**C#**

**Fundamentos**

**Variáveis e Constantes**

Para criar uma variável em C# primeiro se passa o tipo da variável,o nome dela e por último o valor da variável, exemplo:

double raio = 4.5;

Aqui foi criado uma variável do tipo “double” com o nome “raio” com o valor “4.5”.

Para criar uma constante, basta colocar a palavra “const” antes do tipo, que fara com que a variável nunca possa muda o seu valor, exemplo:

const double PI = 3.14;

Aqui foi criado a variável “PI” do tipo “double” sendo uma constante, com o valor “3.14”.

Para imprimir o valor da variável no console, basta usar:

Console.Write(raio);

Ou:

Console.WriteLine(raio);

A diferença é que o “Console.Write” imprime o valor e continua na mesma linha, e o “Console.WriteLine” imprime o valor e pula para a linha de baixo, e para usar dessa forma precisa “importar” o “System” no programa usando:

using System;

No começo do código, para que não precisa escrever “System.Console.Write“ ou “System.Console.WriteLine” toda vez.

**Tipos numéricos:**

**Inteiros:**

byte -> Inteiro de 8 bits sem sinal

sbyte -> Inteiro de 8 bits com sinal

short -> Inteiro de 16 bits com sinal

ushort -> Inteiro de 16 bits sem sinal

int -> Inteiro assinado de 32 bits

uint -> Inteiro assinado de 32 bits sem sinal

long -> Inteiro assinado de 64 bits

ulong -> Inteiro assinado de 64 bits sem sinal

byte -> Não aceita valor negativo, entre 0 até +255

sbyte -> Entre -128 até +127

short -> Entre -32.768 até +32.767

ushort -> Entre 0 até +65.535

int -> Entre -2.147.483.648 até +2.147.483.647, mais usado dos inteiros

uint -> Não aceita valor negativo, entre 0 até +4.294.967.295

long -> Entre -9.223.372.036.854.775.808 até +9.223.372.036.854.775.807

ulong -> Não aceita valor negativo, entre 0 até +18.446.744.073.709.551.615

**Reais:**

float -> 4 bytes

double -> 8 bytes

decimal -> 16 bytes

float -> Entre -3,40228235E+37 até +3,40228235E+37

double -> Entre -1,7976931348623157E+308 até +1,7976931348623157E+308

decimal -> Entre -79.228.162.514.264.337.593.543.950.335 até +79.228.162.514.264.337.593.543.950.335

Para declarar uma variável do tipo “float” precisa fazer a conversão de “double” para “float”:

float preco = 1299.99F;

Pois por padrão se usa o “double”, e precisa fazer a conversão direta de “double” para “float”.

Para ver o valor máximo e minimo de um tipo numérico, basta colocar o tipo dele usando o método “MaxValue” para o maior valor, e “MinValue” para o menor valor:

Retorna o maior valor do tipo “byte”:

Console.WriteLine(byte.MaxValue);

Retorna o menor valor do tipo “byte”:

Console.WriteLine(byte.MinValue);

Outra forma de escrever os números é separando eles por “\_”, sendo opcional, apenas para facilitar a leitura:

uint populacaoBrasileira = 207\_600\_000;

**Boleanos(Verdadeiro ou Falso):**

bool -> true ou false

bool estaChovendo = true;

**Charactere:**

char -> Representa uma letra letra da tabela unicode ou um código da mesma, e é delimitado por aspas simples(‘’).

Ex:

char letra = ‘b’;

**String(Texto):**

string -> Representa uma cadeia de caracteres, usada para escrever texto, nome e etc, e é delimitado por aspas duplas(“”).

Ex:

string texto = ”Seja Bem vindo ao curso de C#”;

**Inferência**

Em C# você também pode declarar uma variável usando o “var”, definindo seu tipo de forma implícita:

var nome = “Gabriel”;

var idade = 32;

Mas ao usar o “var”, você é obrigado a inicializar a variável no momento da sua declaração, pois é pelo valor atribuído a variável que o C# define o seu tipo de forma implícita, ou seja, você não pode declarar uma variável com “var” e inicializar ela na linha seguinte ou em outra linha.

Errado:

var a;

a = 9;

Você pode declarar uma variável sem inicializar ela definindo seu tipo explicitamente:

int b;

b = 2;

Mas sempre deve inicializar a variável no momento da sua declaração se for usar o “var”.

**Interpolação**

Em C# você pode interpolar as strings, ou seja, em vez de fazer um texto dessa forma:

string nome = “Notebook Gamer”;

string marca = “Dell”;

double preco = 5800.00;

Console.WriteLine(“O “ + nome + “ da marca “ + marca + “ custa “ + preco + “.”);

Você pode fazer:

Console.WriteLine(“O {0} da marca {1} custa {2}.”, nome, marca, preco);

Colocando a posição da variável entre chaves({ }) começando por 0, e depois passando as variáveis na ordem separadas por vírgula.

Ou pode passar a variável diretamente dentro das chaves:

Console.WriteLine($“A marca {marca} é legal.”);

Colocando um “dólar($)” antes das aspas, e entre as chaves colocar a variável ou o valor que fique naquela posição da string.