**C# e .NET**

**O que é C# (C SHARP)?**

É uma linguagem de programação

**O que é .NET?**

(DOTNET) é uma plataforma de desenvolvimento para criar diversos tipos de aplicações, utilizando algumas linguagens de programação, tais como, C# (C SHARP), VB (VISUAL BASIC) e F# (F SHARP).

**O que tem no .NET?**

* **BCL – BASE CLASS LIBRARY**

Nessa biblioteca tem um conjunto de funcionalidades para utilizar nos nossos programas.

* **CLR – COMMON LANGUAGE RUNTIME (MÁQUINA VIRTUAL)**

É uma máquina virtual que executa os programas realizados em .net, possui também o garbage collection é um processo de coletar os objetos não utilizados e são automaticamente desalocados da memória.

**Qual a diferença de .NET Framework, .NET Core e Xamarin?**

.Net Framework é usado para desenvolvimento de aplicações Windows usando o Windows Forms (Sistema de telas), WPF (Sistema de telas) e de Aplicações Web usando ASP .NET MVC.

.NET Core é usado para criar aplicações Web para diferentes tipos de sistemas operacionais (Windows, Linux e Mac).

Xamarin é um framework que permite criar aplicações nativas (As aplicações nativas são desenvolvidas para um sistema específico (Android, iOS, Windows) e são descarregadas para um dispositivo móvel a partir de uma loja de aplicações.) para Android, IOS.

**Compilação e interpretação**

O C# é uma linguagem pré-compilada que executa em uma máquina virtual. O programador escreve o código que é pré-compilado, gerado um bytecode (Common Intermediate Language (CIL)) que é rodado em uma máquina virtual, depois disso é convertido em código de máquina que é executado no computador.

**Como é a estrutura de uma aplicação em C#?**

Uma aplicação construída em uma linguagem orientada a objeto como o C#, é formada por um conjunto de classes, essas classes podem ser agrupadas em Namespace **(Agrupamento lógico de classes relacionadas)**, em uma aplicação pode ter várias classes que vão representar as entidades de negócio (Cliente, produtos, etc.), elas ficam agrupadas em um Namespace chamado Entities (Entidades), da mesma forma pode ter outras classes, como, serviços do sistema, que ficam organizados em um Namespace chamado services e a classe repositories que armazena os dados.

1. **Using System;**

É a biblioteca onde tem funções básicas de manipulações de programas.

1. **Namespace;**

É onde as classes ficam organizadas, não é obrigatório o uso, porém por questão de organização é recomendado deixar nossas classes dentro do namespace.

1. **Class program;**

Todo código fonte escrito em c# tem que estar dentro de uma classe.

1. **Static void Main (string [] args);**

É uma declaração padrão em c#, para identificar o entre point da aplicação, ou seja, o ponto de entrada, onde a execução começa.

**Dicas do Visual Studio**

Indentação automática: CTRL + K + D

Comentar a linha toda: CTRL + K + C

**A tabela a seguir lista os tipos de valores internos do C#:**

**TIPOS INTERNOS (REFERÊNCIA C#)**

|  |  |
| --- | --- |
| Palavra-chave Type do C# | Tipo .NET |
| bool | System.Boolean |
| byte | System.Byte |
| sbyte | System.SByte |
| char | System.Char |
| decimal | System.Decimal |
| double | System.Double |
| float | System.Single |
| int | System.Int32 |
| uint | System.UInt32 |
| nint | System.IntPtr |
| nuint | System.UIntPtr |
| long | System.Int64 |
| ulong | System.UInt64 |
| short | System.Int16 |
| ushort | System.UInt16 |