INSTITUTO SUPERIOR DE ENGENHARIA DE COIMBRA

CTSP em Tecnologias e Programação de Sistemas de Informação

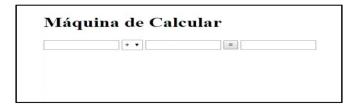
Programação Web II

Os exercícios propostos destinam-se a adquirir competências em Programação Web, com C#. A prática só funciona se, de facto, construir uma solução. Não assuma que o docente lhe disponibilizará uma solução. Como tal, deve, por si, realizar os exercícios propostos e interagir com o docente para avaliar as soluções, por si, desenvolvidas.

FICHA 4

Exercício 1

- 1 Crie um novo website com o nome "MaquinaCalcularSimples".
- 2 Crie uma página ASP.NET com o nome "Calculadora.aspx".
- 3 Construa o layout da página, conforme imagem abaixo:



Textbox Esquerda: ID=txtValA
 DropDownList: ID=selOperator
 Textbox Direita: ID=txtValB

Botão: ID=btnCalc

Textbox Resultado: ID=txtResult

- 4 Adicione a pasta "App_Code" (ASP.NET folder) ao projeto.
- 5 Adicione um ficheiro do tipo **classe** à pasta "**App_Code**" e dê-lhe o nome de "*MaquinaCalcularSimples.cs*".
- 6 Programe a nova página (Calculadora.aspx), de forma a criar uma calculadora que faça as operações de <u>soma</u>, <u>subtração</u>, <u>multiplicação</u> e <u>divisão</u> entre dois números.

Regras:

- O resultado deve ser apresentado com dois dígitos decimais.
- No caso de uma divisão por zero, o resultado é "Infinito".
- No caso de ser inserido um valor não-numérico, o resultado é "Valores Inválidos".

Considere que:

a) Na <u>classe **MaquinaCalcularSimples (MaquinaCalcularSimples.cs)**</u>, deverá criar 4 métodos que permitam efetuar os cálculos.

Recebem 2 parâmetros e retornam um decimal:

public decimal Add(decimal a, decimal b)

```
public decimal Subtract(decimal a, decimal b)

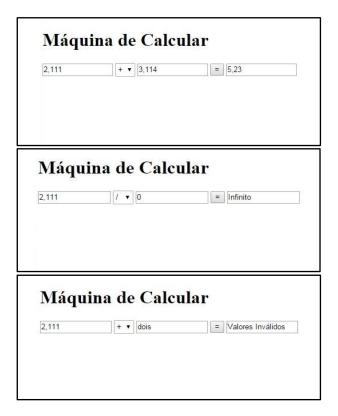
public decimal Multiply(decimal a, decimal b)

public decimal Divide(decimal a, decimal b)
```

b) As operações algébricas deverão ser implementadas no método
 public string Calc(string Valor_A, String Operacao, string Valor_B, int NumDigDec = 2)

da <u>classe *MaquinaCalcularSimples*</u>, definida na pasta "App_Code".

Exemplos para validar o exercício:

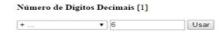


Exercício 2

a) Altere a calculadora desenvolvida no exercício anterior, de modo a que o <u>número de dígitos</u> <u>decimais do resultado seja selecionável</u>.

Considere que, por omissão, o resultado é apresentado com 1 dígito decimal.

No caso de optar por mais do que quatro dígitos, é aberta uma caixa de texto (ID= txtNumDigitosMore) para se colocar o número de dígitos decimais que se pretende para a representação do resultado. Se não colocar um valor numérico nessa caixa de texto então, por omissão, o valor do número de dígitos decimais para a representação do resultado é igual a 5.

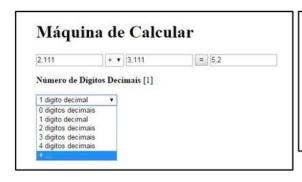


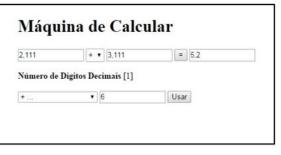
Label [1]: ID=LabelNumDigitos

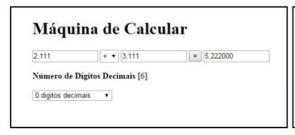
DropDownList: ID=selNumDigitos

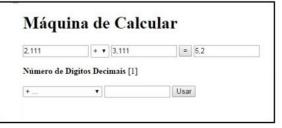
Textbox: ID= txtNumDigitosMore

Botão "Usar": ID= btnNumDigitosUsar

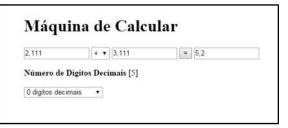












b) Na caixa de texto "**txtNumDigitosMore**", apenas permita a introdução de números dentro do intervalo [0-28].

DICAS:

 No evento Page_Load, deverá definir o valor da Textbox "txtNumDigitosMore" com o valor 1, na primeira vez que entra na página.

if (!Page.IsPostBack)

• Verificar se os valores introduzidos são numéricos:

int verifNum = 0;
if (int.TryParse(txtNumDigitosMore.Text, out verifNum))

Arredondamento

myFinalResult = decimal.Round(myOperationResult, NumDigDec, MidpointRounding.AwayFromZero).ToString(precisao);