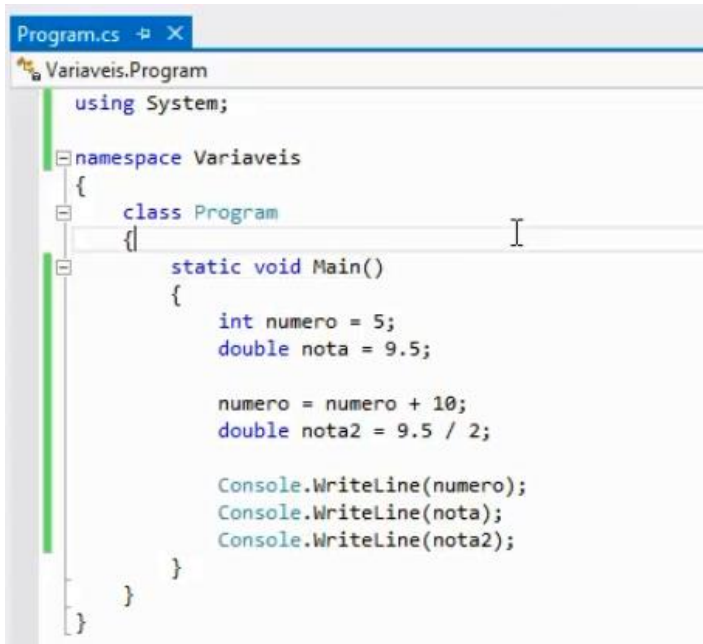


EXEMPLOS EM C#

Uso de variáveis e atribuições.



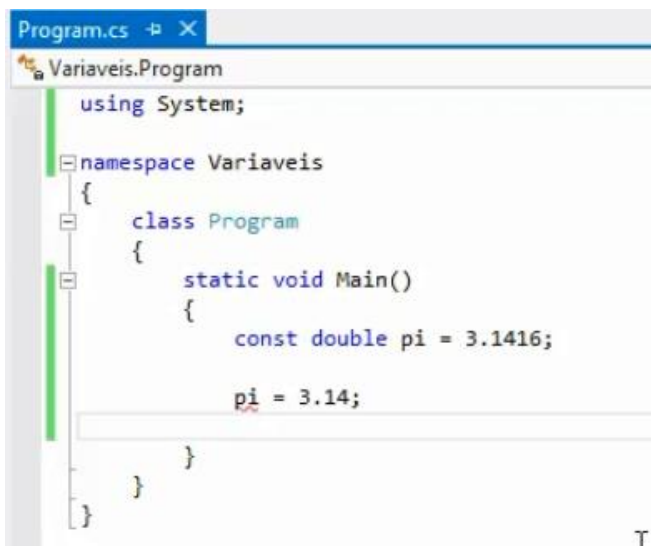
```
Program.cs [X]
Variaveis.Program
using System;

namespace Variaveis
{
    class Program
    {
        static void Main()
        {
            int numero = 5;
            double nota = 9.5;

            numero = numero + 10;
            double nota2 = 9.5 / 2;

            Console.WriteLine(numero);
            Console.WriteLine(nota);
            Console.WriteLine(nota2);
        }
    }
}
```

Definindo constantes



```
Program.cs [X]
Variaveis.Program
using System;

namespace Variaveis
{
    class Program
    {
        static void Main()
        {
            const double pi = 3.1416;

            pi = 3.14;
        }
    }
}
```

Se atribuirmos um outro valor a essa constante dará erro.

E após a atribuição da constante devemos dar um valor senão também acusará erro.

Casting implícito e explícito

```
Program.cs  ▸ ×
Casting.Program

// Casting implícito
int i = 10;
double d = i;

// Casting implícito
short s = 5;
long l = s;

// Casting explícito
double d2 = 3.5;
int i2 = (int)d2;

long l2 = 25;
int i3 = (int)l2;

long l3 = 438746387462735;
int i4 = (int)l3;
Console.WriteLine(i4);
}
```

Bloco IF

```
Program.cs  ▸ ×
EstruturasControle.Program

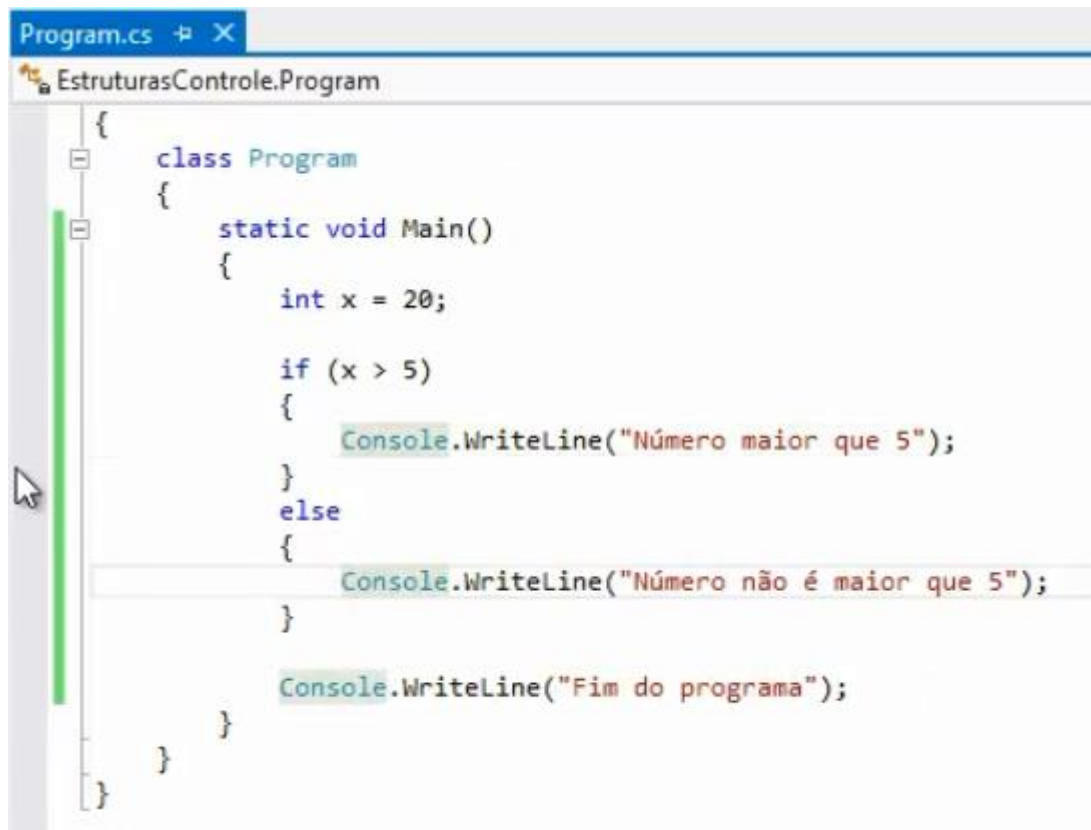
using System;

namespace EstruturasControle
{
    class Program
    {
        static void Main()
        {
            int x = 2;

            if (x > 5)
            {
                Console.WriteLine("Número maior que 5");
            }

            Console.WriteLine("Fim do programa");
        }
    }
}
```

Bloco IF ELSE

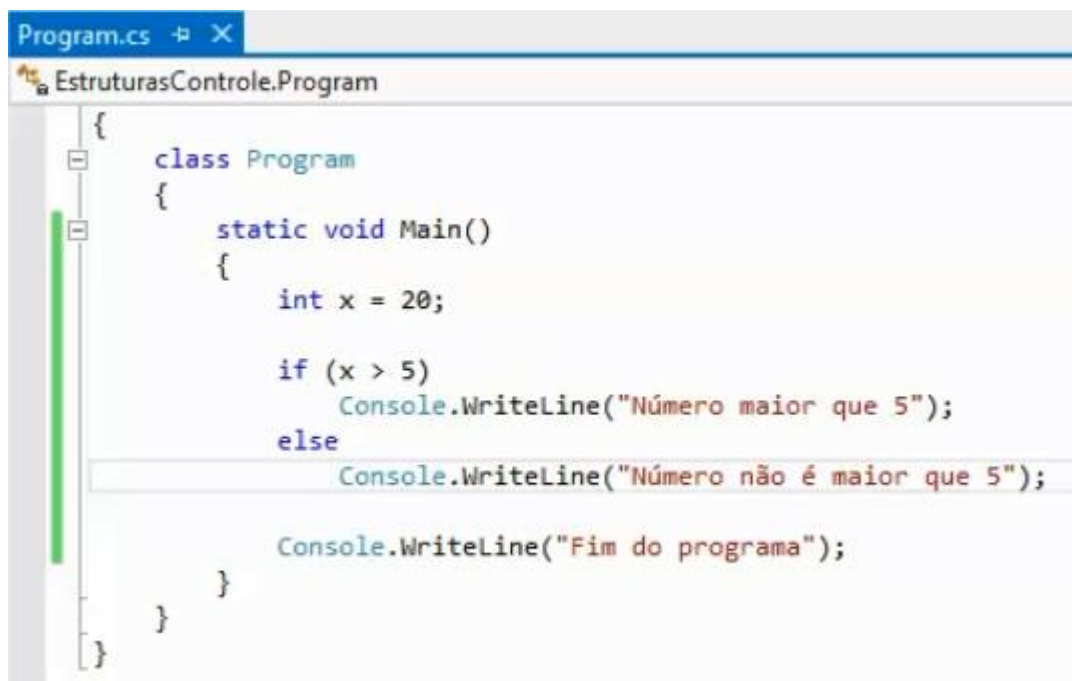


```
Program.cs + X
EstruturasControle.Program
{
    class Program
    {
        static void Main()
        {
            int x = 20;

            if (x > 5)
            {
                Console.WriteLine("Número maior que 5");
            }
            else
            {
                Console.WriteLine("Número não é maior que 5");
            }

            Console.WriteLine("Fim do programa");
        }
    }
}
```

Podemos até simplificar



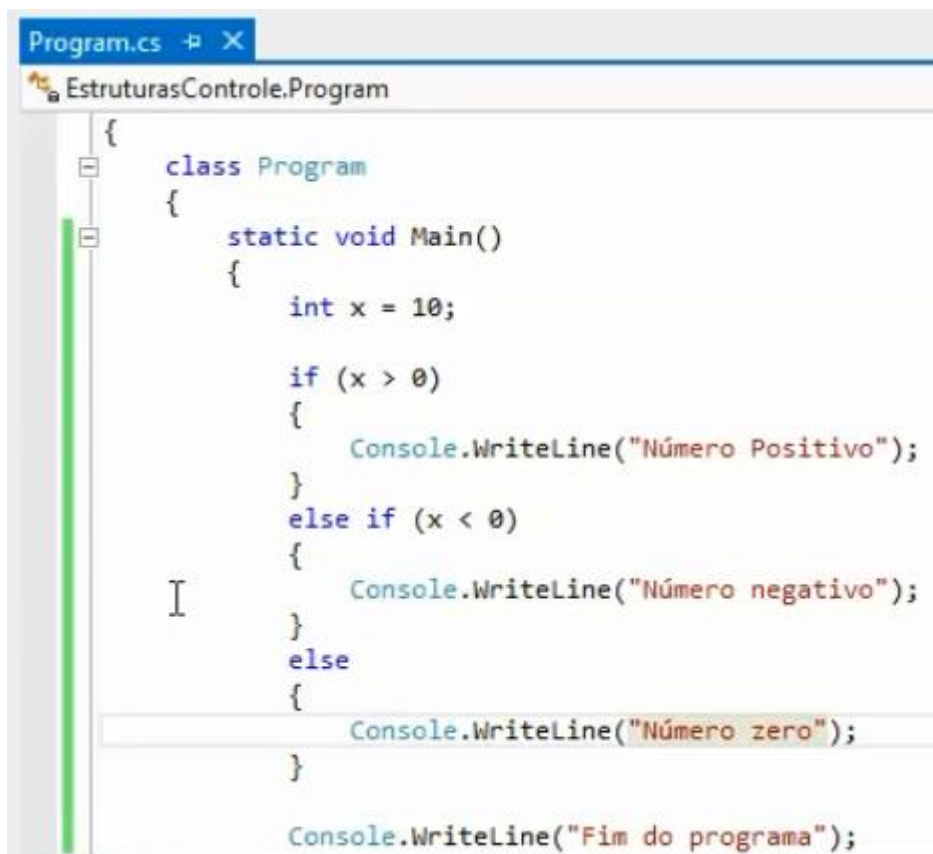
```
Program.cs + X
EstruturasControle.Program
{
    class Program
    {
        static void Main()
        {
            int x = 20;

            if (x > 5)
                Console.WriteLine("Número maior que 5");
            else
                Console.WriteLine("Número não é maior que 5");

            Console.WriteLine("Fim do programa");
        }
    }
}
```

Mas é melhor fazer como o exemplo anterior para maior segurança.

Uso de mais um IF ELSE



The image shows a screenshot of a C# code editor window titled "Program.cs". The code defines a class named "Program" with a static method "Main()". Inside "Main()", an integer variable "x" is initialized to 10. An if-else-else if structure is used to check the value of "x". The first condition is "if (x > 0)", which is true, so the code inside the first block is executed: "Console.WriteLine(\"Número Positivo\");". The other conditions, "else if (x < 0)" and "else", are not executed. Finally, the code prints "Fim do programa" and ends.

```
{
    class Program
    {
        static void Main()
        {
            int x = 10;

            if (x > 0)
            {
                Console.WriteLine("Número Positivo");
            }
            else if (x < 0)
            {
                Console.WriteLine("Número negativo");
            }
            else
            {
                Console.WriteLine("Número zero");
            }

            Console.WriteLine("Fim do programa");
        }
    }
}
```

Uso da Estrutura switch case

```
Program.cs  [X]
EstruturasControle.Program
{
    static void Main()
    {
        int mes = 2;

        switch (mes)
        {
            case 1:
                Console.WriteLine("31 dias");
                break;
            case 2:
                Console.WriteLine("28 dias");
                break;
            case 3:
                Console.WriteLine("31 dias");
                break;
            case 4:
                Console.WriteLine("30 dias");
                break;
            case 7:
                Console.WriteLine("31 dias");
                break;
            case 8:
                Console.WriteLine("31 dias");
                break;
            case 9:
                Console.WriteLine("30 dias");
                break;
            case 10:
                Console.WriteLine("31 dias");
                break;
            case 11:
                Console.WriteLine("30 dias");
                break;
            case 12:
                Console.WriteLine("31 dias");
                break;
            default:
                Console.WriteLine("Mês inválido");
                break;
        }
    }
}
```

```
}
}
}
```