

Prof. Roberto de Castro

Escola Técnica Estadual “São Paulo” - ETESP Linguagem C# - Visual Studio 2019

1 – Conceitos Preliminares

- O que é “computador”?
- O que é “programa”?
- O que é “Linguagem de Programação”?
- Programa “fonte” e “programa executável”.

2 – A linguagem “C”

Desenvolvida nos laboratórios Bell na década de 70, a partir da Linguagem B (criada no final dos anos 60 por Ken Thompson), que foi reformulada por Brian Kernighan e Dennis M. Ritchie e posteriormente renomeada para C.

Podendo ser considerada como uma linguagem de médio nível, pois possui instruções que a tornam ora uma linguagem de alto nível ora uma linguagem de baixo nível pois possui instruções tão próximas da máquina, que só o Assembler possui.

Devemos lembrar que a linguagem C foi desenvolvida a partir da necessidade de se escrever programas que utilizassem recursos próprios da linguagem de máquina de uma forma mais simples e portátil que o assembler.

3 - História da POO (Programação Orientada a Objetos)

1967 – Simula → A primeira linguagem a introduzir conceitos de OO (Orientação a Objetos), idealizada em 1966, na Noruega, como uma extensão da linguagem ALGOL 60.

1972 – Smalltalk → Desenvolvida no Centro de Pesquisas da Xerox durante a década de 70 e incorporou conceitos da linguagem Simula.

1980 – C++ → Derivada da linguagem “C”.

1983 – Ada → Criada para uso militar nos EUA, através de um concurso realizado pelo U.S. Department of Defense (DoD). O principal projetista da equipe foi o francês Jean Ichbiah.

1984 – Eiffel → Criada por Bertrand Meyer que tinha uma extensa experiência com programação orientada a objeto, particularmente com SIMULA.

1986 – Object Pascal → É uma linguagem orientada a objetos, isto é, todas as informações são tratadas como objetos e todos estes objetos pertencem a uma classe, que são categorias de objetos. Delphi / Kylix / Lazarus são exemplos de ferramentas que utilizam esta linguagem.

1995 – Java → O Java é ao mesmo tempo um ambiente e uma linguagem de programação desenvolvida pela Sun Microsystems Inc, puramente orientada a objetos. Os aplicativos em Java são compilados em um código de bytes (“bytecodes”) independente de arquitetura. Esse código pode ser executado em qualquer plataforma que suporte um interpretador Java.

2001/2002 – Plataforma .NET → Ambiente de desenvolvimento integrado da Microsoft, que integra linguagens Visual Basic, C, C++, C# e F#.

4 – O ambiente de Desenvolvimento do Visual Studio IDE - Integrated Development Environment (Ambiente de Desenvolvimento Integrado)

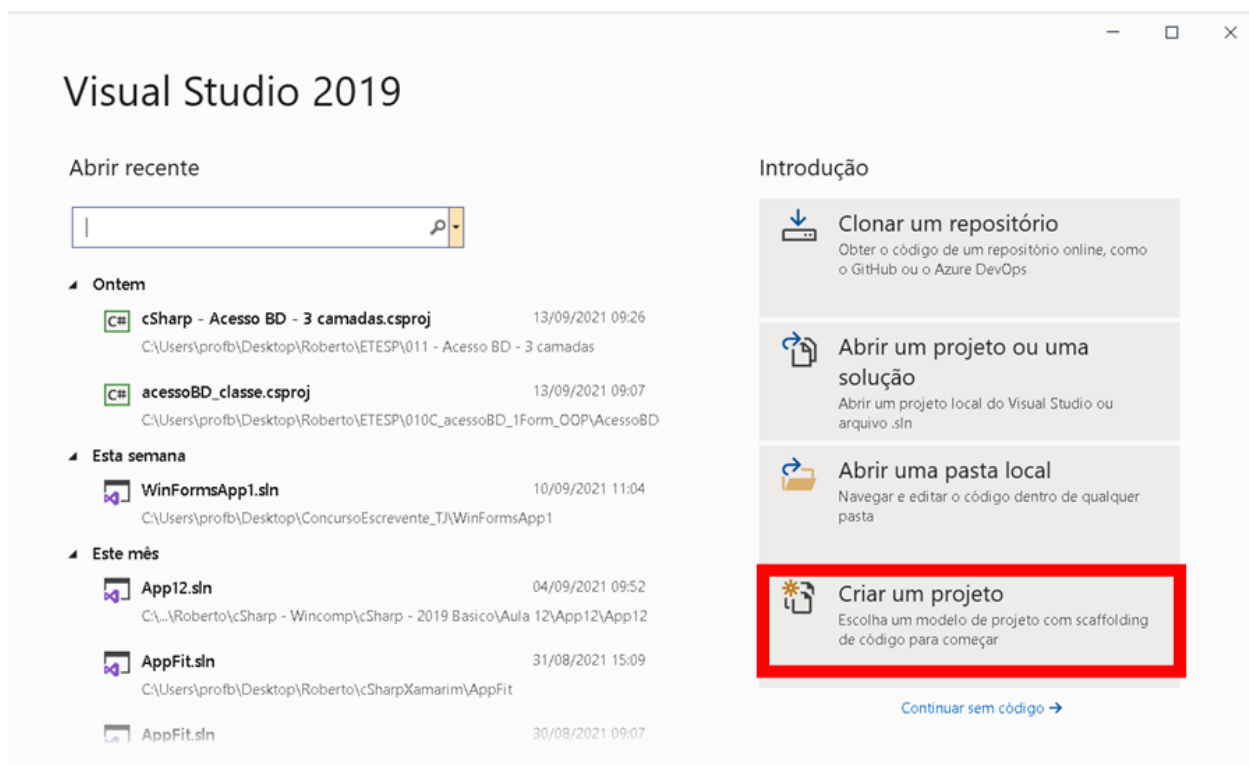
Neste ponto vamos nos preparar para desenvolver (criar) o nosso primeiro projeto. Para que isto seja possível, o Visual Studio 2019 deverá estar instalado em seu computador. Veja a documentação completa no site da Microsoft → <https://docs.microsoft.com/pt-br/visualstudio/install/install-visual-studio?view=vs-2019>

Depois de instalado, vamos iniciar o projeto.

1. Abra o Visual Studio.

A janela de início é exibida com várias opções: clonar um repositório, abrir um projeto recente ou criar um projeto.

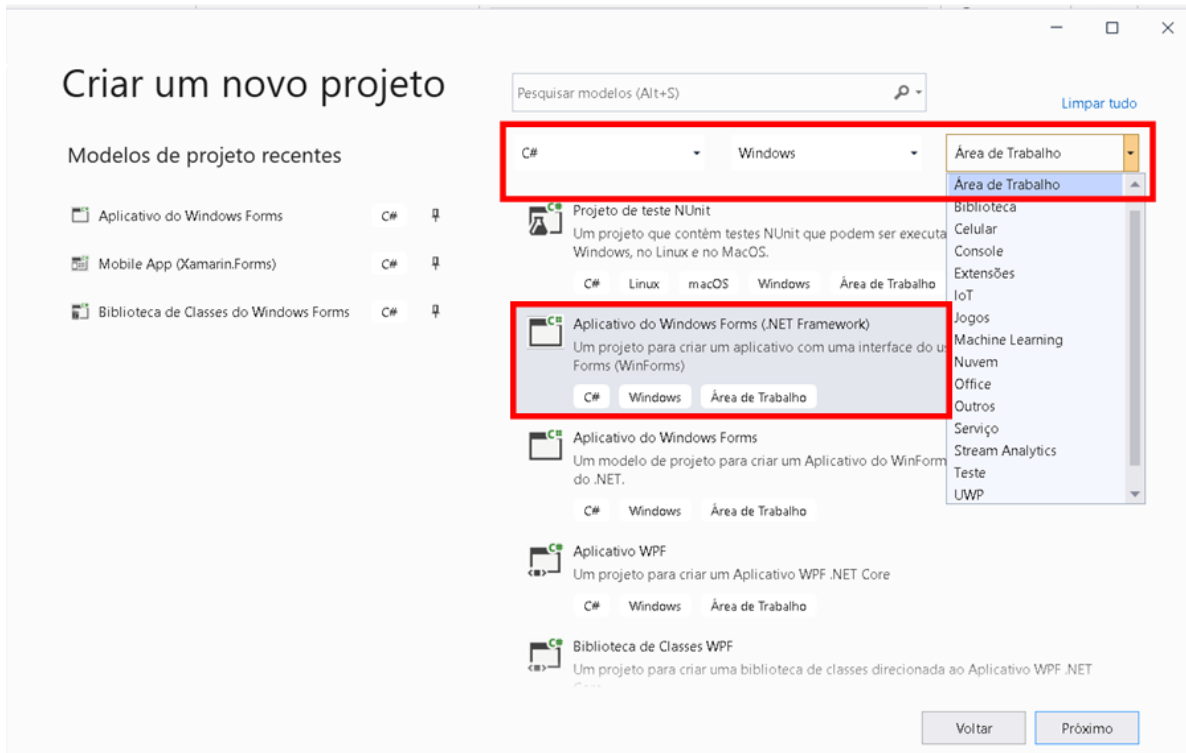
2. Escolha **Criar um projeto**:



A janela “**Criar um novo projeto**” é aberta e exibe diversos *modelos* de projeto. Um modelo contém os arquivos básicos e as configurações necessárias para um determinado tipo de projeto.

O desenvolvimento deste nosso primeiro projeto será do tipo “C# → Windows → Área de Trabalho (Desktop)”. Podemos fazer esta “seleção/filtro” através das caixas de pesquisa, seguindo as etapas abaixo:

Prof. Roberto de Castro



NOTA IMPORTANTE: Observe que temos duas opções para a criação de projetos C# Windows Forms (WFA).

A primeira, em destaque na imagem acima (.NET Framework), que é a que utilizaremos neste curso e a exibida logo abaixo na imagem: Aplicativo do Windows Forms.

No primeiro caso (Aplicativo do Windows Forms – NET Framework), em resumo, refere-se a projetos que serão criados exclusivamente para aplicações Windows.

O segundo caso (Aplicativo do Windows Forms – modelo de projeto), em resumo, refere-se a projetos que serão criados para serem executados em multiplataforma: Windows, Linux e Mac.

Para se aprofundar no assunto: <https://docs.microsoft.com/pt-br/dotnet/standard/choosing-core-framework-server>

Continuando....

Selecione a opção “Aplicativo do Windows Forms (.NET Framework)” e clique em “Proximo”

Na janela “Configurar seu **novo projeto**”, digite “Exe001” na caixa “Nome **do Projeto**” e informe o local onde o projeto será gravado (clique em ...). Esta é uma etapa muito importante, pois o Visual Studio previamente indica um local, mas o ideal é você gravar seus projetos em uma pasta que deverá ser criada para esta finalidade:

Configurar seu novo projeto

Aplicativo do Windows Forms (.NET Framework) C# Windows Área de Trabalho

Nome do projeto
Exe001

Local
C:\Users\profb\Desktop\Roberto\ETESP\0-2021\

Nome da solução ⓘ
Exe001

☐ Colocar a solução e o projeto no mesmo diretório

Framework
.NET Framework 4.7.2

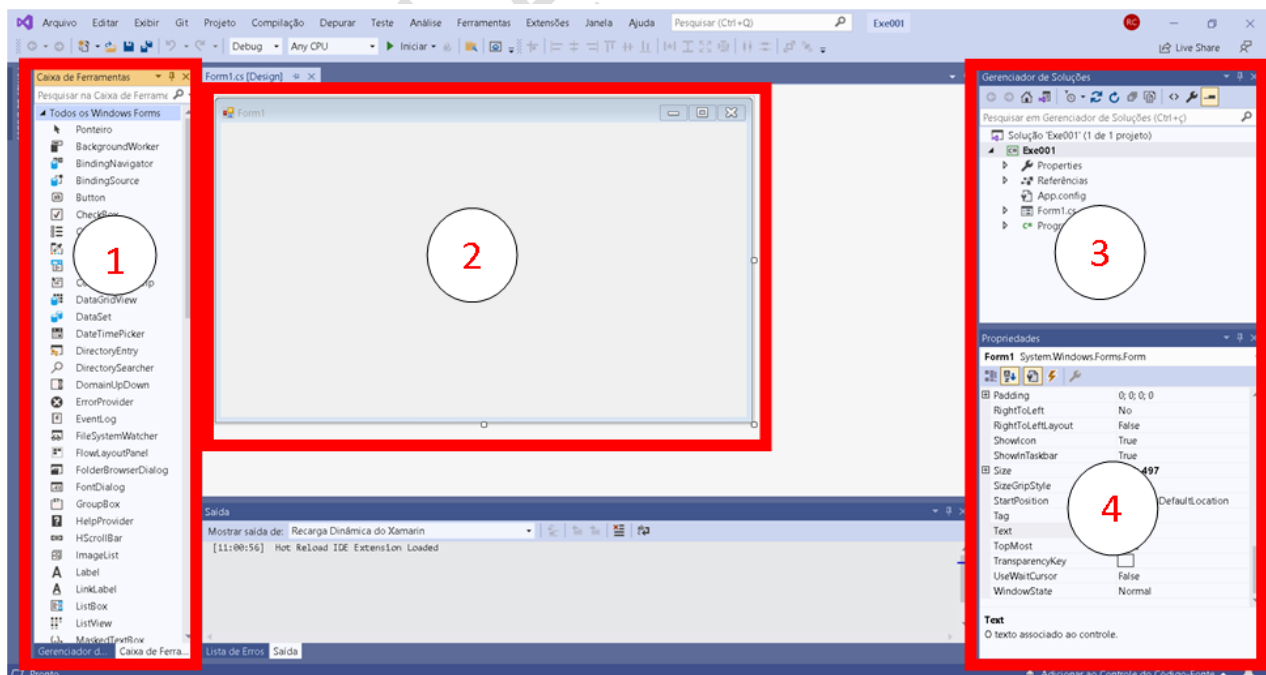
Voltar Criar

← Digite o nome do projeto: Exe001

← Clique nos “pontos” e informe o local de gravação.

Clique em “Criar”.

O Visual Studio exibirá o “ambiente de desenvolvimento”:



Prof. Roberto de Castro

No lado “esquerdo”, sinalizado com o número 1, temos a “ToolBox” (Caixa de Ferramentas). Caso ela não esteja visível em seu ambiente, clique no menu “View” (Exibir) → Toolbox (Caixa de Ferramentas).

Nesta “área” encontram-se os “diversos controles” (componentes visuais) que você poderá inserir em seu projeto, como por exemplo: Labels, Caixas de Texto, Botões, Menus....

Gradativamente, a cada projeto que iremos desenvolver, estudaremos novos controles (componentes visuais), ok?

Na “área central”, sinalizado com o número 2, temos o “Form → formulário”. Ele será o “palco” para o desenvolvimento de nossos projetos. Será nele que iremos “inserir” os controles (componentes visuais) para construir nossos projetos: Botões, Caixas de Texto.... O “Form” é o primeiro “contêiner” que estamos conhecendo.

Na área sinalizada com o número 3, temos a janela do “Gerenciador de Soluções” (Solution Explorer). Em resumo, esta “janela” contém “todos os arquivos” necessários para o desenvolvimento do projeto. Gradativamente iremos nos familiarizando com estas informações. Caso esta janela não esteja visível, dê em clique no menu “Exibir → Gerenciador de Soluções”.

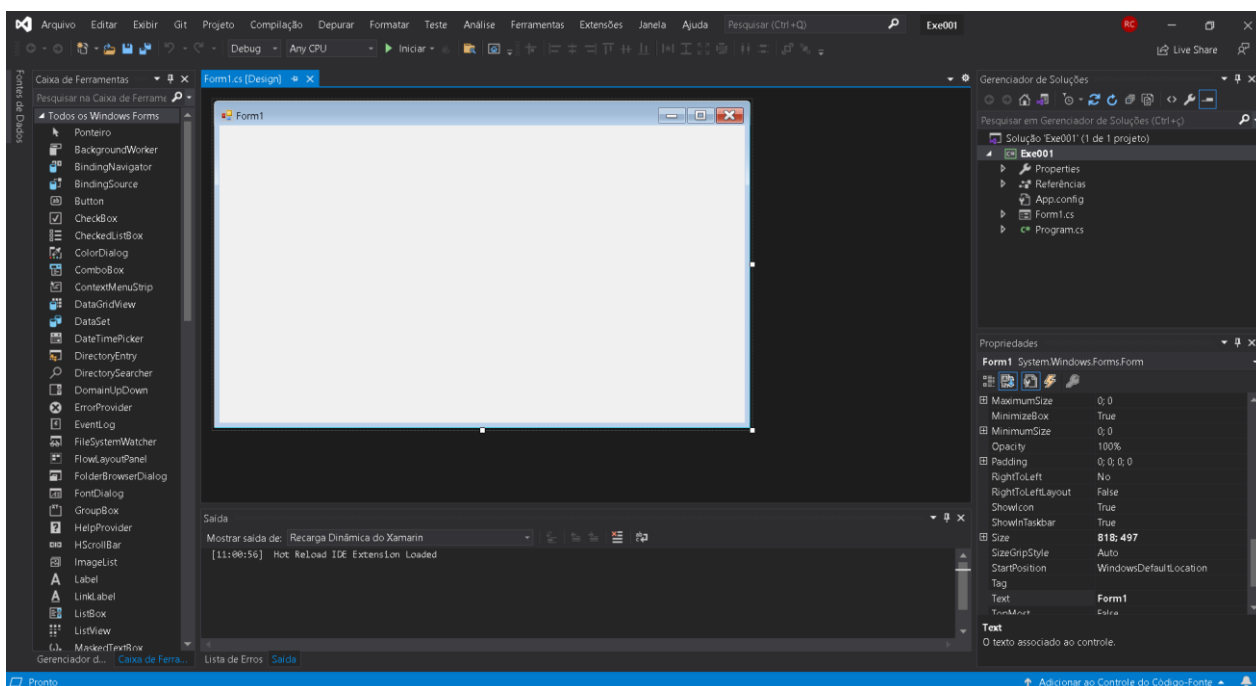
E finalmente, na área 4, temos a “janela de propriedades – properties”. As “propriedades” serão utilizadas, principalmente, para “personalizar” os controles (componentes visuais) que iremos inserir no formulário - projeto, como por exemplo: mudar uma cor, inserir uma borda, mudar a fonte, mudar a cor e o tamanho da fonte, entre outras opções que veremos ao longo do curso.

Personalizar o Visual Studio

Você poderá, opcionalmente e se desejar, personalizar a IDE do Visual Studio, como por exemplo, para alterar o tema para **Escuro**:

1. Na barra de menus, escolha **Ferramentas > Opções**.
2. Na página opções **gerais** do ambiente, altere a seleção de **tema de cores** para **escuro** e, em seguida, escolha **OK**.

Veja o resultado:

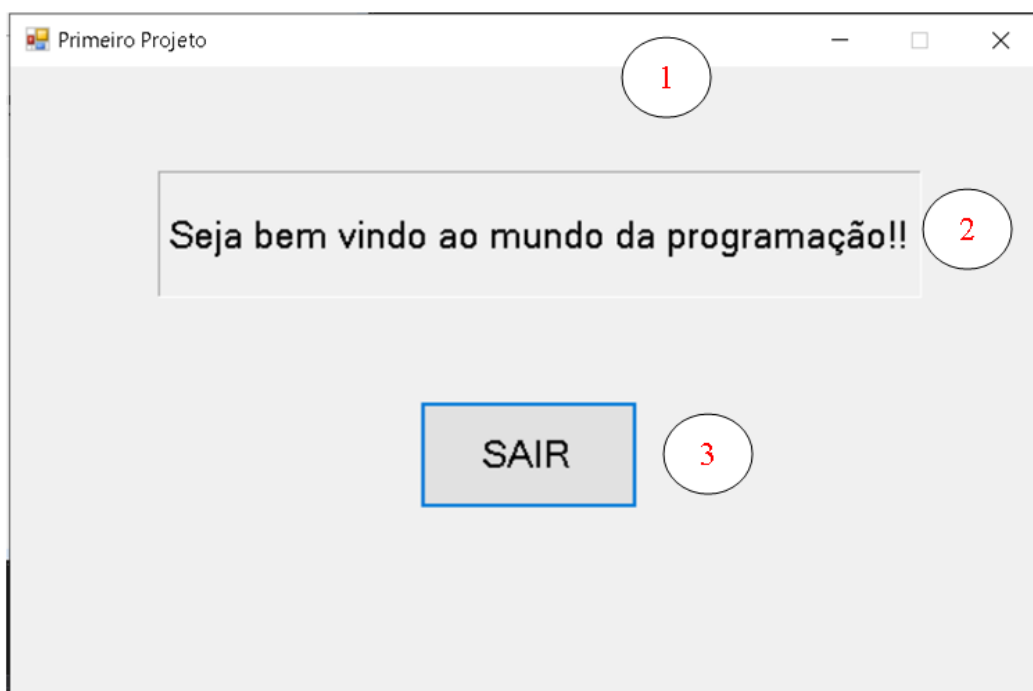


Criando o Primeiro Projeto

Em todas as nossas atividades seguiremos algumas etapas para o desenvolvimento dos projetos. São basicamente 4 (quatro) etapas:

- Primeira Etapa → Desenho da interface gráfica;
- Segunda Etapa → Configuração das propriedades e;
- Terceira Etapa → Programação dos “eventos”.
- Quarta etapa → Testes

Seguindo a metodologia acima, nosso primeiro projeto terá como aparência final, o modelo abaixo:



Vamos construir o projeto “passo-a-passo”:

Primeira etapa → Desenho da interface gráfica

1. Inicialmente temos, em nosso projeto “Windows Form Application” o próprio formulário (1). Será nele que iremos inserir todos os demais controles (componentes gráficos).
2. Dentro do formulário temos um controle, chamado “label” (2) que utilizamos para “exibir mensagens”.
3. Abaixo do label, temos o controle chamado “button” (3). Em regra, os “botões” sinalizam que “alguma tarefa será executada” quando o usuário do projeto clicar com o mouse. No caso em questão, quando o projeto estiver sendo executado, ao clicar no botão, o projeto será finalizado.

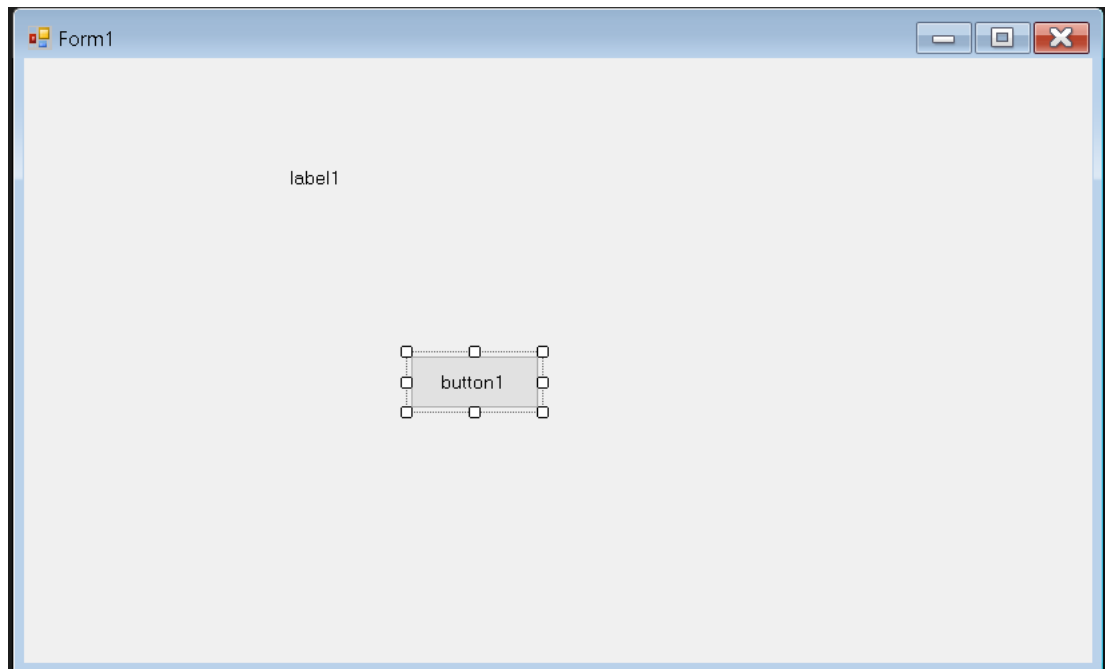
Após a identificação de cada componente visual (formulário / label / botão), devemos “inserir” estes componentes em nosso projeto. OBS: O formulário já existe. Precisamos inserir o label e o botão.

Prof. Roberto de Castro

Localize na caixa de ferramentas o controle chamado “Label”. Dê um clique, segure e arraste para o formulário.

Vamos realizar o mesmo procedimento com o botão. Localize na caixa de ferramentas o controle chamado “Button”. Dê um clique, segure e arraste para o formulário.

Neste momento seu “projeto” deve ter a aparência abaixo:



IMPORTANTE: O tamanho do controle poderá ser “alterado/ajustado” quando ele estiver selecionado (na imagem acima, o “button1” está selecionado). Para alterar o tamanho e/ou altura, basta fazer os ajustes com o mouse (arrastar, clicando nas bordas). No momento, o controle “label” NÃO poderá ser ajustado, pois está configurado como “ajuste automático”. Veremos como resolver este problema logo mais adiante.

Segunda Etapa → Configuração das propriedades

Até o momento inserimos no formulário os “controles” (componentes visuais) que serão necessários para o projeto (um label e um botão), porém, a “aparência” deles está diferente da nossa proposta inicial (o label não contém a mensagem de “Boas vindas” e o botão também não contém a mensagem de “sair”). Vamos fazer, agora, estas alterações, seguindo o “quadro abaixo”:

CONTROLE	PROPRIEDADE	CONTEÚDO
Form	Name	FrmExe001
	Text	Primeiro Projeto
Label	Name	LblMensagem
	AutoSize	False
	BorderStyle	Fixed3D
	Text	Seja bem vindo ao mundo da programação!!
Button	Name	BtnSair
	Text	Sair

Prof. Roberto de Castro

Para configurar os controles utilizaremos a “janela” de propriedades (é a janela que está no lado direito inferior).

Dê um único clique no formulário (pode ser, por exemplo, na área do título) para selecioná-lo.

A “caixa de propriedades”, neste momento, contém as propriedades padrão do formulário (componente que está em foco – selecionado).

Para este exemplo vamos alterar duas propriedades: a “Name” e a propriedade “Text” (seguindo as orientações do quadro acima).

A propriedade “Text” refere-se ao que estamos visualizando na barra de títulos do formulário, enquanto a propriedade “Name” é uma referência “interna” a este componente.

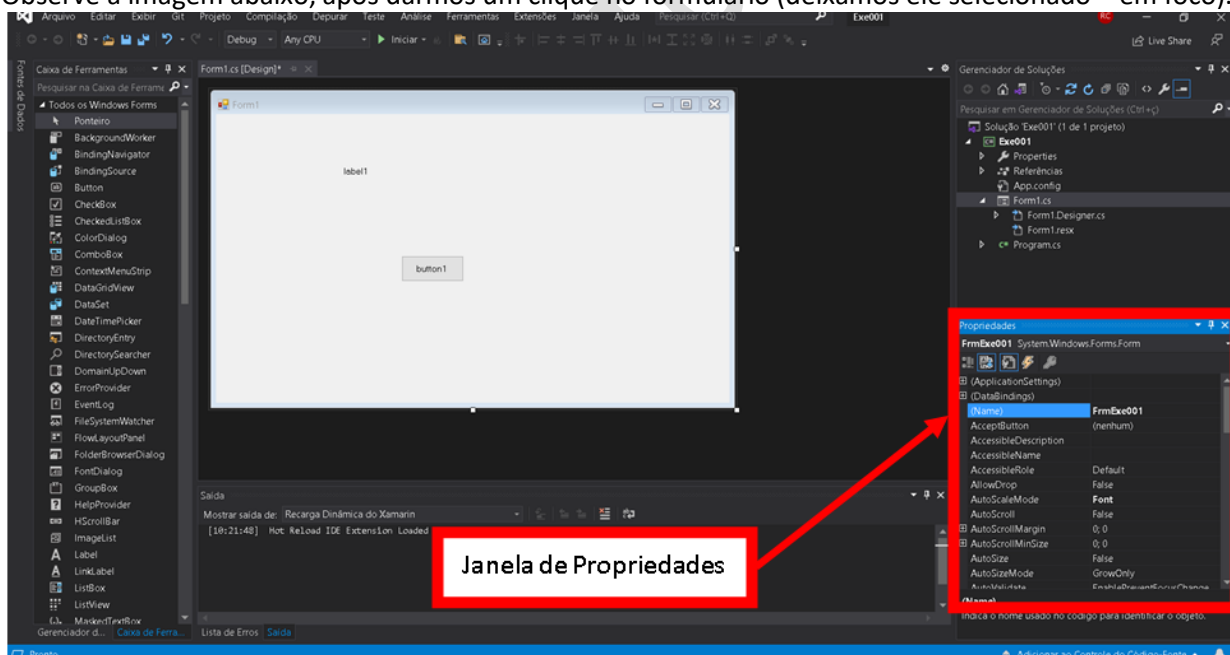
Em regra, cada item que consta no formulário (incluindo o próprio formulário) configuraremos a propriedade “Name”. Gradativamente estudaremos as justificativas para este procedimento, ok?

Também temos uma “pequena” regra para atribuir “nomes” aos controles.

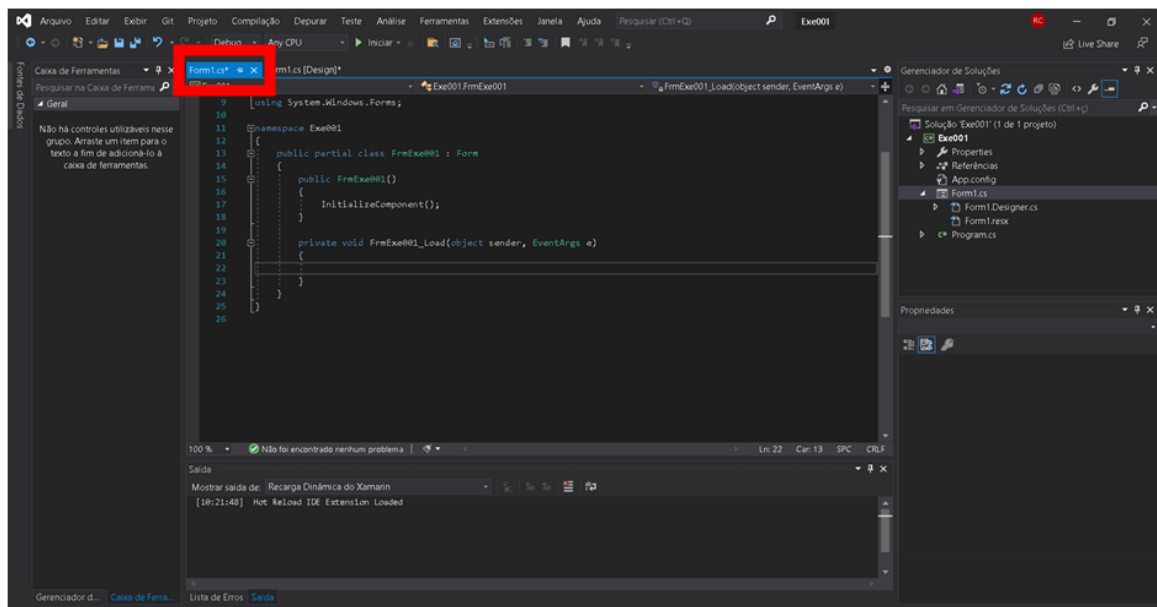
Os três primeiros caracteres do nome serão uma “abreviação” que indicará qual é o tipo de controle. A primeira letra escrita em “caixa alta” (maiúsculo). Na sequência, o próximo caractere também será escrito em caixa alta (maiúsculo) e será um nome de “livre escolha” do programador (ou da instituição onde trabalha). A boa prática sugere sempre a utilização de nomes “sugestivos” que indiquem seu conteúdo/finalidade.

IMPORTANTE: Não podemos utilizar nomes contendo espaços. Evite também nomes extensos.

Observe a imagem abaixo, após darmos um clique no formulário (deixamos ele selecionado – em foco).



DICA: Caso você não esteja visualizando a imagem acima, mas a que está sendo exibida abaixo:



Isto indica que você “saiu” do modo de design e foi para o modo de “código” - programação (isto ocorreu, provavelmente, devido a dois cliques no formulário).

Sem problemas....

Para retornar ao modo de design basta dar um clique na “aba” → Form1.cs Design (destacada na figura acima), ou clicar no Menu Exibir → Design.

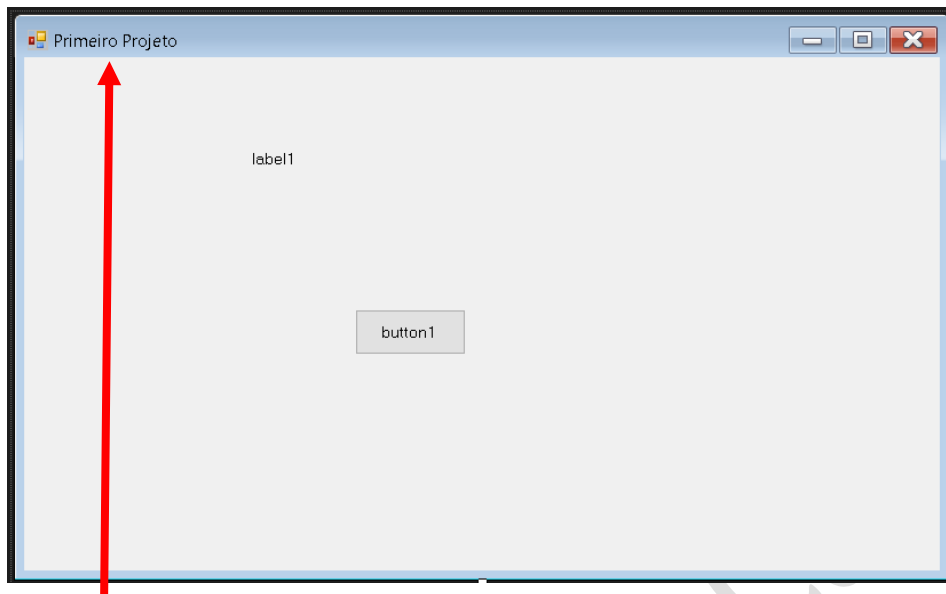
Resolvida a questão, vamos continuar na tarefa de configuração das propriedades...

A propriedade “Name” está na terceira linha da janela de propriedades (todas as demais propriedades estão em ordem alfabética). Excluímos o texto que estava lá e digitamos “FrmExe001” (que significa: Formulário do Exercício 001). Dê um <Enter> ao final da digitação.

Após digitar o nome “FrmExe001” vamos alterar a propriedade “Text”. Utilize a barra de rolagem vertical para chegar até a propriedade Text.

Exclua o texto que está descrito lá e digite o texto (seguindo a orientação do quadro: “Primeiro Projeto” (dê um <ENTER>).

Neste momento o seu projeto deverá ter a aparência abaixo:



Observe que a área de título do formulário agora exibe a mensagem "Primeiro Projeto". Isto ocorre pois alteramos a propriedade "text" do formulário. A propriedade "name" não causa nenhuma alteração visual, mas veremos com mais detalhes quando iniciarmos a etapa de programação.

Vamos agora alterar as propriedades "Name", "AutoSize", "BorderStyle" e "Text" do "Label", para isto devemos "dar um clique no label" (deixar ele em foco) e localizar as propriedades na caixa de propriedades.

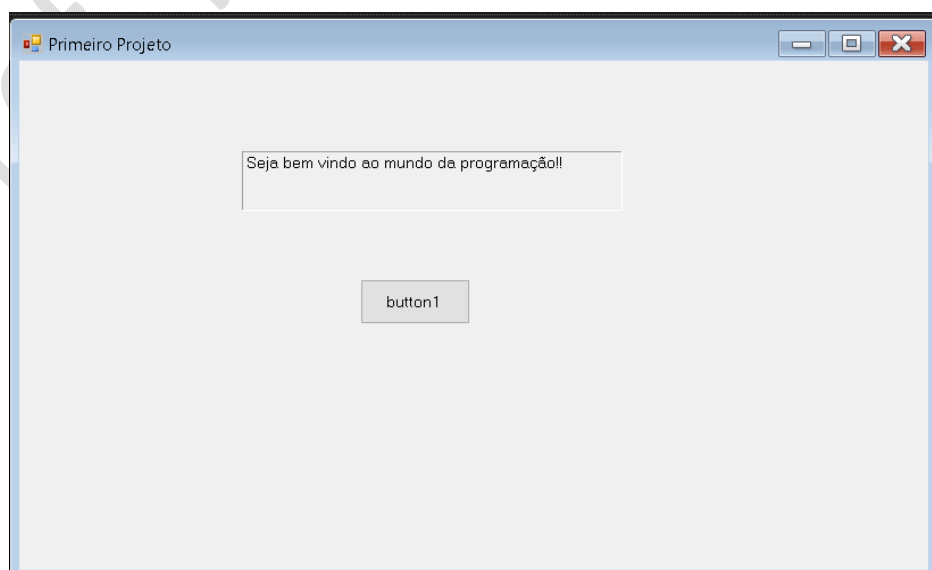
O nome do label será "LblMensagem".

A propriedade "AutoSize" está pré-configurada, por padrão (default), como TRUE (que significa que o Label assumirá o tamanho necessário que o "texto" exigir - ajuste automático. É por esta razão que não conseguimos ajustar o tamanho do Label da mesma forma que ajustamos o Botão). Altere a opção para FALSE. A partir deste momento você conseguirá alterar/ajustar o tamanho do Label com o mouse...

Faremos agora a alteração da propriedade "BorderStyle", que por padrão vem "None" (que significa Sem Borda). Vamos selecionar a opção "Fixed3D" da propriedade "BorderStyle".

E por último, a propriedade "Text" do Label. Digite: Seja bem vindo ao mundo da programação!!

O seu projeto, neste momento, deverá ter a aparência abaixo:

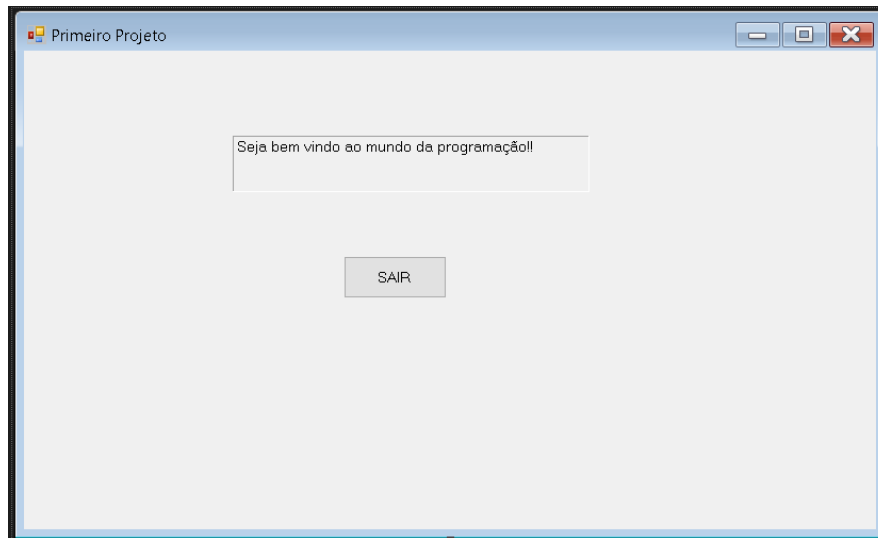


Prof. Roberto de Castro

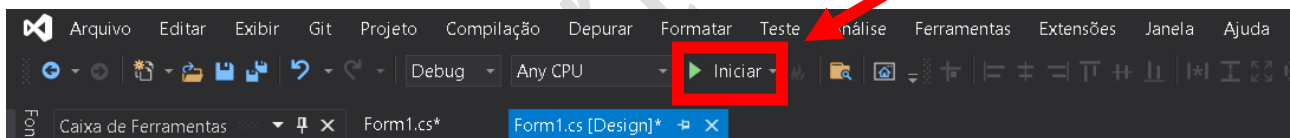
Vamos agora para os últimos ajustes visuais, desta segunda etapa de desenvolvimento do projeto: A configuração da propriedade “Name” e “Text” do Botão, para isto devemos “dar um clique no botão” (deixar ele em foco) e localizar as propriedades na caixa de propriedades.

O “Name” do botão será “BtnSair” e a propriedade “text” será “Sair”.

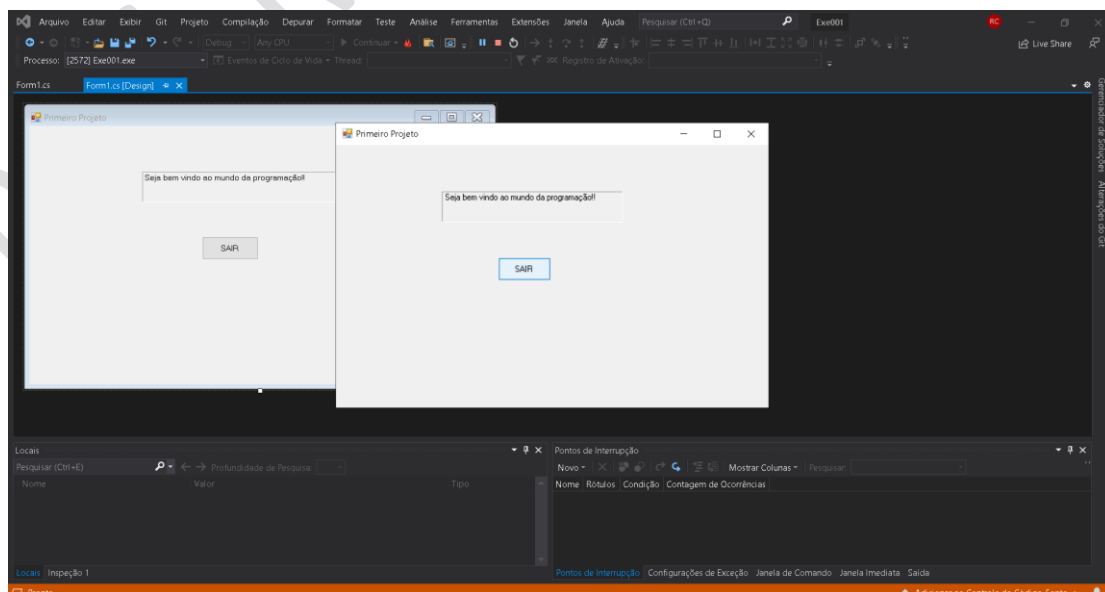
O seu projeto deverá ter a aparência abaixo:



Apesar de não incluirmos, até o momento, nenhuma programação (o botão SAIR não tem nenhuma funcionalidade), nosso projeto poderá ser “testado”... Dê um clique em “Iniciar”:



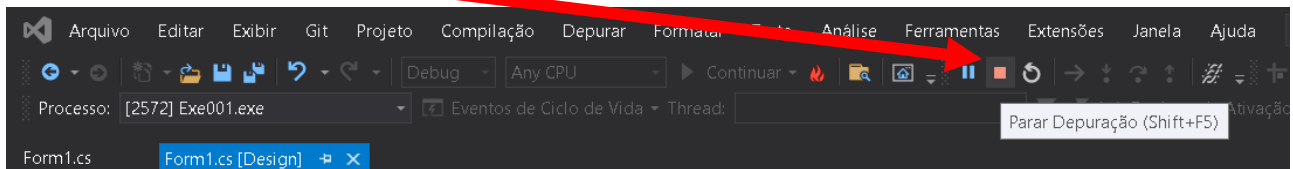
O seu projeto será “compilado” e será executado (se não apresentar nenhum tipo de erro...). O resultado será exibido:



Prof. Roberto de Castro

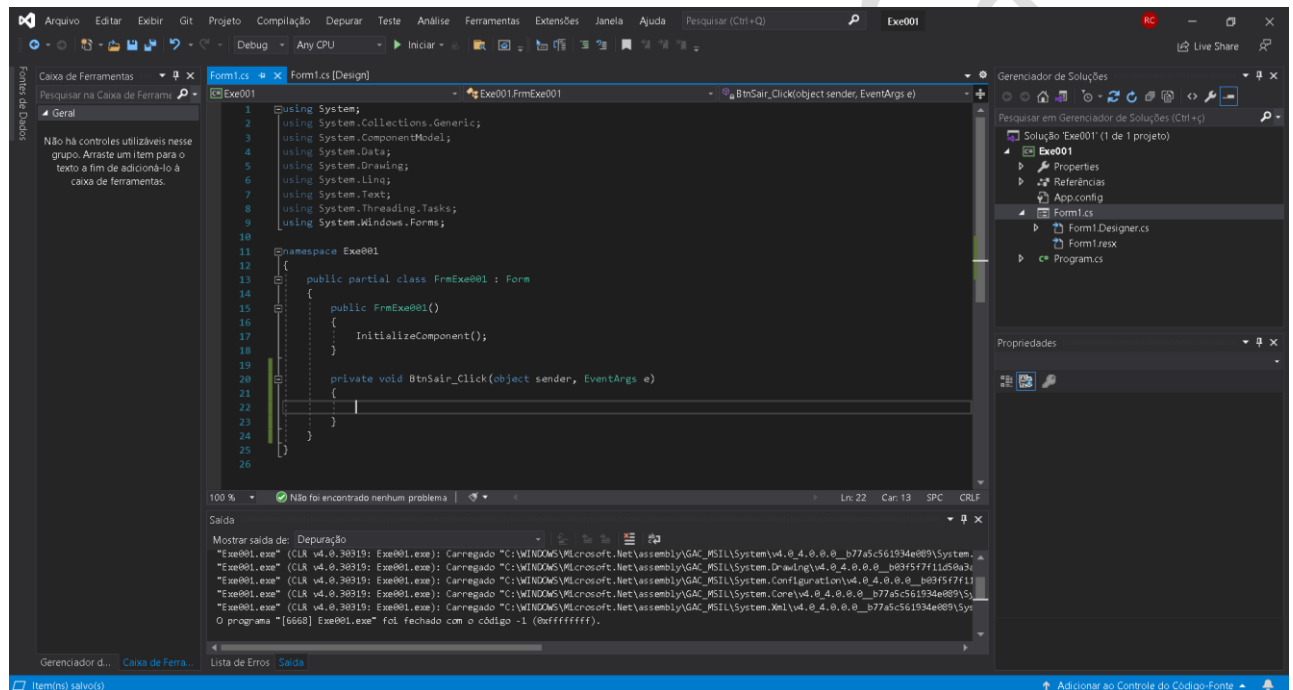
Se desejar, para uma melhor visualização do resultado, poderá minimizar o Visual Studio que está atrás e deixar somente o “formulário”.

Lembre-se: O botão SAIR até o momento não tem nenhuma funcionalidade. Para encerrar o seu projeto, deverá clicar no “PARAR”:



Agora vamos para a terceira etapa de desenvolvimento do projeto: Programação dos Eventos.

Dê “dois” cliques no “botão Sair”. Isto sinaliza que iremos sair do “ambiente de design” e vamos para o “ambiente de programação”. A imagem abaixo ilustra o “cenário atual”:



Mesmo sem escrever nenhuma linha de programação até o momento, nosso programa já apresenta alguns comandos.

As linhas de 1 a 9 foram inseridas “automaticamente” pelo Visual Studio. Estas linhas estão sinalizando que poderemos utilizar vários comandos em nosso projeto e que o Visual Studio já está pronto para recebê-los. Vamos manter assim por enquanto.

DICA: Ao finalizar o projeto em desenvolvimento, os “using” que não foram necessários, estarão “opacos”, sinalizando que poderão ser excluídos sem problema... Mas se desejar, poderá manter todos, pois não causarão nenhum tipo de problema.

A linha 11 indica o nome do projeto: Exe001.

Vamos focar agora com o que realmente nos interessa: a programação do botão SAIR.

Ao darmos dois cliques no botão Sair, o Visual Studio criou uma área (chamamos de método) na qual iremos digitar os comandos que serão executados pelo botão Sair:

```
private void BtnSair_Click(object sender, EventArgs e)
{

}
}
```

O entendimento completo deste cabeçalho “private void BtnSair_Click...” será abordado gradativamente ao longo do curso.

Para o momento, resumidamente, devemos entender que:

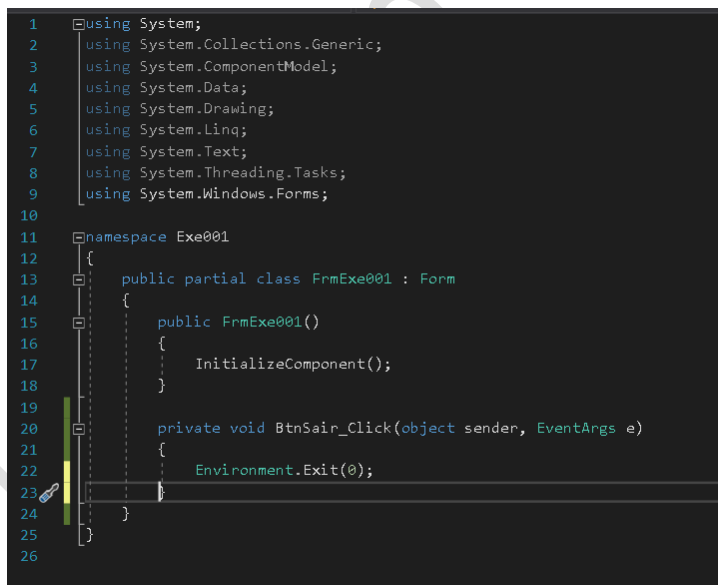
1. Iremos inserir comandos para o controle “BtnSair” (lembra do name dado botão sair ?????, pois é, aí está!!!). Os “parênteses” sinalizam/delimitam o início e o término do bloco de comandos. Na medida em que vamos digitando comandos, esta delimitação (início/término) vai aumentando automaticamente;
2. que serão executados quando o botão for “clicado” (evento Click).

Em regra, o evento “Click” para o “botão” é o mais utilizado, mas veremos alguns outros eventos mais à frente.

O que desejamos, no momento, é finalizar o programa quando o usuário clicar no botão sair for. O comando que “finaliza um programa” → Application.Exit();

A partir deste ponto, muita atenção na digitação se faz necessária, pois um pequeno erro na digitação, poderá impedir a execução do programa e causar erros que poderiam ser evitados. Observe o “ponto e vírgula” no final (é obrigatório).

O seu trecho de código deverá estar conforme modelo abaixo:



```
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Text;
8  using System.Threading.Tasks;
9  using System.Windows.Forms;
10
11 namespace Exe001
12 {
13     public partial class FrmExe001 : Form
14     {
15         public FrmExe001()
16         {
17             InitializeComponent();
18         }
19
20         private void BtnSair_Click(object sender, EventArgs e)
21         {
22             Environment.Exit(0);
23         }
24     }
25 }
26
```

Nota de alerta: Caso o seu projeto apresente outros cabeçalhos do tipo “private void.....”, provavelmente isto deve ter ocorrido após você ter clicado duas vezes em algum controle (componente visual) que está no formulário. **NÃO TENTE EXCLUI-LO**, pois isto gerará um erro e o seu projeto apresentará um grave erro, por esta razão, mantenha o “cabeçalho” no ambiente de desenvolvimento, pois não gerará nenhum tipo de problema... Assim que avançarmos mais um pouco com o desenvolvimento de projetos, veremos como resolver o problema, ok? Vamos continuar...

Podemos testar novamente o projeto. Clique em Iniciar.

Quando o projeto for iniciado, caso desejar finalizar, basta dar um clique no botão SAIR, que agora possui uma funcionalidade: sair do projeto.

Prof. Roberto de Castro

Pronto, você acabou de ser apresentado ao primeiro comando da linguagem C# (pronuncia-se “C Sharp”), que é a uma das linguagens de programação da Plataforma .NET, derivada de C/C++ orientada a objetos, vem ganhando inúmeros adeptos (programadores de outras linguagens) em virtude de sua característica e semelhança ao C, C++ ou Java. Orientada a Objetos, foi criada praticamente a partir do zero, pelo engenheiro de software Anders Hejlsberg, desenvolvedor do compilador Pascal e do Delphi.

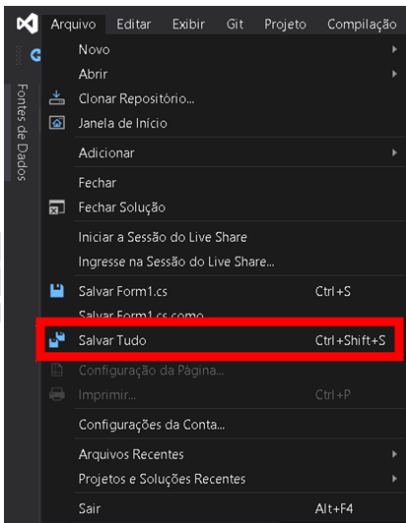
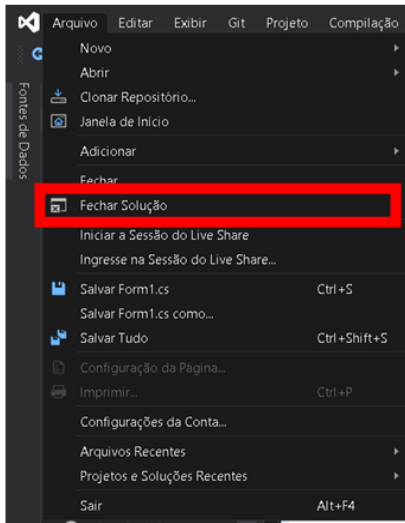
Características do C#

- Simplicidade: os projetistas de C# costumam dizer que essa linguagem é tão poderosa quanto o C++ e tão simples quanto o Visual Basic;
- Completamente orientada a objetos;
- Fortemente tipada: isso ajudará a evitar erros por manipulação imprópria de tipos, atribuições incorretas.

Neste momento você poderá fazer algumas implementações visuais no seu projeto:

- Alterar o tamanho da fonte do Label (ver propriedade Font).
- Fazer o alinhamento do texto dentro do Label (ver propriedade TextAlign).
- Alterar a cor da “fonte” do Label (ver propriedade ForeColor).
- Alterar a cor de fundo do Label (ver propriedade BackColor).
- Desabilitar o botão de “maximizar” do formulário (ver propriedade MaximizeBox → tornar como False).
- Quando for executado, exibir o formulário CENTRALIZADO na tela do Windows (ver propriedade StartPosition → a alterar para: CenterScreen).

IMPORTANTE: Todo o trabalho feito até o momento poderá ser perdido se por acaso faltar energia e o computador ser desligado, pois não SALVAMOS o projeto em nenhuma etapa (e isto é um GRANDE risco).

Clique na opção: Arquivo → Salvar Tudo	Pronto, seu projeto poderá agora ser “Fechado”. Clique em Arquivo → Fechar Solução.
	

Sair do Visual Studio...