



Programação Visual

Aplicações Windows com uso de Conceitos Visuais

▶
▶
1
10



Prof. Eng. Ederson Cichaczewski, MSc

Conversa Inicial

■ Tema 1

- Ambiente de Desenvolvimento Visual Studio .NET

■ Tema 2

- Windows Forms





Tema 3



**Interface gráfica
com C#**



Tema 4



**Interface gráfica
com VB.NET**



Tema 5



**Desenvolvimento de
aplicações gráficas**

Contextualizando

- Como são desenvolvidos os aplicativos baseados em Windows?
- Ferramenta: Visual Studio .NET.
- Windows forms com C# e Visual Basic .NET.



4
53



Tema 1: Programação com IDE Microsoft Visual Studio .NET

5
30



Conhecendo a ferramenta Microsoft Visual Studio .NET

- .NET Framework: máquina virtual.

- Conceitos:

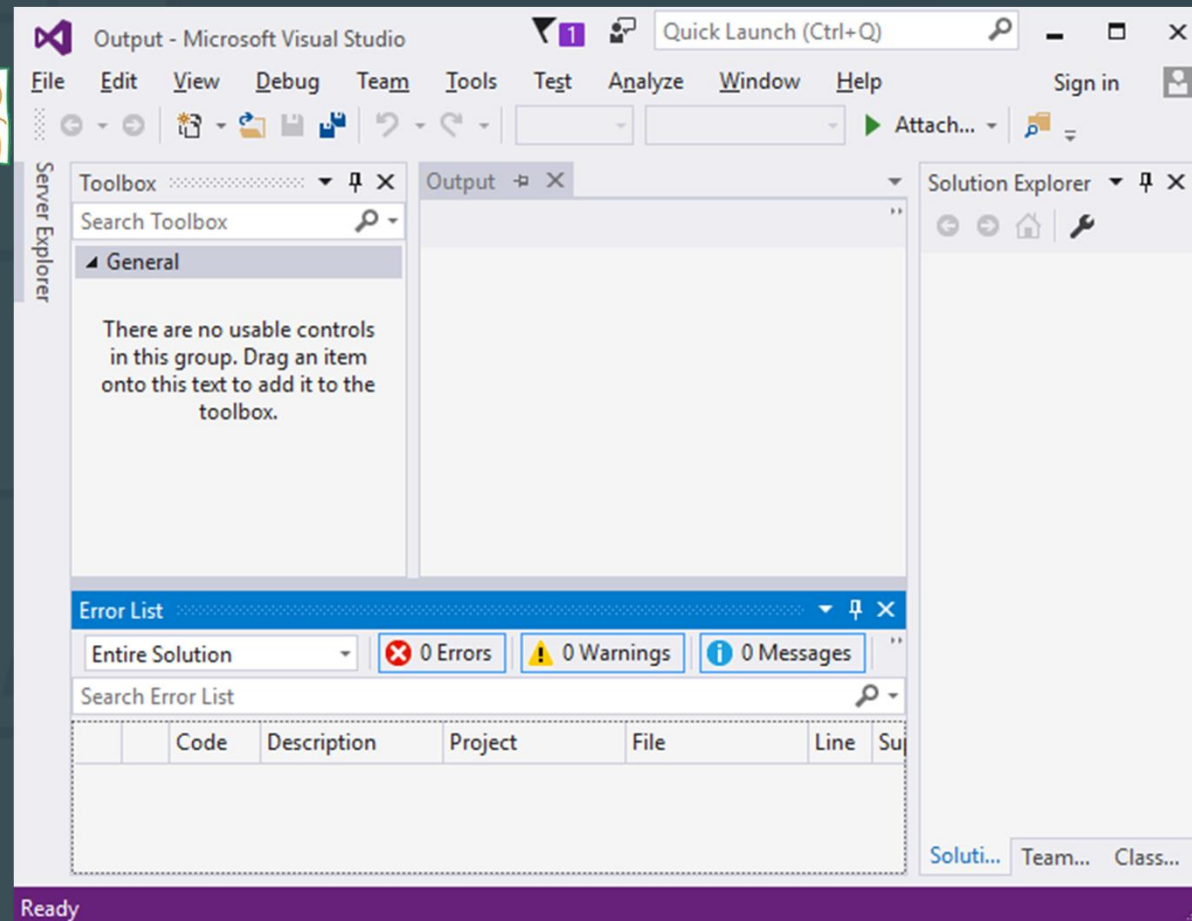
- Solution (.sln)
- Project (.csproj, .vbproj)
- Project Items (.cs, .vb)

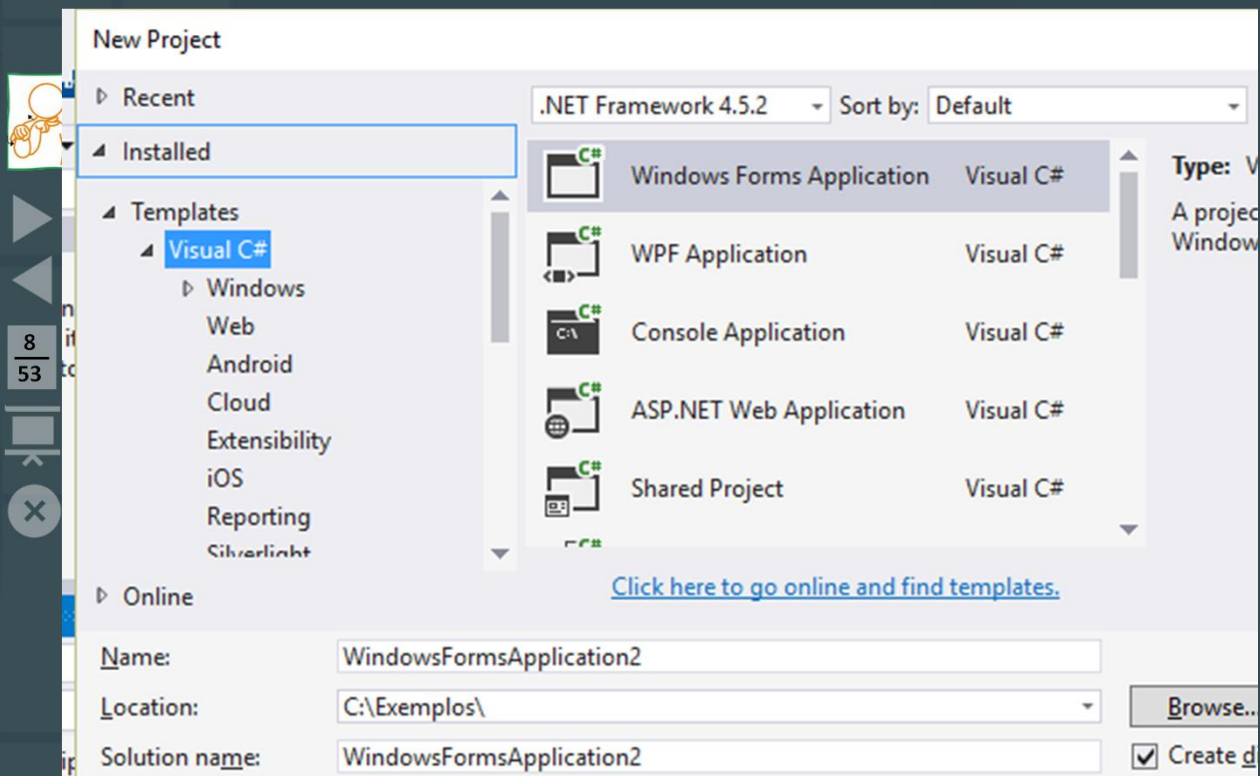
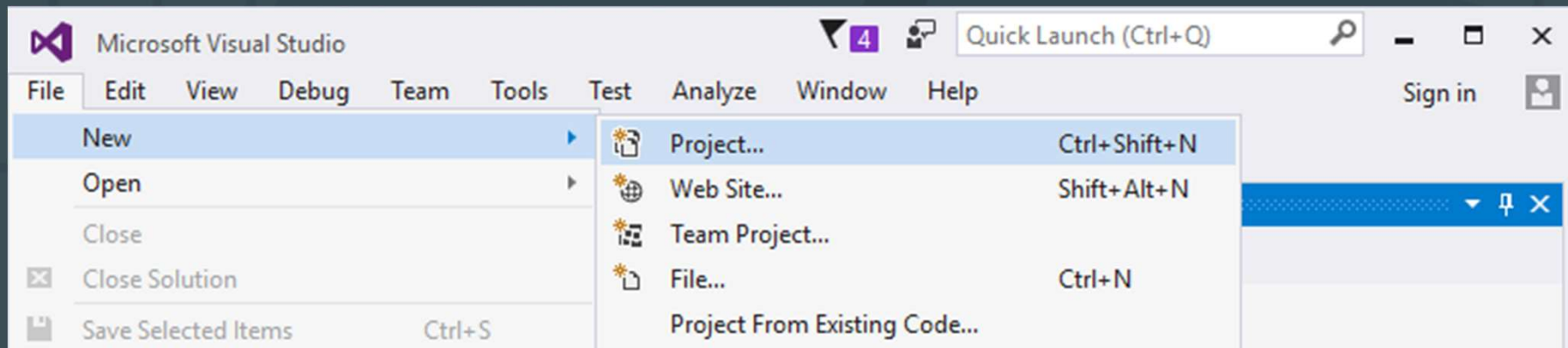


6
53

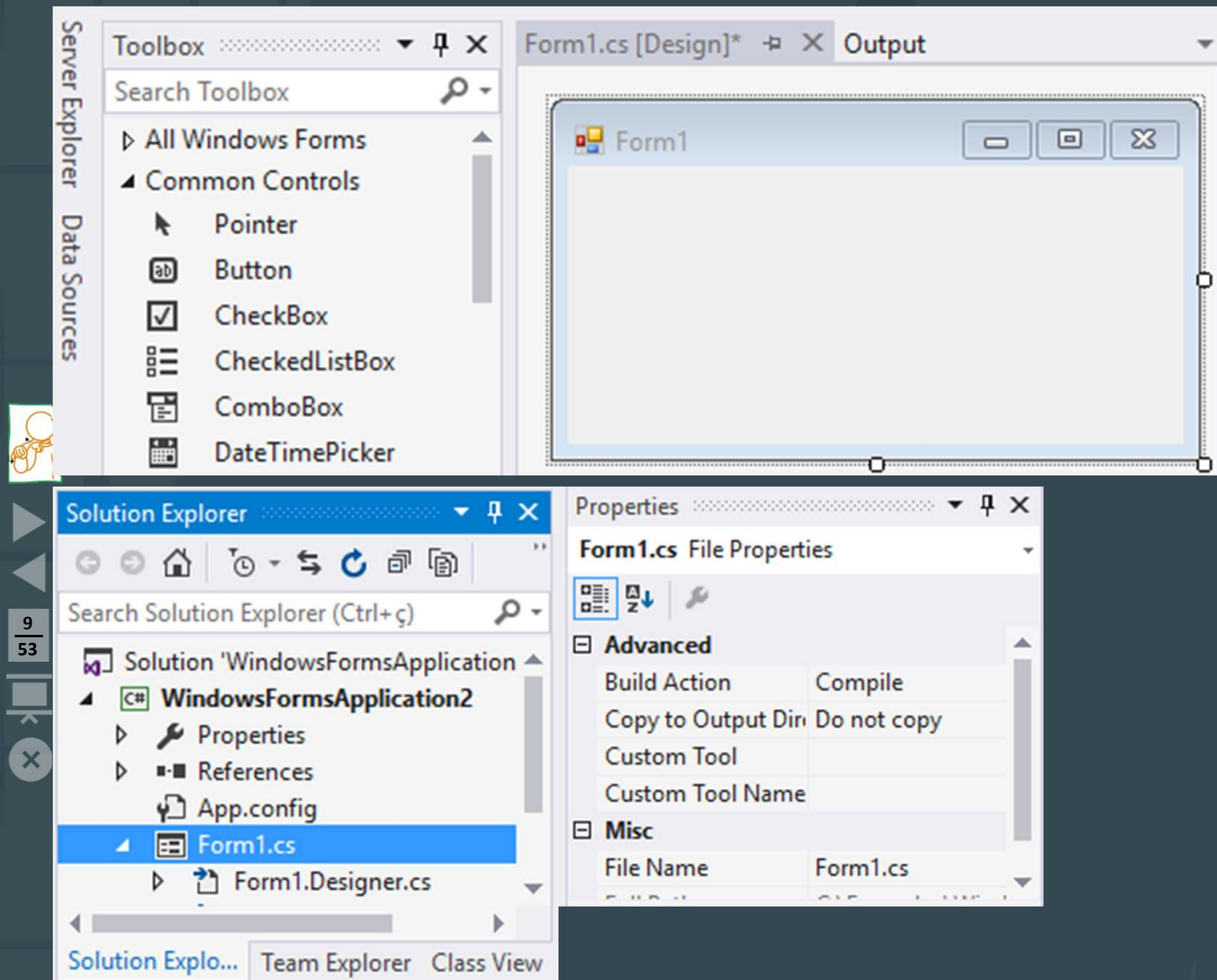


■ IDE Microsoft Visual Studio .NET 2015 versão Community (free)





■ Novo Projeto



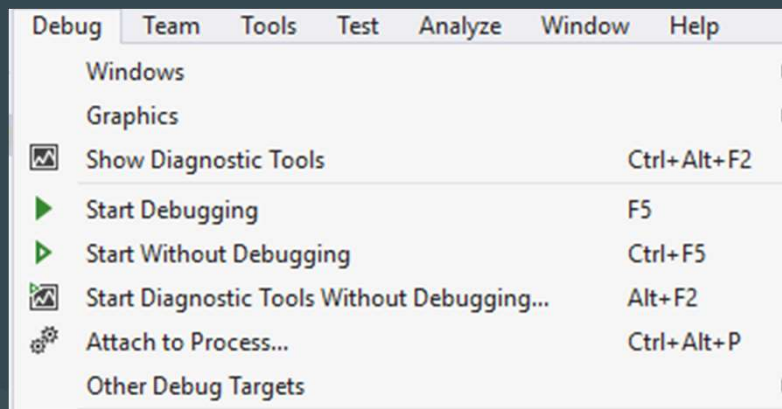
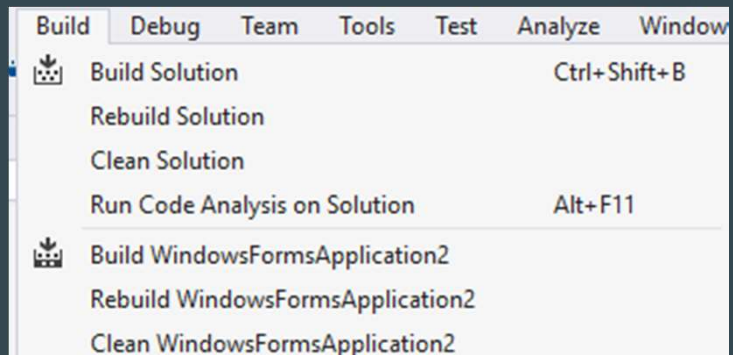
■ Principais recursos

Output

Show output from: Build

```
1>----- Build started: Project: WindowsFormsApplication2, Configuration: Del
1> WindowsFormsApplication2 -> C:\Exemplos\WindowsFormsApplication2\Windowsf
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====
|
```

Output Error List



■ **Compilando e rodando**

Tema 2: Windows Forms

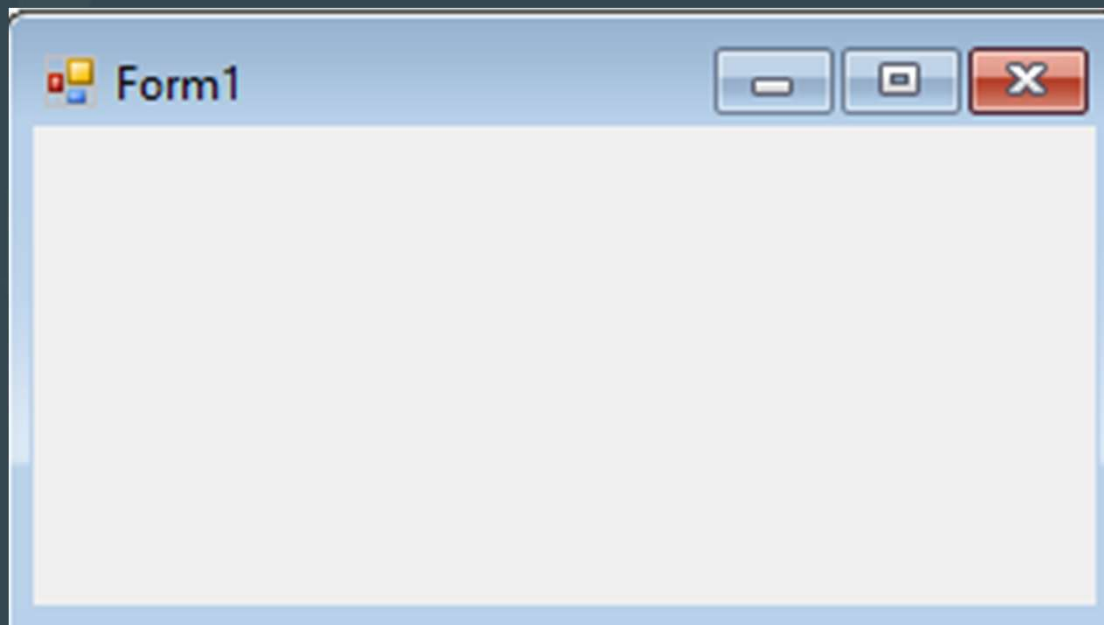
11
30



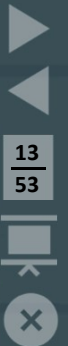
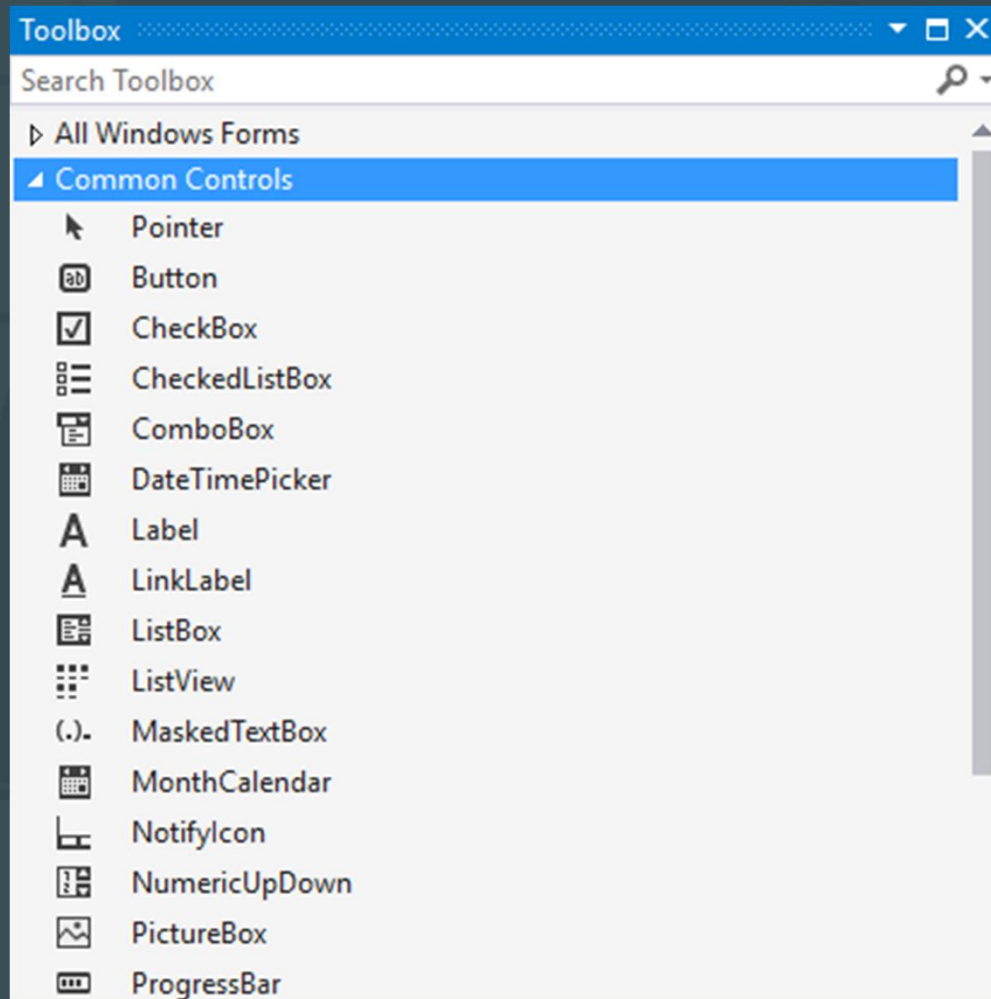
- **Form – formulário: superfície visual que mostra informações;**
- **Orientado a eventos.**



12
53



- **Controles (controls):**
componente que mostra dados
ou aceita entrada de dados.



- **Janelas (form): a janela é o form em si, principal ou secundárias.**



- **Tamanho: pode ser redimensionada, ter uma barra de rolagem lateral e inferior;**

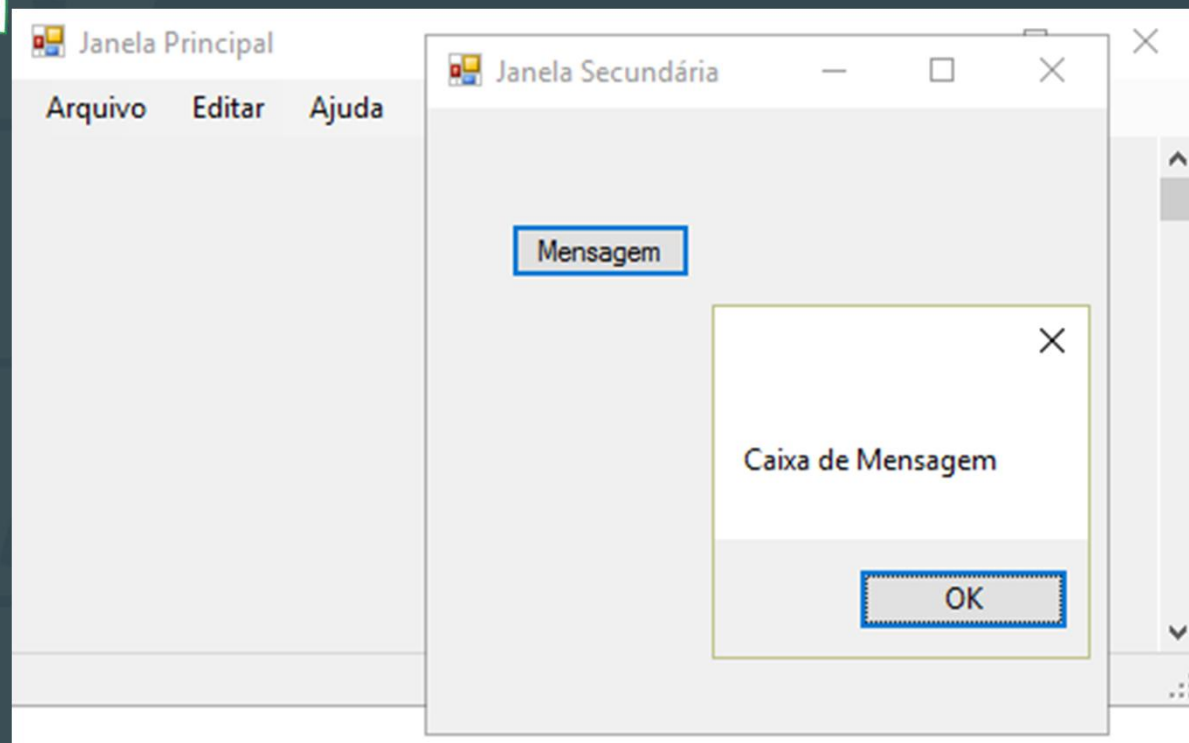
- **Partes:** barra superior de menus e barra inferior de status e dicas;
- **Tipos:** caixa de diálogo, mensagens, pop-up, impressão, salvar, etc.



15
53

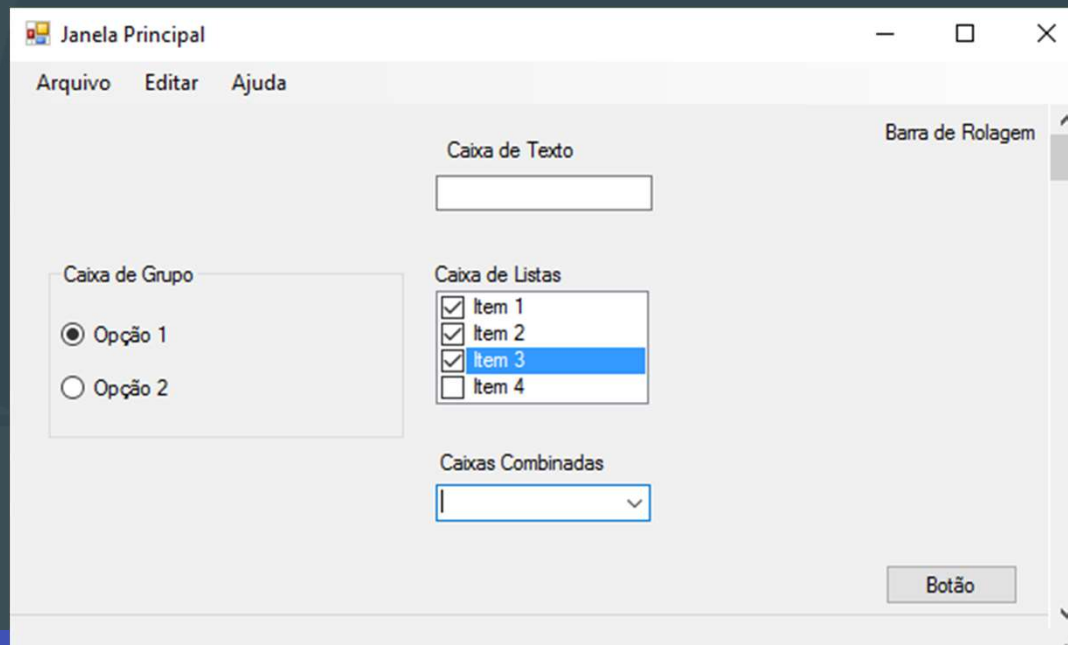


- Exemplo de janela principal, janela secundária e caixa de mensagem.



Componentes

- Exemplo com: menu, barra de status, caixa de texto, caixa de grupo, caixa de listas, caixas combinadas, botão e barra de rolagem.

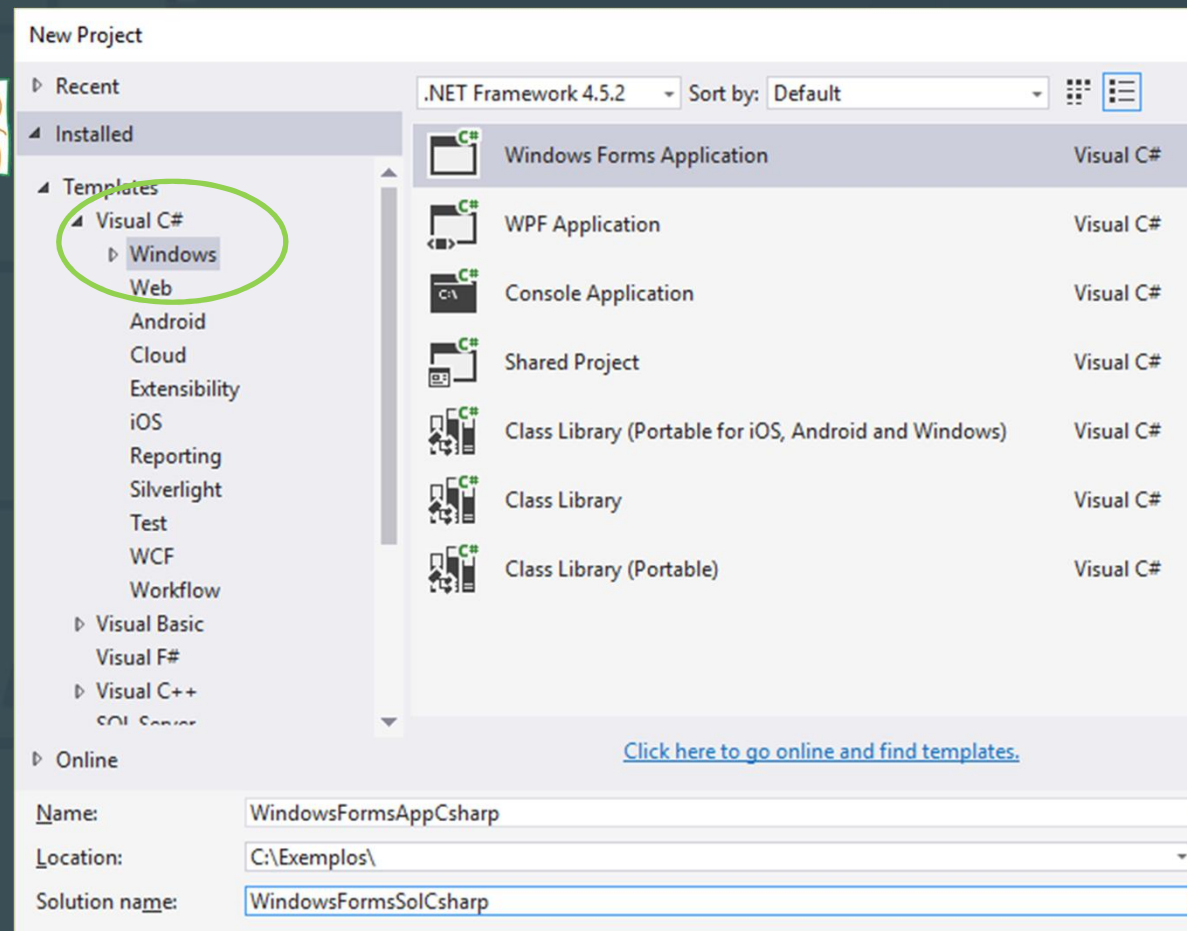


Tema 3: Interface Gráfica com C#

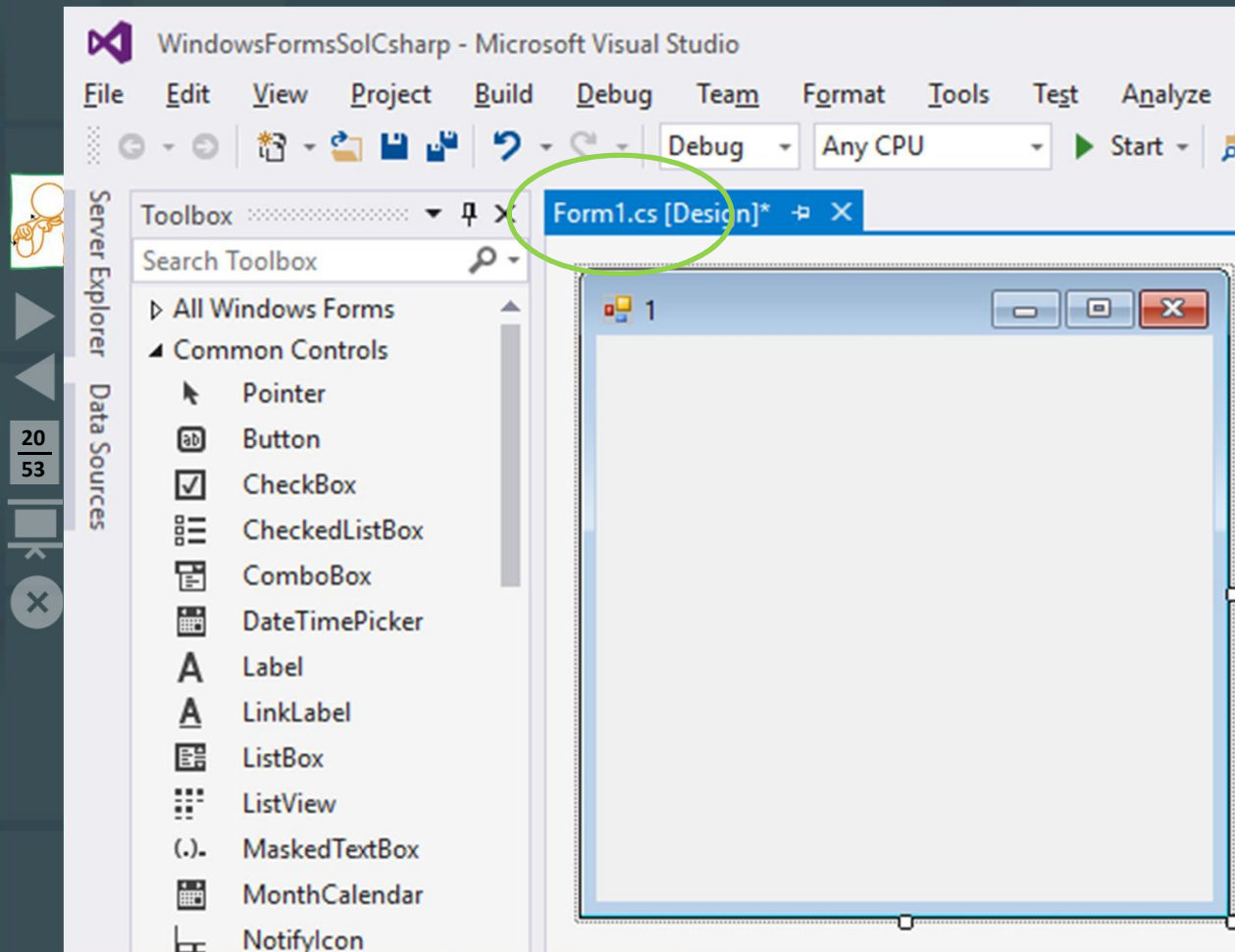
18
30



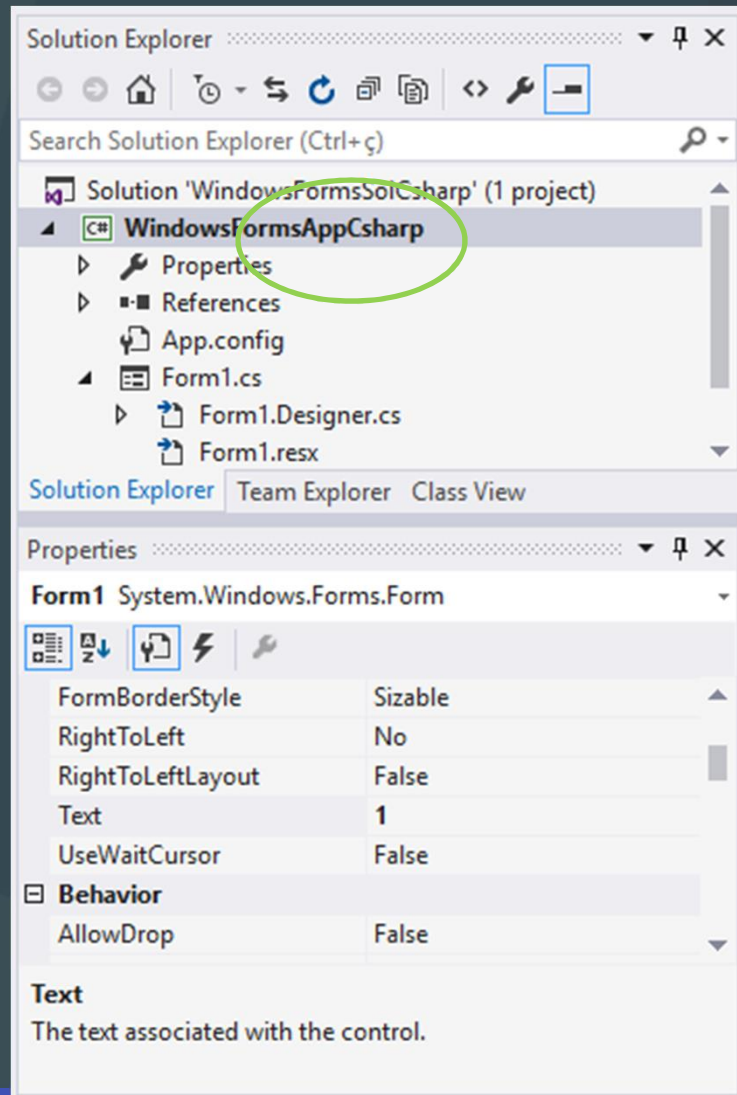
■ Vejamos os passos básicos para desenvolver uma aplicação Windows Form com C#:



- **Ambiente de Desenvolvimento**
 - **Toolbox e Form.**



- Ambiente de Desenvolvimento
- Solution Explorer e Properties.

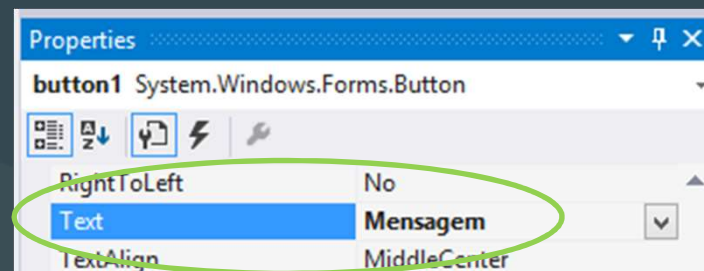
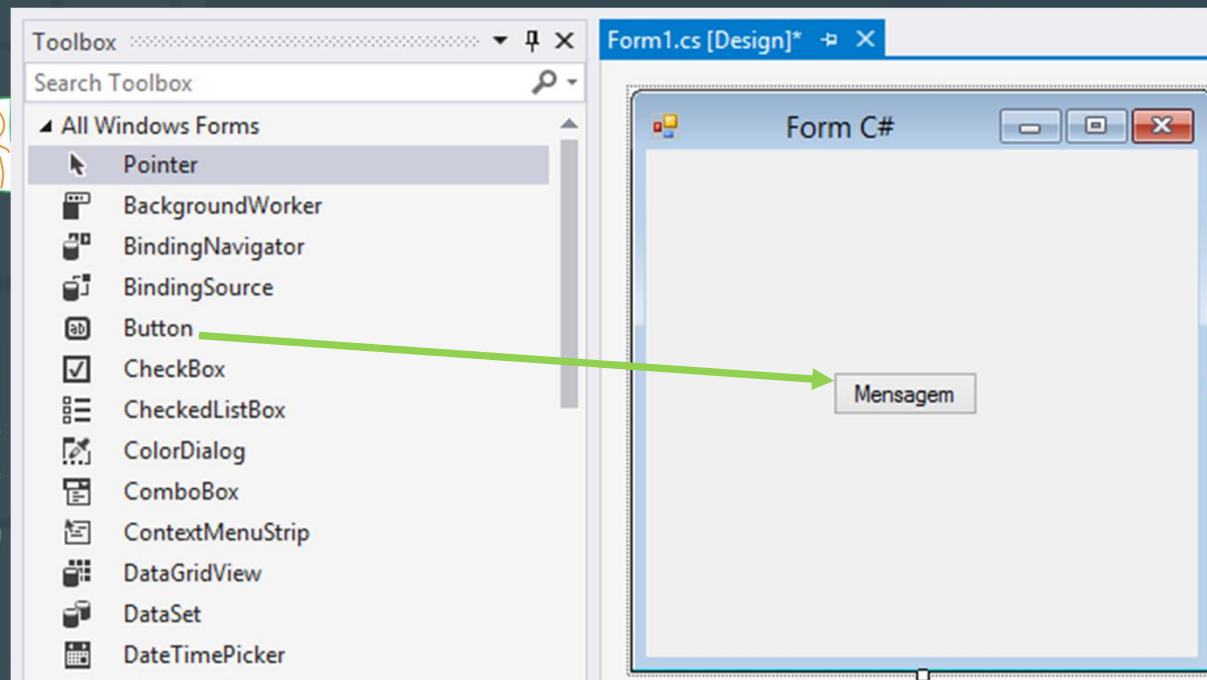


21
53



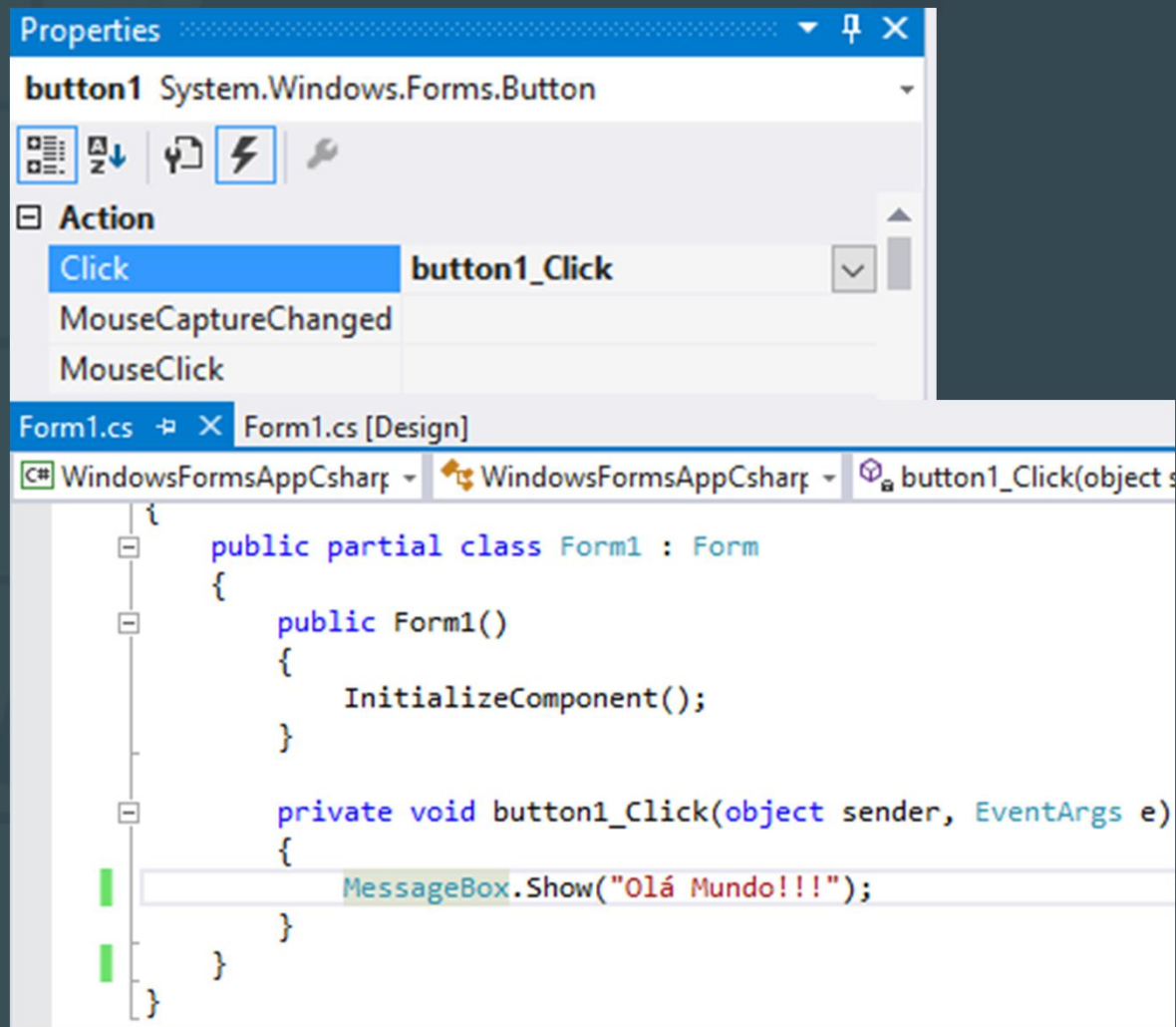
■ Aplicação

- Inserir um botão e alterar a Propriedade Text para "Mensagem".



- **Aplicação**

- **Editar o evento click do botão para mostrar a mensagem "Olá Mundo!!!".**

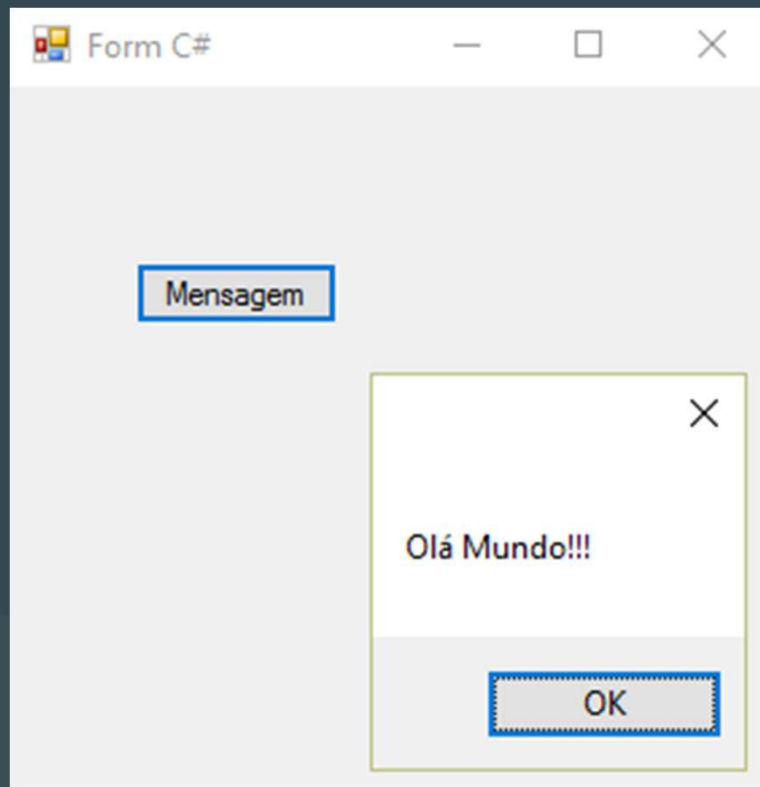


■ Aplicação

- Rodar a aplicação e clicar no botão Mensagem, fará abrir uma caixa de mensagem com a frase implementada.



24
53

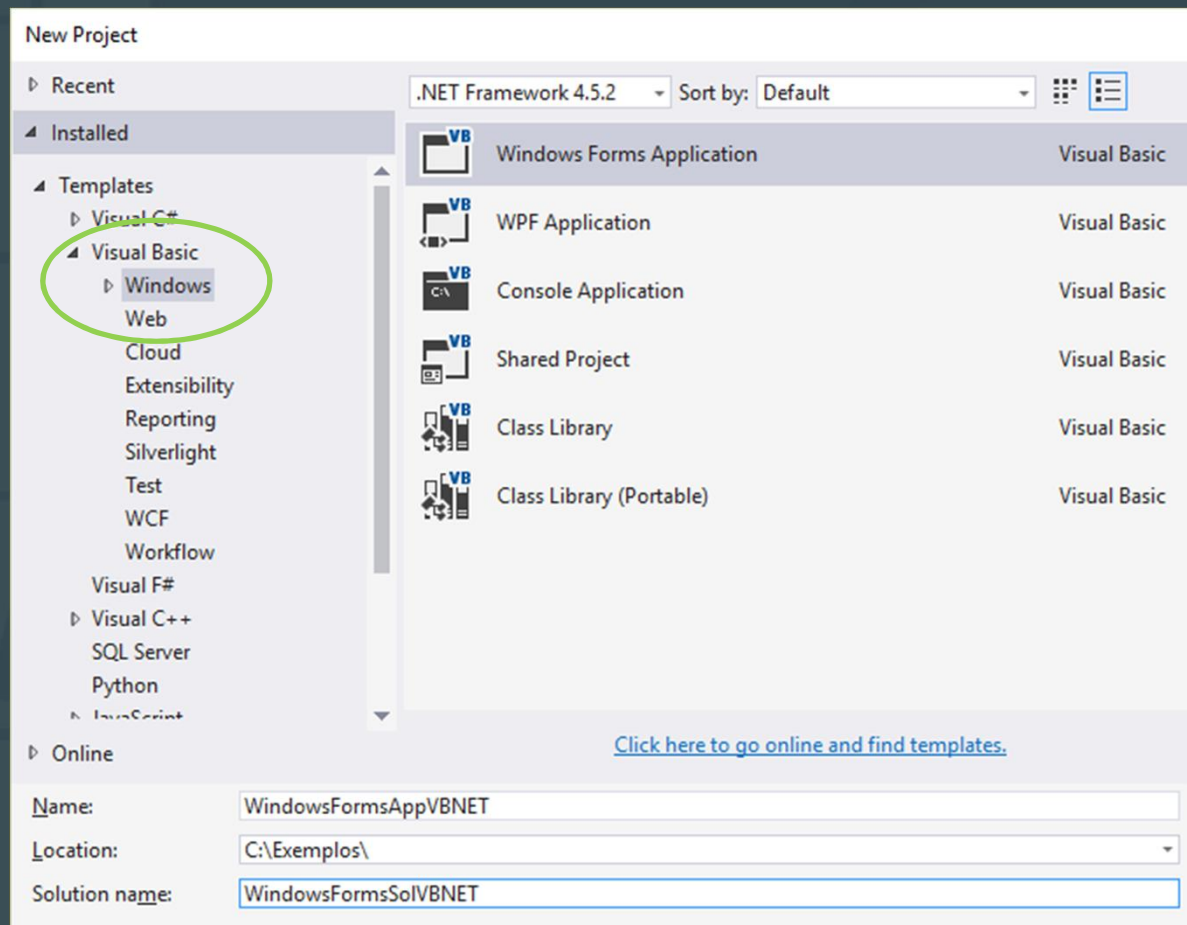


Tema 4: Interface Gráfica com VB.NET

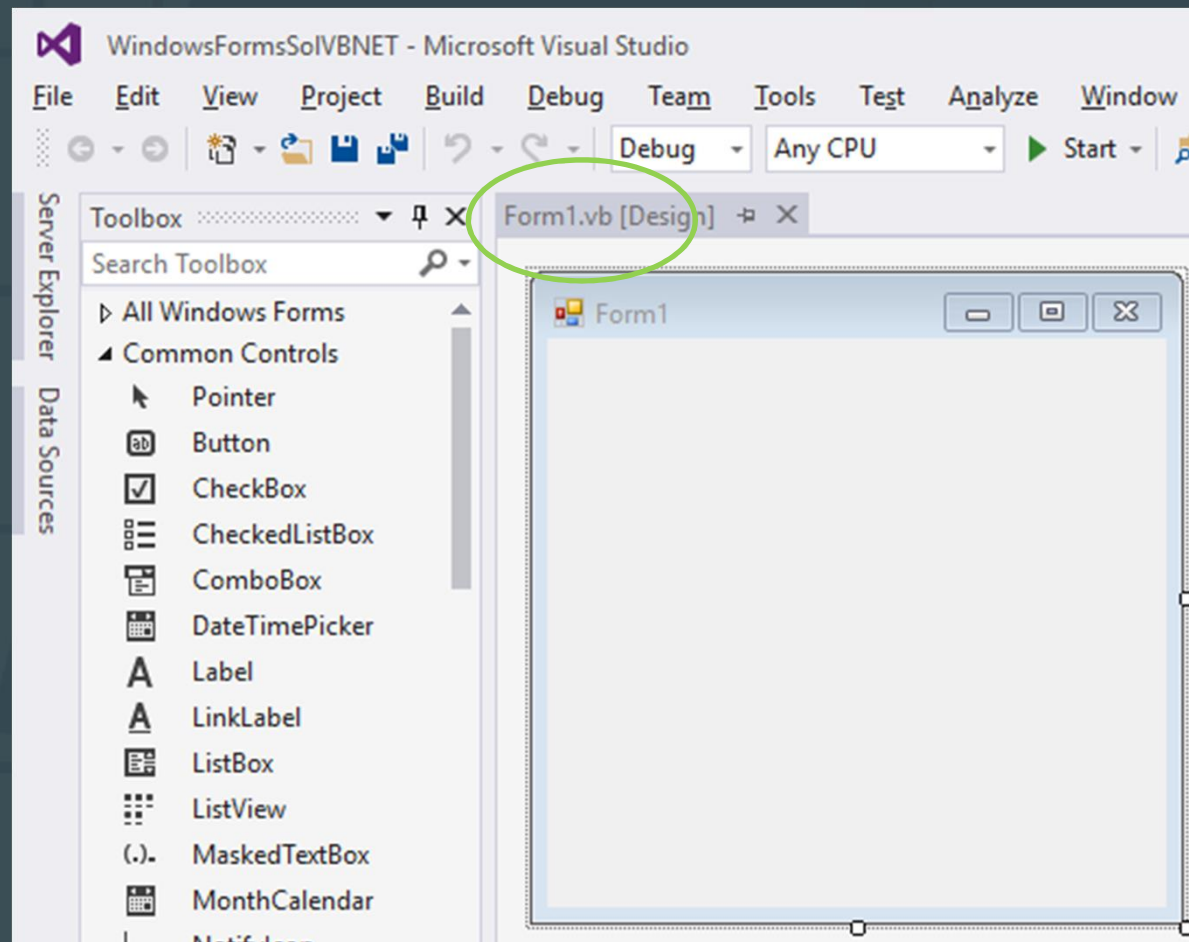
25
30



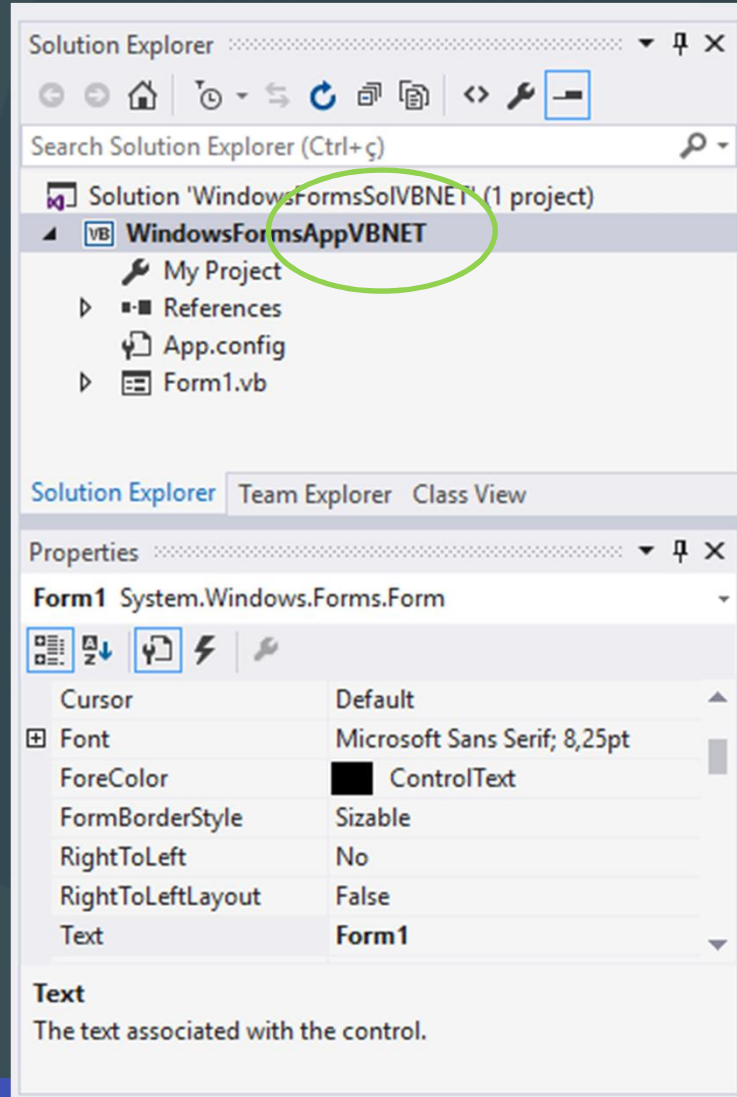
Vamos ver os passos básicos para desenvolver uma aplicação Windows Form com VB.NET.



- **Ambiente de Desenvolvimento**
 - **Toolbox e Form.**

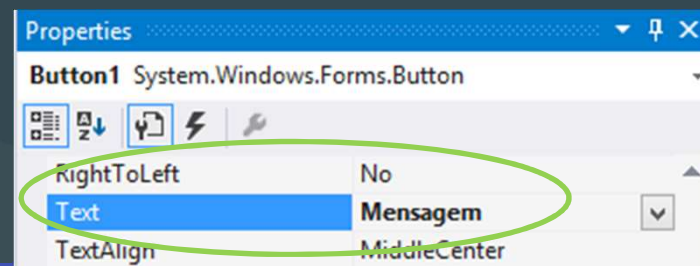
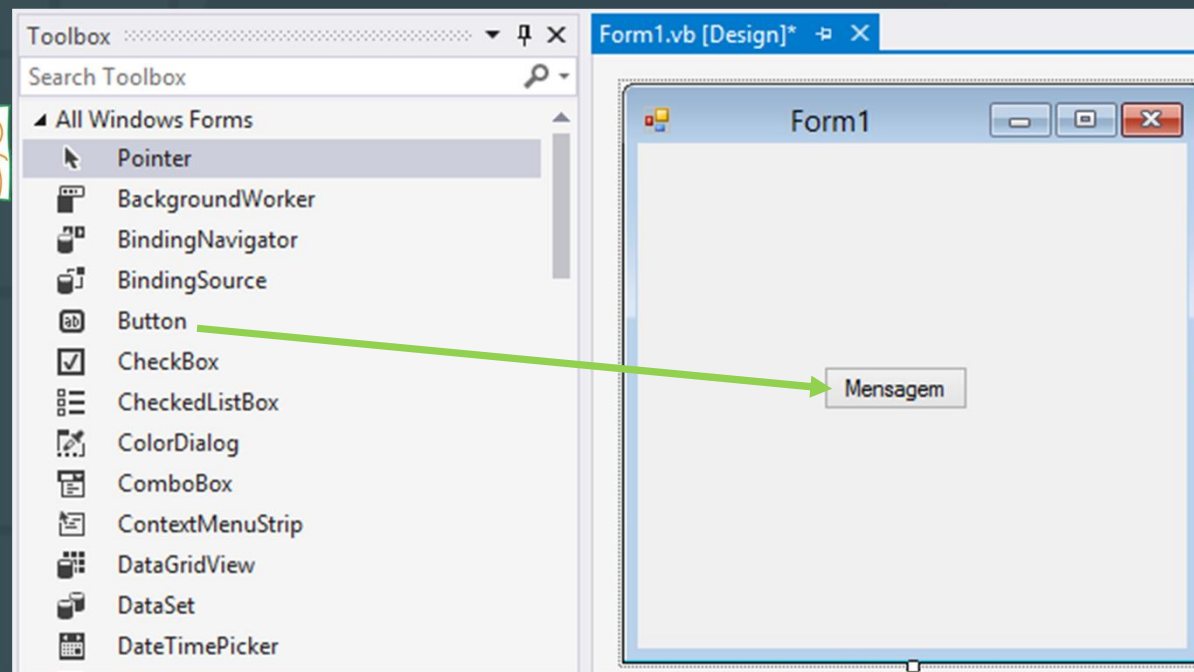


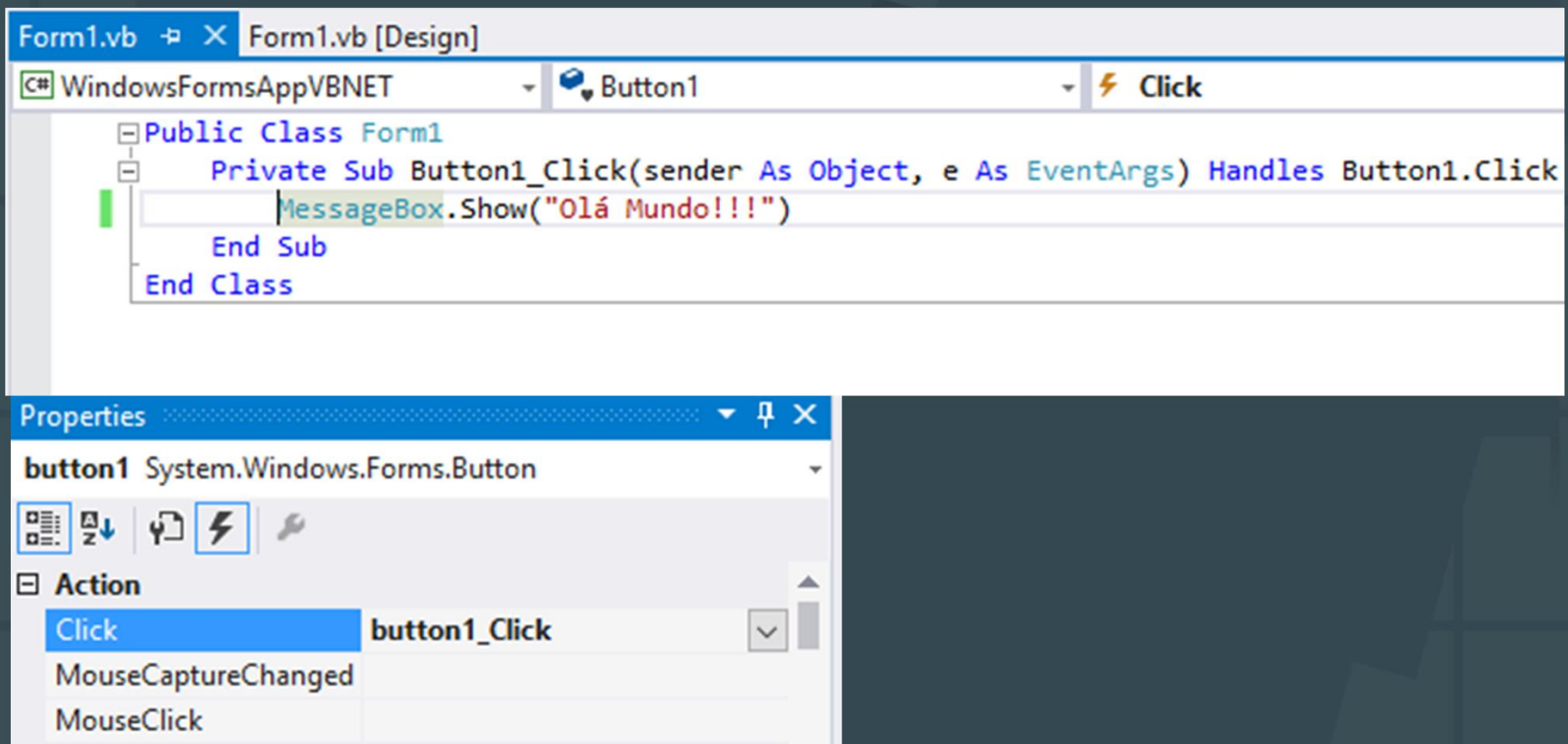
- Ambiente de Desenvolvimento
- Solution Explorer e Properties.



■ Aplicação

- Inserir um botão e alterar a Propriedade Text para "Mensagem".



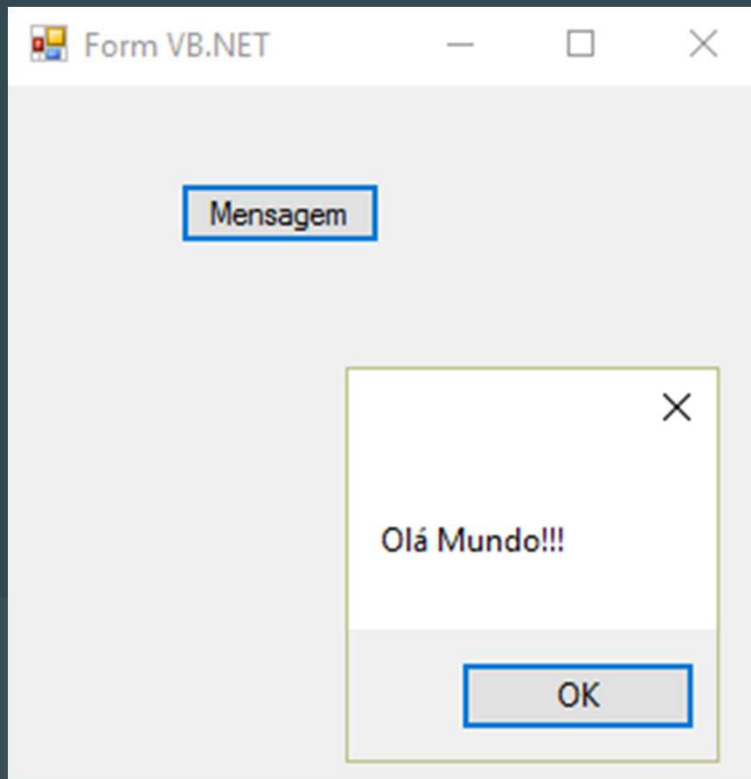


■ Aplicação

- Editar o evento click do botão para mostrar a mensagem "Olá Mundo!!!".

- **Aplicação**

- **Rodar a aplicação e clicar no botão Mensagem, irá abrir uma caixa de mensagem com a frase implementada.**



Tema 5:

Desenvolvimento de aplicações gráficas

32
30



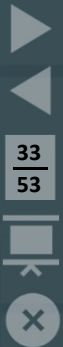
Visualizador de Imagens

- Vamos fazer um exemplo completo de programação visual com

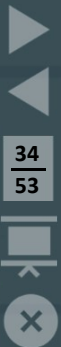


Windows Forms:

- Aplica-se tanto para C# como para VB.NET;
- O projeto base é C#;
- Criar um projeto chamado Picture Viewer.



- Definir as propriedades do form:
 - clicar no form;
 - ir à janela Properties;
 - propriedade Appearance Text: "Picture Viewer";
 - propriedade Layout Size: 550 x 350.

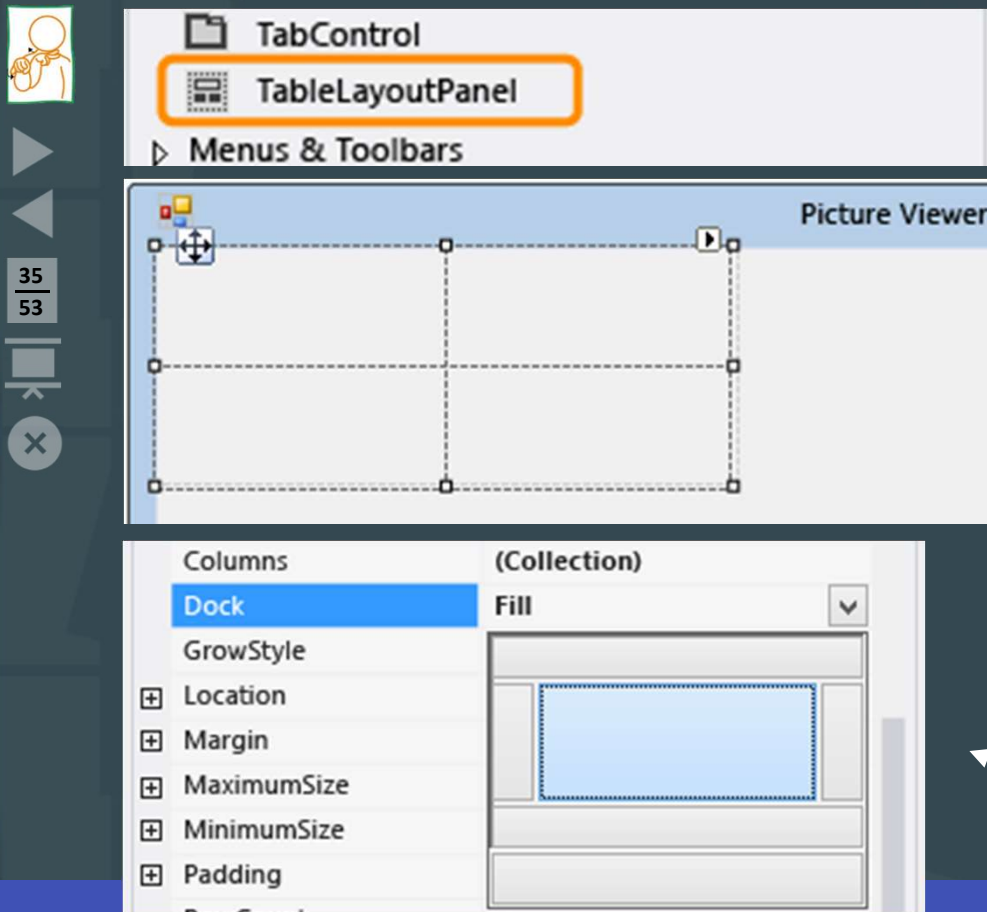


The screenshot shows the 'Properties' window in Visual Studio. The title bar says 'Properties'. Below it, the selected control is 'Form1 System.Windows.Forms.Form'. There are icons for 'Properties', 'Events', 'Resources', 'Appearance', and 'Layout'. The 'Appearance' icon is selected. The list of properties includes:

SizeGripStyle	Auto
StartPosition	WindowsDefaultLocation
Tag	
Text	Picture Viewer
TopMost	False
TransparencyKey	<input type="checkbox"/>
UseWaitCursor	False
WindowState	Normal

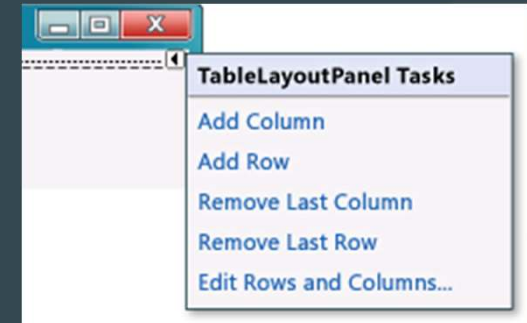
Below the list, there is a section titled 'Text' with the description 'The text associated with the control.'

- Organizar o layout do form:
 - adicionar o componente **TableLayoutPanel**;
 - propriedade **Layout Dock: Fill**.



■ Configurando o Layout do Painel :

- clique no componente **TableLayoutPanel**;
- no canto superior direito há uma seta para a direita, clique nela;
- no menu de opções, clique em **Edit Rows and Columns**;



- clique em **Show Columns** e deixe a **coluna 1** com **15%** e a **coluna 2** com **85%**;
- clique em **Show Rows** e deixe a **row 1** com **90%** e a **row 2** com **10%**.



37
53

Column and Row Styles

Show: Columns

Member	Size Type	Value
Column1	Percent	15.00 %
Column2	Percent	85.00 %

Add Delete Insert

Size Type

☐ Absolute 20 pixels

☒ Percent 15.00 %

☐ AutoSize

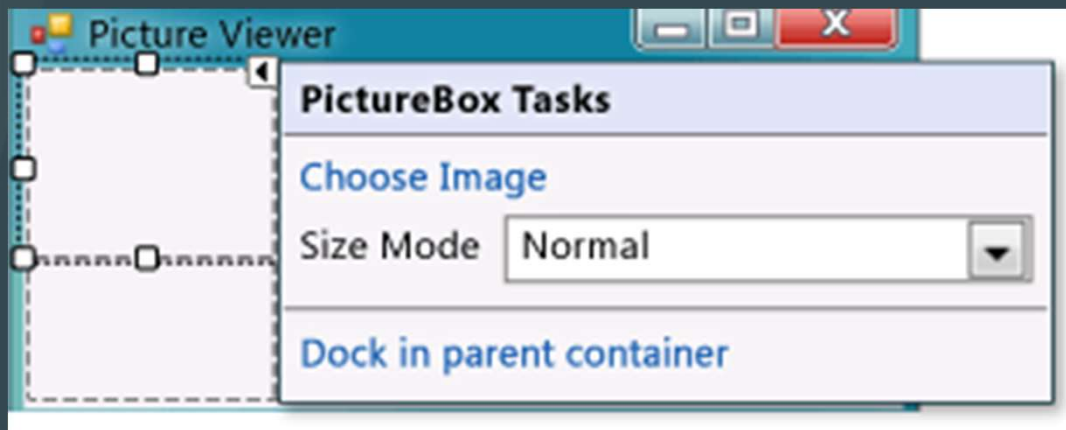
i Column and row spanning:
If you want a control to [span multiple rows or columns](#), set the RowSpan and ColumnSpan properties on the control.

OK Cancel

- **Inserindo um componente de imagem:**
 - **puxar o componente PictureBox sobre a parte superior do Layout Panel;**



38
53









- na seta no canto superior direito, clique em “Dock in parental container” para ocupar a célula;
- Para que ocupe também a célula ao lado, vá à janela Properties e deixe o item **ComunSpan** com valor 2.

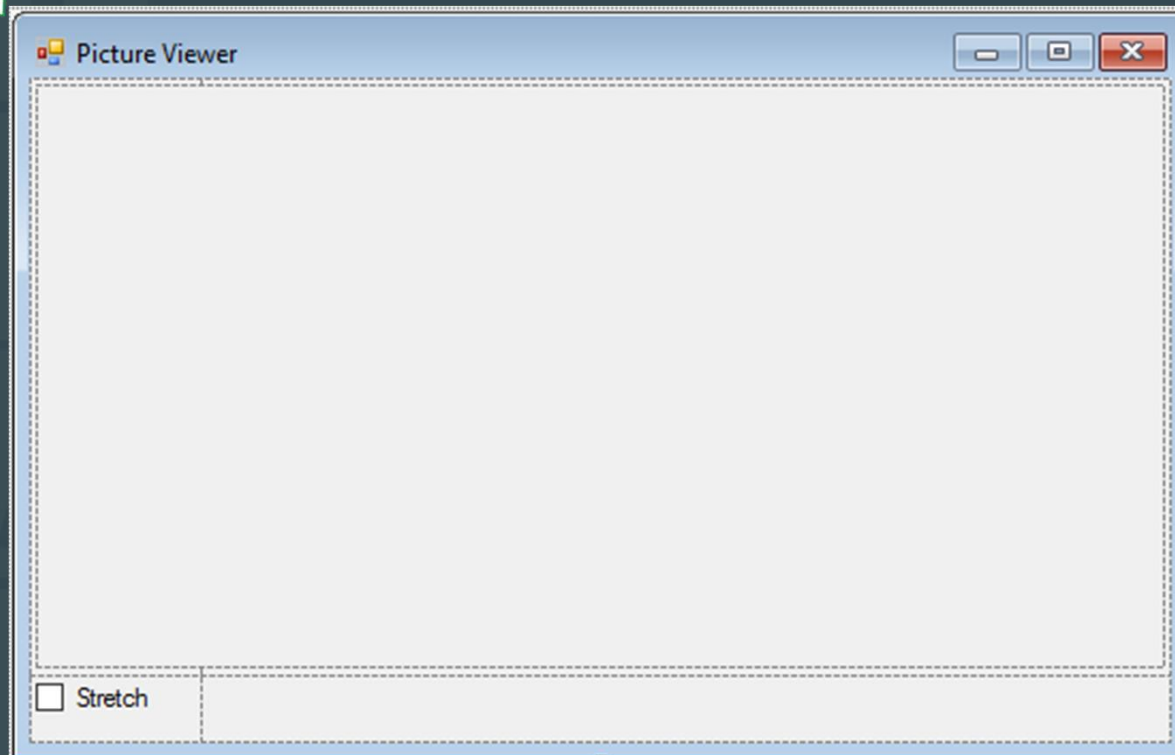
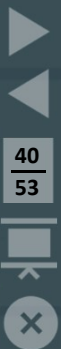


39
53



Properties		
PictureBox1 System.Windows.Forms.PictureBox		
    		
	Cell	0,0
	Column	0
	ColumnSpan	2
	Dock	Fill

- **Adicionando uma caixa de seleção:**
 - da janela Toolbox, puxe o item **CheckBox** para o form;
 - vá para a janela **Properties** e deixe **Text** com **Stretch**.



- **Adicionando botões:**

- **antes, para organizá-los, adicionaremos o componente `FlowLayoutPanel`;**
- **configure a sua propriedade `Dock` para `Fill`, e `FlowDirection` para `RightToLeft`;**



41
53



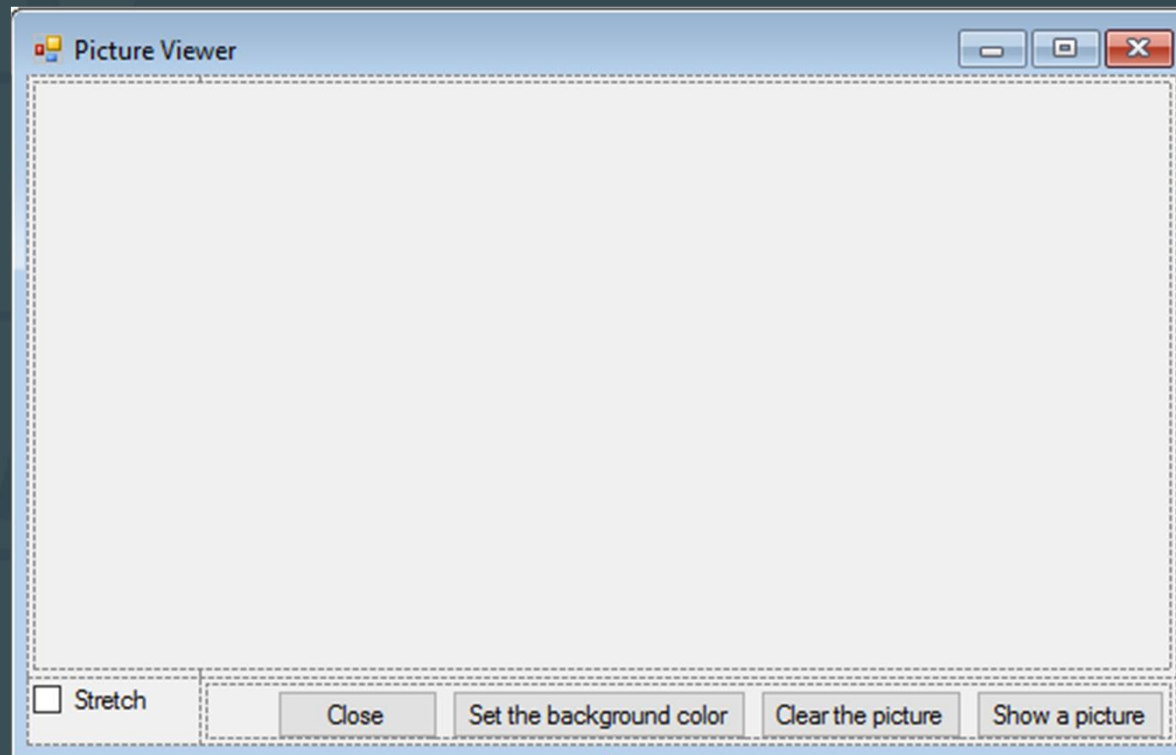
- vá para a janela Toolbox e puxe 4 botões para o form sobre o FlowLayoutPanel. Defina a propriedade AutoSize deles como True;
- na propriedade Text de cada botão, escreva suas funções (do botão 1 ao 4: Close, Set the background color, Clear the Picture e Show a picture).



42
53



- **Dar nome aos botões:**
 - para cada botão, selecione a sua propriedade Name, deixando-os assim: closeButton, backButton, clearButton e showButton.



43
53



■ Adicionando Caixas de Diálogo:

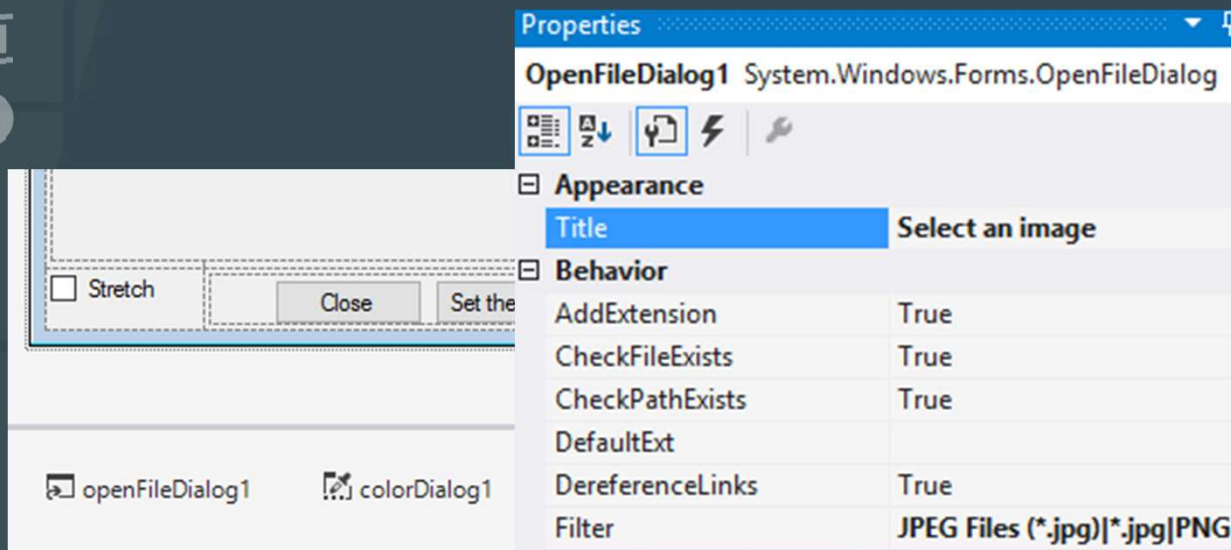
- vá para Toolbox, e no grupo Dialog dê um duplo clique em OpenFileDialog para adicioná-lo ao form;
- faça o mesmo para o componente ColorDialog;



44
53



- clique sobre o OpenFileDialog adicionado, e vá à janela Properties e no item Filter digite: JPEG Files (*.jpg)|*.jpg|PNG Files (*.png)|*.png|BMP Files (*.bmp)|*.bmp|All files (*.*)|*.*
- no item Title escreva: Select an image.



- **Adicionando Código (Eventos):**
 - **para abrir um arquivo de imagem: dê um duplo clique no botão Show a picture, e então digite o código abaixo em C#:**


```
if (openFileDialog1.ShowDialog() == DialogResult.OK)
{
    pictureBox1.Load(
        openFileDialog1.FileName);
}
```

- **caso a aplicação seja em VB.NET, digite o código abaixo em VB.NET:**

```
If OpenFileDialog1.ShowDialog() =
DialogResult.OK Then
    PictureBox1.Load(
    OpenFileDialog1.FileName)
End If
```

Adicionando Código (Eventos):

- para configurar a cor de fundo: dê um duplo clique no botão Set the background color, e então digite o código abaixo em C#:



```
if (colorDialog1.ShowDialog() ==  
DialogResult.OK)  
    pictureBox1.BackColor =  
    colorDialog1.Color;
```

- 
- caso a aplicação seja em VB.NET, digite o código abaixo:

```
If ColorDialog1.ShowDialog() =  
    DialogResult.OK Then  
    PictureBox1.BackColor =  
        ColorDialog1.Color  
  
End If
```

■ Adicionando Código (Eventos):

- alongar a imagem: vá no evento **CheckedChanged** do componente **CheckBox**, dê um duplo clique, e então digite o código abaixo em C#:

```
if (checkBox1.Checked)
    pictureBox1.SizeMode =
PictureBoxSizeMode.StretchImage;
else
    pictureBox1.SizeMode =
PictureBoxSizeMode.Normal;
```



48
53



- caso a aplicação seja em **VB.NET**, digite o código abaixo:

```
If CheckBox1.Checked Then
    PictureBox1.SizeMode =
PictureBoxSizeMode.StretchImage
Else
    PictureBox1.SizeMode =
PictureBoxSizeMode.Normal
End If
```


- **Adicionando Código (Eventos):**
 - para Limpar a imagem e fechar: dê um duplo clique no botão Clear the picture, então digite o código abaixo em C#:

pictureBox1.Image = null;



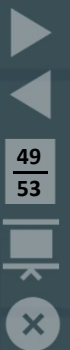
- Caso a aplicação seja em VB.NET, digite o código abaixo:

PictureBox1.Image = Nothing

- Para fechar a aplicação, dê um duplo clique no botão Close, então digite :

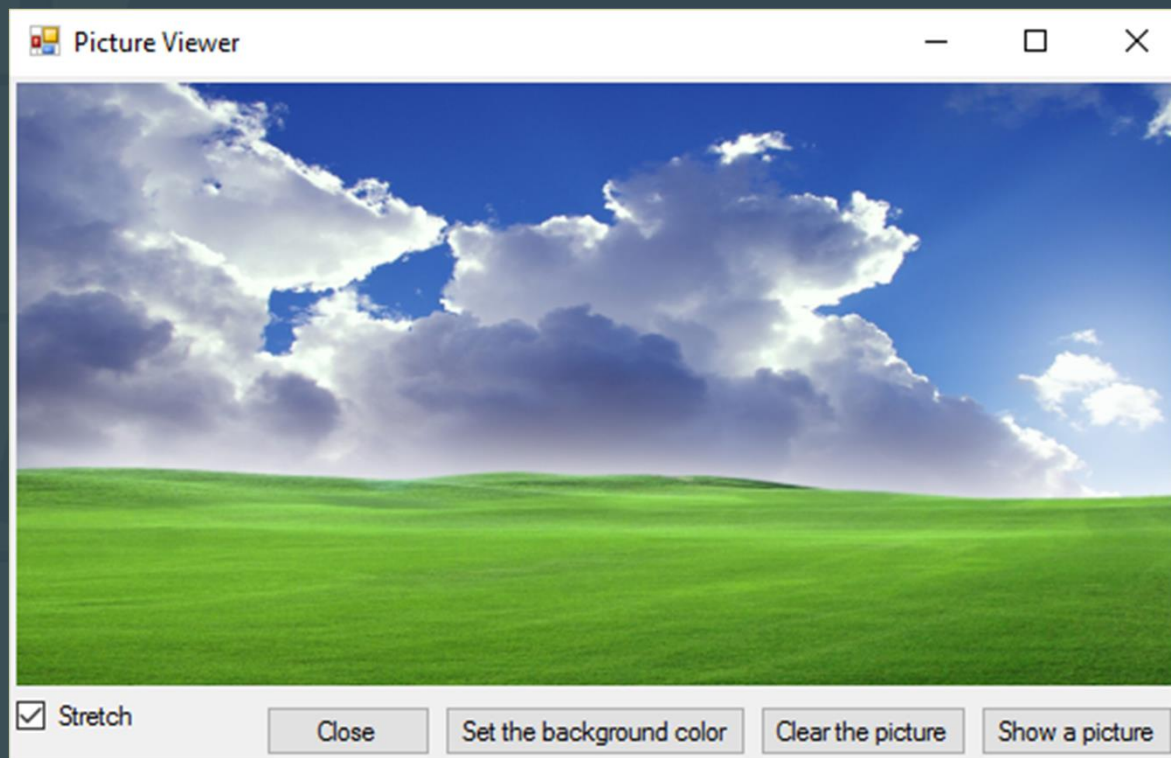
C#: this.Close();

VB.NET: Close()



Aplicação concluída!!!

- agora é só clicar na tecla F5 para compilar e rodar a aplicação;
- teste-a clicando no botão Show a picture, escolha uma imagem para abrir e deixe marcada a opção Stretch..



Síntese

51
30



■ **Nesta primeira aula vimos:**

- **Programação gráfica com Visual Studio;**
- **Windows Forms;**
- **Interface gráfica C#;**
- **Interface Gráfica VB;**
- **Desenvolvimento de uma aplicação.**



52
53

