

Principais componentes c#

TextBox:

Componente usado para receber entradas de texto. No exercício abaixo, as caixas txtPagador, txtRecebedor, txtValor, e txtExtenso são todas TextBox, que capturam informações fornecidas pelo usuário.

Propriedade importante:

Text: Armazena o texto que o usuário insere.

Name: Identifica o componente, como txtPagador.

Label:

Usado para exibir texto fixo no formulário, como rótulos para identificar campos.

Não tem interatividade, apenas exibe informações.

Propriedade importante:

Text: Define o texto visível no Label.

ComboBox:

Componente que permite ao usuário selecionar um item de uma lista suspensa.

No exercício, o cmbReferente possui itens como "serviços prestados em treinamento".

Propriedade importante:

Items: Lista de opções que podem ser selecionadas.

SelectedIndex: Índice do item selecionado.

RichTextBox:

Similar ao TextBox, mas permite formatação de texto como negrito, itálico, e sublinhado.

No exercício, o ricTexto é usado para gerar o recibo com formatação.

Propriedade importante:

AppendText: Adiciona texto ao conteúdo atual.

SelectionFont: Define a formatação da seleção.

Button:

Componente que aciona ações no código quando clicado.

No exercício, o btnGerar é usado para gerar o recibo.

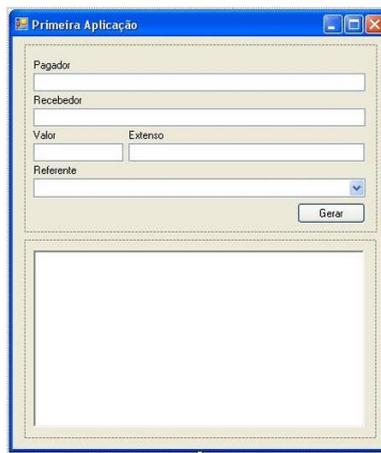
Propriedade importante:

Click: Evento que dispara ao clicar no botão.

Exercício:

Iremos construir uma pequena aplicação de geração de recibo para utilizar alguns componentes como o **TextBox** e o **ComboBox** e suas propriedades comum.

1. Inicie um novo projeto C# do tipo WinForms .NetFrameWork.
2. Adicione os componentes (Panel, Label, TextBox, ComboBox, RichTextBox e Button) como mostrado na figura a seguir:



3. Dê os seguintes nomes aos componentes inseridos:
 - txtPagador;
 - txtRecebedor;
 - txtValor;
 - txtExtenso;
 - cmbReferente;
 - btnGerar;
 - ricTexto;
4. Entre na propriedade Items do ComboBox e digite em cada linha:
 - serviços prestados em treinamento
 - suporte técnico
 - serviços prestados no desenvolvimento de sistemas
5. Clique duas vezes no botão Gerar e digite o seguinte código:

```
// Variáveis
string linha;
const string branco = "\n";
int posicao;

// Limpando o RichText
ricTexto.Clear();

linha = "R E C I B O " + branco;
ricTexto.AppendText(linha);
ricTexto.AppendText(branco);
linha = "Eu " + txtRecebedor.Text + ", recebi a importância de "
        + txtValor.Text + "(" + txtExtenso.Text + "), referente a(o) "
        + cmbReferente.Text + "." + branco;
ricTexto.AppendText(linha);
ricTexto.AppendText(branco); ricTexto.AppendText(branco); ricTexto.AppendText("-----");
----" + branco); ricTexto.AppendText(txtRecebedor.Text);
ricTexto.Focus();
ricTexto.Select(0,11);

ricTexto.SelectionFont = new Font(ricTexto.SelectionFont, FontStyle.Bold);
```

6. Execute o código pressionando a tecla F5, preencha os campos e clique no botão Gerar. Veja o resultado.
-