



## VISUAL STUDIO PARTE II

### INTRODUÇÃO

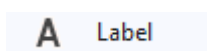
Quando se utiliza componentes visuais é importante salientar um detalhe:

**Quando se falar em componente visual, utiliza-se o termo controle.**

**Quando se utiliza um componente que não é visual, é interessante referenciá-lo como componente.**

Alguns componentes possuem a propriedade dock a qual permite ao controle se redimensionar à medida que seu controle pai é redimensionado.

### COMPONENTE LABEL



É o componente que apresenta texto. Apesar de possuir alguns eventos, estes são pouco usados.

Propriedades mais utilizadas:

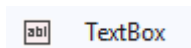
Text = mensagem

AutoSize = False permite receber múltiplas linhas. Neste caso, combinando-se essa propriedade com o redimensionamento do componente, ajuste da borda e também com o alinhamento do texto. Podendo, inclusive utilizar esse componente como um título específico.

É possível inserir imagem junto com texto e alinhá-los diferentemente dentro do componente.

Os componentes que admitem borda possuem, geralmente, a propriedade **Dock** que permite posicionar o componente em um local específico do formulário.

### COMPONENTE TEXTBOX



É um componente para entrada de texto simples.

Propriedades mais utilizadas:

Multiline = True permite inclusão de várias linhas. Neste caso, torna-se interessante redimensionar o componente. Com essa propriedade True, torna-se interessante ativar as propriedades AcceptsReturn e AcceptsTab para receber, respectivamente, as teclas Enter e Tab como caracteres dentro do



texto. Pode ser interessante também ativar a propriedade de barra de rolagem (ScrollBars) como vertical para auxiliar na edição de um texto mais longo.

PasswordChar = Recebe um caractere para ser visualizado quando o campo receber senha. Mas, em vez de informar um caractere aqui, utilize a propriedade **UseSystemPasswordChar** como True para que o caractere fique no formato de bolinhas.

CharacterCasing = Permite deixar a entrada com letras maiúsculas (caixa alta) com a opção Upper ou com a opção Lower para entrada de letras minúsculas (caixa baixa).

ReadOnly = True não permite ao usuário digitar informações no componente.

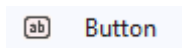
Enabled = False deixa o componente bloqueado

Visible = False não mostra o componente em tempo de execução

**MaxLength** = IMPORTANTÍSSIMO. Permite estabelecer a quantidade de caracteres a serem digitados

Existem algumas propriedades que podem ser úteis como as que iniciam com AutoComplete, pois podem auxiliar o usuário em certas digitações.

## COMPONENTE BUTTON



Propriedades mais utilizadas:

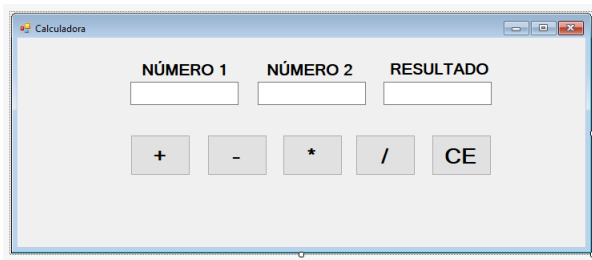
Text (permite & para tecla aceleradora), Name, Image.

Quando possuir texto e imagem é possível alinhá-los usando TextAlign e ImageAlign deixando, por exemplo, o texto na parte inferior e a imagem na parte superior. Há também a propriedade TextImageRelation que permite ajustar imagem e texto.

Existem eventos para o Button como Click, KeyPress, Enter e outros.

## ATIVIDADES PRÁTICAS

01.Crie um projeto e desenvolva uma calculadora que efetue as quatro operações usando duas caixas de texto para entrada de dados e a terceira caixa de texto como resultado, como ilustrado a seguir:



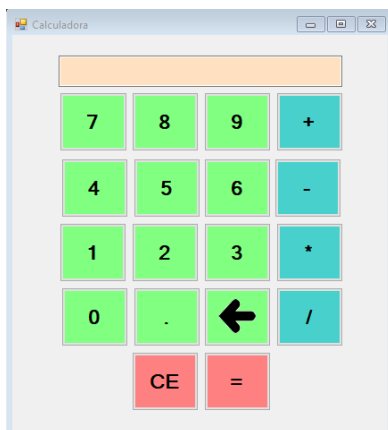
Nota:

Utilize a função `Convert.ToDouble` para converter os valores de cada caixa de texto para variável do tipo `double` e a função `toString` para converter um valor `double` para o label da resposta.

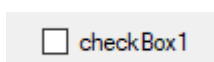
02. Crie um projeto que permita simular uma calculadora com as seguintes características:

- dez teclas numéricas
- quatro teclas das operações básicas
- tecla de limpar, tecla de resultado
- um visor para mostrar o resultado parcial e total

Crie um formulário semelhante a:



## COMPONENTE CHECKBOX



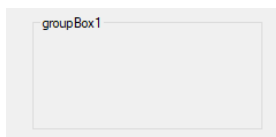


É um componente que permite ser marcado (checked=true) ou desmarcado (checked=false).

Propriedades mais utilizadas:

Name, Text, Checked (True ou False), CheckState (Unchecked, Checked e Indetermined), CheckAlign

## COMPONENTE GROUPBOX

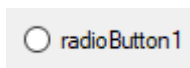


Permite agrupar diversos componentes. Um desses componentes é o RadioButton, sendo que se um estiver marcado, os demais ficam desmarcados. Isso faz com que fiquem mutuamente exclusivos. Um GroupBox possui a propriedade Text que permite criar um título para o grupo. Cada RadioButton deve possuir um identificador (name) específico. Além disso, o GroupBox permite agrupar elementos para melhorar o design. Por exemplo: acomodar todos os elementos de entrada de dados de endereço; todos os componentes relacionados com telefone.

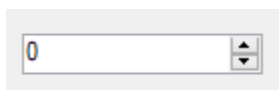
## COMPONENTE RADIOBUTTON

Propriedades mais utilizadas de RadioButton:

Text e Checked (True ou False).



## COMPONENTE NUMERICUPDOWN



Permite informar dados através das setas indicativas para baixo ou para cima.

Propriedades:

Increment (valor que será incrementado/decrementado a cada clique), Value (valor atual), InterceptArrowsKeys (True/False, ativa ou desativa as setas),



DecimalPlaces (destaca quantidade de casas decimais) e UpdownAlign (posiciona as setas na esquerda ou direita).

## COMPONENTE MASKEDTEXTBOX

(.)- MaskedTextBox

Possui propriedades semelhantes ao TextBox. Serão citadas apenas as propriedades específicas desse componente.

Propriedades:

Mask = Permite estabelecer máscara de entrada com o botão de reticências:

Descrição da Máscara	Formato de Dados	Tipo de Validação
Phone number	(574) 555-0123	(nenhum)
Phone number no area code	555-0123	(nenhum)
Short date	12/11/2003	DateTime
Short date and time (US)	12/11/2003 11:20	DateTime
Social security number	000-00-1234	(nenhum)
Time (European/Military)	23:20	DateTime
Time (US)	11:20	DateTime
Zip Code	98052-6399	(nenhum)
<Personalizado>		(nenhum)

Máscara:  ☒ Usar ValidatingType

Visualização:

OK Cancelar

Essa janela permite configurar uma máscara de entrada. Pode-se criar uma nova máscara acessando a opção em destaque. Em seguida, informe a máscara na caixa de texto **Máscara**. Se usar 0, indica que o dígito é obrigatório. Se usar 9 indica que é opcional. Para formatar um ponto use a vírgula.

InsertKeyMode = Permite mudar a forma de inserção (ora empurrando o conteúdo já existente, ora subscrevendo-o, conforme a opção selecionada).

PromptChar = É o caractere que é apresentado quando não há nenhuma digitação

SkipLiterals = Se True evita que o usuário digite os elementos separadores da máscara como ponto e outros



TextMaskFormat = ExcludePromptAndLiterals (permite obter só o que foi digitado), IncludePrompt (inclui os caracteres não digitados), IncludeLiterals (inclui os pontos) e IncludePromptAndLiterals(inclui todos os literais existentes).

Tabela de mascaramento

Elemento de mascaramento	Descrição
0	Dígito, obrigatório. Esse elemento aceitará qualquer dígito único entre 0 e 9.
9	Dígito ou espaço, opcional.
#	Dígito ou espaço, opcional. Se essa posição estiver em branco na máscara, ela será renderizada como um espaço na <a href="#">Text</a> propriedade. São permitidos sinais de mais (+) e de menos (-).
L	Letra, obrigatória. Restringe a entrada para as letras ASCII de a-z e A-Z. Esse elemento Mask é equivalente a [a-zA-Z] em expressões regulares.
?	Letra, opcional. Restringe a entrada para as letras ASCII de a-z e A-Z. Este elemento Mask é equivalente a [a-zA-Z]? em expressões regulares.
&	Caractere, obrigatório. Se a <a href="#">AsciiOnly</a> propriedade for definida como true, esse elemento se comparecerá com o elemento "L".
C	Caractere, opcional. Qualquer caractere que não seja de controle. Se a <a href="#">AsciiOnly</a> propriedade for definida como true, esse elemento se comparecerá com o elemento "?".
.	Espaço reservado de decimal. O caractere de exibição real usado será o símbolo decimal apropriado para o provedor de formato, conforme determinado pela propriedade do controle <a href="#">FormatProvider</a> .
,	Espaço reservado para milhares. O caractere de exibição real usado será o espaço reservado de milhares apropriado para o provedor de formato, conforme determinado pela <a href="#">FormatProvider</a> Propriedade do controle.
:	Separador de hora. O caractere de exibição real usado será o símbolo de hora apropriado para o provedor de formato, conforme determinado pela propriedade do controle <a href="#">FormatProvider</a> .
/	Separador de data. O caractere de exibição real usado será o símbolo de data apropriado para o provedor de formato, conforme determinado pela propriedade do controle <a href="#">FormatProvider</a> .



Elemento de mascaramento	Descrição
\$	Símbolo de moeda. O caractere real exibido será o símbolo de moeda apropriado para o provedor de formato, conforme determinado pela propriedade do controle <a href="#">FormatProvider</a> .
<	Deslocar para baixo. Converte todos os caracteres que seguem para minúsculas.
>	Mover para cima. Converte todos os caracteres que seguem para letras maiúsculas.
	Desabilitar um deslocamento anterior ou deslocar para baixo.
\	Fuga. Escapa um caractere de máscara, transformando-o em um literal. " \ \" é a sequência de escape para uma barra invertida.
Todos os outros caracteres	Literais Todos os elementos não máscara aparecerão como eles mesmos <a href="#">MaskedTextBox</a> . Os literais sempre ocupam uma posição estática na máscara em tempo de execução e não podem ser movidos ou excluídos pelo usuário.

## COMPONENTE COMBOBOX



Permite apresentar uma lista dinâmica ou estática de informações, mostrando apenas um item da lista.

Propriedades:

Itens = São os itens da lista do combobox (estático)

DropDownStyle = DropDown (lista de itens que aceita digitação); DropDownList (não aceita digitação)

Sorted = para ordenar (true) os itens na lista

DropDownHeight = Permite informar a altura que ficará a lista de itens

É interessante, às vezes, combinar as propriedades que se iniciam com AutoComplete.



Ressalta-se que existem propriedades que só são acessadas via código.

Links de apoio:

[http://www.macoratti.net/11/01/c\\_cbo1.htm](http://www.macoratti.net/11/01/c_cbo1.htm)

<https://docs.microsoft.com/pt-br/dotnet/desktop/winforms/controls/add-and-remove-items-from-a-wf-combobox?view=netframeworkdesktop-4.8>

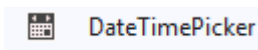
Para obter o valor que está sendo visualizado => propriedade Text

Para obter o valor do conteúdo que não é visualizado => SelectedValue

Para obter a posição do elemento na lista => SelectedIndex (de 0 até total\_de\_elementos – 1)

Com SelectedItem obtém-se o objeto correspondente ao item selecionado.

## COMPONENTE DATETIMEPICKER



Permite selecionar data.

As propriedades que se iniciam com Calendar permitem combinar as cores que serão visualizadas no calendário, nos finais de semana e outros.

A propriedade Format estabelece a formação do conteúdo sendo Short (permite apresentar apenas a data), Long (data + hora), Time (apenas as horas) e Custom (customizado).

Com as propriedades MinDate e MaxDate estabelece-se um intervalo de datas para serem informadas na interface gráfica.

A propriedade ShowCheckBox permite habilitar um check para a data, que oferece a propriedade checked (false ou true).

A propriedade ShowUpDown ativa (quando True) as setas para cima/baixo para se selecionar um conteúdo, que é interessante para selecionar a hora.

A propriedade Value fornece o valor atual que está no controle.

## COMPONENTE DATAGRIDVIEW





O controle de DataGridView fornece uma maneira poderosa e flexível para exibir dados em um formato de tabela, permitindo exibir somente leitura de dados ou pode ajustar a escala para mostrar exibições editáveis de grandes conjuntos de dados. Permite ser estendido de várias maneiras para compilar comportamentos personalizados em seus aplicativos. Oferece recursos para especificar seus próprios algoritmos de classificação com programação, podendo criar seus próprios tipos de células. É possível personalizar facilmente sua aparência escolhendo dentre várias propriedades. Muitos tipos de armazenamentos de dados podem ser usados como uma fonte de dados ou sem nenhuma fonte de dados.