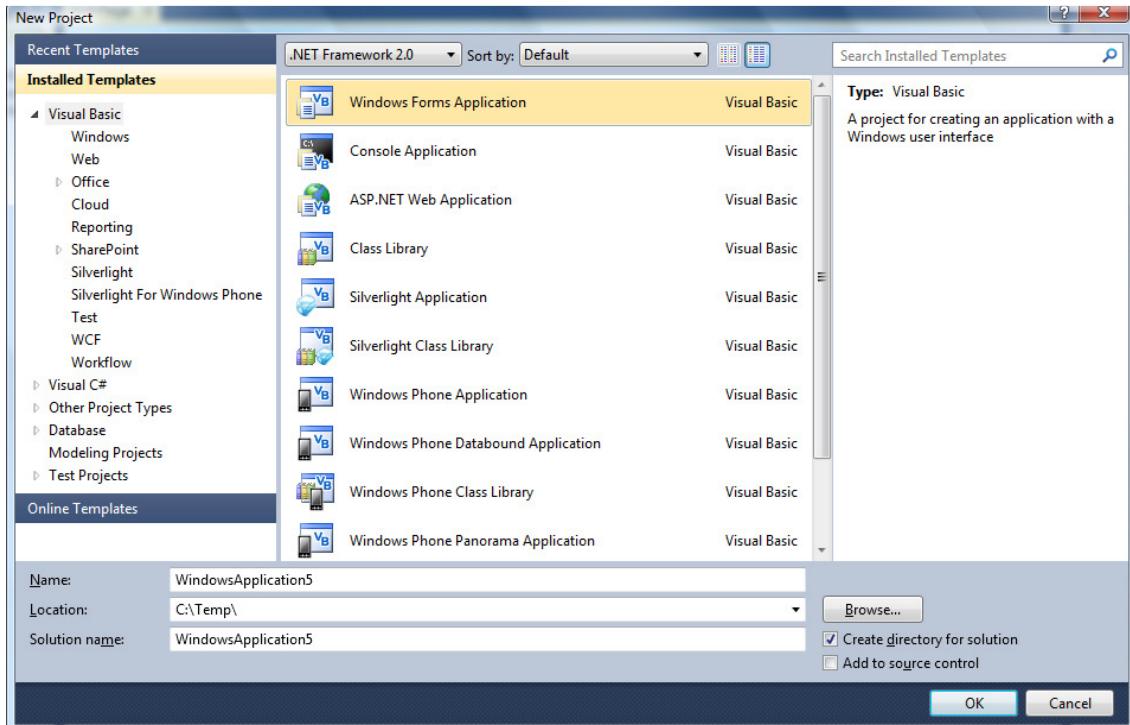
        Dim fnt As New Font("LBQR1b", 4)
        RichTextBox1.Font = fnt

        RichTextBox1.Text = Qr.QrEncode(TextBox1.Text, "L", 0)

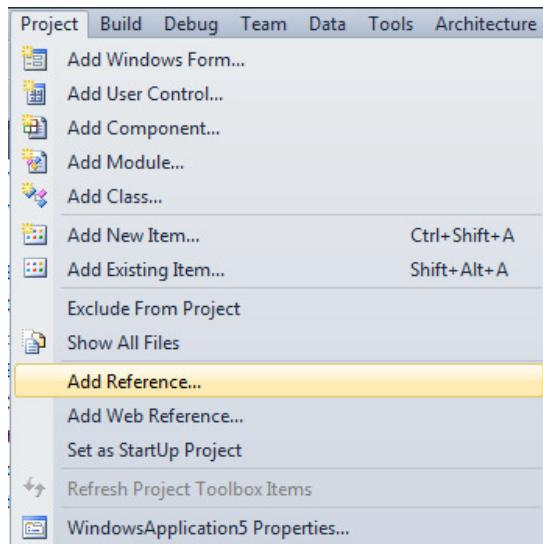
    End Sub
End Class
```

Criando Código de Barras com Crystal Reports para Visual Studio .Net

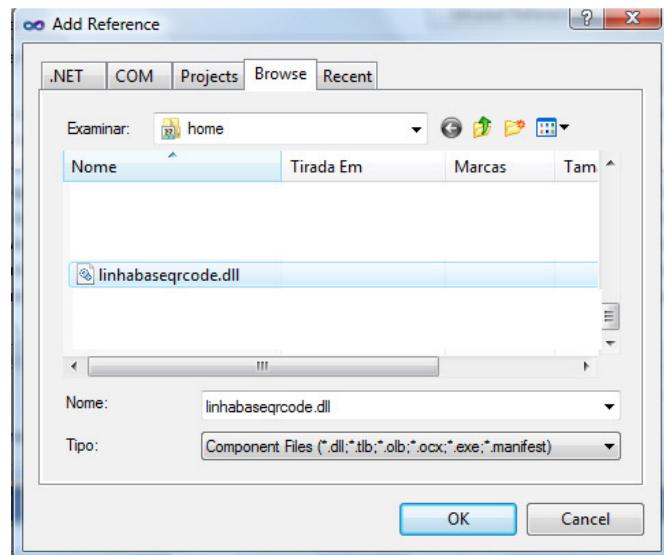
– Abra o Visual Studio.Net e crie uma nova aplicação Windows.



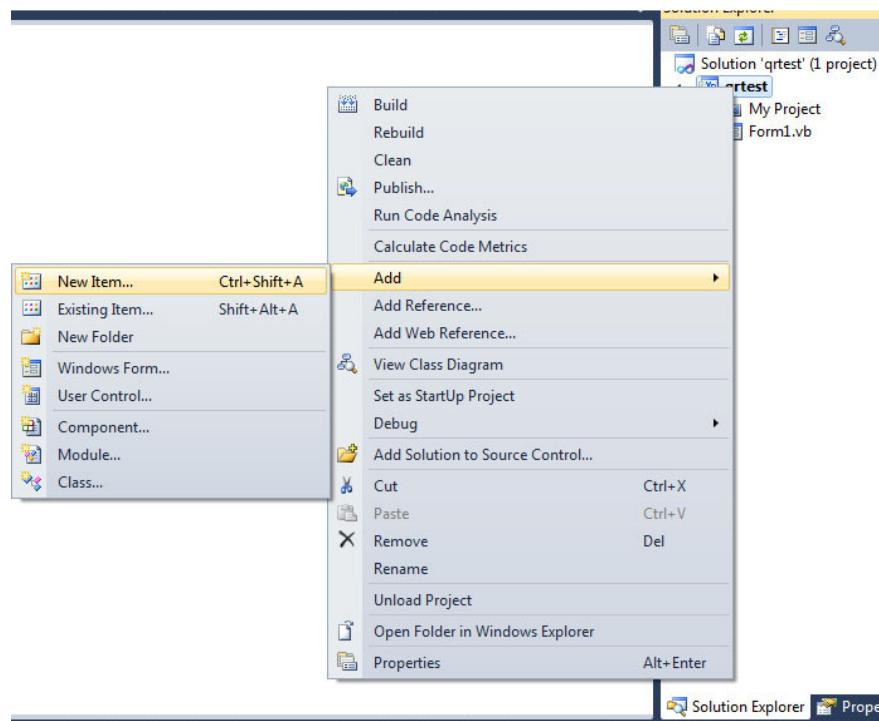
– Clique em Project – Add Reference para adicionar referência ao arquivo LinhabaseqrCode.dll.



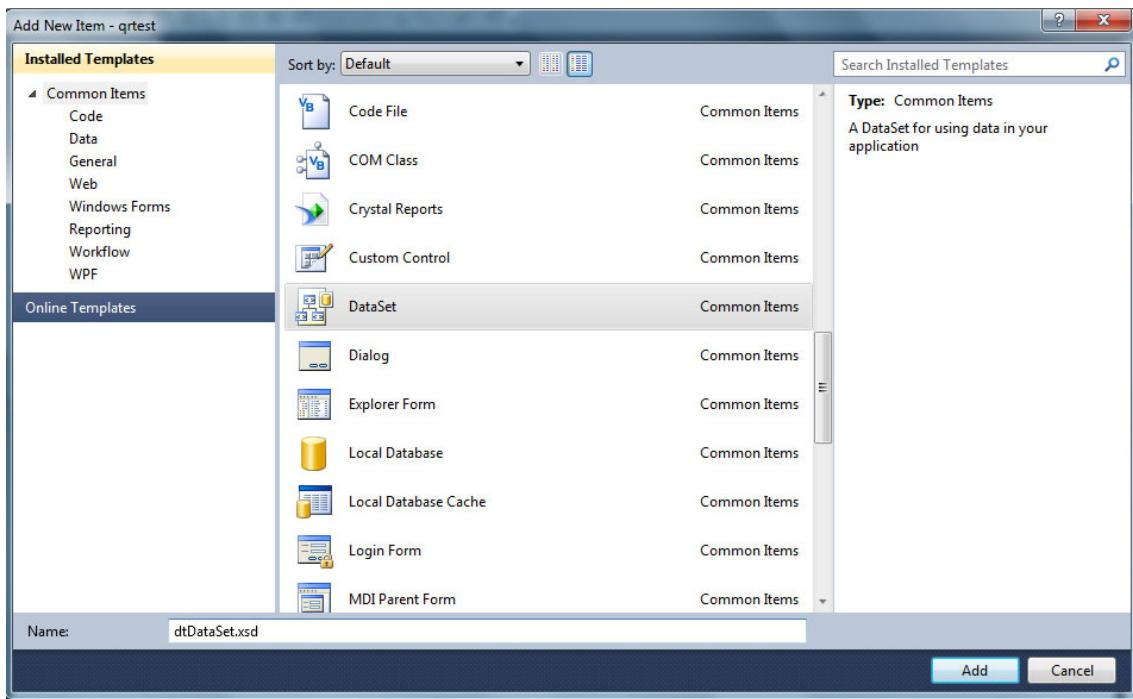
– Localize o arquivo LinhabaseqrCode.dll.



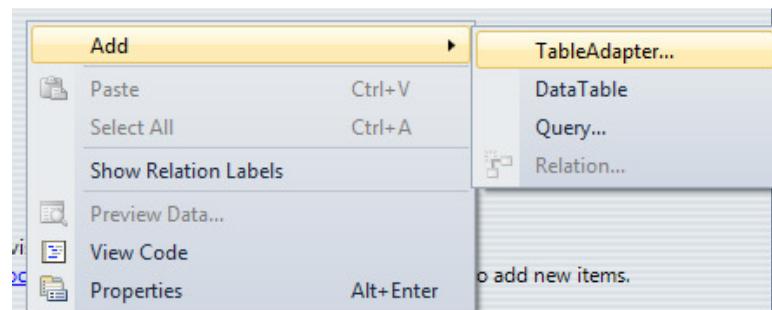
– Em Solution Explores clique com o botão direito selecione Add – New Item.



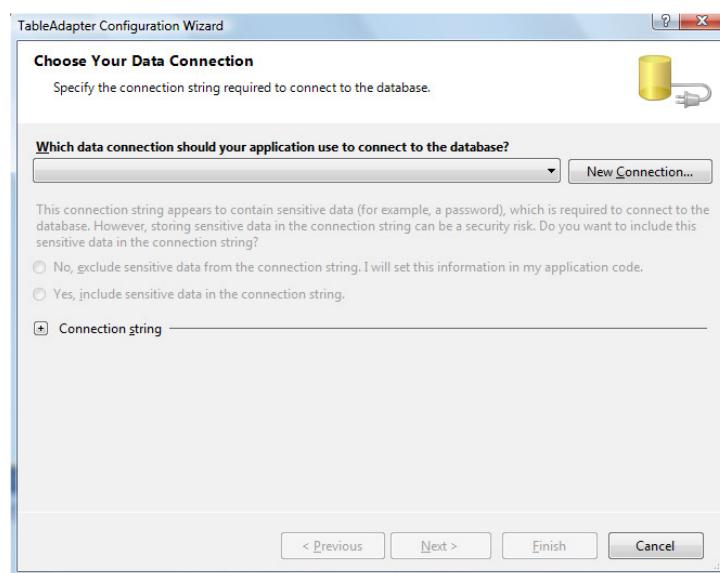
– Adicione um Dataset ao projeto.



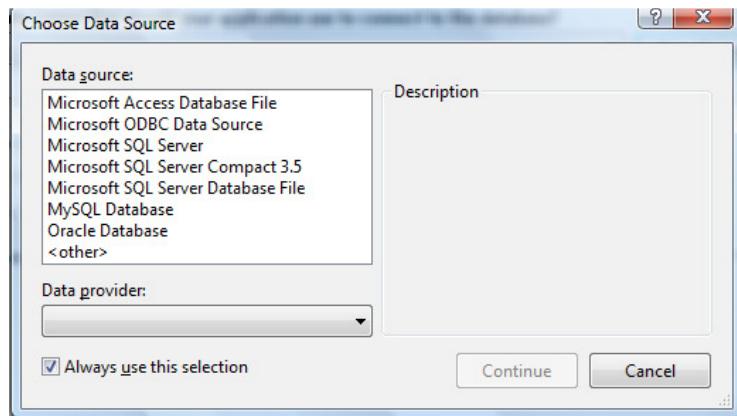
– Clique com o botão direito na área do Dataset e escolha Add – Table Adapter.



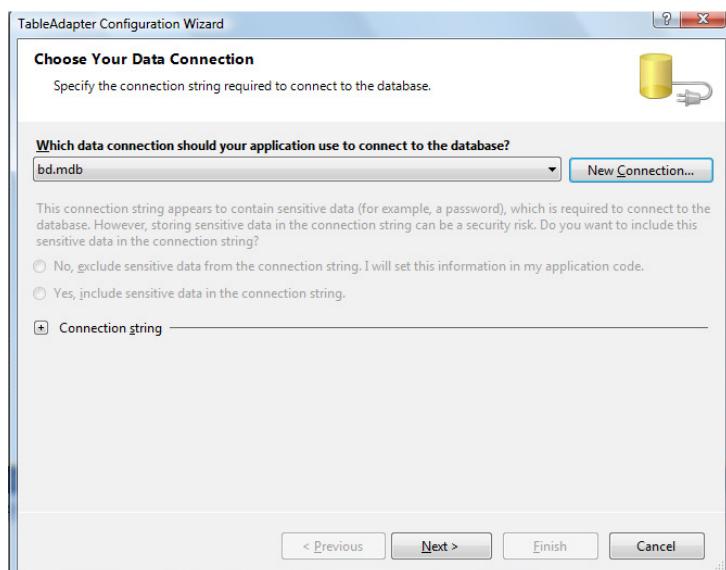
– Escolha a conexão com sua base de dados.



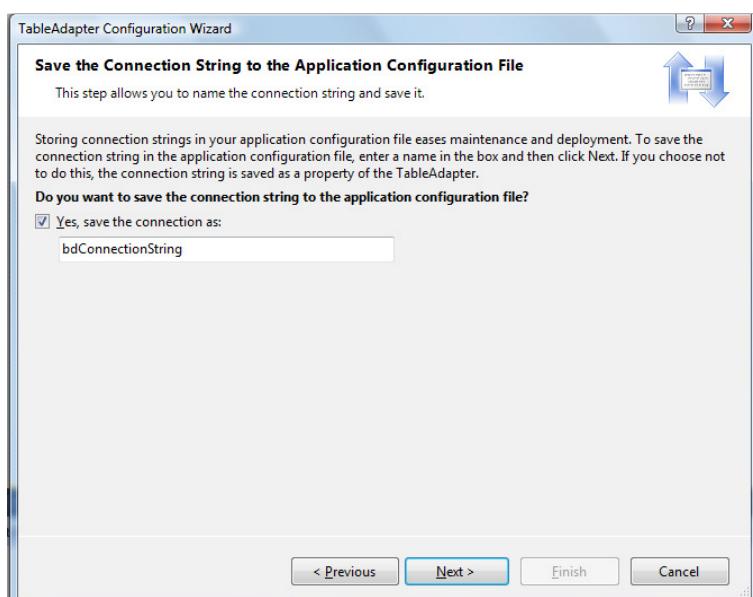
– Clique no botão New Connection e selecione o Data Source.



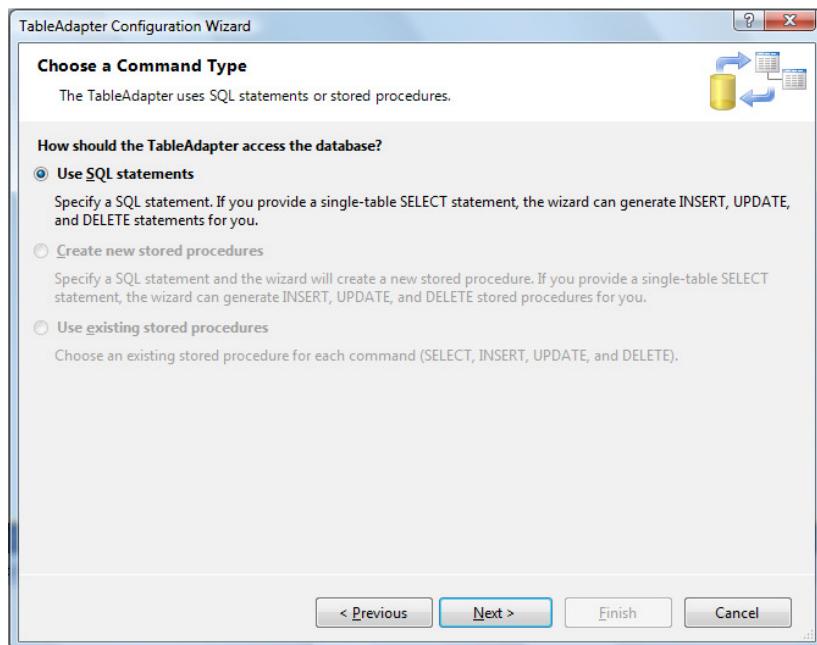
– Após ter configurado a conexão com a base de dados clique no botão Next.



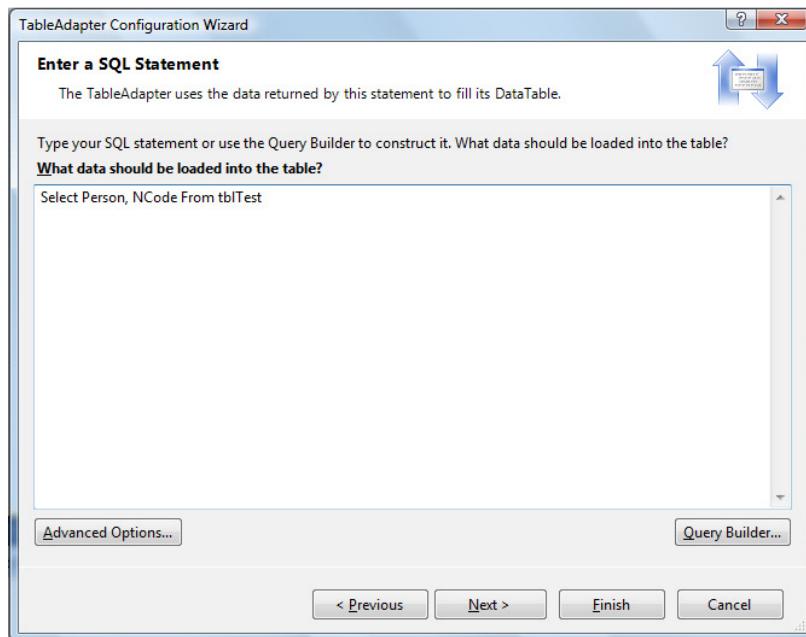
– Salve a conexão com a base de dados e clique no botão Next.



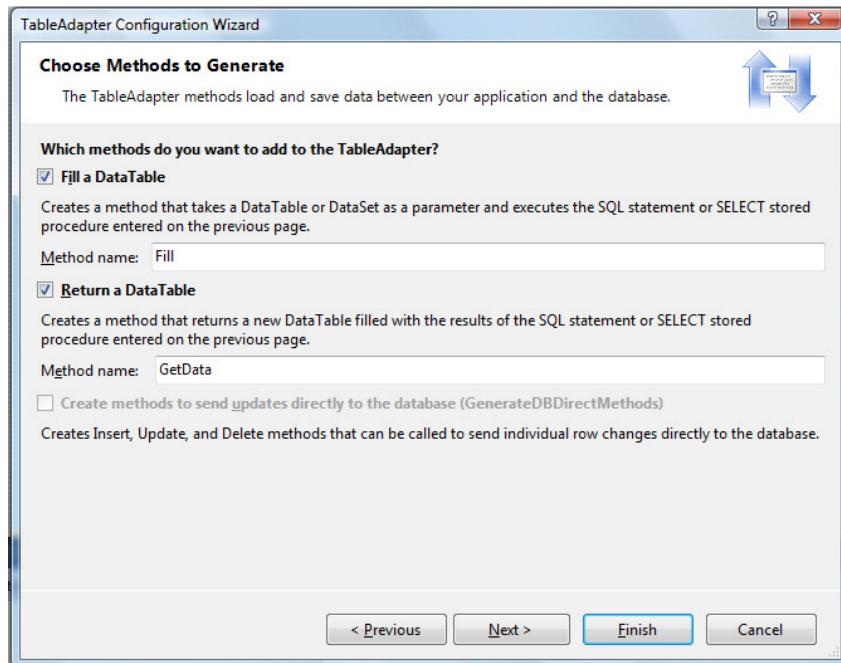
– Selecione o tipo de Comando a ser utilizado no TableAdapter.



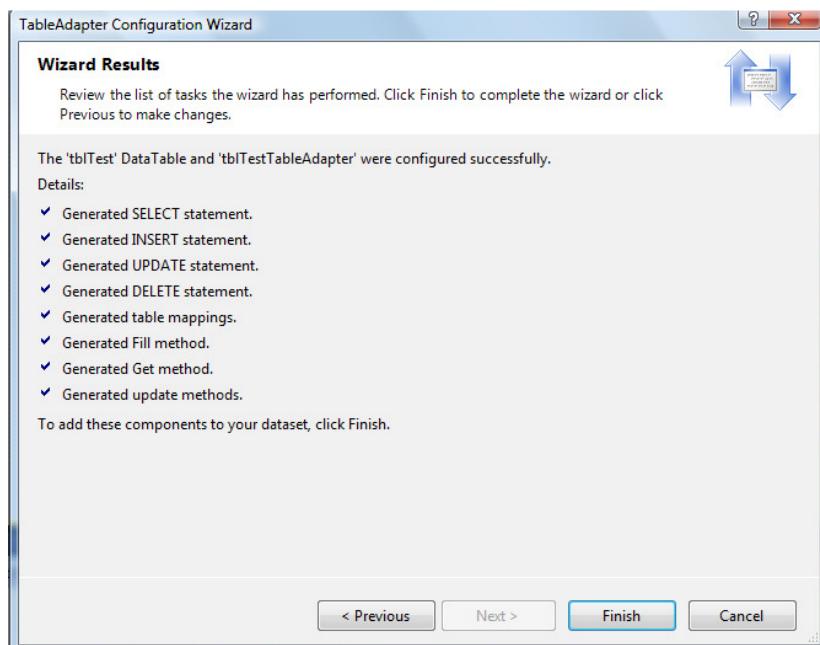
– Digite o comando SQL para execução do query.



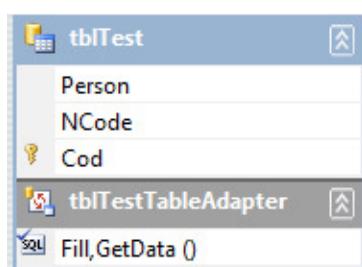
– Escolha os métodos que serão utilizados junto ao TableAdapter.



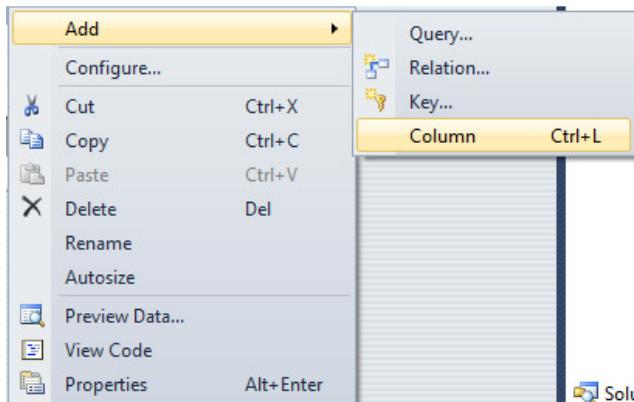
– Finalize o Wizard.



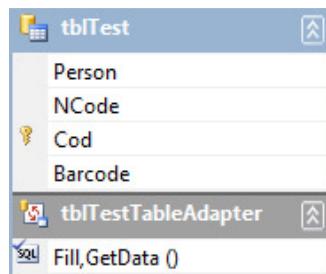
– O DataSet está pronto e configurado no projeto.



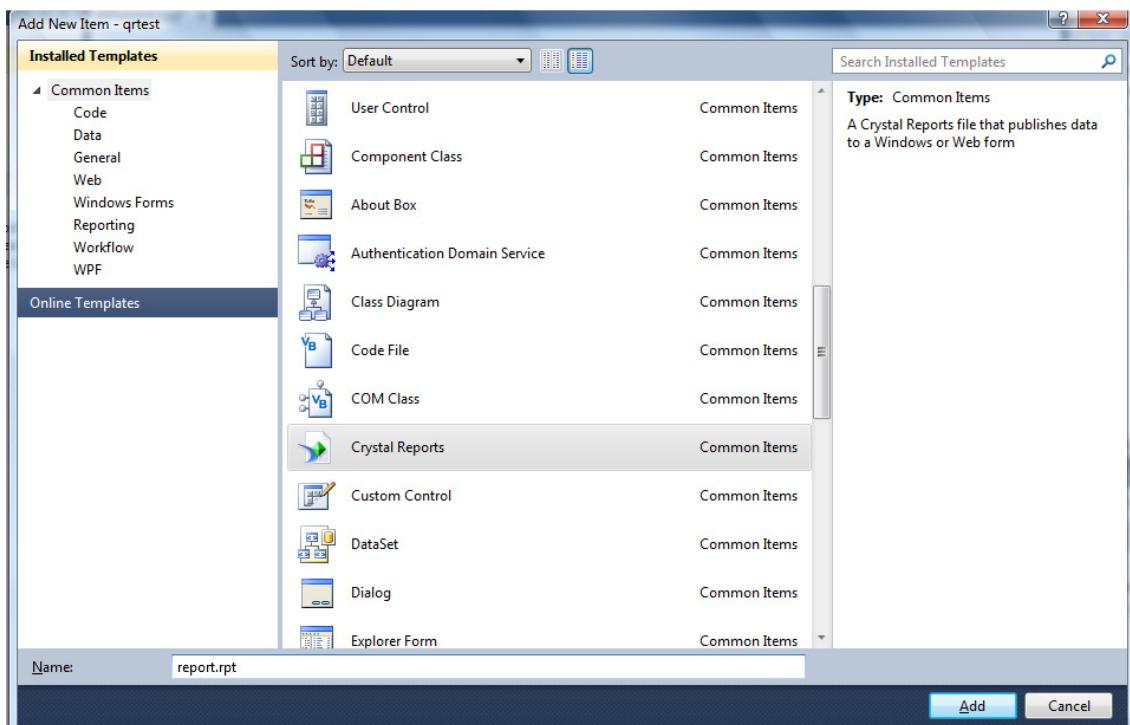
– Clique com o botão direito no DataSet e escolha Add – Column.



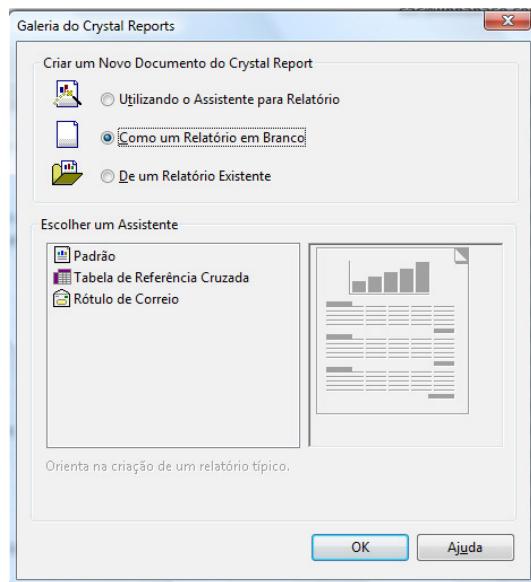
– Adicione uma coluna tipo String que será utilizada para armazenar o mapeamento do código de barras.



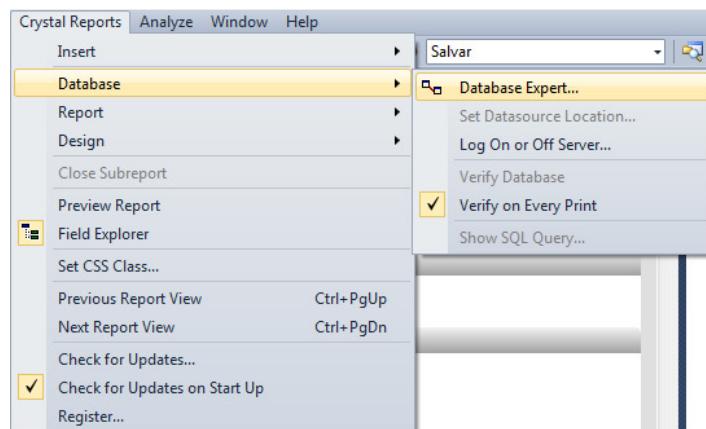
– Adicione ao projeto um relatório Crystal Reports.



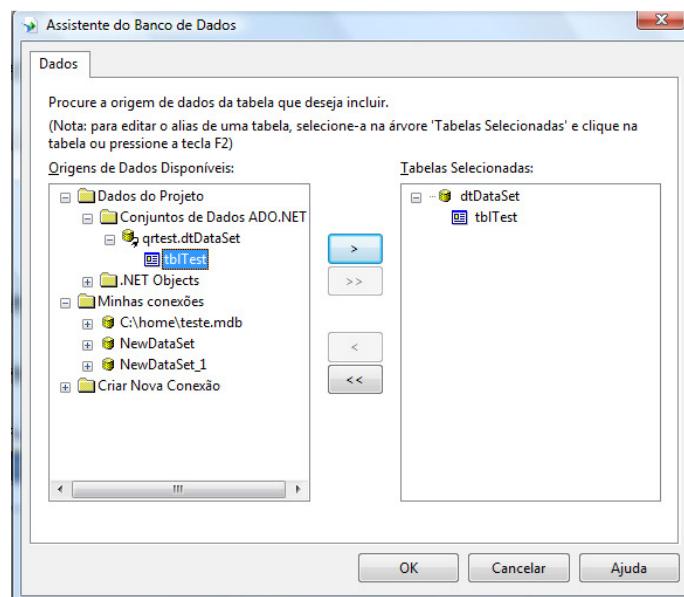
– Crie um relatório em branco.



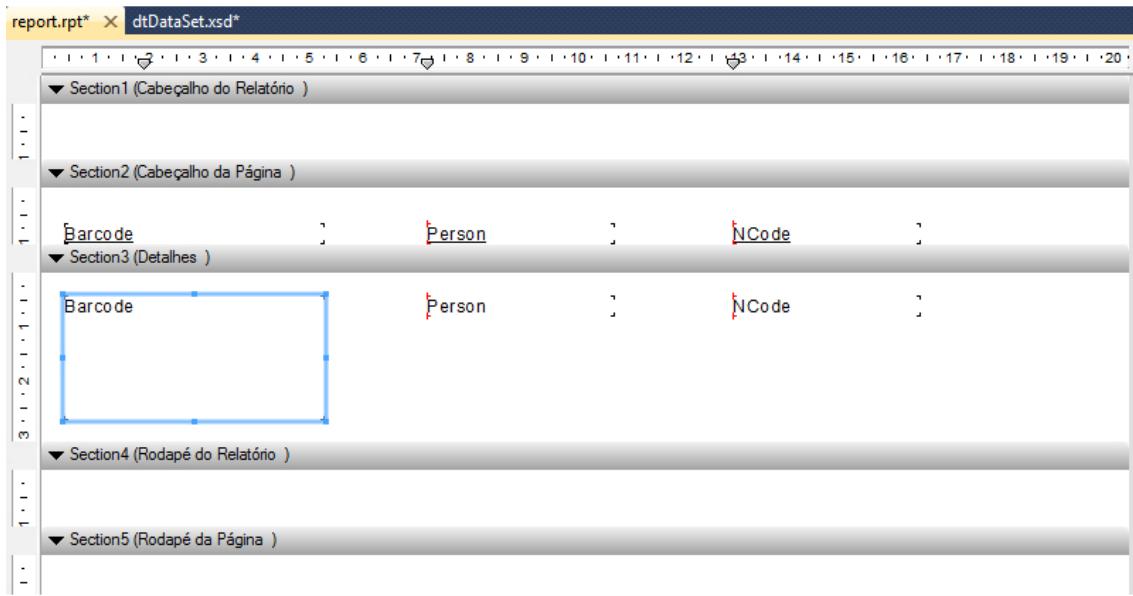
– Selecione o menu Crystal Reports – Database – Database Expert.



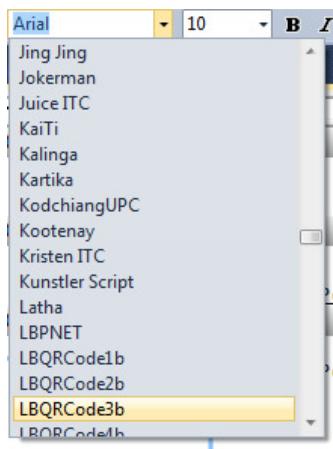
– Selecione o Dataset adicionado no projeto como a fonte de dados do relatório.



– Adicione ao relatório os campos desejados e configure o tamanho dos campos.

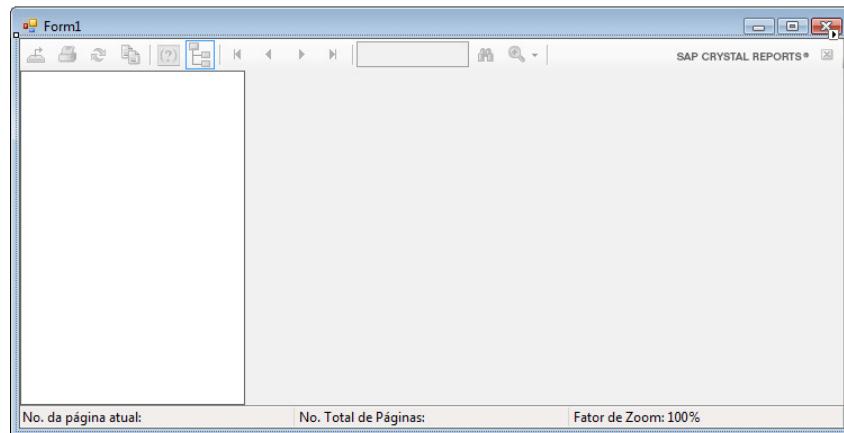


– Clique no campo do código de barras e selecione a fonte de código de barras e ajuste seu tamanho.



OBS: O tamanho do campo do código de barras deve ser ajustado de modo que todo o código de barras seja reproduzido de forma correta com o relatório em modo de execução. Veja em Ajuste do Tamanho do Campo do Código de Barras abaixo.

– Adicione ao Form do projeto o controle CrystalReportViewer.



– Adicione o código de seu projeto.

```
Imports LinhaBaseQrMapping.Map

Public Class Form1

    Private Sub Form1_Load(sender As System.Object, e As System.EventArgs)
Handles MyBase.Load
        Dim ta As New dtDataSetTableAdapters.tblTestTableAdapter()
        Dim dt As New dtDataSet.tblTestDataTable()

        ta.Fill(dt)

        Dim row As dtDataSet.tblTestRow
        Dim QRMapping As New LinhaBaseQrMapping.Map
        Dim Mapeamento As String

        For Each row In dt.Rows
            Mapeamento = QRMapping.QrEncode(row.Person.ToString, "L", 8)
            row.Barcode = Mapeamento
        Next

        Dim report As New report()
        report.SetDataSource(CType(dt, DataTable))

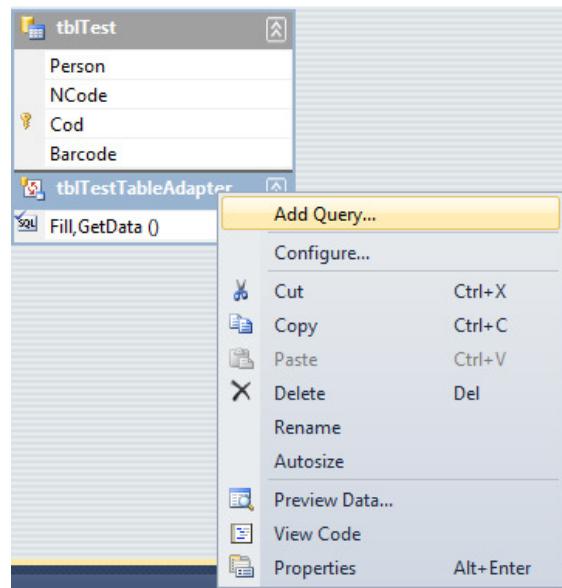
        CrystalReportViewer1.ReportSource = report
    End Sub
End Class
```

Barcode	Person	NCode
	John	1234
	Mary	2345
	Sofia	3456

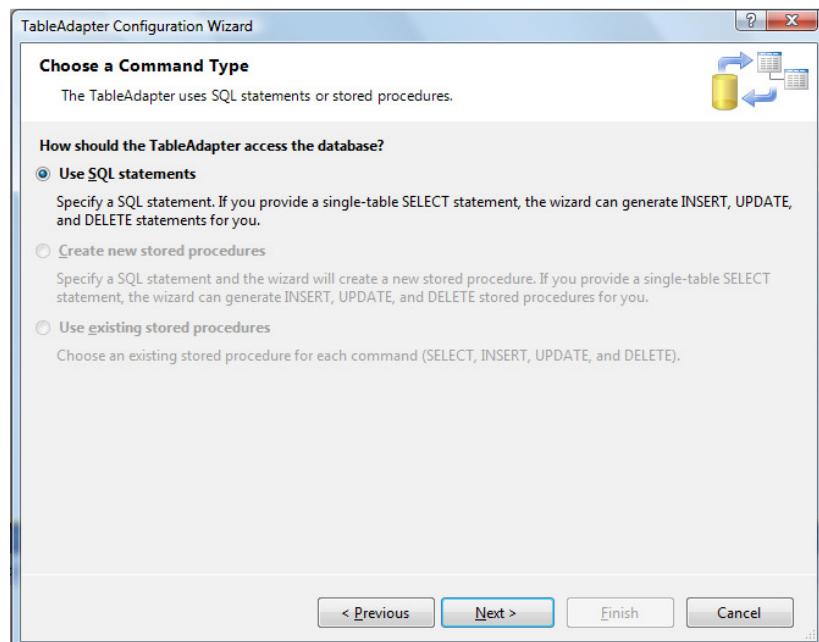
Criando Código de Barras com Crystal Reports para Visual Studio .Net com Dataset executando um query com valor variável em modo de execução

No exemplo anterior o Dataset foi preenchido com um comando SQL do tipo Select realizando um fill completo da base de dados. Mas também é possível preencher o Dataset com um valor qualquer definido em modo de execução. Para isso basta criar um Query com parâmetro.

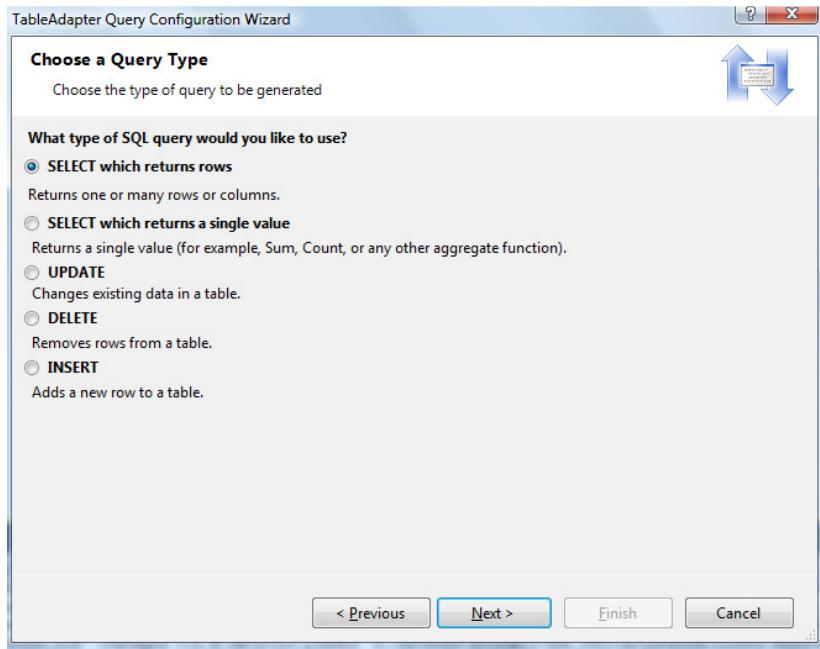
- Clique com o botão direito sobre o TableAdapter e escolha Add Query.



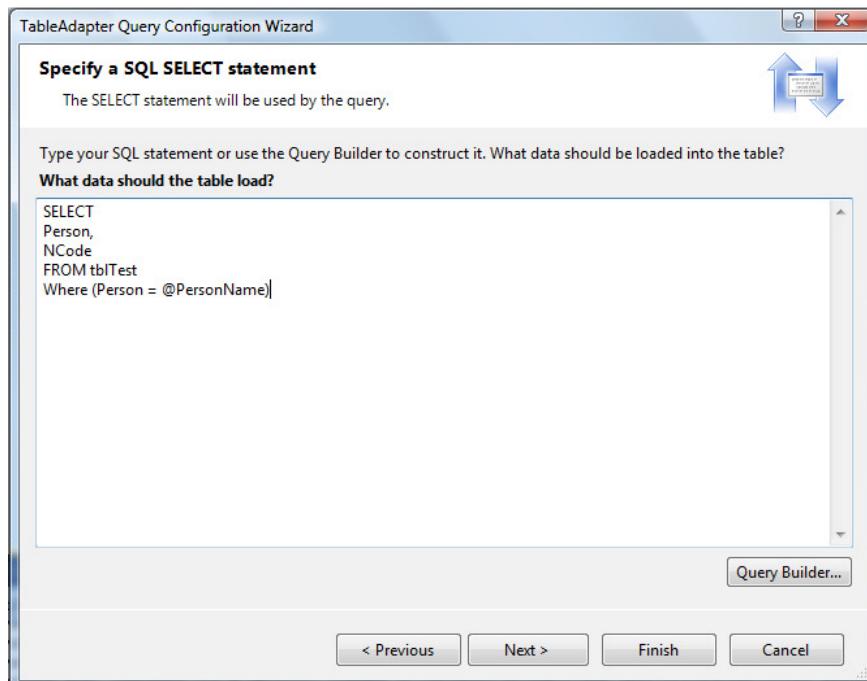
- Selecione o tipo de Comando a ser utilizado no TableAdapter.



– Selecione o tipo de query a ser utilizado.



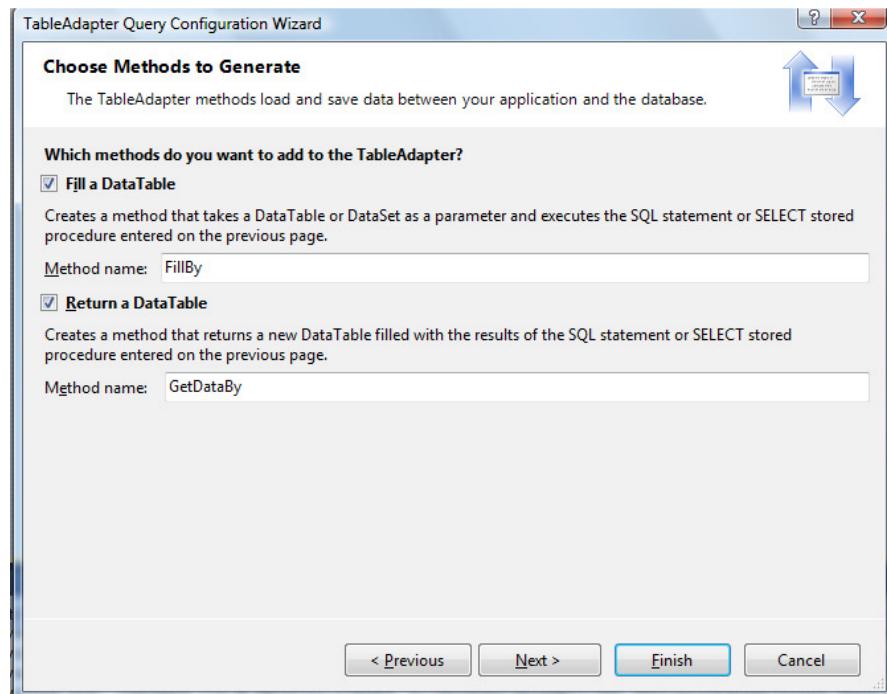
– Digite o comando do query e clique no botão Next.



Foi acrescentado a cláusula Where onde foi definido que a pesquisa será realizada com base no valor a ser fornecido para a coluna Person.

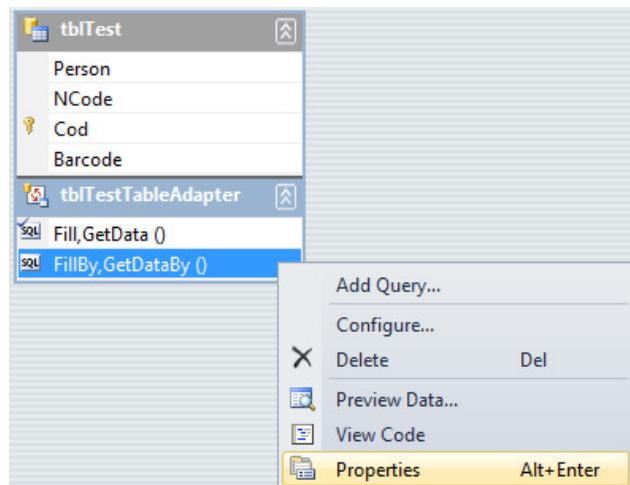
Select Person, NCode From tblTest Where (Person = @PersonName)

– Selecione os métodos a serem utilizados no TableAdapter e clique no botão Finish.

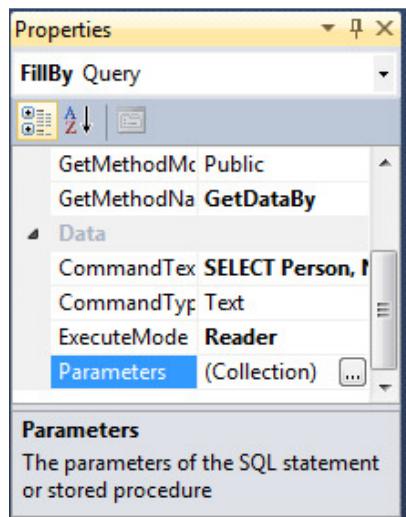


– Clique no botão Ok para ignorar qualquer mensagem de erro junto ao query.

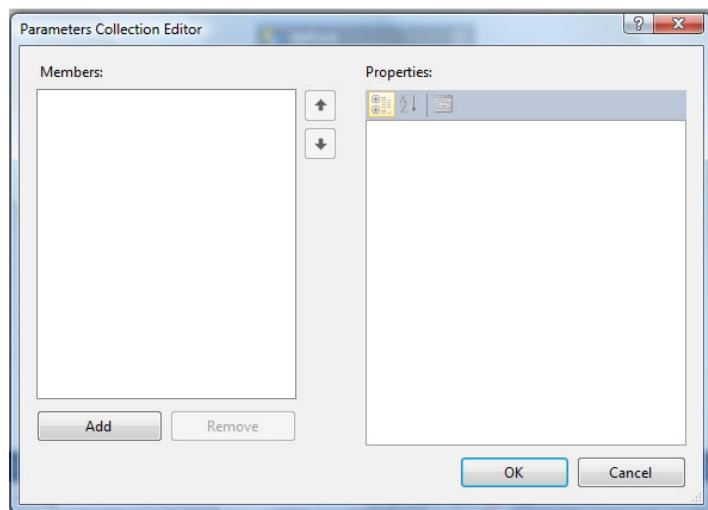
– Clique com o botão direito na linha que contém os métodos do query e selecione Properties.



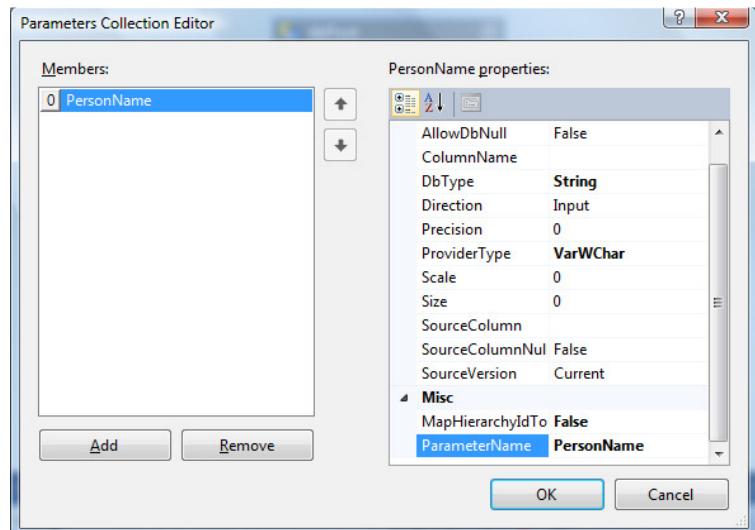
– Selecione Parameters.



– Clique no botão Add.



– Em DbType escolha String e em ParameterName digite o nome para o parâmetro do query e clique no botão Ok.



– O código ficará como apresentado abaixo:

```
Imports LinhaBaseQrMapping.Map

Public Class Form1

    Private Sub Form1_Load(sender As System.Object, e As System.EventArgs)
Handles MyBase.Load
        Dim ta As New dtDataSetTableAdapters.tblTestTableAdapter()
        Dim dt As New dtDataSet.tblTestDataTable()

        ta.FillBy(dt, "Mary")

        Dim row As dtDataSet.tblTestRow
        Dim QRMapping As New LinhaBaseQrMapping.Map
        Dim Mapeamento As String

        For Each row In dt.Rows
            Mapeamento = QRMapping.QrEncode(row.Person.ToString, "L", 8)
            row.Barcode = Mapeamento
        Next

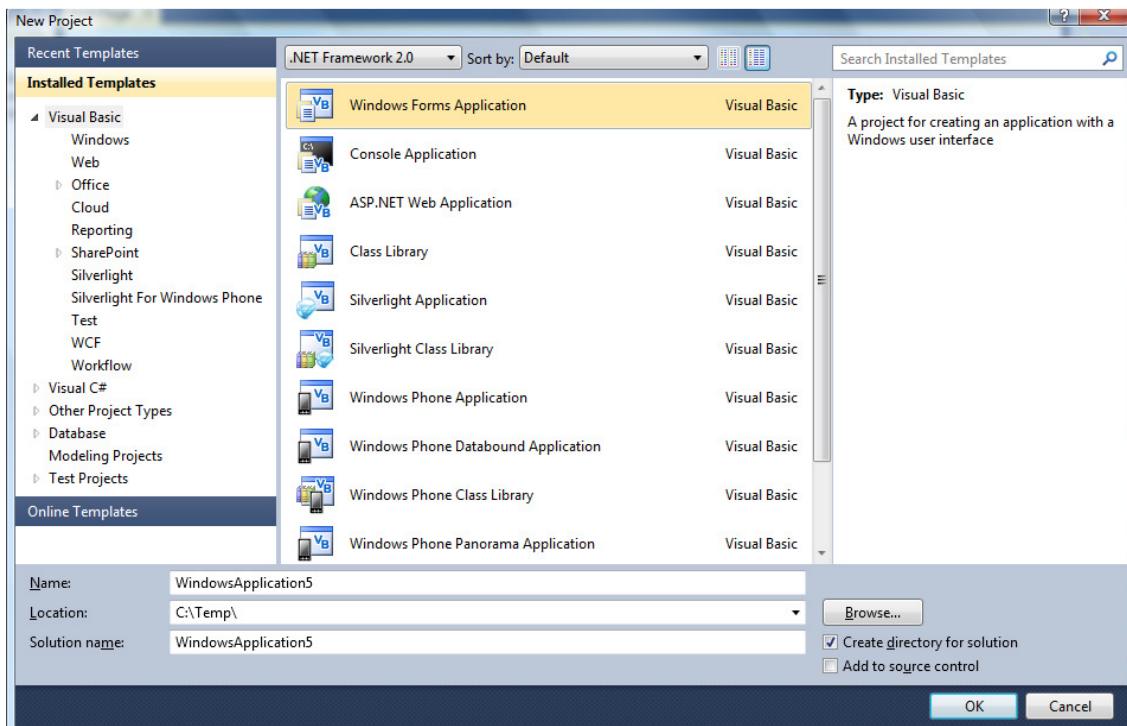
        Dim report As New report()
        report.SetDataSource(CType(dt, DataTable))

        CrystalReportViewer1.ReportSource = report
    End Sub
End Class
```

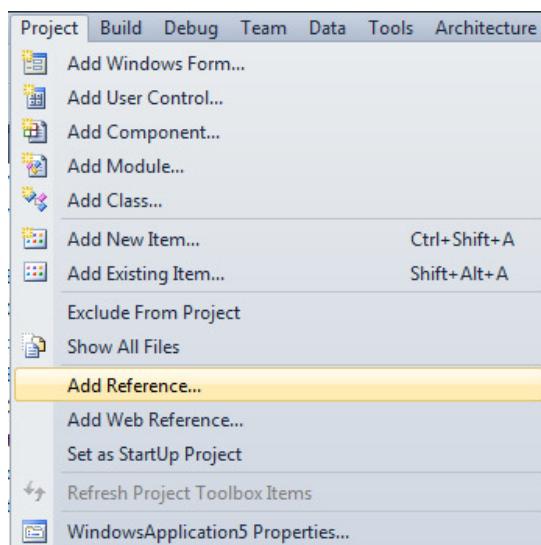


Criando Código de Barras com Report Viewer para Visual Studio .Net

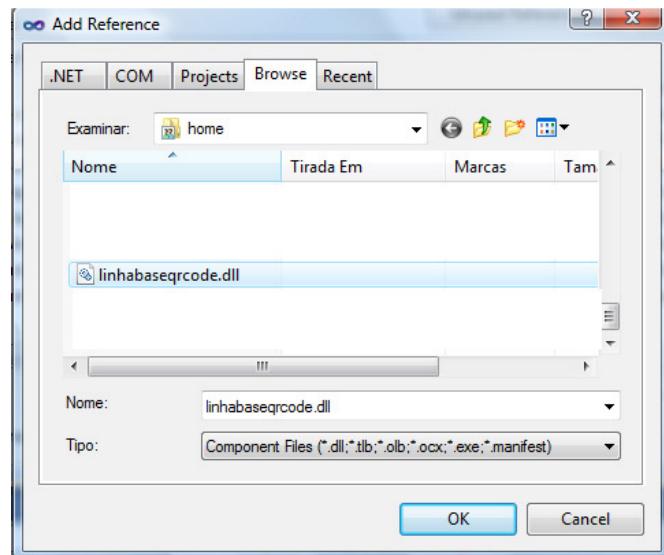
– Abra o Visual Studio.Net e crie uma nova aplicação Windows.



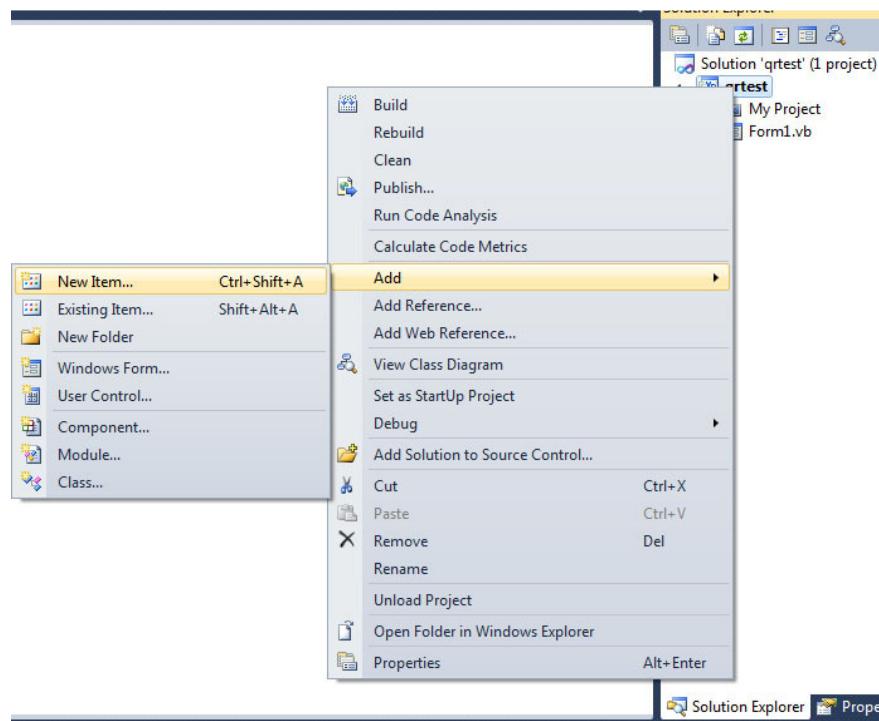
– Clique em Project – Add Reference para adicionar referência ao arquivo LinhabaseqrCode.dll.



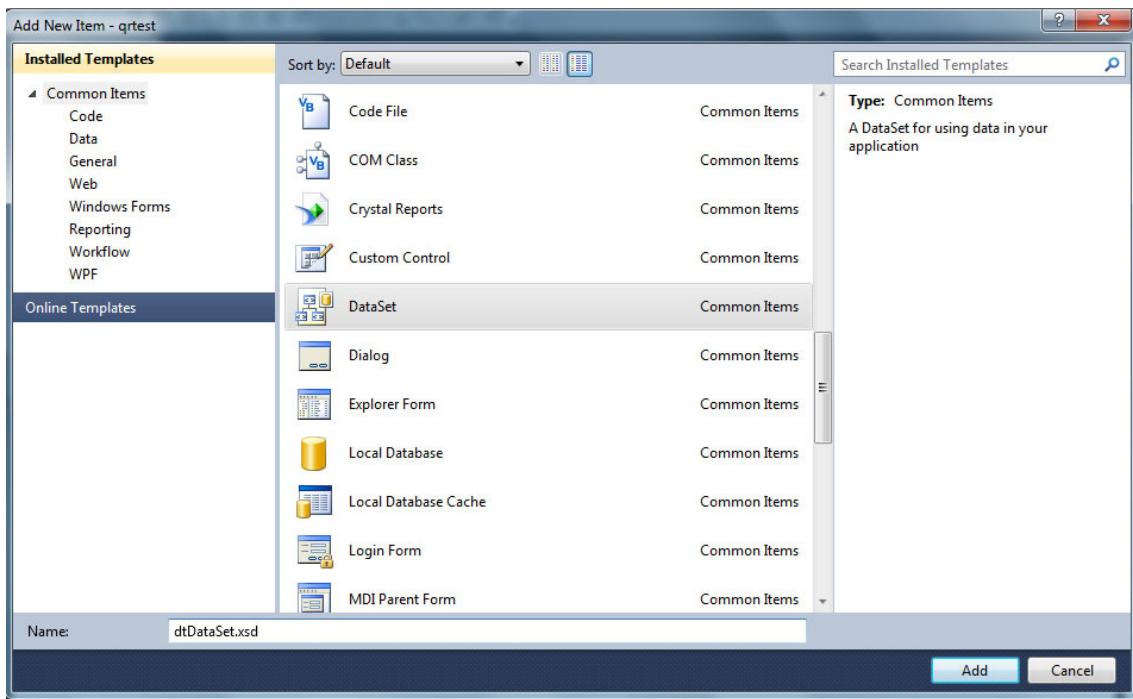
– Localize o arquivo LinhabaseqrCode.dll.



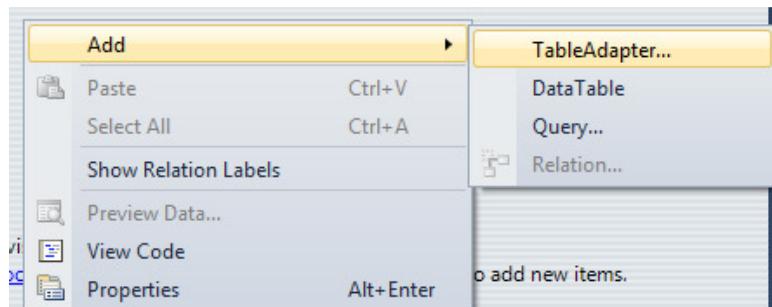
– Em Solution Explores clique com o botão direito selecione Add – New Item.



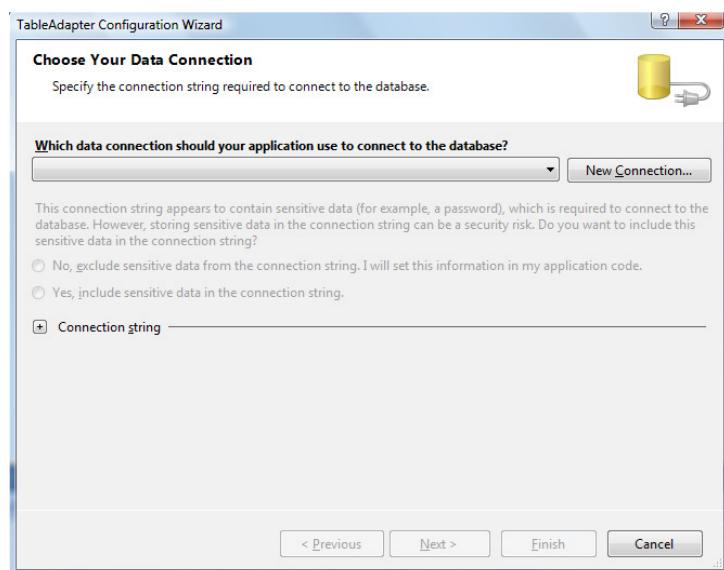
– Adicione um Dataset ao projeto.



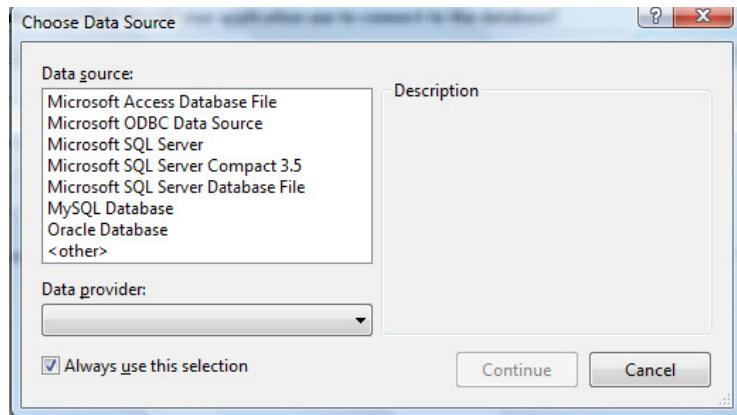
– Clique com o botão direito na área do Dataset e escolha Add – Table Adapter.



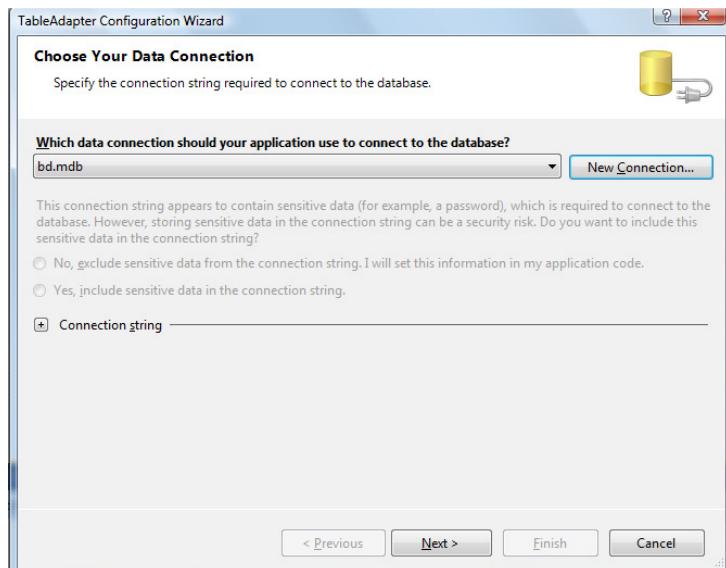
– Escolha a conexão com sua base de dados.



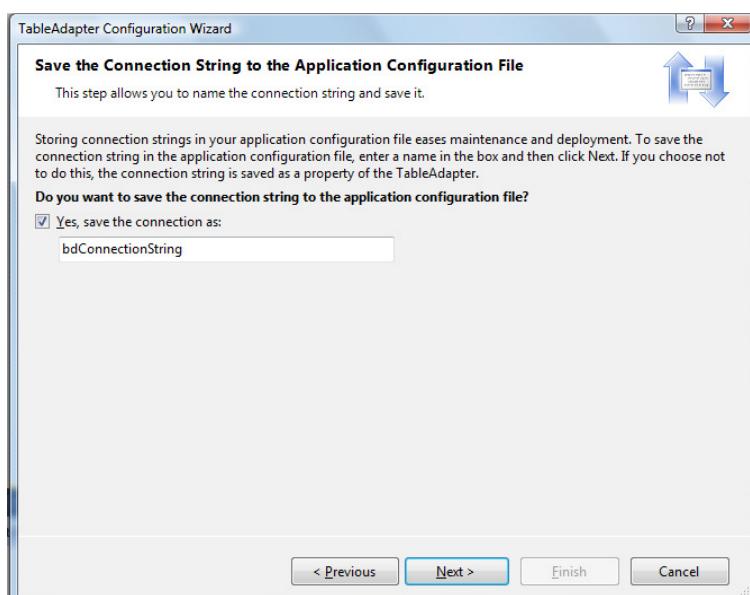
– Clique no botão New Connection e selecione o Data Source.



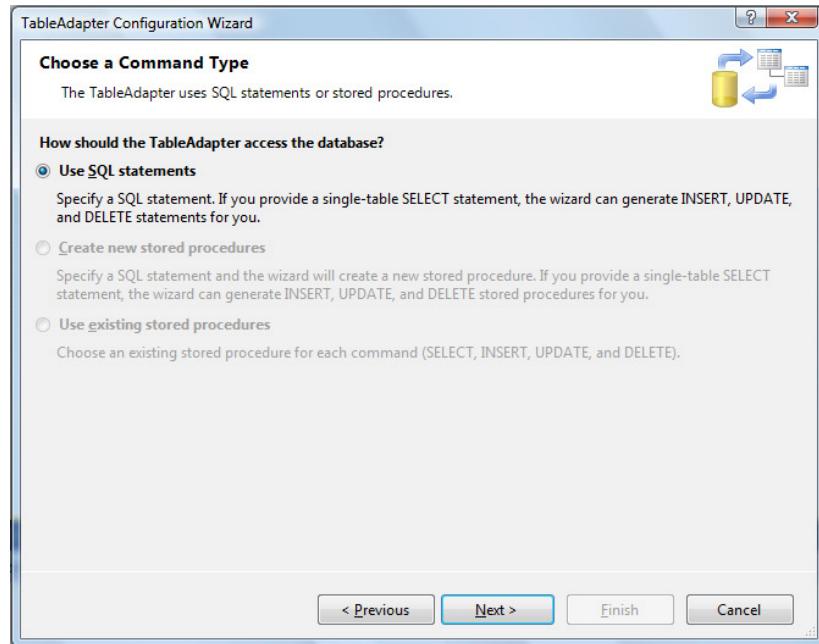
– Após ter configurado a conexão com a base de dados clique no botão Next.



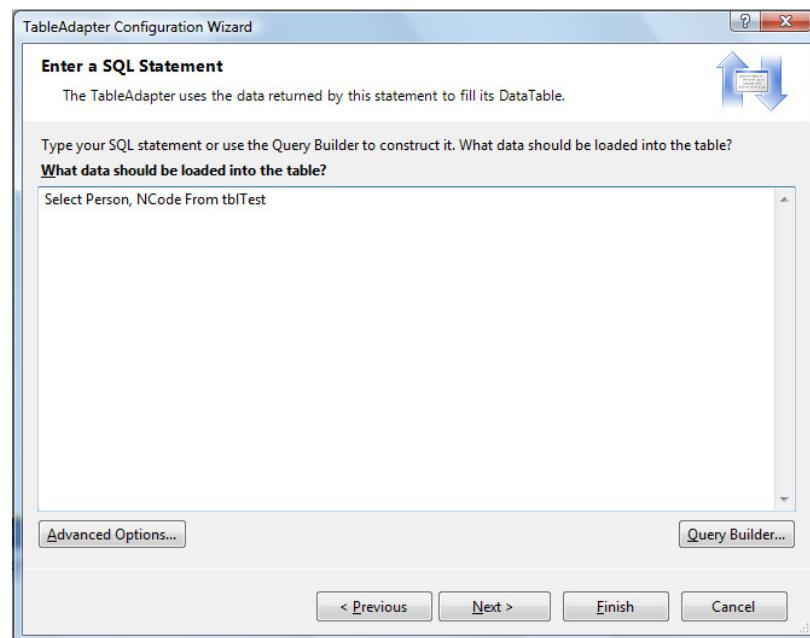
– Salve a conexão com a base de dados e clique no botão Next.



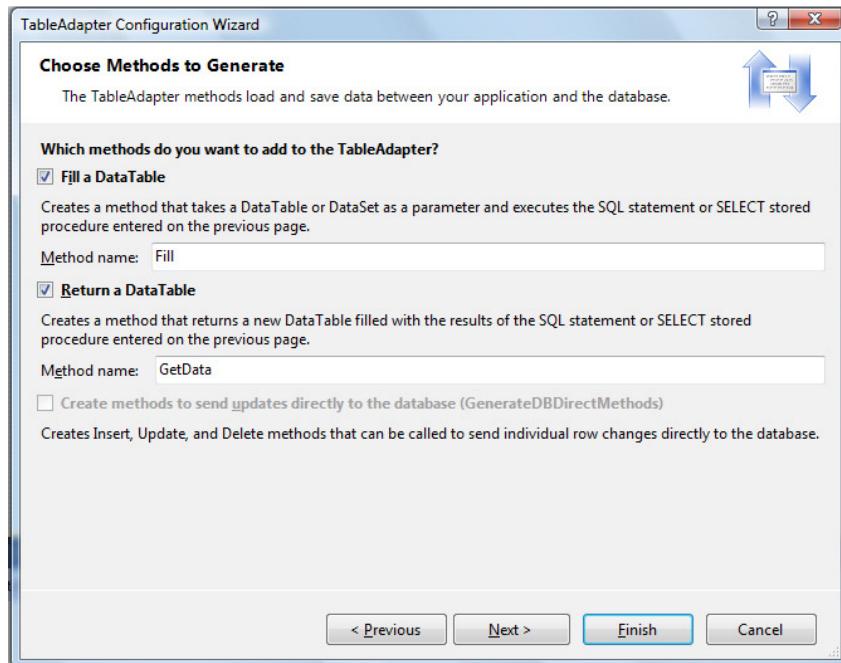
– Selecione o tipo de Comando a ser utilizado no TableAdapter.



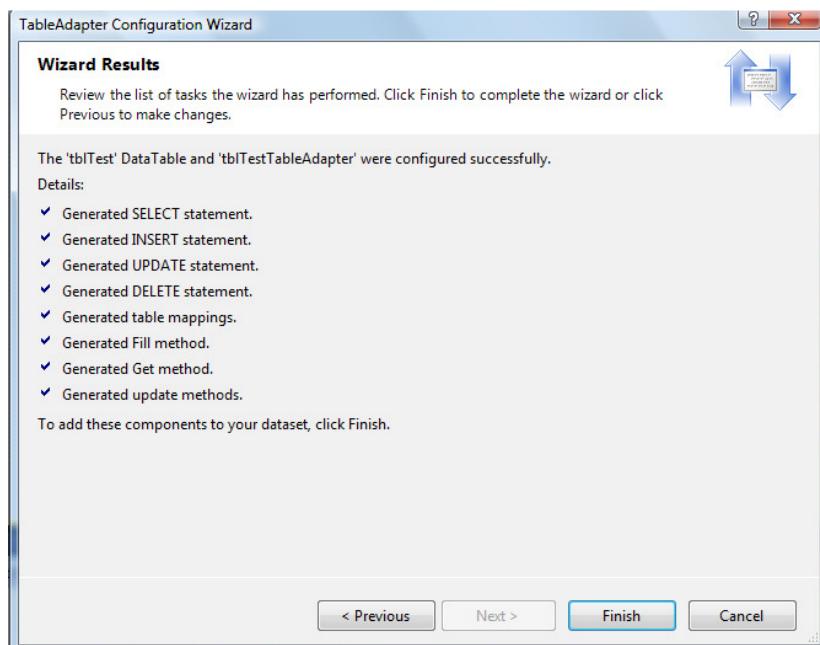
– Digite o comando SQL para execução do query.



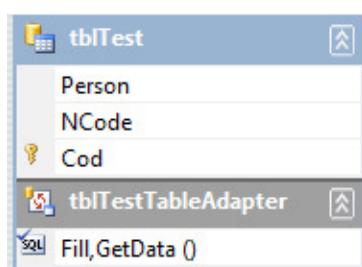
– Escolha os métodos que serão utilizados junto ao TableAdapter.



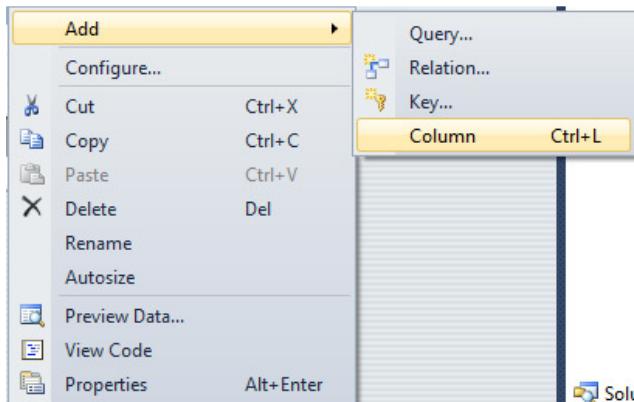
– Finalize o Wizard.



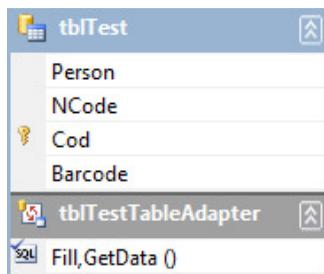
– O DataSet está pronto e configurado no projeto.



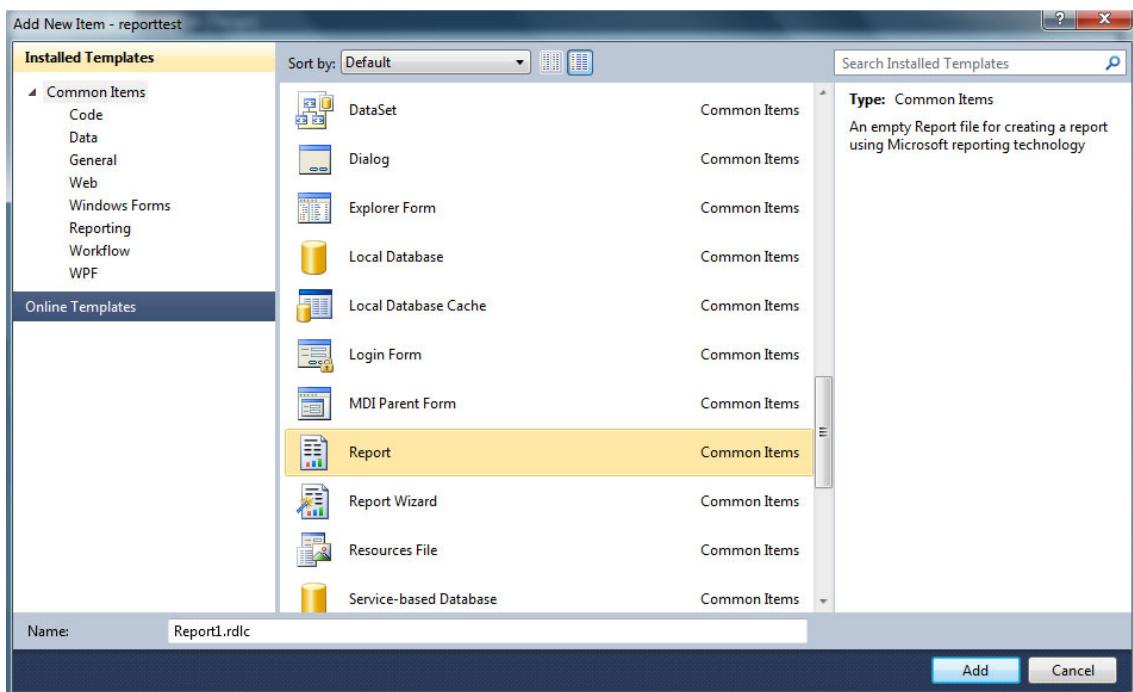
– Clique com o botão direito no DataSet e escolha Add – Column.



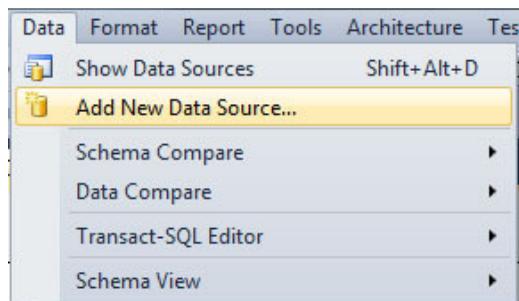
– Adicione uma coluna tipo String que será utilizada para armazenar o mapeamento do código de barras.



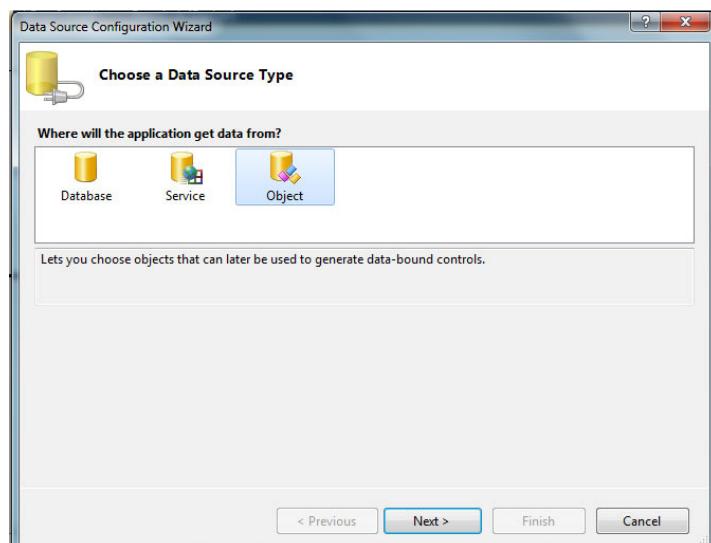
– Adicione ao projeto um relatório tipo Report.



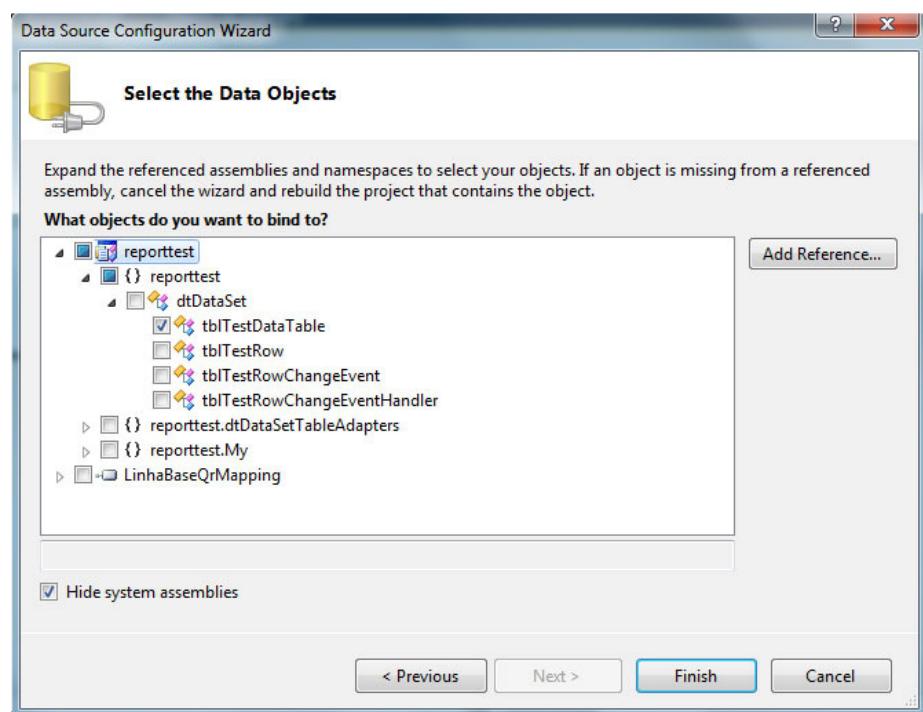
– Selecione o menu Data – Add New Data Source.



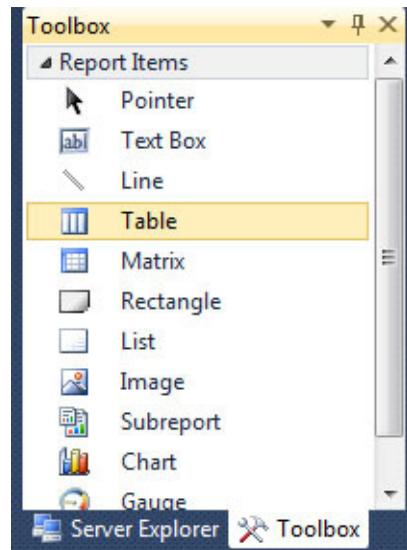
– Selecione Object e clique no botão Next.



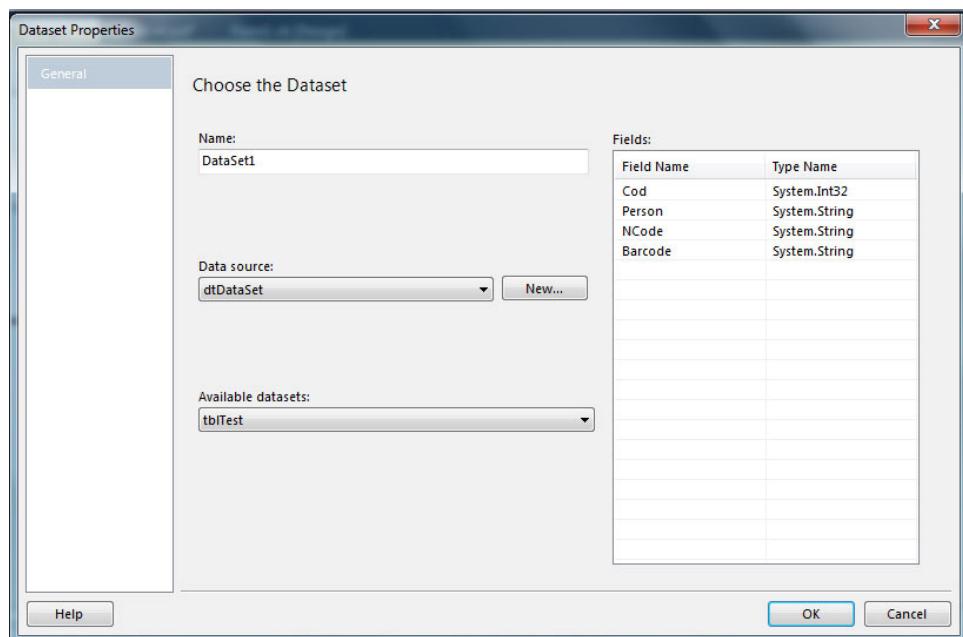
– Selecione o objeto que representa o DataTable do Dataset adicionado ao projeto e clique no botão Finish.



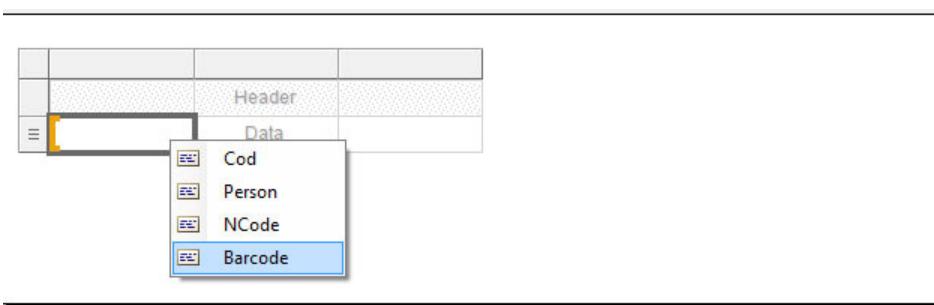
– no Toolbox do Report selecione Table e insira no relatório.



– Na tela Dataset Properties selecione o Data Source que foi criado no projeto e clique no botão Ok.

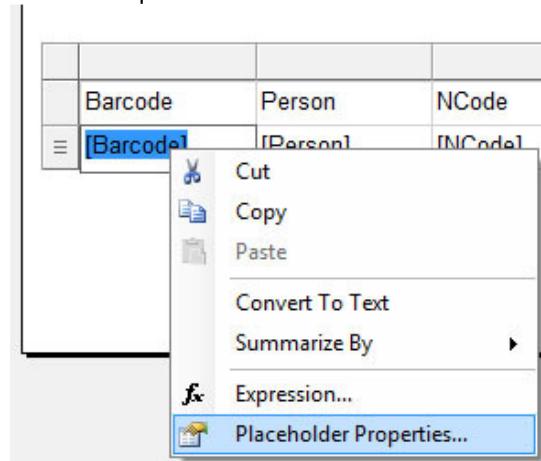


– Adicione a tabela os campos que deseja incluir no relatório.

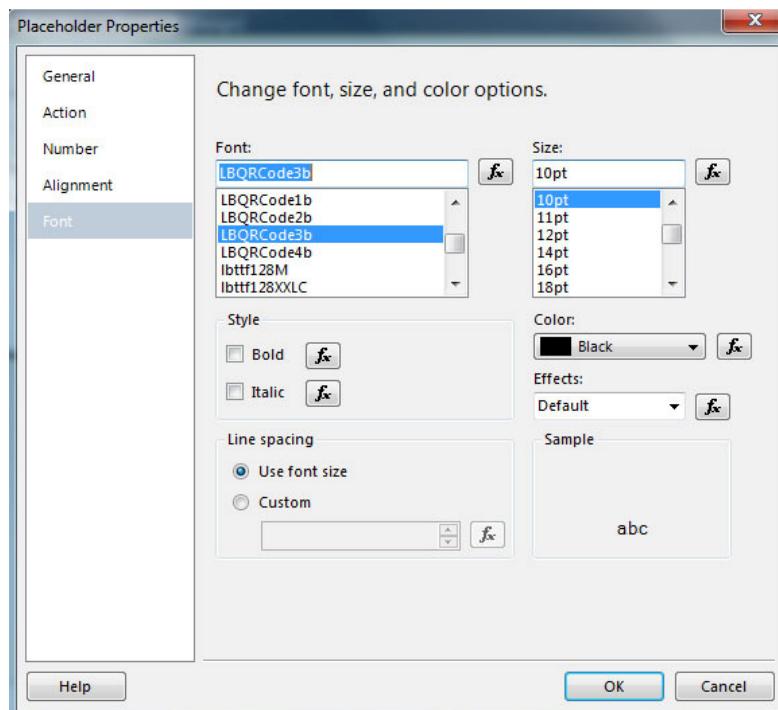


Barcode	Person	NCode
[Barcode]	[Person]	[NCode]

– Na coluna do código de barras clique com o botão direito e escolha Placeholder Properties.

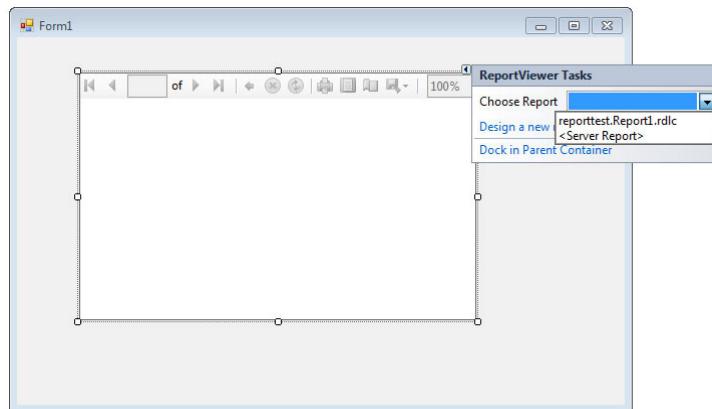


– Escolha a fonte de código de barras apropriada e ajuste seu tamanho.



OBS: O tamanho do campo do código de barras deve ser ajustado de modo que todo o código de barras seja reproduzido de forma correta com o relatório em modo de execução. Veja em Ajuste do Tamanho do Campo do Código de Barras abaixo.

- Adicione ao Form o controle ReportViewer, selecione o report do projeto e ajuste a propriedade Dock para Fill.



- Ajuste o código do Form.

```
Imports LinhaBaseQrMapping.Map

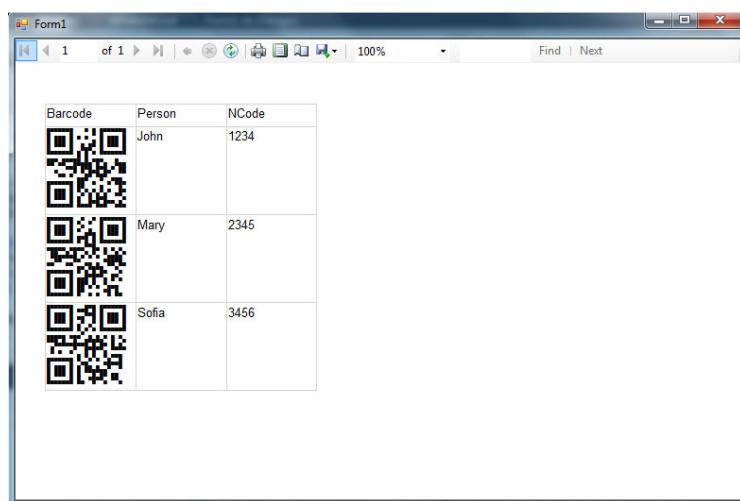
Public Class Form1

    Private Sub Form1_Load(sender As System.Object, e As System.EventArgs)
Handles MyBase.Load
    tblTestTableAdapter.Fill(dtDataSet.tblTest)

    Dim row As dtDataSet.tblTestRow
    Dim QRMapping As New LinhaBaseQrMapping.Map
    Dim Mapeamento As String

    For Each row In dtDataSet.tblTest.Rows
        Mapeamento = QRMapping.QrEncode(row.Person.ToString, "L", 8)
        row.Barcode = Mapeamento
    Next

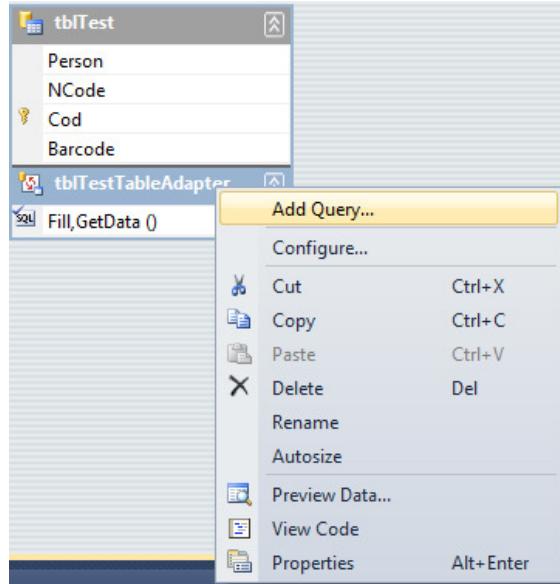
    Me.ReportViewer1.RefreshReport()
End Sub
End Class
```



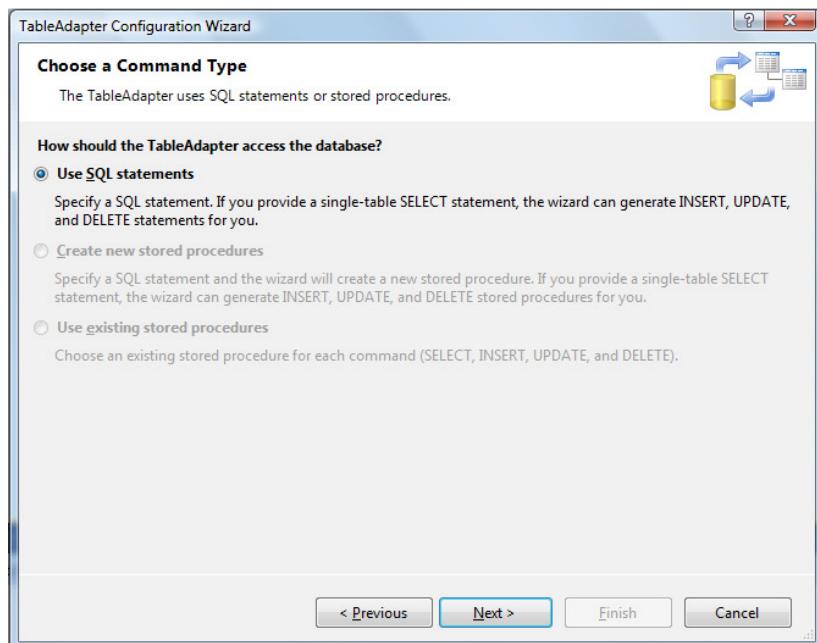
Criando Código de Barras com Report Viewer para Visual Studio .Net com Dataset executando um query com valor variável em modo de execução

No exemplo anterior o Dataset foi preenchido com um comando SQL do tipo Select realizando um fill completo da base de dados. Mas também é possível preencher o Dataset com um valor qualquer definido em modo de execução. Para isso basta criar um Query com parâmetro.

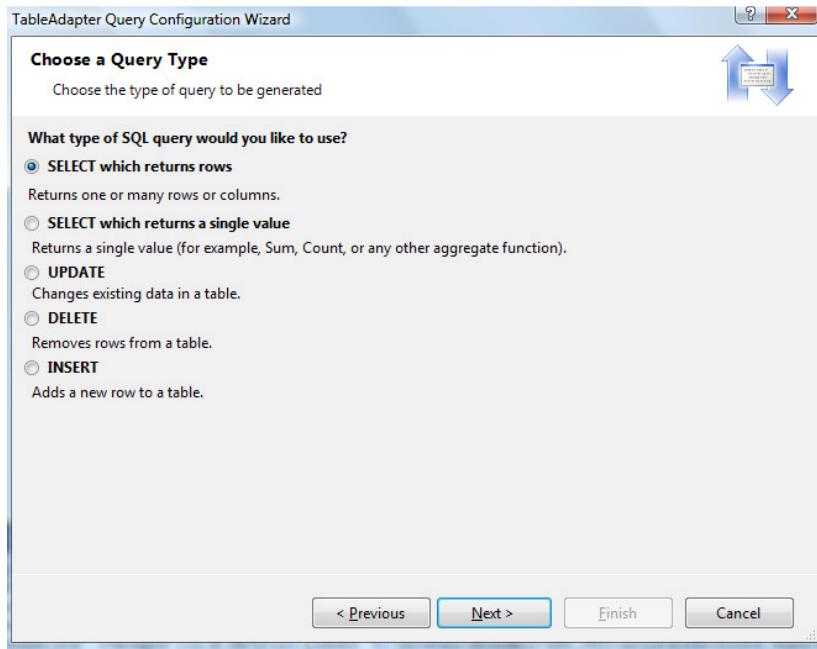
- Clique com o botão direito sobre o TableAdapter e escolha Add Query.



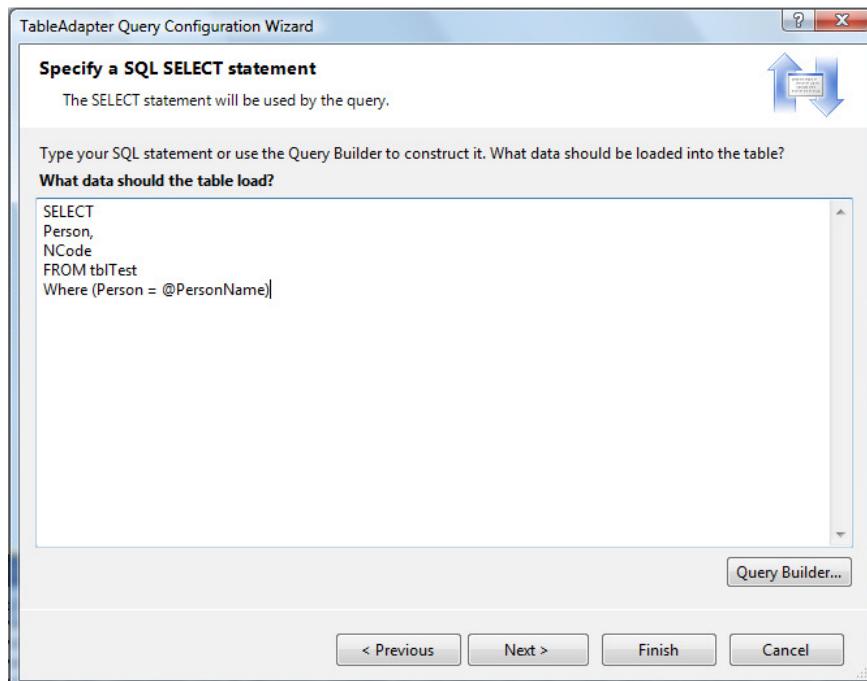
- Selecione o tipo de Comando a ser utilizado no TableAdapter.



– Selecione o tipo de query a ser utilizado.



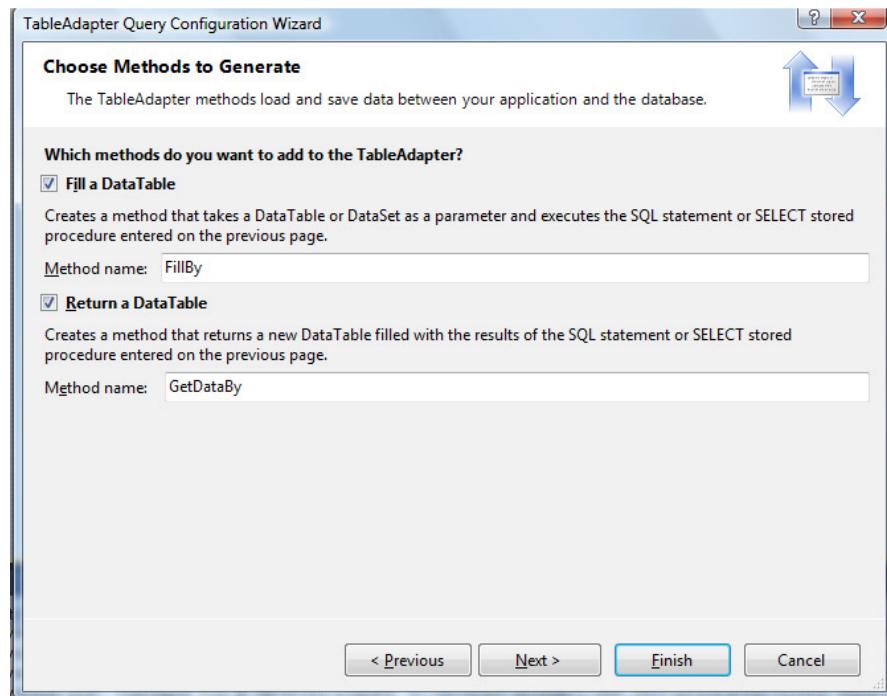
– Digite o comando do query e clique no botão Next.



Foi acrescentado a cláusula Where onde foi definido que a pesquisa será realizada com base no valor a ser fornecido para a coluna Person.

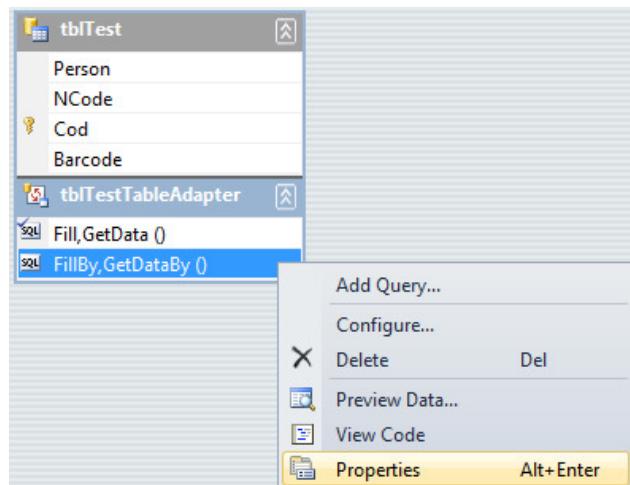
Select Person, NCode From tblTest Where (Person = @PersonName)

– Selecione os métodos a serem utilizados no TableAdapter e clique no botão Finish.

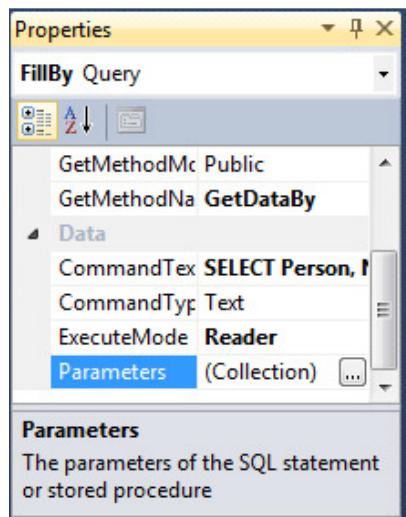


– Clique no botão Ok para ignorar qualquer mensagem de erro junto ao query.

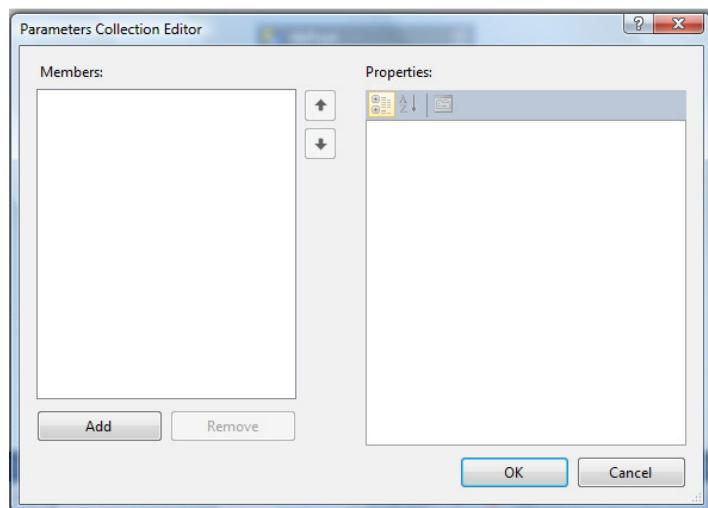
– Clique com o botão direito na linha que contém os métodos do query e selecione Properties.



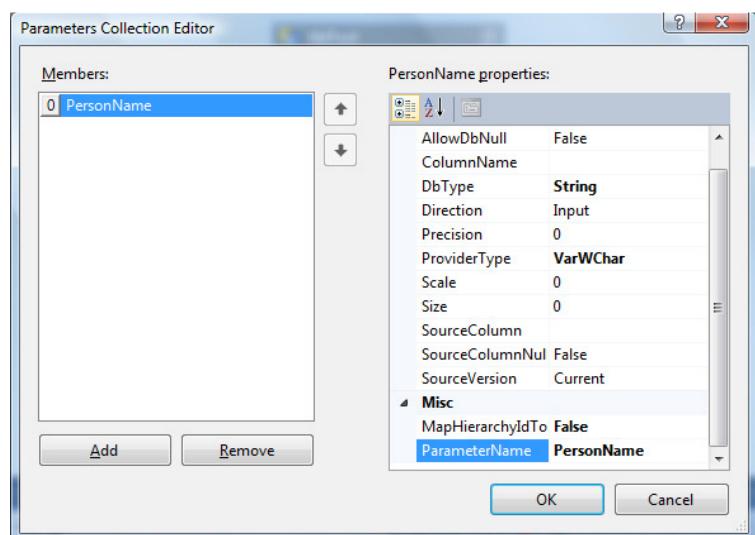
– Selecione Parameters.



– Clique no botão Add.



– Em DbType escolha String e em ParameterName digite o nome para o parâmetro do query e clique no botão Ok.



– O código ficará como apresentado abaixo:

```
Imports LinhaBaseQrMapping.Map

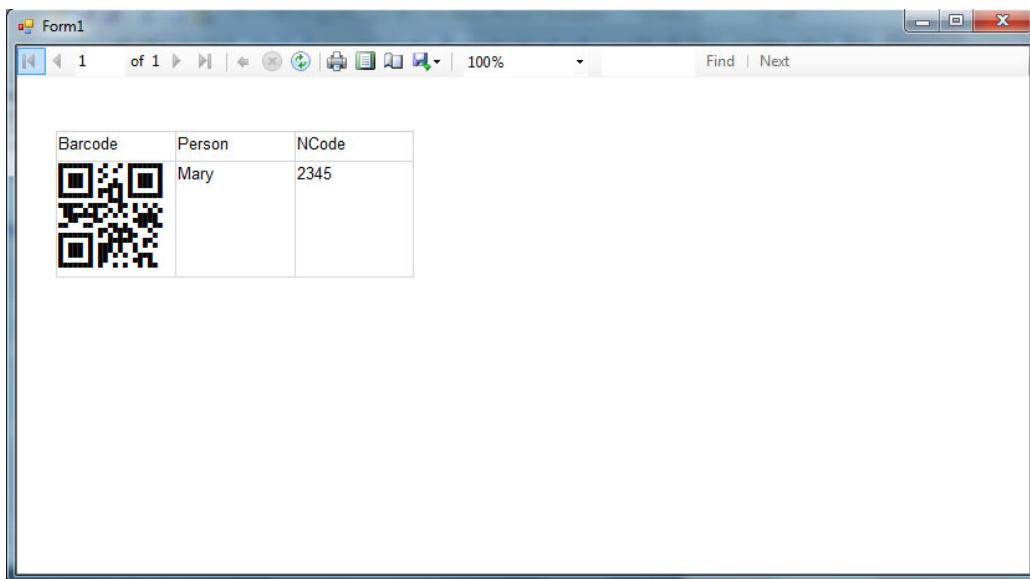
Public Class Form1

    Private Sub Form1_Load(sender As System.Object, e As System.EventArgs)
Handles MyBase.Load
    tblTestTableAdapter.FillBy(dtDataSet.tblTest, "Mary")

    Dim row As dtDataSet.tblTestRow
    Dim QRMapping As New LinhaBaseQrMapping.Map
    Dim Mapeamento As String

    For Each row In dtDataSet.tblTest.Rows
        Mapeamento = QRMapping.QrEncode(row.Person.ToString, "L", 8)
        row.Barcode = Mapeamento
    Next

    Me.ReportViewer1.RefreshReport()
End Sub
End Class
```



Ajuste do Tamanho do Campo do Código de Barras

O campo onde será reproduzido o código de barras deve ter seu tamanho ajustado corretamente para que todo o código de barras seja reproduzido no relatório em modo de execução.

Barcode



Person

John

Campo do código de barras ajustado corretamente para o tamanho da fonte de código de barras escolhida.

Barcode



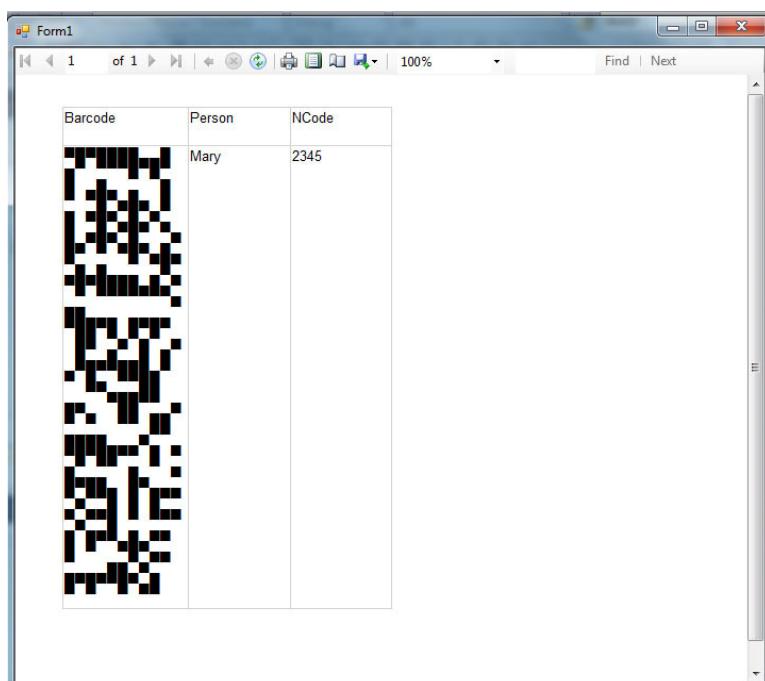
Person

John

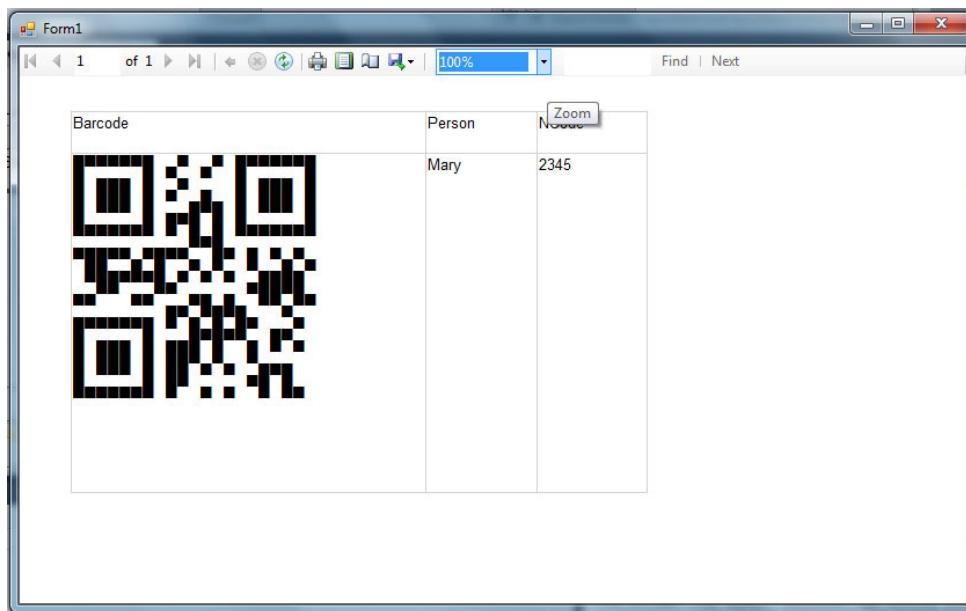
NCode

1234

Campo do código de barras ajustado de maneira incorreta para o tamanho da fonte de código de barras escolhida. Código de barras reproduzido com corte.



Campo do código de barras ajustado de maneira incorreta no Report Viewer para o tamanho da fonte de código de barras escolhida.



Campo do código de barras ajustado corretamente no Report Viewer para o tamanho da fonte de código de barras escolhida.

Apêndices

Erros

O código de barras não pode ser lido pelo equipamento decodificador

Resposta:

1 – Verifique se o tamanho do código de barras não é muito pequeno ou grande demais para o equipamento leitor. Os leitores de código de barras trabalham dentro de características de sensibilidade dado por cada fabricante.

2 – Verifique se as cores escolhidas para o fundo e as barras não são próximas. É necessário que sejam escolhidas cores onde existam diferenças significativas entre as barras e o fundo, como barras pretas e fundo branco.

4 – Verifique se o tipo de código de barras escolhido pode ser lido pela leitora. Nem todos os códigos de barras são compatíveis com o modelo de leitora utilizado.

5 – Verifique se a impressora utilizada está imprimindo um código de barras de boa qualidade.

6 – Verifique se a quantidade de caracteres contidos no código de barras ultrapassa a quantidade máxima de caracteres permitidos pelo modelo da leitora. A maioria das leitoras de código de barras trabalha com um buffer interno que limita a quantidade máxima de caracteres por código de barras.

7 – Verifique se o Word ou outro editor de texto realizou alguma alteração nos caracteres ou correção ortográfica automática.

Informações

Figuras Ilustrativas

As figuras ilustrativas foram tiradas de uma determinada versão do Pacote de Fontes de Código de Barras, podendo ser ligeiramente diferentes em outras versões do pacote.

Suporte

O suporte a este produto se restringe apenas as fontes fornecidas no pacote. A criação, modificação, implantação e/ou programação do código fonte ou de outros que possam se originar dos exemplos contidos neste manual para qualquer linguagem de programação e ou sistema é de total responsabilidade do usuário.

O conteúdo deste pacote é protegido pelas leis federais 9.609 de 19 de fevereiro de 1998 e 9.610 de 19 de fevereiro de 1998 e por tratados internacionais. A reprodução ou a distribuição não autorizadas deste pacote de programas e ou arquivos, ou de qualquer porção deste, pode gerar penalidades civis e criminais.

