

[◀ VOLTAR](#)

# A linguagem C#

Apresentar os conceitos da Linguagem C#. Conceituar diretiva using, namespaces e algumas regras da linguagem C#

## NESTE TÓPICO

- > Introdução aos Conceitos da Linguagem C#
- > Principais Elementos de um Programa em C#
- > Namespace em C# - Marcar (Espaços de nomes) tópico



## Introdução aos Conceitos da Linguagem C#

O C# (pronuncia-se *Cê Sharp* ou *See Sharp*) é uma linguagem de programação simples, moderna, poderosa, orientada a objeto, criada pela Microsoft para os desenvolvedores criarem aplicações utilizando o .NET Framework.

O C# é uma evolução das linguagens C e C++ que além de utilizar muitas características da linguagem C++, possui um mecanismo denominado *Garbage Collector* (Coletor de Lixo) o qual gerencia a memória utilizada pelas aplicações de forma automática .

O C# fornece os recursos que são mais importantes para os programadores, como programação orientada a objeto, strings, elementos gráficos, componentes de interface gráfica com o usuário (GUI), tratamento de exceções, múltiplas linhas de execução, multimídia (áudio, imagens, animação e vídeo), processamento de arquivos, estruturas de dados pré-empacotadas, processamento de banco de dados, redes cliente/servidor com base na Internet e na World Wide Web e computação distribuída.

A linguagem C# foi projetada especificamente para a plataforma .NET como uma linguagem que permite aos programadores migrarem facilmente devido ter suas raízes em C, C++ e Java.

O C# é uma linguagem visual orientada a eventos e totalmente orientada a objeto. Sendo uma linguagem orientada a objeto, ela suporta conceitos como encapsulamento, herança e polimorfismo. Todas as variáveis e métodos, incluindo o método principal (**Main**), o ponto de execução de uma aplicação, são encapsulados em definições de classes.

Como uma linguagem de programação orientada a eventos, com o C# você escreverá programas que respondem aos eventos do usuário como por exemplo, quando o usuário clicar em um botão ou escrever um texto em uma caixa de texto respondendo aos eventos do teclado.

Os programas desenvolvidos em C# são criados em um IDE (*Integrated Development Environment* – Ambiente de Desenvolvimento Integrado) sendo o Visual Studio da Microsoft o mais utilizado. Ele é uma suíte (pacote) de programas para criação e desenvolvimento de software do .Net Framework.

O IDE possibilita ao programador, criar, executar, testar e depurar programas C#, reduzindo o tempo necessário para produzir um programa funcional rapidamente.

Podemos criar vários tipos de projetos em C#, como por exemplo: aplicações para Console, aplicações gráficas para Windows (este tipo de aplicação é o que vamos abordar), aplicações para Web.

## Principais Elementos de um Programa em C#

Um programa desenvolvido em C# contém quatro elementos principais: pelo menos uma declaração de **namespace**, uma **classe**, um único método **Main**, e uma declaração de **programa**. Os arquivos de código-fonte dos programas tem a extensão **.cs**.

Depois de criar os seus arquivos no Visual Studio, você precisa compilar esses arquivos. Quando você compila os seus programas, o compilador gera um arquivo executável, com extensão **.exe**. Também existem outros tipos de compilação, como por exemplo, quando criamos uma biblioteca **.dll** (**Dynamic Link Library** – **Biblioteca de ligação dinâmica**).

Mas não se preocupe porque o IDE do Visual Studio facilita todo o desenvolvimento, cria códigos, compila os programas, mas você ainda terá que escrever os seus códigos e a lógica do desenvolvimento dos seus programas.

## Namespace em C# - (Espaços de nomes)

Um **namespace** agrupa vários recursos do C# em categorias relacionadas. A plataforma **.NET** (**.NET Framework**) fornece um rico conjunto de **namespaces** para serem usados com a Linguagem C#. Esses namespaces contém códigos que podem ser reutilizados tornando a programação mais fácil e rápida.

O C# contém muitas classes predefinidas e agrupadas em **namespace** (espaços de nomes) que estão no **.NET Framework Class Library**, ou seja, são as bibliotecas de classes da plataforma .NET.

## Diretiva using

A **diretiva using** é utilizada para especificar os **namespaces** que usamos em cada programa. Veja o exemplo da seguinte declaração:

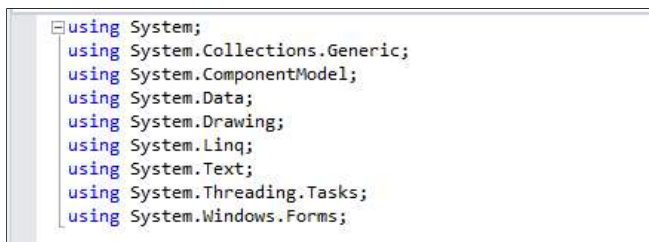
```
1. using System;
```

Essa declaração informa ao compilador que estamos utilizando o **namespace System**, que nos permite, por exemplo, escrever a instrução de forma abreviada **Console.WriteLine**, em vez de **System.Console.WriteLine**, em todo o programa.

A instrução **Console.WriteLine** é utilizada para que o programa apresente (ou imprima) uma linha de texto na saída padrão do computador. A classe **Console** fornece o métodos que permitem aos programas C# apresentarem strings (textos) ou outros tipos de informações no **prompt** de comando do Windows, por exemplo.

Quando você estiver desenvolvendo os seus aplicativos, o Visual Studio adicionará algumas referencias comuns quando você criar um projeto.

A figura a seguir apresenta alguns **namespaces** adicionados automaticamente em um projeto.

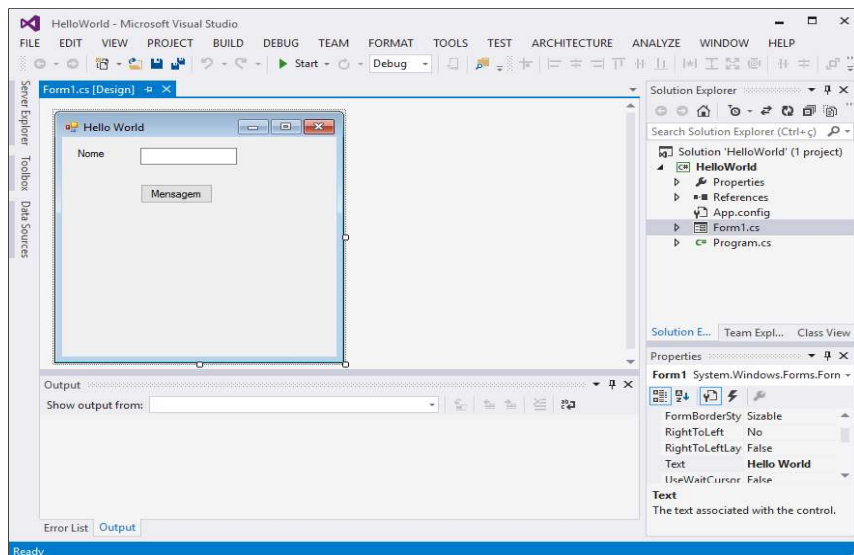


```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
using System.Windows.Forms;
```

Como vamos desenvolver aplicativos que utilizam interface gráfica, o **namespace System.Windows.Forms** contém muitas classes que ajudam os desenvolvedores de C# a definirem as GUIs (Graphical User interface - Interface Gráfica do Usuário) para seus aplicativos.

Os componentes de GUI, como por exemplo, botões, , caixa de texto, facilitam a entrada de dados por parte do usuário e a formatação ou apresentação de saída de dados para o usuário.

A figura a seguir apresenta a Interface Gráfica do IDE Visual Studio.



## Algumas Regras da Linguagem C#

A linguagem C# é case sensitive, ou seja, o C# diferencia letras maiúsculas de minúsculas. Portanto, **Total** é diferente de **total**, que é diferente de **toTal**.

Um *par de chaves* ( { } ) delimita o corpo de toda definição de classe ou método.

As instruções (comandos) devem terminar com *ponto-e-vírgula* ( ; ).

## Quiz

Exercício

A linguagem C#

INICIAR ➤

## Referências

DEITEL, H. M. *et al.* **C# Como Programar**. São Paulo: Pearson Makron Books, 2003.

OLSSON, Mikael. *C# Quick Syntax Reference*. Berkeley, CA: Apress, 2013.

SHARP, John. *Microsoft Visual C# 2013 Step by Step*. Microsoft Press, 2013.



Avalie este tópico



ANTERIOR

Conhecendo a plataforma .NET

Biblioteca  
(https://www.uninove.br/conhecendo-a-uninove/biblioteca/sobre-a-biblioteca/apresentacao/) Portal Uninove  
(http://www.uninove.br) Mapa do Site



Estrutura de um programa em Visual Studio

© Todos os direitos reservados

Ajuda?  
PRÓXIMO  
(https://ava.uninove.br/curso=) >