

Escola Técnica Estadual “São Paulo” - ETESP
Linguagem C# - Visual Studio 2019

7 - Operadores aritméticos

/	Divisão
+	Soma
-	Subtração
*	Multiplicação
%	Resto da divisão

Exemplos de operações possíveis:

- 1) Divisão → resultado = valor1 / valor2;
- 2) Soma → resultado = valor1 + valor2;
- 3) Subtração → resultado = valor1 - valor2;
- 4) Multiplicação → resultado = valor1 * valor2;
- 5) Resto da divisão → resto = valor1 % valor2;

7.1 - Operadores reduzidos (+= -= *= /=)

Em alguns casos poderemos “simplificar” a expressão aritmética, por exemplo, considere a expressão:

valor1 = valor1 + valor2.

Esta operação poderá ser “simplificada” para:

valor1 += valor2, o mesmo ocorrerá nos casos abaixo:

valor1 = valor1 - valor2 → expressão simplificada para: valor1 -= valor2

valor1 = valor1 * valor2 → expressão simplificada para: valor1 *= valor2

valor1 = valor1 / valor2 → expressão simplificada para: valor1 /= valor2

Vamos a um exemplo prático de “Operações Aritméticas”... crie o projeto Exe009.

Prof. Roberto de Castro

INTERFACE GRÁFICA

Faça a configuração das “propriedades”. Observe que, inicialmente, o GroupBox das “operações aritméticas” está “travado”. Será “destravado somente após a “verificação dos valores”.

PROGRAMAÇÃO

Principais etapas:

- 1) Declaração das variáveis: Para este projeto teremos TRÊS variáveis numéricas do tipo “double” (variáveis com casas decimais). São elas: valor1, valor2 e resultado.
- 2) Neste projeto teremos OBRIGATORIAMENTE o procedimento de “conversão” do dado digitado nas caixas de texto (TxtValor1 e TxtValo2) para um dado numérico (esta conversão de tipo será realizada pelo botão “Validar Valores”. LEMBRE-SE: O dado digitado em uma caixa de texto será SEMPRE considerado um dado do tipo texto (string) e a conversão será obrigatória!!!

```
valor1 = Convert.ToDouble(TxtValor1.Text);
```

```
valor2 = Convert.ToDouble(TxtValor2.Text);
```

- 3) Efetuar a operação (programação dos botões: Soma / Subtrai / Multiplica / Divide:

```
resultado = valor1 + valor2; //Exemplo para a soma
```

```
resultado = Math.Pow(valor1, valor2); //Exemplo do cálculo da potência
```

- 4) Após o cálculo, exibir o resultado no label “LblResultado”. IMPORTANTE: Devemos converter o número para texto (string):

```
LblResultado.Text = resultado.ToString();
```

Em projeto anterior foi apresentada uma solução para “bloquear” a digitação de dados não numéricos (a caixa de texto aceitava somente números. No projeto incluímos a programação no evento KeyPress para evitar problemas de “erros de conversão”).

Para este projeto apresento a estrutura de tratamento de erro: Try..... Catch. (esta programação será incluída no botão “Validar Valores”).

Basicamente funciona assim:

Abrimos o bloco Try e inserimos o bloco de programação que poderia gerar o erro.

Abrimos o bloco Catch e inserimos o bloco de programação que irá “tratar/resolver” o erro.

Existe um formato mais completo, que poderemos ver mais adiante.

Veja o exemplo abaixo:

DICA: A estrutura poderá ser criada automaticamente, digitando Try e pressionando duas vezes a tecla TAB.

```
Try
{
    Código de programação que “poderá” gerar algum tipo de erro;
}
Catch
{
    Programação que será executada caso ocorra algum erro. (Mensagem de erro)
}
```

PROGRAMAÇÃO

```
using System;
using System.Windows.Forms;

namespace Exe009
{
    public partial class FrmExe009 : Form
    {
        //Declaração das propriedades
        double valor1 = 0;
        double valor2 = 0;
        double resultado = 0;

        public FrmExe009()
        {
            InitializeComponent();
        }

        private void BtnSair_Click(object sender, EventArgs e)
        {

```

```
Application.Exit();
}

private void BtnLimpar_Click(object sender, EventArgs e)
{
    TxtValor1.Text = "";
    TxtValor2.Text = "";
    LblResultado.Text = "";
    valor1 = 0;
    valor2 = 0;
    resultado = 0;
    GrpValores.Enabled = true;
    GrpOperacoes.Enabled = false;
    TxtValor1.Focus();
}

private void BtnValidar_Click(object sender, EventArgs e)
{
    try
    {
        valor1 = Convert.ToDouble(TxtValor1.Text);
        valor2 = Convert.ToDouble(TxtValor2.Text);
        MessageBox.Show("Valores ok!!!", "Selecione a operação");
        GrpOperacoes.Enabled = true;
        GrpValores.Enabled = false;
    }
    catch
    {
        MessageBox.Show("Digite apenas números!!!", "ERRO - ATENÇÃO");
        TxtValor1.SelectAll();
        TxtValor1.Focus();
    }
}

private void BtnSomar_Click(object sender, EventArgs e)
{
    resultado = valor1 + valor2;
    LblResultado.Text = resultado.ToString();
}

private void BtnSubtrair_Click(object sender, EventArgs e)
{
    resultado = valor1 - valor2;
    LblResultado.Text = resultado.ToString();
}

private void BtnMultiplicar_Click(object sender, EventArgs e)
{
    resultado = valor1 * valor2;
```

```
LblResultado.Text = resultado.ToString();
}

private void BtnDividir_Click(object sender, EventArgs e)
{
    if (valor2 != 0)
    {
        resultado = valor1 / valor2;
        LblResultado.Text = resultado.ToString();
    }
    else
    {
        LblResultado.Text = "***ERRO ***";
    }
}

private void BtnCalcularPotencia_Click(object sender, EventArgs e)
{
    resultado = Math.Pow(valor1, valor2);
    LblResultado.Text = resultado.ToString();
}
}
```

Observe a programação do botão “Validar Valores”.

Incluimos a estrutura de “tratamento de erro Try...” e dentro dela a programação que “poderá gerar um erro em tempo de execução” e interromper o programa. O erro ocorrerá no caso de ser digitado um dado não numérico na caixa de texto e o programa tentar converter o dado para número.

Incluimos na estrutura “catch” o código de programação que será executado no caso de ocorrer o erro, neste caso, a digitação de um dado não numérico. O programa capturará o erro e exibirá a mensagem de alerta, impedindo desta forma que o programa seja encerrado.

Faça o teste.... Execute o programa. Digite informações: números, textos... Observe os resultados.

Prof. Roberto de Castro

Exercicio proposto

Exe009 → Versão 2 (com a utilização de “array” de botões e estrutura switch ... case)

INTERFACE GRÁFICA

SOLUÇÃO

```
using System;
using System.Windows.Forms;

namespace Exe009
{
    public partial class FrmExe009Versao2 : Form
    {
        //Declaração das propriedades
        double valor1 = 0;
        double valor2 = 0;
        double resultado = 0;
        string operacao = "";

        public FrmExe009Versao2()
        {
            InitializeComponent();
        }

        private void BtnSair_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }

        private void FrmExe009Versao2_Load(object sender, EventArgs e)
        {
            timer1.Enabled = true;
            LblData.Text = DateTime.Now.ToString("dd/MM/yyyy");
        }
    }
}
```

Prof. Roberto de Castro

```
LblHora.Text = DateTime.Now.ToString("HH:mm:ss");
LblDia.Text = DateTime.Now.ToString("dddd").ToUpper();
}
```

```
private void BtnLimpar_Click(object sender, EventArgs e)
{
    TxtValor1.Text = "";
    TxtValor2.Text = "";
    LblResultado.Text = "";
    valor1 = 0;
    valor2 = 0;
    resultado = 0;
    operacao = "";
    TxtValor1.Focus();
}
```

//Este método é executado por todas as operações aritméticas

```
private void BtnSomar_Click(object sender, EventArgs e)
{
    try
    {
        valor1 = Convert.ToDouble(TxtValor1.Text);
        valor2 = Convert.ToDouble(TxtValor2.Text);

        operacao = (sender as Button).Text;

        switch (operacao)
        {
            case "+":
                {
                    resultado = valor1 + valor2;
                    LblResultado.Text = resultado.ToString();
                    break;
                }

            case "-":
                {
                    resultado = valor1 - valor2;
                    LblResultado.Text = resultado.ToString();
                    break;
                }

            case "*":
                {
                    resultado = valor1 * valor2;
                    LblResultado.Text = resultado.ToString();
                    break;
                }

            case "/":
```

```
{
    if (valor2 != 0)
    {
        resultado = valor1 / valor2;
        LblResultado.Text = resultado.ToString();
        break;
    }
    else
    {
        throw new Exception("Divisão por zero???");
    }
}

default:
{
    resultado = Math.Pow(valor1, valor2);
    LblResultado.Text = resultado.ToString();
    break;
}
}
}
catch (Exception erro)
{
    MessageBox.Show("Verificar digitação dos valores " + erro, "Atenção");
    TxtValor1.SelectAll();
    TxtValor1.Focus();
}
}

private void timer1_Tick(object sender, EventArgs e)
{
    LblHora.Text = DateTime.Now.ToString("HH:mm:ss");
}
}
```