

Microsoft Visual Studio Express 2012 for Windows Desktop

Apresentação da ferramenta

Professor: Danilo Giacobbo

Página pessoal: www.danilogiacobo.eti.br

E-mail: danilogiacobo@gmail.com

Introdução

- ❑ Visual Studio Express 2012 for Windows Desktop é a nova geração da IDE (Integrated Development Environment) de mesmo nome da Microsoft, voltada para o desenvolvimento de aplicações para desktop.
- ❑ Com ele, você pode criar programas explorando todas as possibilidades do Windows e utilizando linguagens nativas da plataforma, como C#, VB.NET, e C++.
- ❑ Essa versão da ferramenta é grátis para uso não comercial e requer um registro no site da Microsoft para obter uma chave de licença.
- ❑ Pode ser utilizado para desenvolver aplicações dos mais variados tipos e para diversas finalidades. O programa disponibiliza tanto o uso de console de comandos para programação quanto da ferramenta gráfica.



Características

- ✓ Esta versão, especificamente, é mais voltada para o desenvolvimento de aplicações para serem utilizadas em computadores com o Windows.
- ✓ Ele oferece suporte às linguagens mais adequadas para se criar aplicativos para o sistema Windows, incluindo C#, VB.NET, e C++. O programa também possibilita desenvolver para o Windows 8, Windows Presentation Foundation (WPF), Windows Forms (WinForms) e Win32.
- ✓ Grande quantidade de frameworks e outras tecnologias para facilitar o seu desenvolvimento, além de ferramentas de depuração (debug) e controles para as diferentes aplicações que podem ser criadas.
- ✓ Foi otimizado o suporte ao IntelliSense e para XAML (especialmente no quesito layout).
- ✓ Oferece a possibilidade de integrar o seu aplicativo com a computação nas nuvens (por meio de APIs).



Vantagens e Desvantagens

Vantagens

- Interface amigável
- Integrado a outros produtos da Microsoft
- Suporte a várias linguagens de programação

Desvantagens

- Inteiramente em inglês
- Programa pesado, pode deixar o computador lento



Instalação

O Microsoft Visual Studio Express 2012 for Windows Desktop pode ser baixado do seguinte endereço:

<http://www.microsoft.com/en-us/download/details.aspx?id=34673>

Você poderá selecionar se quer instalar o aplicativo a partir da Internet ou baixar uma imagem do mesmo para ser instalada mais tarde sem precisar de conexão com a internet.

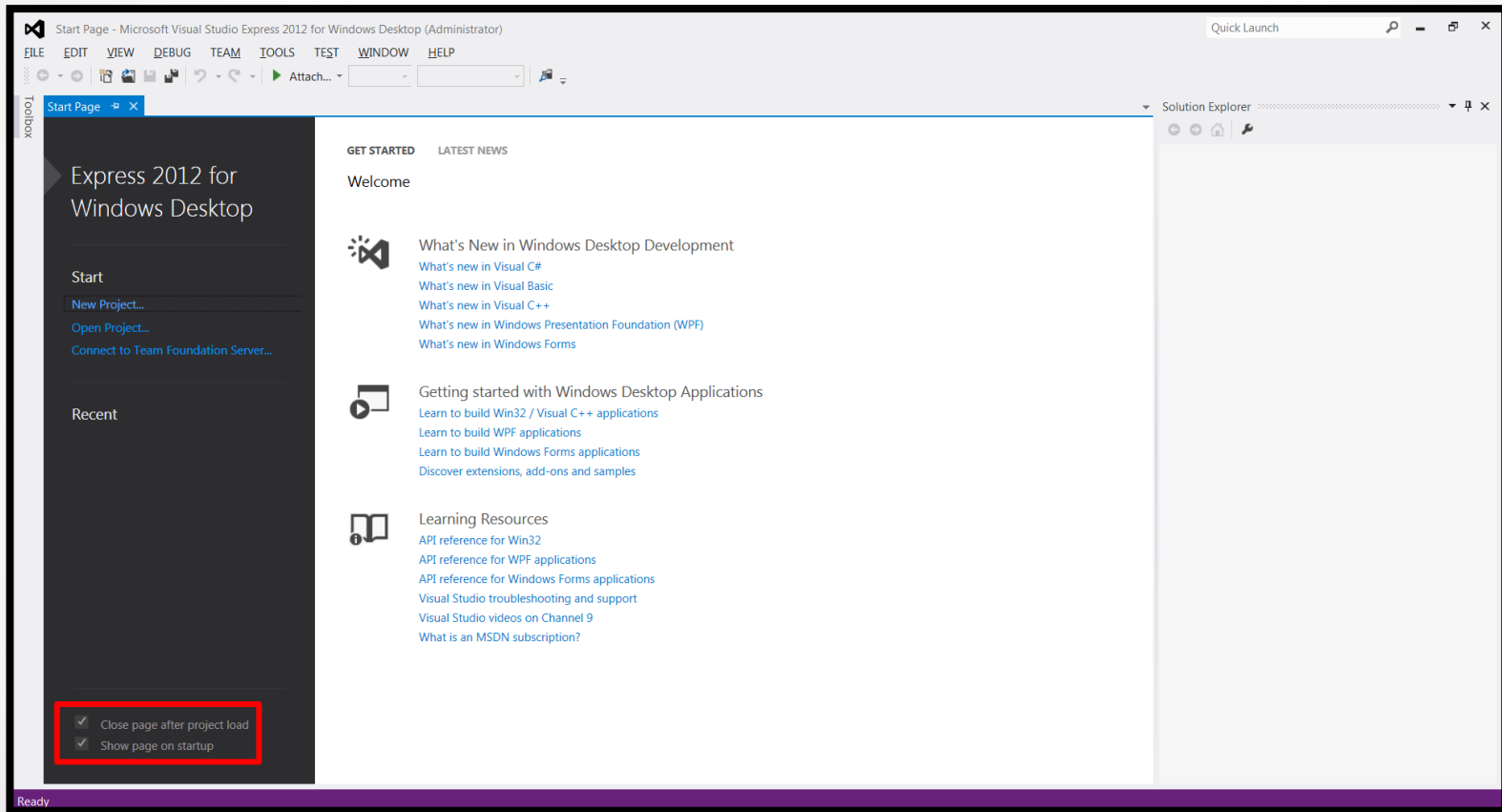
Depois de instalada a ferramenta será necessário obter uma chave de licença para usar o software. Para isso é necessário ter um cadastro no site da Microsoft.

Após este procedimento basta digitar a chave recebida por e-mail e ativar o programa. Ele estará então pronto para uso.



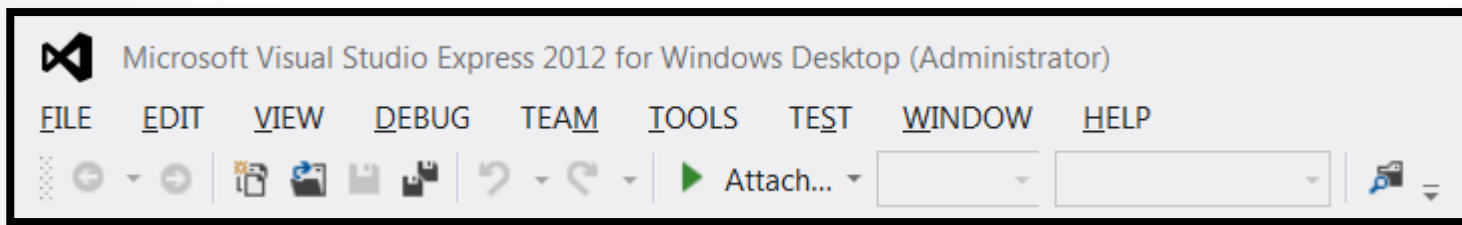
Tela Inicial

Quando a ferramenta for iniciada a seguinte tela será aberta:



Barra de Ferramentas

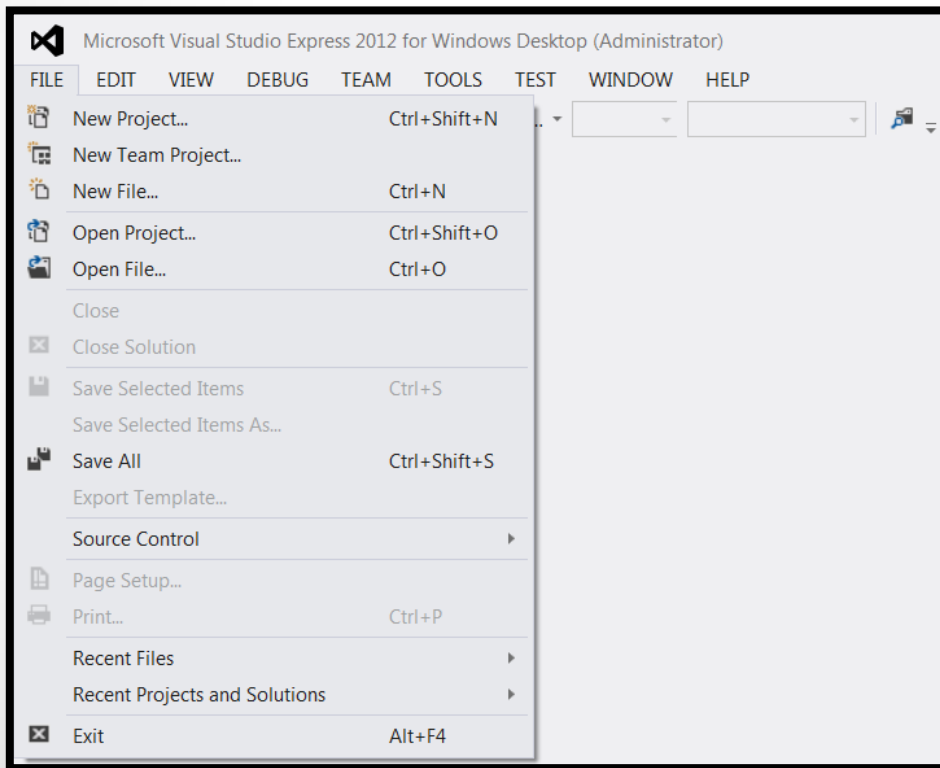
O menu e a barra de ferramentas ficam no canto superior da tela do aplicativo. As opções mais utilizadas ficam acessíveis por meio de ícones mas eles também podem ser customizados.



Nos slides a seguir serão explicados as principais opções e funcionalidades da ferramenta bem como algumas teclas de atalho úteis para tornar mais produtiva a utilização da mesma.

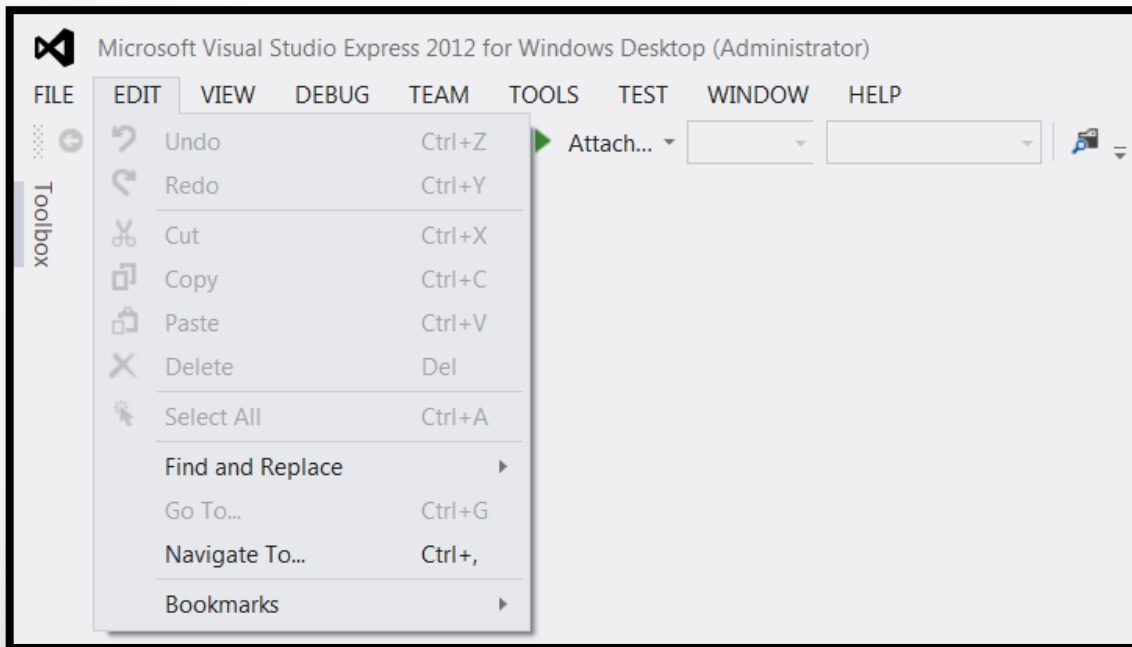
Barra de Ferramentas - FILE

- No menu FILE ficam as opção de **abertura** e **salvamento** de **projetos**.
- As **teclas de atalho** aparecem do lado de cada opção.
- As opções mais usadas são **Criar Novo Projeto** e **Salvar Tudo**.



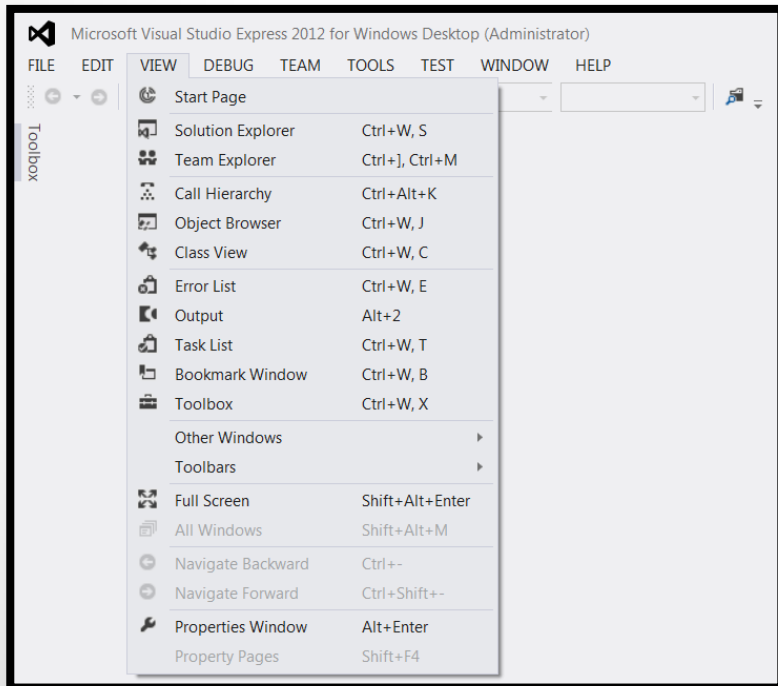
Barra de Ferramentas - EDIT

- No menu **EDIT** ficam as opções de edição que existem em qualquer ferramenta que trabalhe com arquivos do tipo texto.
- As **teclas de atalho** aparecem do lado de cada opção.
- As opções mais usadas são **Copiar, Colar, Recortar, Selecionar Tudo e Localizar e Substituir**.



Barra de Ferramentas - VIEW

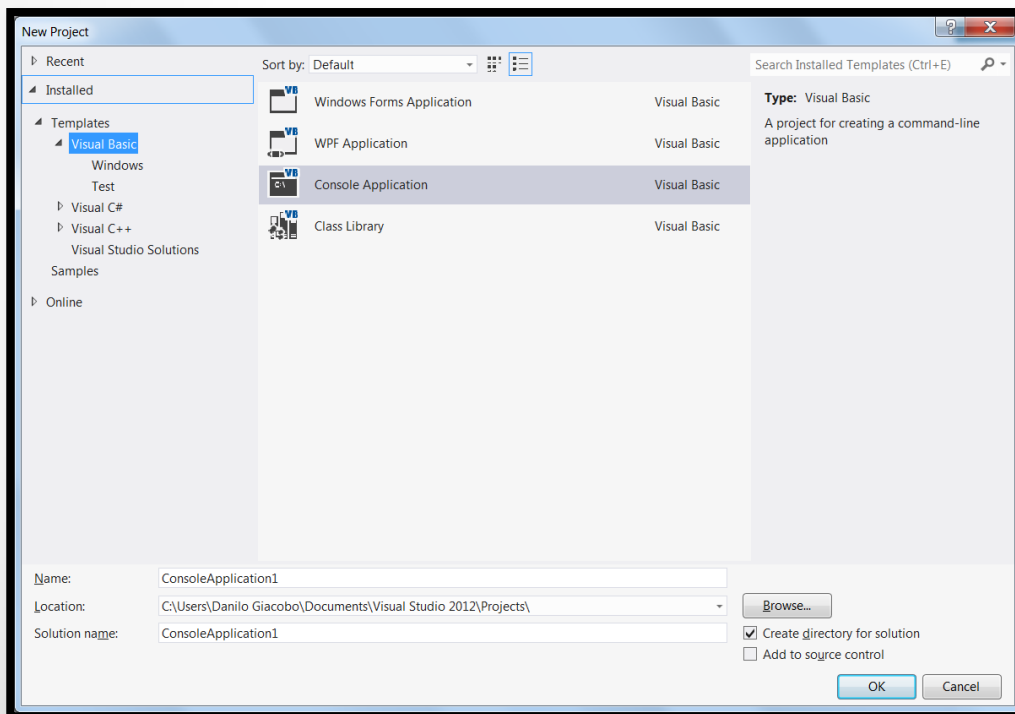
- No menu **V**IEW ficam as opções de visualização da ferramenta. Você pode customizar quais partes do aplicativo você gostaria que aparecesse quando da criação de um novo projeto.
- As **teclas de atalho** aparecem do lado de cada opção.
- As opções mais usadas são: **Mostrar a Caixa de Ferramentas**, o **Explorador da Solução** e a **Janela de Propriedades**.



Criando um projeto Console

Para verificar se a ferramenta está funcionando corretamente crie um pequeno projeto de teste do tipo Console conforme explicado a seguir:

- Clique no meu **FILE** -> **New Project...**
- A seguinte tela será aberta:



Criando um projeto Console

- Selecione a linguagem **Visual Basic**
- Selecione a opção **Console Application**
- Escolha um **nome para o seu projeto** e **onde o mesmo será salvo**.
- Pressione o botão **OK** para criar o projeto.

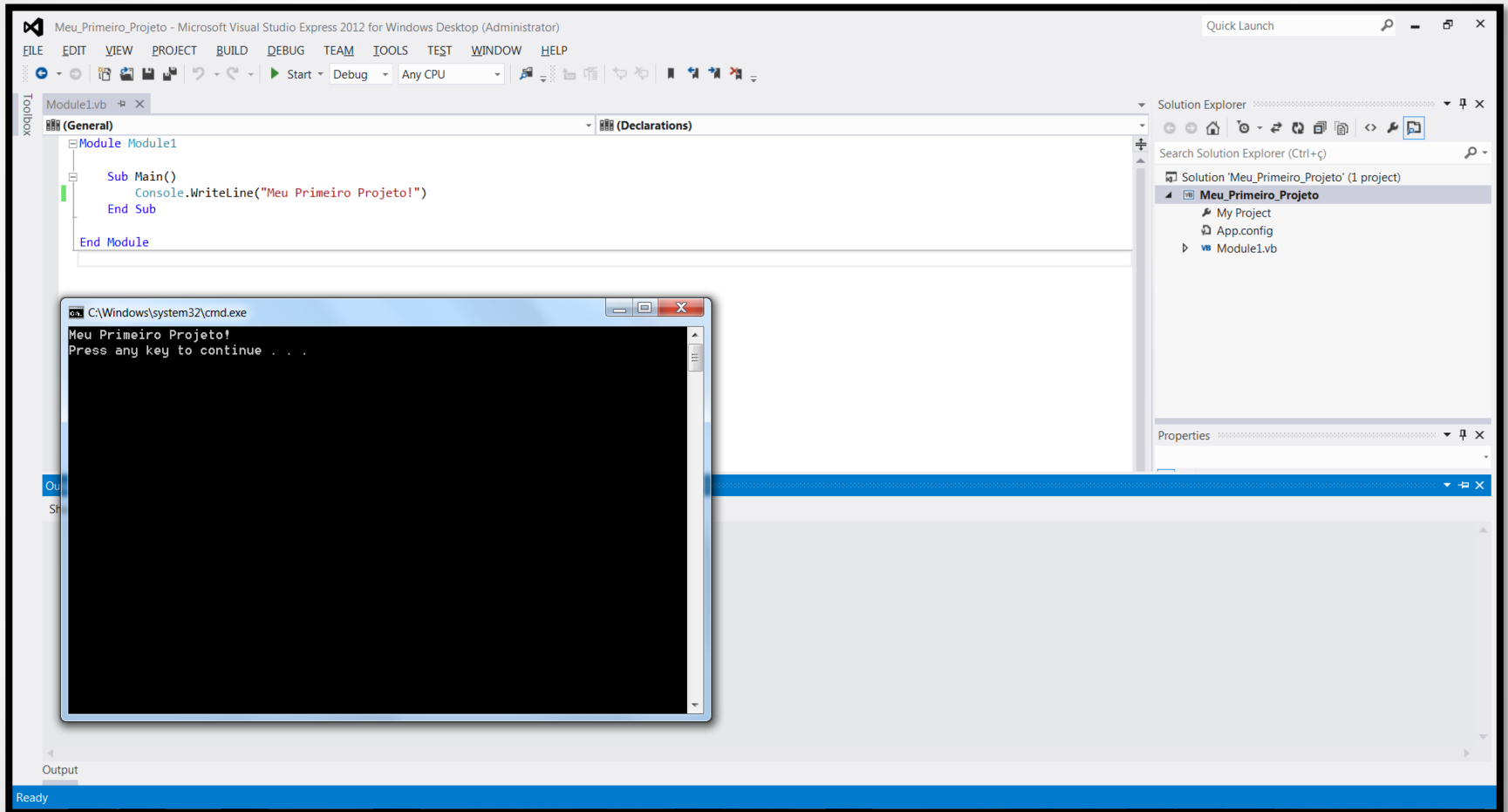
DICA: Não escolha um nome muito grande e um diretório com muitas pastas pois a ferramenta pode gerar um erro por causa disso.

- Dentro do código do módulo chamado **Module1.vb** e no corpo do procedimento **Main**, digite o seguinte código:

```
System.Console.WriteLine("Meu Primeiro Projeto!")
```

Para compilar e executar o programa basta clicar no botão **Start** ou pressionar as teclas **Ctrl + F5**.

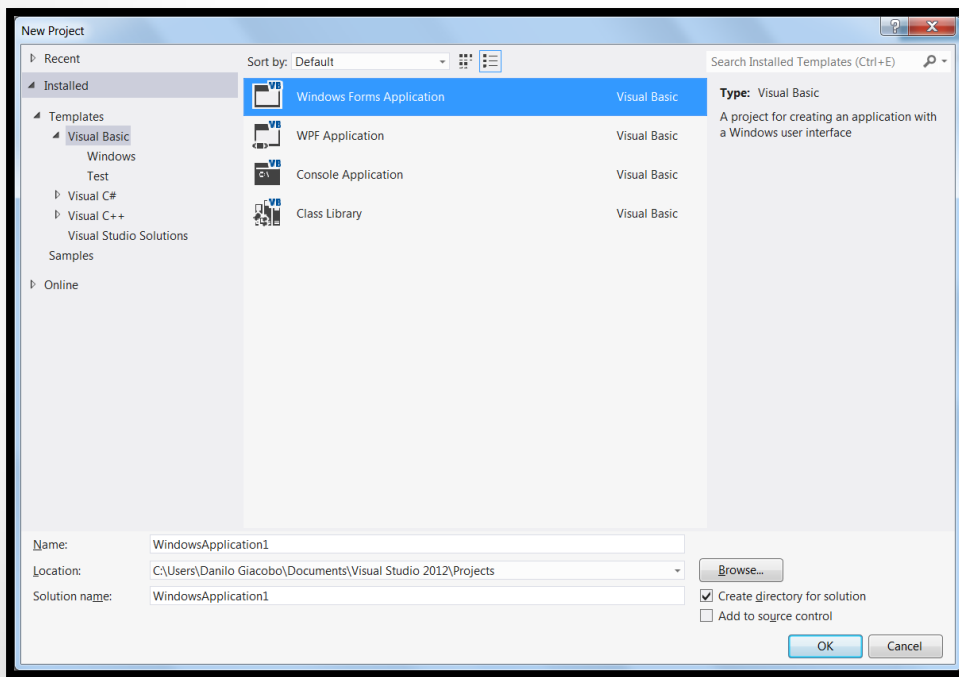
Criando um projeto Console



Criando um projeto Gráfico

Para criar uma aplicação baseada na interface gráfica do Windows siga os seguintes passos:

- Clique no meu **FILE** -> **New Project...**
- A seguinte tela será aberta:



Criando um projeto Gráfico

- Selecione a linguagem **Visual Basic**
- Selecione a opção **Windows Forms Application**
- Escolha um **nome para o seu projeto** e **onde o mesmo será salvo**.
- Pressione o botão **OK** para criar o projeto.

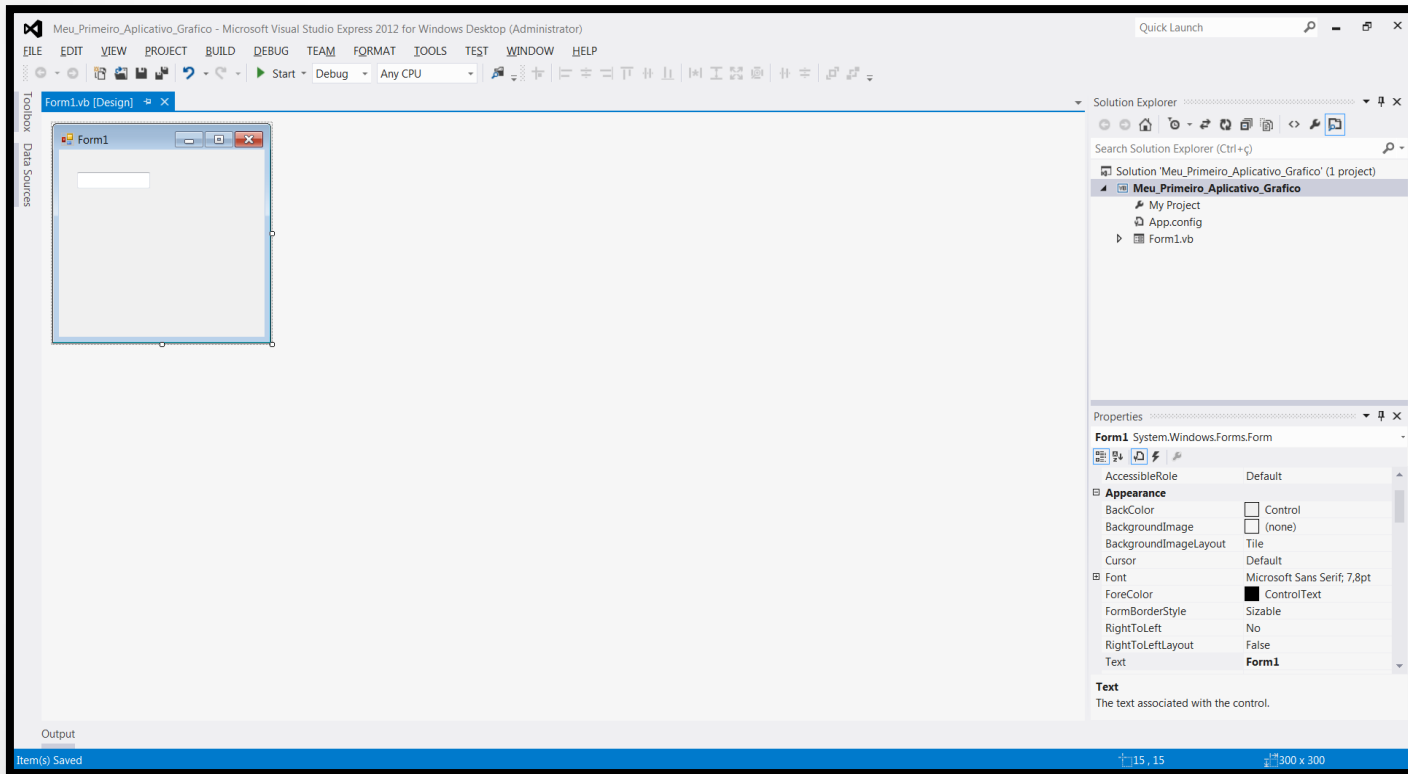
DICA: Não escolha um nome muito grande e um diretório com muitas pastas pois a ferramenta pode gerar um erro por causa disso.

- Um novo **form** será criado onde você poderá desenhar sua aplicação gráfica por meio dos componentes disponibilizados na caixa de ferramenta.
- Abra a **Toolbox** e arraste um **componente** do tipo **TextBox** para a tela.

Dica: você pode clicar duas vezes sobre o componente que o mesmo será inserido no form.

Criando um projeto Gráfico

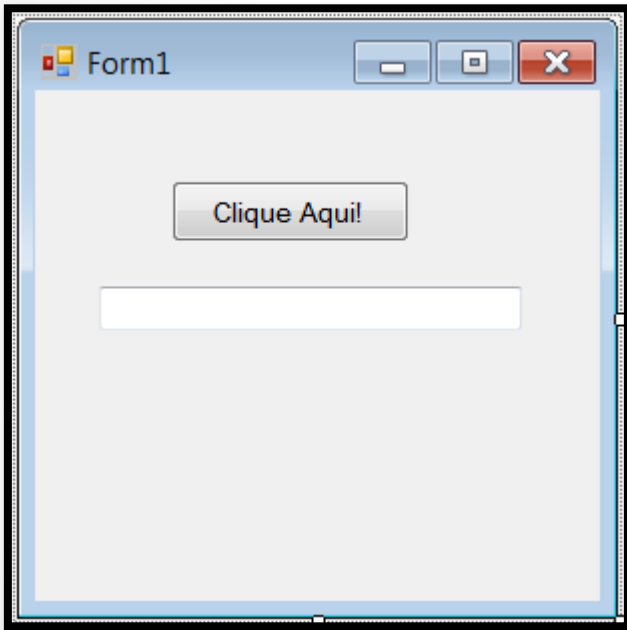
Depois dos passos realizados anteriormente o resultado deverá ser semelhante a imagem abaixo:



Dica: Se o arquivo estiver com um * do lado do nome significa que ele ainda não foi salvo.

Criando um projeto Gráfico

- Abra a **Toolbox** e arraste um **componente** do tipo **Button** para a tela.
- Altere o texto do botão para **Clique Aqui!**.
- Organize os componentes no form conforme mostra a figura abaixo:



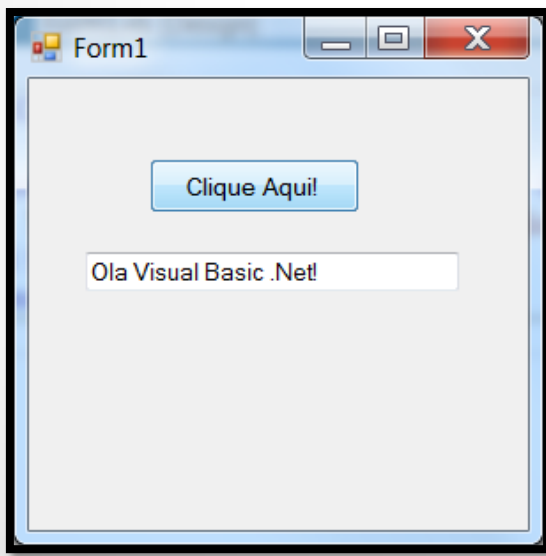
- Clique duas vezes sobre o botão. O código do form será aberto. Nessa tela você poderá digitar o código para manipular as ações ocorridos no botão.

Criando um projeto Gráfico

- Dentro do procedimento provado chamado **Button1_Click**, adicione o seguinte código:

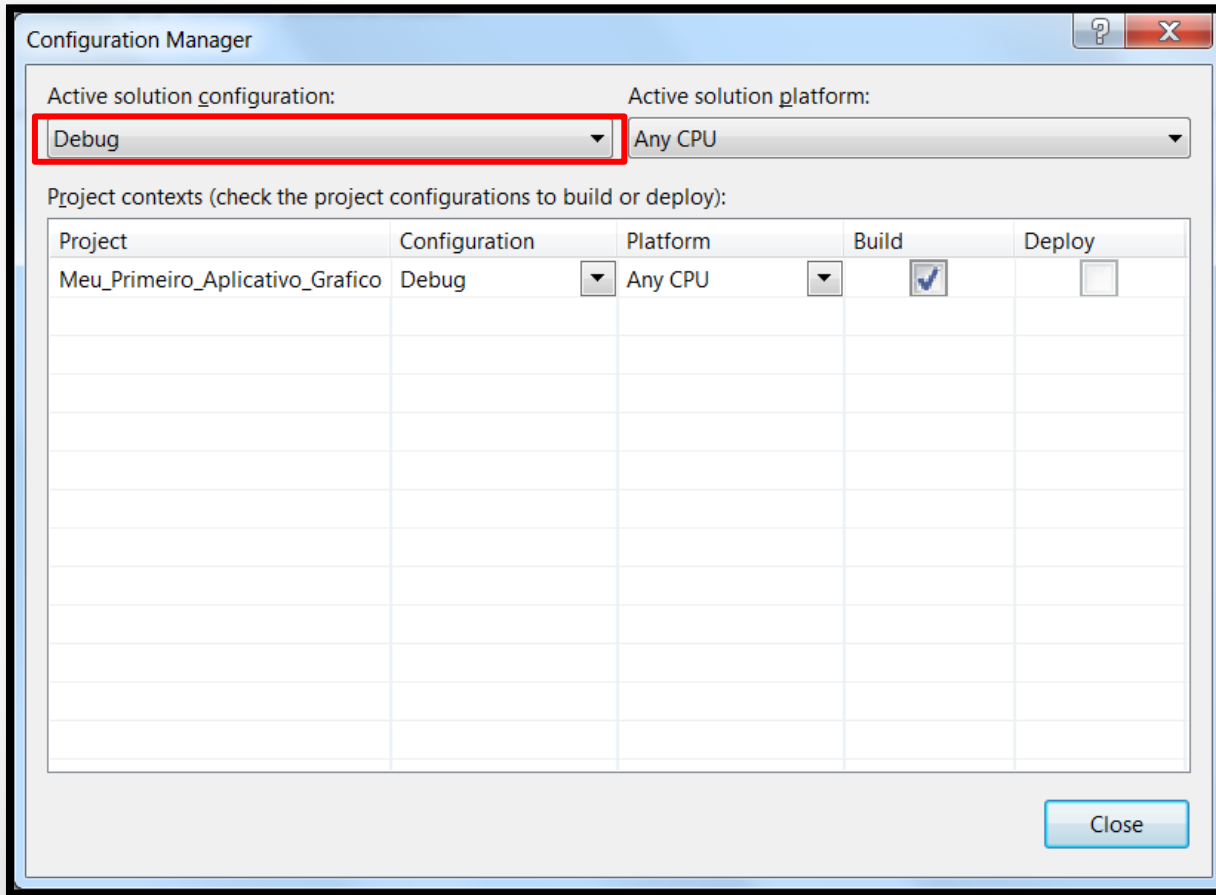
```
TextBox1.Text = "Ola Visual Basic .Net!"
```

Para compilar e executar o programa basta clicar no botão **Start** ou pressionar a tecla **F5**.



Versões de Debug e Release

Selecione o menu **BUILD** e depois a opção **Configuration Manager...**



#Region e #End Region

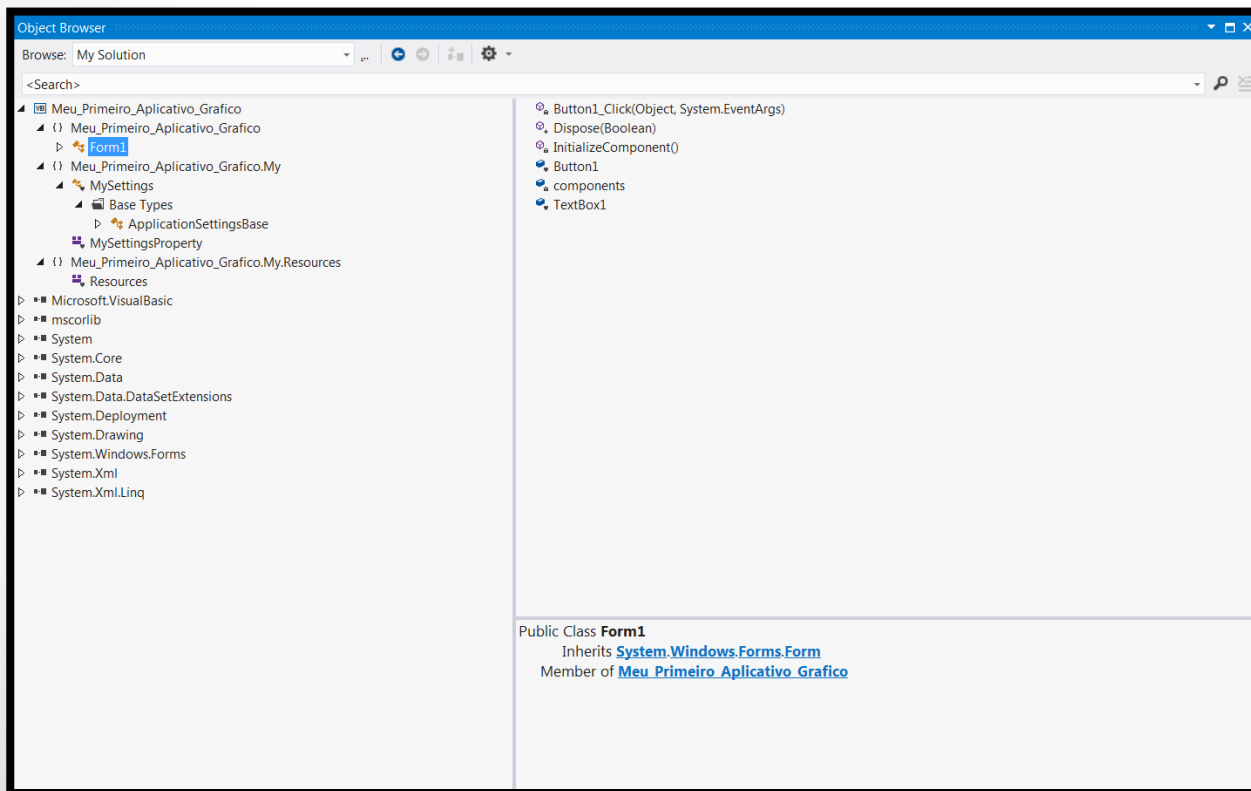
Você pode usar as diretivas **#Region** e **#End Region** no código, permitindo que você expanda e contraia partes do código. Exemplo:

```
1 #Region "Comentários"  
2  
3 'Exemplo de criação de uma região  
4 'Autor: Danilo Giacobbo  
5 'Data: 24/02/2014  
6  
7 #End Region  
8  
9 Public Class Form1  
10  
11 End Class  
12
```

```
1 + Comentários  
8  
9 Public Class Form1  
10  
11 End Class  
12
```

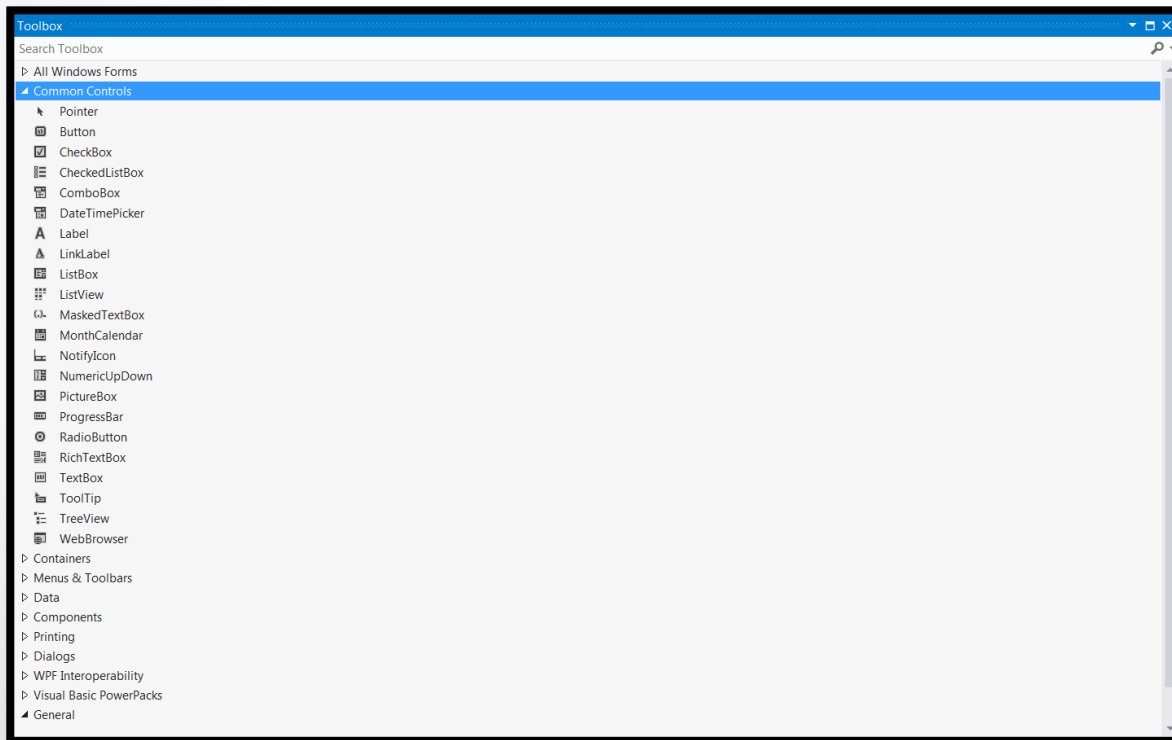
Object Browser

O **Object Browser** permite você olhar todos os membros de um objeto de uma vez só. Ele ajuda também a descobrir quais os objetos “misteriosos” que a ferramenta inclui junto ao seu projeto. Para exibir essa janela basta acessar o menu **VIEW -> Object Browser** ou usar a tecla de atalho **Ctrl + W + J**.



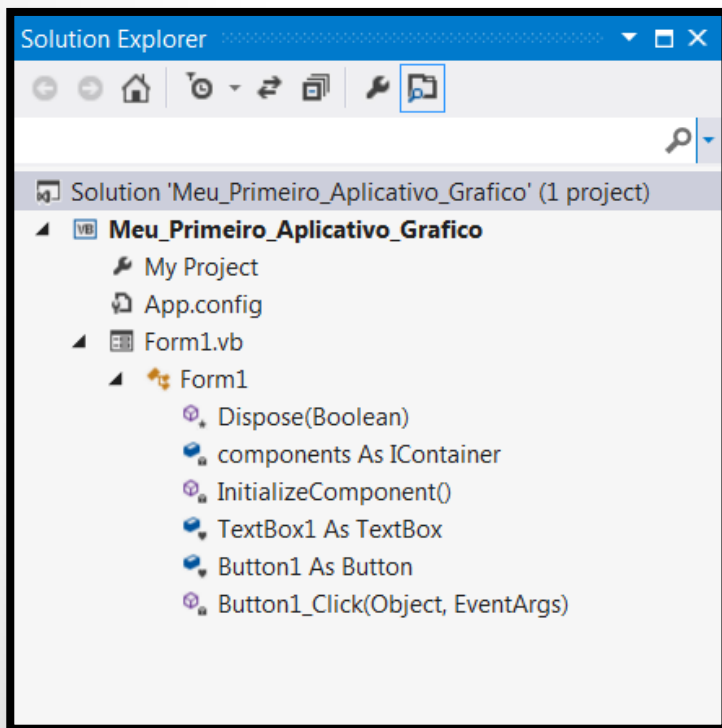
Toolbox

A **Caixa de Ferramentas** é dividida em categorias de componentes. Ela só aparece quando você está com algum formulário aberto. Todos os componentes necessários para a criação de um projeto visual ficam dispostos nessa janela. Para exibir essa janela basta acessar o menu **VIEW** -> **Toolbox** ou usar a tecla de atalho **Ctrl + W + X**.



Solution Explorer

○ **Solution Explorer** exibe os itens que integram o seu projeto. Ele mostra os mesmos de forma hierárquica. Quando você clica em cima de um item ele exibe as propriedades deste. Para exibir essa janela basta acessar o menu **VIEW - > Solution Explorer** ou usar a tecla de atalho **Ctrl + W + S**.



Outras Janelas

Class View

Essa janela fica na mesma aba da Solution Explorer. Ela exibe soluções e projetos em termos de classes. Para exibir essa janela basta acessar o menu **VIEW** -> **Solution Explorer** ou usar a tecla de atalho **Ctrl + W + C**.

Properties

Essa janela tem por objetivo mostrar as propriedades de um determinado objeto visual. Você pode designar valores para cada propriedade. Algumas são apenas de leitura. Para exibir essa janela basta acessar o menu **VIEW** -> **Properties Window** ou usar a tecla de atalho **Alt + ENTER**.

Component Trays

Quando um componente que não é visível em tempo de execução é colocado no formulário o mesmo é exibido em uma área especial da tela abaixo da janela principal do formulário.

Outras Janelas

Output Window

Essa janela mostra o resultado da compilação e execução de programas. Você inclusive pode mandar mensagens para esta janela. Tente escrever esse trecho de código no seu programa para ver o resultado:

```
System.Diagnostics.Debug.WriteLine("Olá Janela de Output")
```

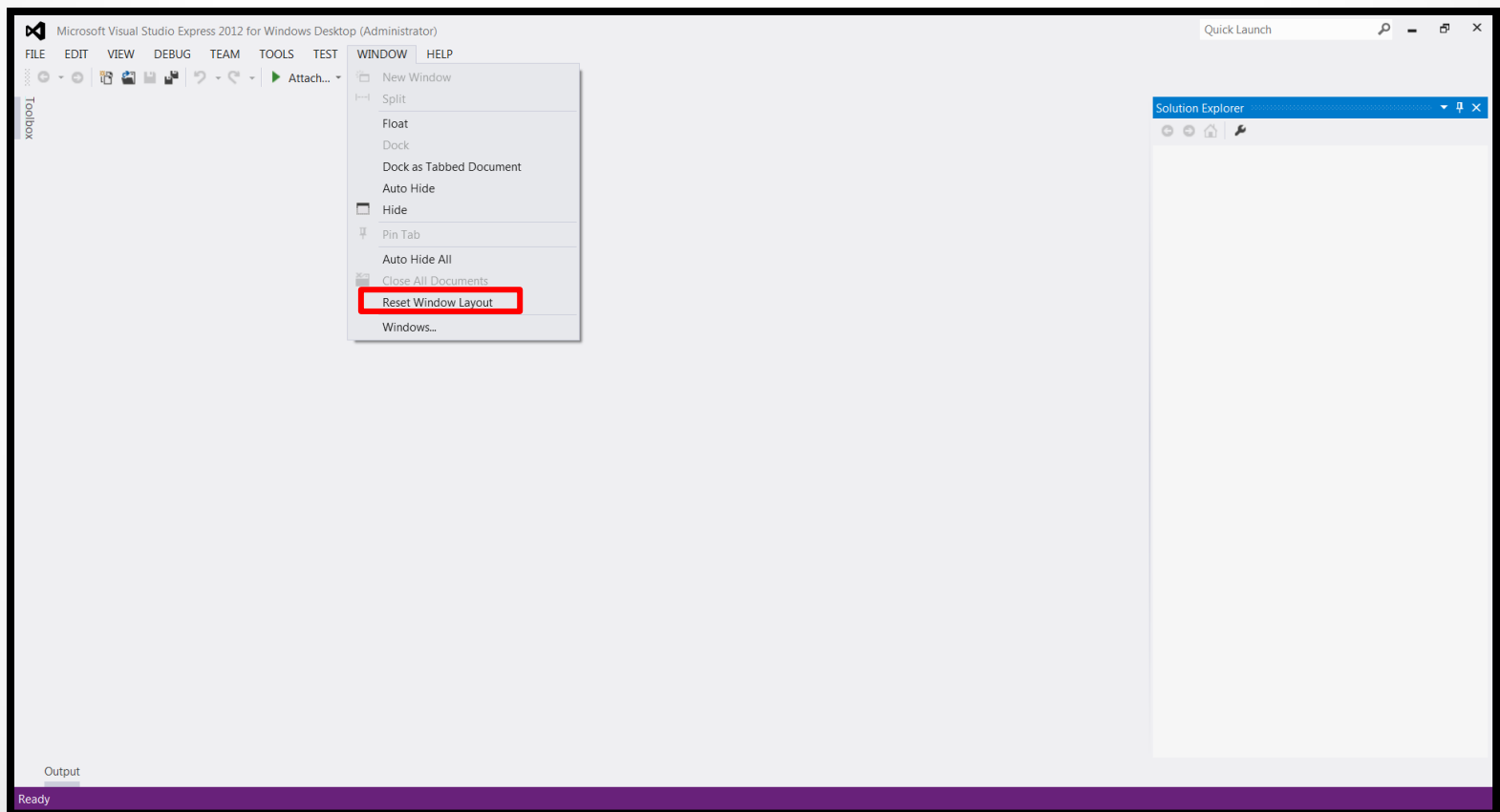
Para exibir essa janela basta acessar o menu **VIEW** -> **Output** ou usar a tecla de atalho **Alt + 2**.

Task List

Essa janela uma lista de tarefas que você pode criar para organizar melhor seu projeto. Ela serve também para o VB .NET marcar os problemas que existem em seu código. Você pode escrever comentários para que outros desenvolvedores saibam o que foi alterado no código. Para exibir essa janela basta acessar o menu **VIEW** -> **Output** ou usar a tecla de atalho **Ctrl + W + T**.

Dicas

- Para restaurar o layout das janelas para o seu estado original siga os seguintes passos:
 - Acesse o menu **Window** e depois clique em **Reset Window Layout**.



Dicas

- Se você digitar algum código que o VB .NET considere um erro de sintaxe ele vai sublinhar o mesmo com uma linha vermelha. Você pode parar o ponteiro do mouse em cima da linha que ele exibirá uma dica explicando porque ele acha que está errado.
- Faça uso do **IntelliSense**: ele mostra os membros de um objeto, lista os argumentos de um procedimento, mostra informações sobre alguns elementos do seu código, completa palavras digitadas e adiciona parênteses e chaves quando necessário.
- A ferramenta ajuda a você alinhar e espaçar seus componentes em uma janela gráfica poupando o trabalho de saber se os objetos estão dispostos corretamente na mesma.
- Se você clicar com o botão direito do mouse em cima de um projeto e depois na opção **Properties** você poderá alterar as configurações do seu aplicativo.

Melhores Práticas

Evite a utilização de “Números Mágicos”:

Exemplo:

```
Function blnVerificaTamanho(dblParametro As Double) As Boolean  
    If dblParametro > 1024 Then  
        blnVerificaTamanho = True  
    Else  
        blnVerificaTamanho = False  
    End If  
End Function
```

O número 1024 é claramente um número “mágico”. Você deve declarar ele como uma constante principalmente se existirem vários deles no projeto.

Melhores Práticas

- Crie um módulo para armazenar funções e constantes que serão utilizadas por todos os códigos do projeto.
- Programe de forma defensiva. Por exemplo, verifique se os dados passados aos procedimentos e funções estão corretos antes de usar.
- Procedimentos ou funções devem ter apenas um propósito. Se ela faz várias tarefas considere então dividir a mesma em procedimentos menores.
- Evite muitas condições e laços de repetição aninhados. Três níveis devem ser o máximo recomendável.
- Variáveis devem ser definidas no menor escopo possível. Variáveis globais podem causar muitos problemas.
- Não passe variáveis globais via parâmetro para procedimentos e funções.

Melhores Práticas

- Quando for criar um longo texto (String), use o caractere “_” para quebrar o texto em várias linhas. Essa prática facilita a visualização do conteúdo da variável. Exemplo:

Dim Msg As String

```
Msg = "Bem, há um problema " _  
& "com o seu programa. Eu não estou certo " _  
& "qual problema é, mas definitivamente " _  
& "alguma coisa está errada."
```

- A Microsoft recomenda a indentação do código com 4 espaços.

- Se você for trabalhar em grupo em um projeto visual considere o uso de um ferramenta de controle de versões (exemplo: Visual SourceSafe).

Referências Bibliográficas

- HOLZNER, Steven. **Visual basic.NET: black book**. Arizona: Coriolis Group Books, 2002. xxxviii, 1144 p ISBN 1-57610-835-X.
- **Visual Studio Express 2012 for Windows Desktop**. Disponível em: <http://www.baixaki.com.br/download/visual-studio-express-2012-for-windows-desktop.htm>. Acesso em 22 de Fevereiro de 2014.