Curso Técnico

de Programação de Jogos Digitais

Aula 02

índice

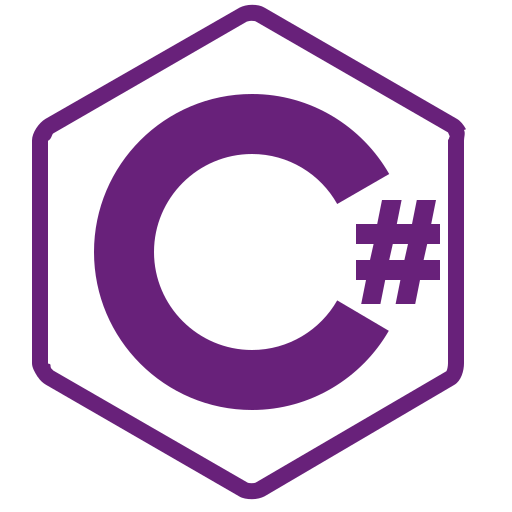
[1. C# 3](#_Toc169071666)

[2. Variáveis e Tipos de Dados em C# 3](#_Toc169071667)

[3. Declaração de variaveis 4](#_Toc169071668)

[4. Referencias 6](#_Toc169071669)

# C#

C# (pronuncia-se "C-sharp") é uma linguagem de programação desenvolvida pela Microsoft como parte da sua plataforma .NET. Projetada para ser simples, robusta e segura, C# combina a expressividade do C++ com a facilidade de uso do Visual Basic, tornando-se uma escolha popular tanto para desenvolvedores iniciantes quanto experientes. Sua sintaxe clara e estruturada facilita a escrita, leitura e manutenção do código, promovendo práticas de desenvolvimento eficientes e escaláveis. Um dos fundamentos essenciais na programação em C# é o uso de variáveis, que são utilizadas para armazenar dados temporários durante a execução de um programa.

# Variáveis e Tipos de Dados em C#

Uma variável é um espaço na memória do computador que armazena um valor que pode ser usado em um programa. Uma variável tem um nome, que serve para identificá-la e acessá-la, e um tipo, que define o tipo de valor que ela pode armazenar e as operações que podem ser feitas com ela. O tipo de uma variável também determina o tamanho da memória que ela ocupa.

Em C#, existem dois tipos principais de variáveis: as variáveis de valor e as variáveis de referência. As variáveis de valor armazenam diretamente o valor na memória, enquanto as variáveis de referência armazenam um endereço de memória que aponta para o valor, que está armazenado em outra parte da memória. As variáveis de valor são mais rápidas e eficientes, mas têm um tamanho fixo e limitado. As variáveis de referência são mais flexíveis e podem armazenar valores de tamanho variável, mas são mais lentas e consomem mais memória.



O nome "C#" (pronuncia-se "C-sharp") é derivado da notação musical. Na música, o símbolo "#" representa um "sustenido", que indica uma nota que é meio tom acima da nota natural. De maneira semelhante, o nome C# sugere uma evolução ou um passo à frente em relação à linguagem C.

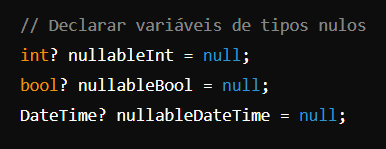
Os tipos de dados são as categorias que definem os valores que podem ser armazenados em uma variável e as operações que podem ser feitas com eles. Em C#, existem vários tipos de dados pré-definidos, que podem ser divididos em três grupos: os tipos simples, os tipos complexos e os tipos especiais.

Os tipos simples são os tipos básicos que representam valores numéricos, lógicos ou textuais. Eles são subdivididos em dois subgrupos: os tipos numéricos e os tipos não numéricos. Os tipos numéricos são os tipos que representam valores numéricos inteiros ou reais, como byte, short, int, long, float, double e decimal. Eles podem ser usados para fazer operações aritméticas, como soma, subtração, multiplicação, divisão e resto. Os tipos não numéricos são os tipos que representam valores lógicos ou textuais, como bool, char e string. Eles podem ser usados para fazer operações lógicas, como comparação e negação, ou operações textuais, como concatenação e extração.

Os tipos complexos são os tipos que representam valores estruturados ou coleções de valores. Eles são subdivididos em dois subgrupos: os tipos estruturados e os tipos coleções. Os tipos estruturados são os tipos que representam valores compostos por vários campos ou propriedades, como struct, enum e class. Eles podem ser usados para criar novos tipos personalizados, com características específicas e comportamentos definidos por métodos. Os tipos coleções são os tipos que representam conjuntos ou sequências de valores, como array, list, dictionary e queue. Eles podem ser usados para armazenar vários valores do mesmo tipo ou de tipos diferentes, e para fazer operações como adicionar, remover, pesquisar e ordenar.



Você sabia que o coletor de lixo (Garbage Collector, GC) do C# é um mecanismo automático de gerenciamento de memória ? Ele otimiza recursos, previne vazamentos de memoria e monitora os seus objetos , identificando e liberando os que não são mais acessíveis.

Os tipos especiais são os tipos que representam valores especiais ou nulos. Eles são subdivididos em dois subgrupos: os tipos nulos e os tipos anônimos. Os tipos nulos são os tipos que permitem armazenar o valor null em uma variável, indicando a ausência de valor ou a referência inválida. Eles são formados pelo símbolo ? após o tipo original, como int?, bool? ou string?.

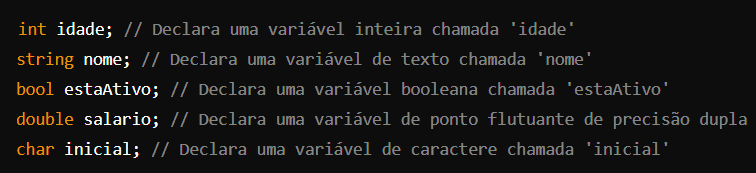
Eles podem ser usados para verificar se uma variável tem ou não um valor atribuído, usando o operador ?? ou o método HasValue. Os tipos anônimos são os tipos que permitem criar objetos sem especificar o tipo explicitamente, usando a palavra-chave var ou a sintaxe new { }. Eles podem ser usados para criar objetos temporários ou simplificados, sem precisar definir uma classe.

# Declaração de variaveis

Para declarar uma variável em C#, é preciso seguir a seguinte sintaxe:

tipo nome = valor;

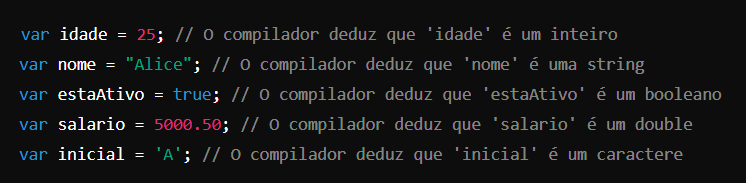
Onde tipo é o tipo de dado da variável, nome é o identificador da variável e valor é o valor inicial da variável. O sinal = indica a atribuição do valor à variável. O ponto-e-vírgula indica o fim da declaração.



Por exemplo:

int x = 10; // declara uma variável de valor do tipo int com o nome x e o valor 10

string y = "Olá";



# Referencias

1. Unity Technologies. (2020). **Unity User Manual.** Unity Technologies.
2. [C#, 2020] **Visual C# Developer Center**, Microsoft Docs.

Microsoft Corporation. (2020).

1. Geig, M. (2018). **Unity 2018 Game Development in 24 Hours**, Sams Teach Yourself. Sams Publishing.
2. Hocking, J. (2015). **Unity in Action: Multiplatform Game Development in C#.** Manning Publications.