

Tópicos desta Aula

1. Introdução a IDE do Visual C# Express	2
2. Opções da Janela Solution Explorer.....	4
3. Definindo o formulário inicial	8
4. Toolbox	10
5. Usando a Janela Propriedades	13
6. Usando o IntelliSense	14
7. Avaliação da aula	17

Introdução a IDE do Visual C# Express

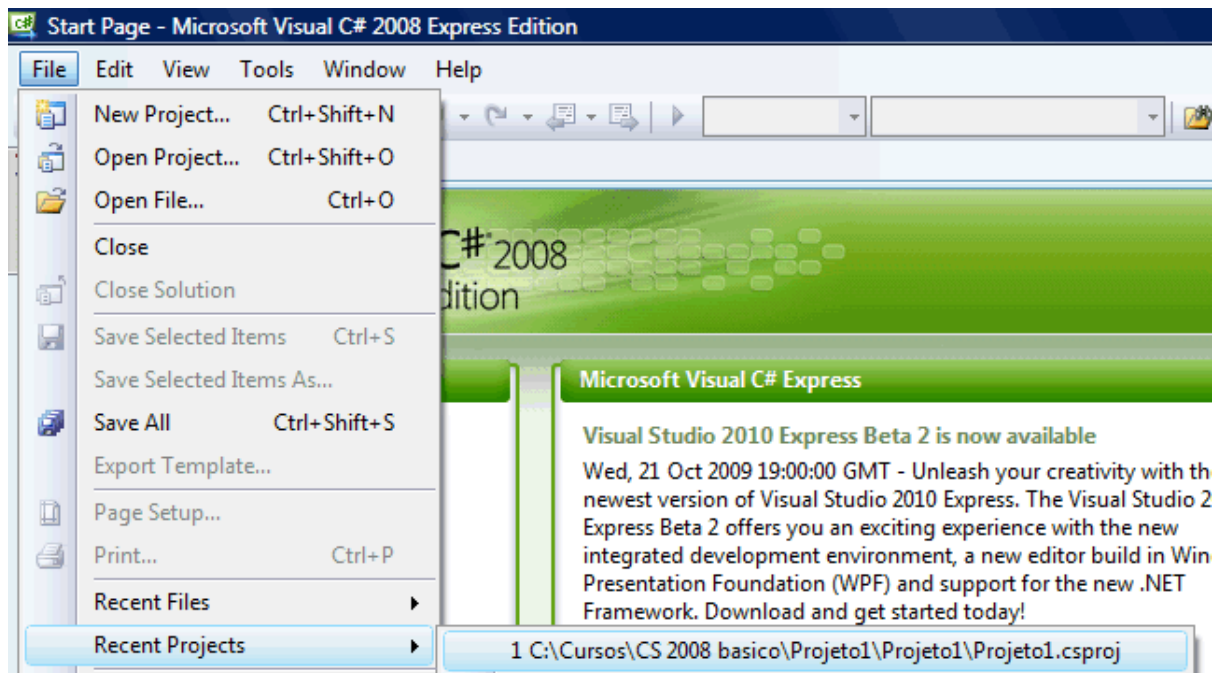
O ambiente de desenvolvimento integrado (IDE) do Visual C# Express torna fácil para você criar programas para o Windows.

O IDE contém várias ferramentas que ajudam você a projetar seus aplicativos e escrever, editar e depurar o código do aplicativo.

Para visualizar melhor a IDE abra seu projeto da aula 1

Dica

Utilize o menu File -> Recent Projects



Usando o Solution Explorer

O Solution Explorer é uma área do ambiente de desenvolvimento integrado (IDE) que contém a solução e o seu projeto e ajuda você a gerenciar seus arquivos de projeto.

Observação

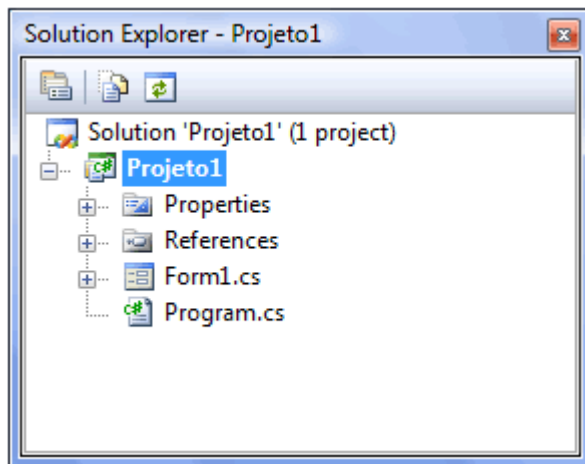
Uma solução pode ser composta por um ou mais projetos.

Os arquivos são exibidos em uma exibição hierárquica, parecido com o Windows Explorer.

Por padrão, o Solution Explorer está localizado no lado direito do IDE.

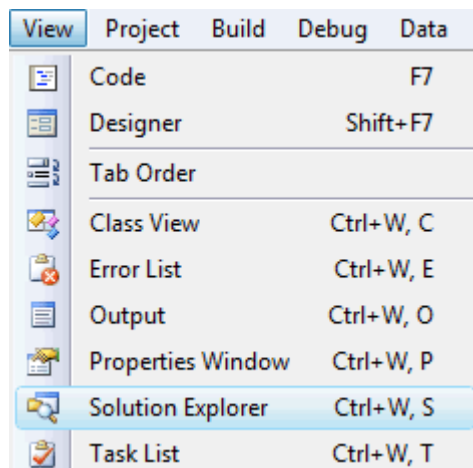
Quando você cria um aplicativo do tipo Windows Forms a solução contém o seu projeto (Projeto1) o qual contém alguns nós: Properties (propriedades), References (referências a algumas bibliotecas) e Form1.cs e o arquivo Program.cs como mostra a figura a seguir.

Visual C# 2008 - Módulo Básico
Aula 02



Dica

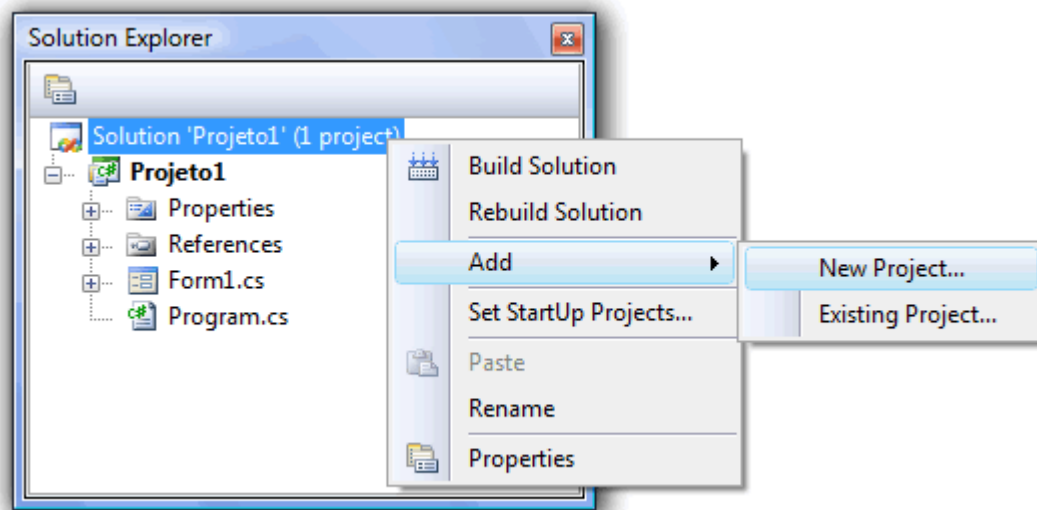
Se o Solution Explorer não estiver visível, você pode clicar no menu View e, em seguida, clicar em Solution Explorer para abri-lo.



Opções da Janela Solution Explorer

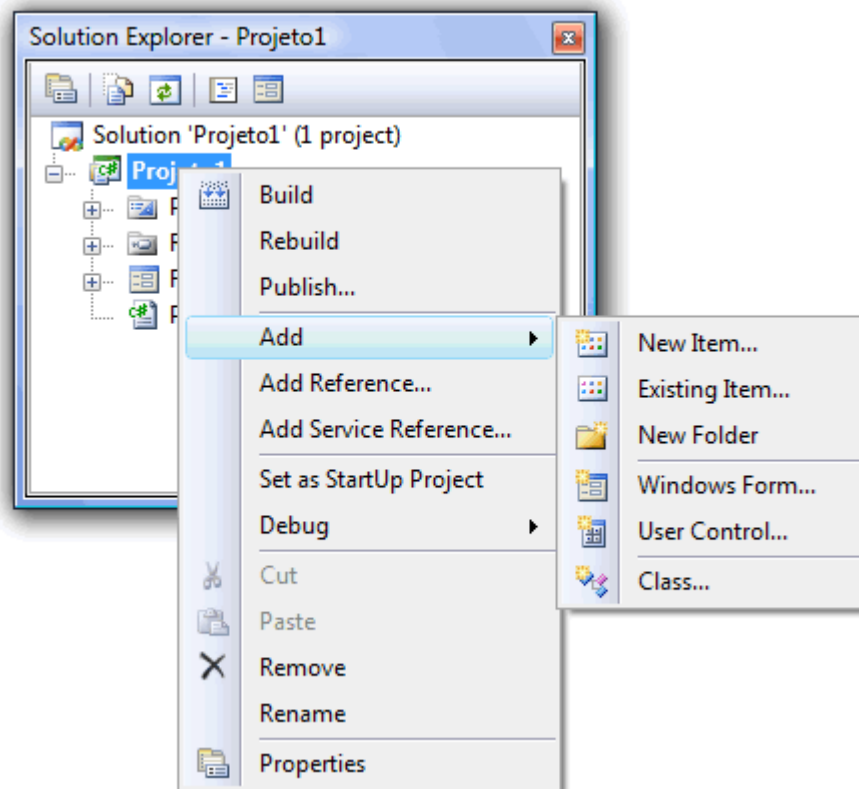
Nesta janela podemos executar muitas operações relativas a solução e ao projeto.

Para executar alguma operação relativa a solução clique com o botão direito do mouse sobre o nome da solução.



Observe que há varias opções para compilar e recompilar a solução, adicionar novos projetos ou projetos já existentes, definir o projeto inicial, além de poder renomear e configurar as propriedades do mesmo.

Para executar alguma operação relativa ao projeto clique com o botão direito do mouse sobre o nome do projeto.



Observe que o menu é bem diferente e apresenta opções exclusivas do projeto, inclusive é aqui que você pode adicionar novos itens ao projeto.

Adicionando um novo item ao projeto

Você pode adicionar itens, como outro Windows Form ou classes, ao seu projeto.

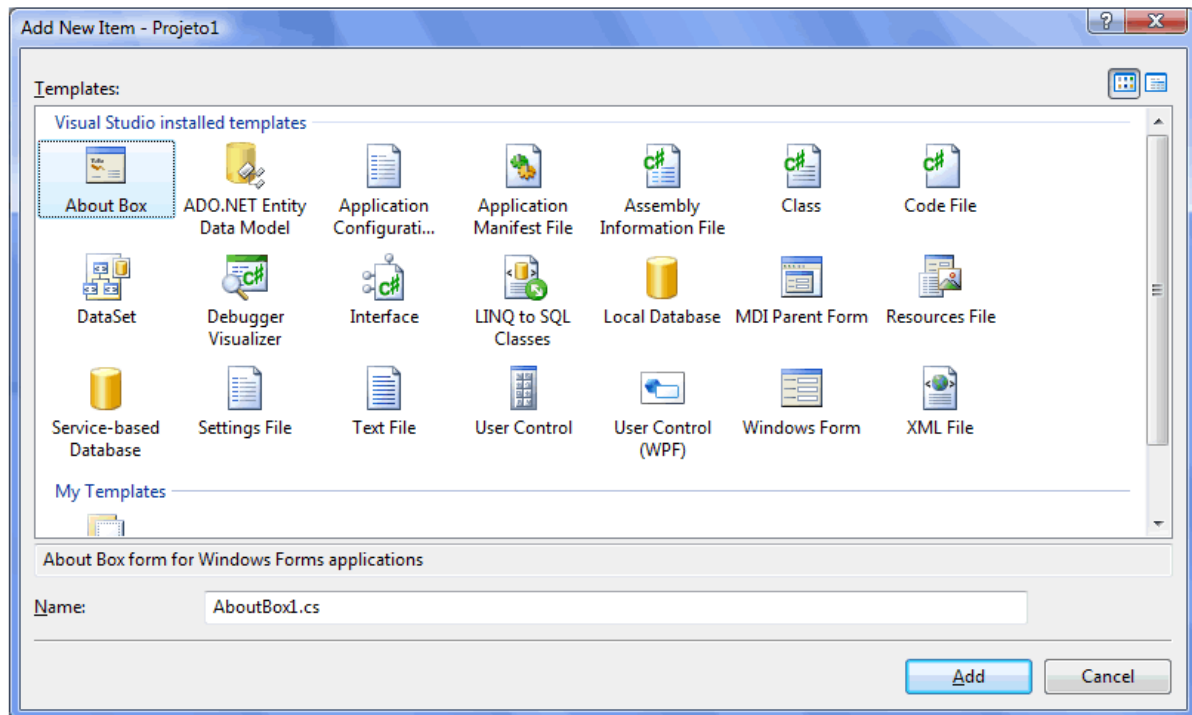
No Solution Explorer, clique com o botão direito do mouse no nome do projeto (Projeto1) , selecione **Add**.

Em seguida, clique em **New Item**.

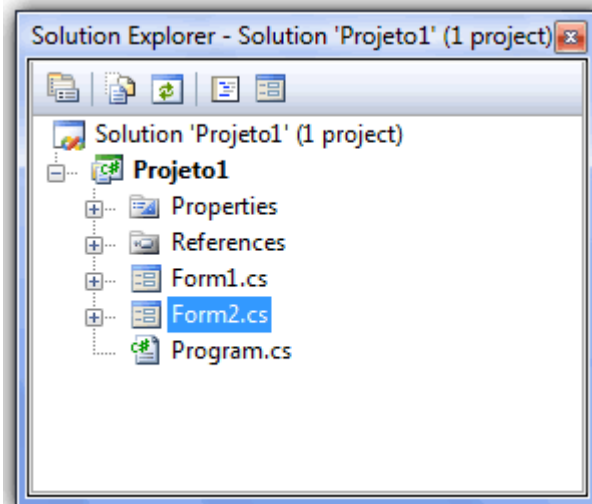
A janela **Add New Item** é exibida.

Visual C# 2008 - Módulo Básico

Aula 02



Para adicionar mais um formulário ao seu projeto selecione o template **Windows Form** e em seguida, clique em **Add**.



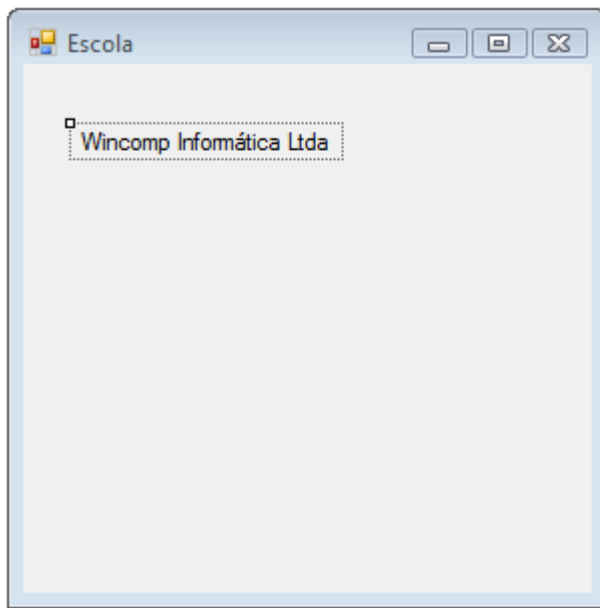
Um novo formulário, denominado **Form2.cs** é adicionado ao seu projeto.

Para editar este formulário basta dar um duplo clique sobre o Form2.cs dentro do Solution Explorer.

Faça as seguintes configurações

- 1- Configure a propriedade **Name** do Form2 para **frmEscola** e a propriedade **Text** para Escola
- 2- Adicione um Label e configure a propriedade **Name** para **lblEscola** e a propriedade **Text** para Wincomp Informática Ltda

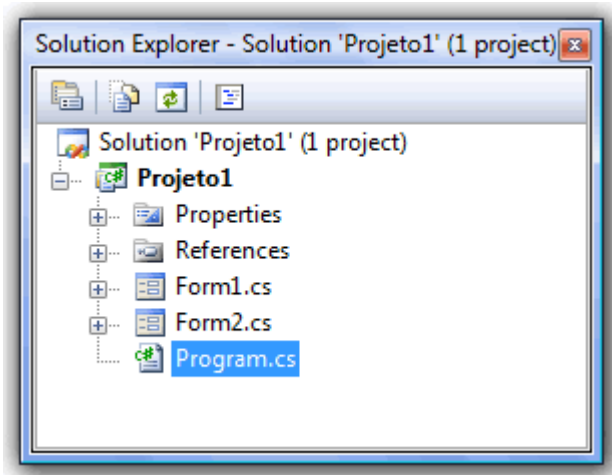
Visual C# 2008 - Módulo Básico
Aula 02



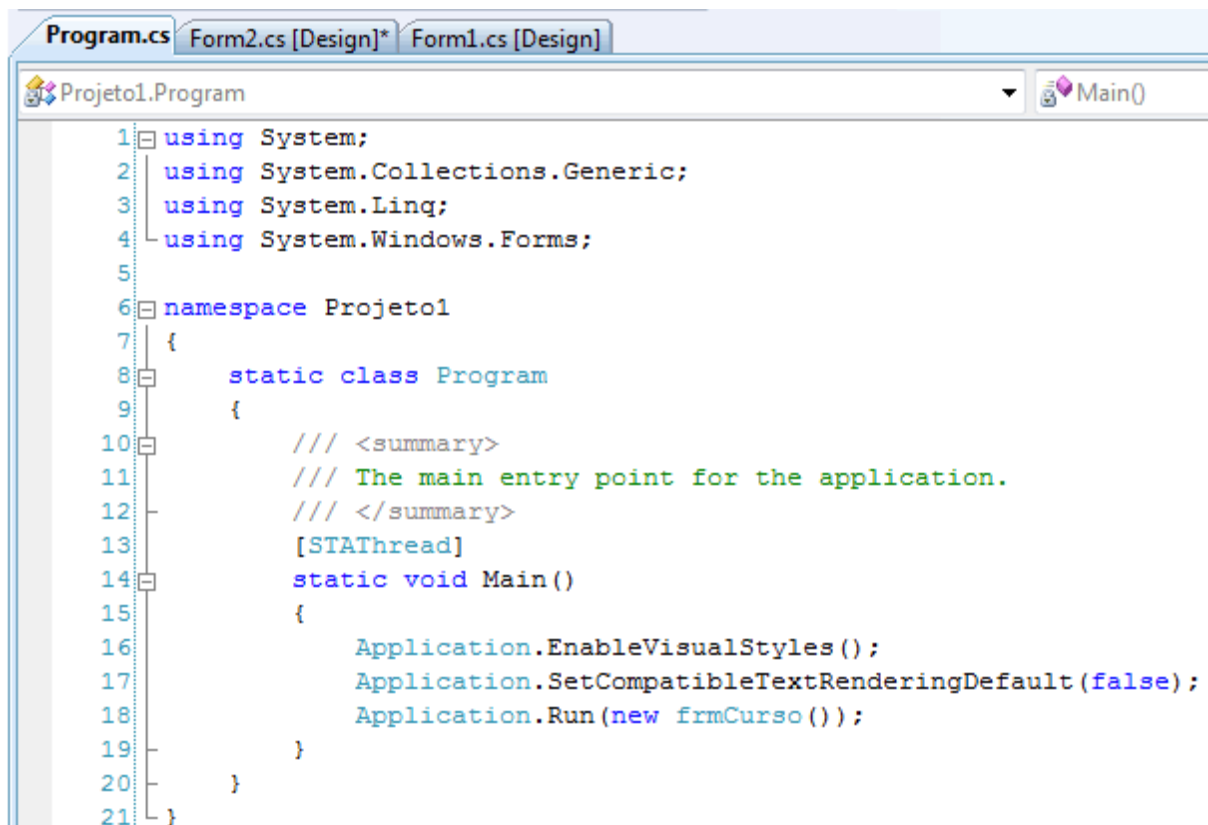
Definindo o formulário inicial

O formulário inicial é o formulário que será executado quando você executar a aplicação (F5).

A definição do formulário inicial é feita através do Program.cs



De um duplo clique neste arquivo para abrir o código do mesmo.



Não se assuste com o código acima, é um pouco complexo no momento.

O importante no momento é saber que quando você executa uma aplicação o Program.cs é carregado para a memória e o método Main (na linha 14) é executado automaticamente.

Visual C# 2008 - Módulo Básico

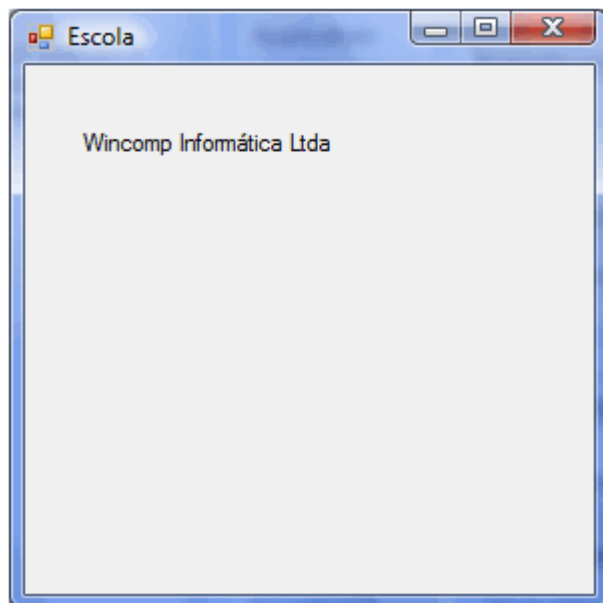
Aula 02

A instrução da linha 18 esta definindo qual formulário a aplicação deve rodar.

Basta alterar de frmCurso para frmEscola (que é o valor da propriedade Name do Form2)

```
14 static void Main()  
15 {  
16     Application.EnableVisualStyles();  
17     Application.SetCompatibleTextRenderingDefault(false);  
18     Application.Run(new frmEscola());  
19 }  
20 }
```

Execute o seu projeto (F5)



A janela Escola é que será apresentada agora.

Para encerrar a aplicação basta fechar esta janela.

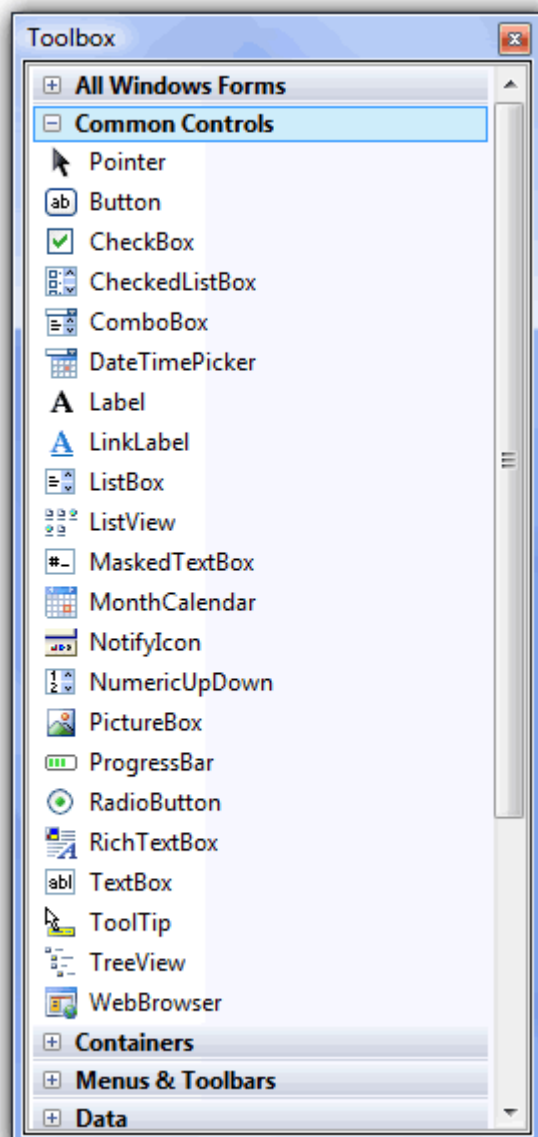
Toolbox

A Toolbox (caixa de ferramentas) é um contêiner para todos os controles que você pode adicionar ao Windows Forms.

Por padrão, a Toolbox está localizada no lado esquerdo do ambiente de desenvolvimento integrado (IDE).

Se a Toolbox não estiver visível, você pode clicar no menu **View** e em seguida, clicar em **ToolBox** para exibi-la.

Controles comuns na Toolbox



Você pode configurar a Toolbox para ocultar automaticamente quando você não a estiver usando, ou você pode configurar a Toolbox para estar sempre visível no IDE.

Basta utilizar o botão Auto Hide



Isso torna mais fácil para você ver todos os controles enquanto você cria seu aplicativo. O ícone a esquerda do X (pino) aparece em uma posição horizontal quando ele é definido como ocultar automaticamente. Caso contrário, ele será exibido em uma posição vertical.

Observação

Os controles não estão visíveis na ToolBox quando você estiver no Editor de Códigos.

Para adicionar controles ao seu aplicativo, você pode arrastá-los diretamente a partir da Toolbox para o formulário.

Guias da Toolbox

Para expandir uma guia, clique no sinal de mais (+) próximo ao nome da guia.

Para recolher uma guia expandida, clique no sinal de menos (-) próximo ao nome da guia.

Breve descrição as guias mais utilizadas em aplicações Windows

Os tipos de controles que são visíveis na Toolbox dependem do tipo de aplicativos que você está criando.

Para projetos Windows Forms veja abaixo uma breve descrição as guias mais utilizadas .

All Windows Forms

Contém todos os controles do Windows Forms que estão disponíveis na Toolbox.

Common Controls

Contem os controles, mais comuns, que normalmente são usados em aplicativos baseados no Windows

Containers

Contém controles que contem outros controles, como os controles GroupBox e os controles Panel.

Menus & ToolBars

Contém controles que permitem que você crie menus e barras de ferramentas para o seu aplicativo, como os controles MenuStrip e ToolStrip.

Data

Contém controles que ajudam você a facilmente trabalhar com dados em seu aplicativo, como o controle DataGridView.

Components

Contém componentes que geralmente não têm uma interface do usuário, tais como o Timer e o ErrorProvider.

Printing

Contém controles que permitem que você forneça recursos de impressão para um aplicativo.

Dialogs

Contém controles que permitem que você use caixas de diálogo comuns no seu aplicativo, como os controles SaveFileDialog, FolderBrowserDialog e FontDialog.

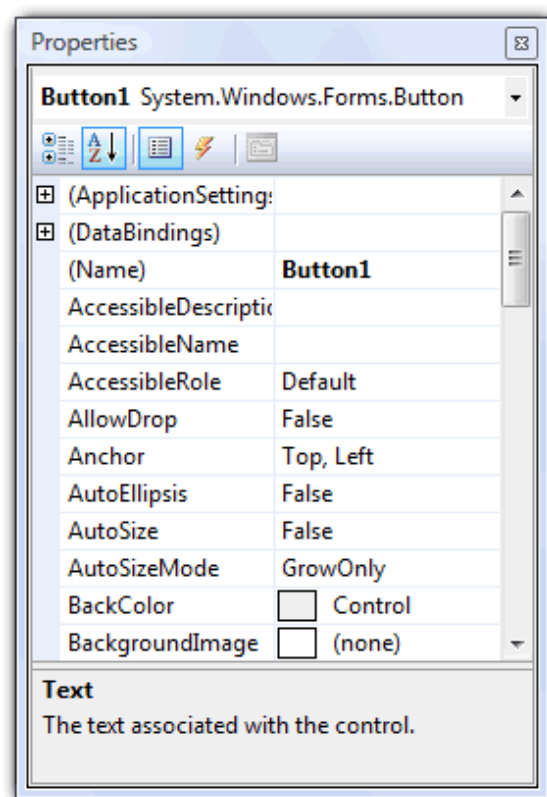
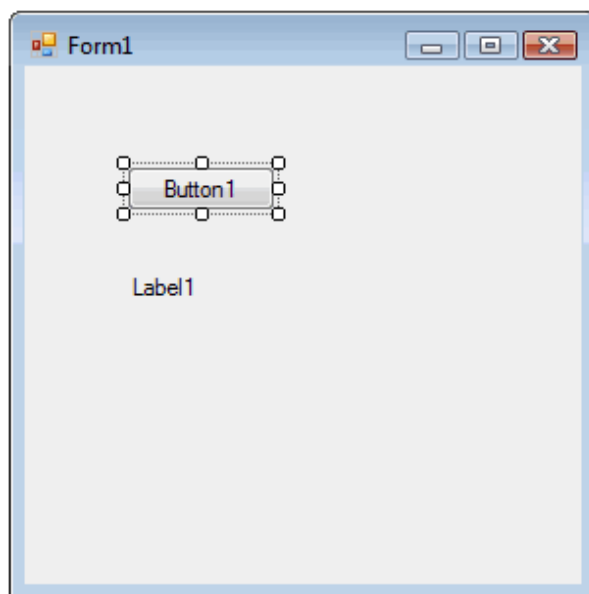
Usando a Janela Propriedades

Todos os objetos no Visual C#, incluindo formulários e controles, têm seus próprios atributos que os descrevem.

Esses atributos são conhecidos como propriedades.

Você pode definir as propriedades de um formulário e de controles no formulário no ambiente de desenvolvimento integrado (IDE) usando a janela *Properties*, ou escrevendo código no Editor de Códigos.

Por exemplo, se você deseja exibir algum texto em um botão, você pode definir a propriedade *Text* do botão, conforme visto anteriormente.



Usando o IntelliSense

O ambiente de desenvolvimento integrado (IDE) do Visual C# ajuda a escrever o código com menos digitação e portanto menos erros, fornecendo listas das palavras-chave disponíveis, variáveis e membros (métodos, propriedades e eventos).

O IDE também completa palavras à medida que você digita o código.

No Visual C#, você pode obter toda a ajuda que você precisa logo no editor de códigos, enquanto você digita o código.

Recursos do IntelliSense

Há vários recursos do IntelliSense que podem facilitar suas tarefas de codificação.

Isso inclui membros de lista, informações de parâmetro, informações rápidas, complementação de palavras e dicas de sintaxe.

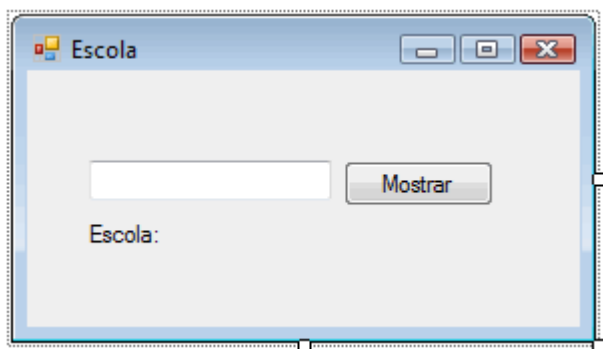
Quando você digita um nome no editor de código ele lista todos os métodos, propriedades e eventos que ficam disponíveis em uma lista suspensa.

Tente o seguinte

Abra o teu Form2, adicione um Label (rótulo), um TextBox (caixa de texto) e e um Button (botão) Configure

- 1- Configure a propriedade Name do Form2 para **frmEscola** e a propriedade Text para **Escola**
- 2- No Label configure a propriedade Name para **lblEscola** e a propriedade Text para **Escola:**
- 3- No TextBox configure a propriedade Name para **txtEscola** e a propriedade Text deixe em branco
- 4- No Button configure a propriedade Name para **cmdMostrar** e a propriedade Text para **Mostrar**

O seu formulário deverá se parecer com a figura abaixo



- 5- De um duplo clique no botão Mostrar para abrir a Janela de Código

Observe que automaticamente foi criado o procedimento de evento abaixo

```
24 private void cmdMostrar_Click(object sender  
25 {  
26     |  
27 }
```

e o cursor já esta na posição que você deve digitar o código.

Observação

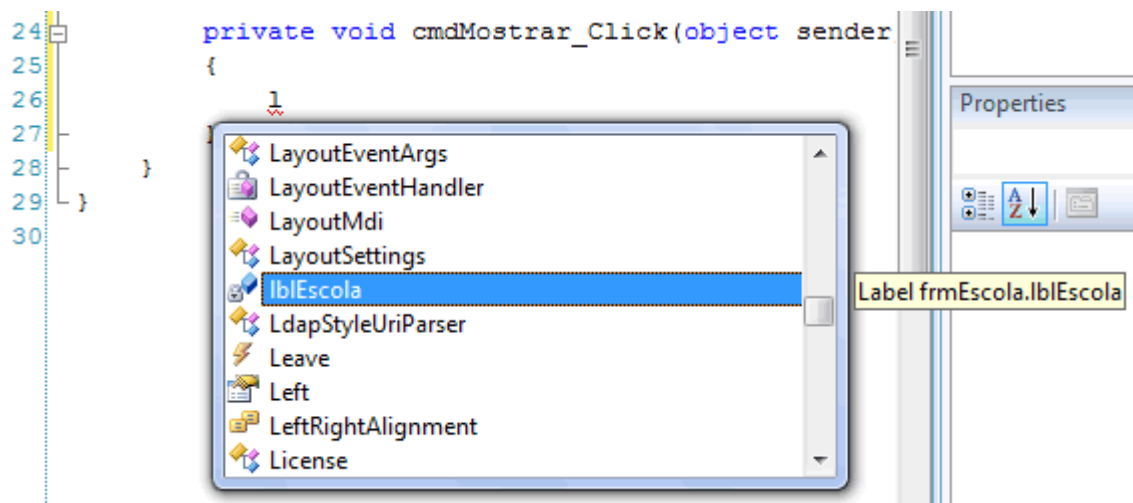
Procedimentos serão abordados mais a frente.

Completar Palavra

O IntelliSense pode completar algumas palavras para você à medida que você as digita.

Quando você tiver digitado suficientes letras de uma palavra para distingui-la de outras palavras, você pode completar a palavra pressionando **Enter**.

Por exemplo ao digitar apenas **I** (letra ele) o editor já vai sugerindo algumas opções disponíveis que iniciam com I



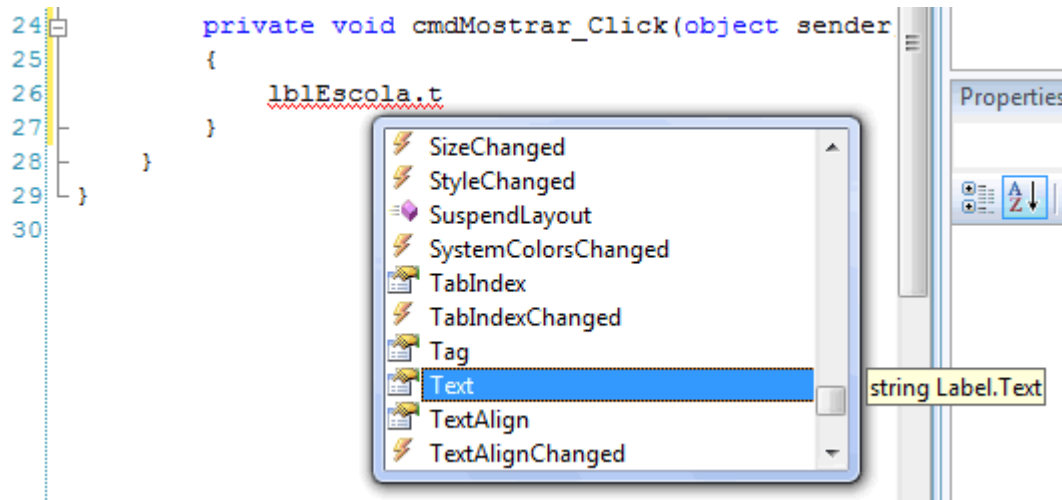
Membros da lista

Um exemplo de código escrito dentro de um método que exibe os membros da lista é mostrado na ilustração a seguir.

Você pode percorrer a lista ou digitar as primeiras letras do membro para ir rapidamente para esse membro na lista.

Visual C# 2008 - Módulo Básico

Aula 02



Em seguida, pressione a tecla **Enter** para adicionar a propriedade Text para o seu código.

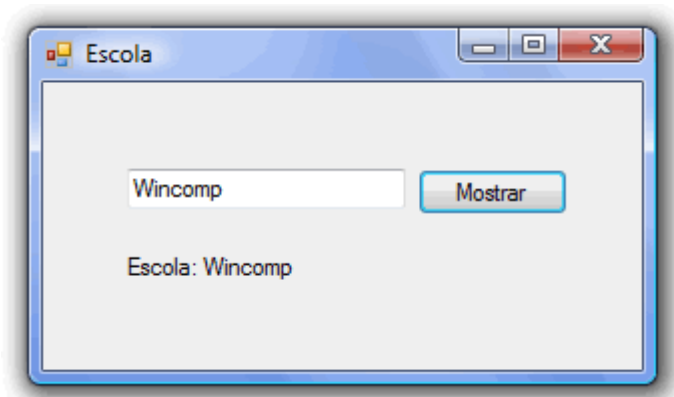
Complemente seu código conforme o exemplo abaixo

```
24 private void cmdMostrar_Click(object sender, EventArgs e)
25 {
26     lblEscola.Text = "Escola: " + txtEscola.Text;
27 }
```

Observações

- 1- No caso estamos concatenando (juntando) a string "Escola: " com a propriedade Text do controle txtEscola e atribuindo para a propriedade Text do controle lblEscola.
- 2- observe que no fim da linha há um ponto e vírgula que é necessário para finalizar uma instrução.
- 3- Utilize a complementação de código para evitar erros de digitação.
- 4- O C# diferencia letras maiúsculas de minúsculas.

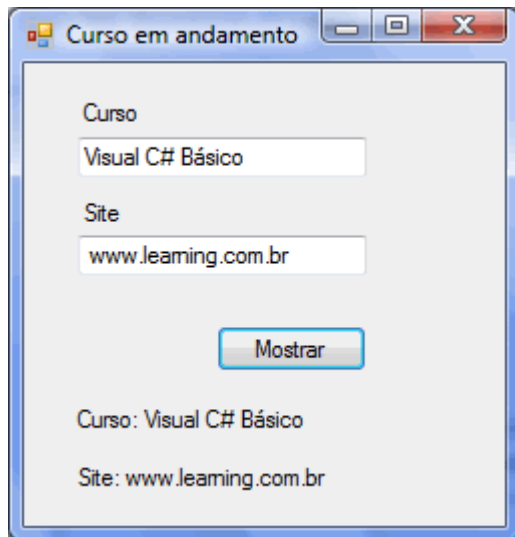
Execute o seu projeto (F5)



Preencha a caixa de texto e clique no botão Mostrar.

Avaliação da aula

1- Criar o seguinte formulário e configurar o mesmo como inicial.



Ao clicar no botão Mostrar o valor das caixas de texto serão apresentados nos Labels abaixo do botão.

Para a avaliação desta aula e para que a próxima aula seja liberada, você deve enviar o projeto acima e também o formulário apresentado nesta aula para o email do seu tutor.

O prazo para correção e conseqüente liberação da próxima aula é de até 24 horas.

Observação

Alguns provedores tem barrado emails que contenham anexos com arquivos compactados que contenham executáveis dentro (.exe ou .dll)

Sugerimos que para evitar problemas renomeie a extensão do arquivo de .zip ou .rar para .txt