

1. TForm

TForm é a componente associada à janela da aplicação padrão (forma). Usa-se a **TForm** para criar uma forma tal como a janela principal, uma caixa de diálogo (*"dialog box"*) ou uma forma filho do tipo *Multiple Document Interface* (*"MDI child"*). A forma pode conter outros objectos tais como Botões (TButton), Caixas de Marcação (TCheckBox) e Etiquetas/Rótulos (TLabel).

Para se adicionar uma forma à aplicação escolher a opção *New Form*, o que pode ser efectuado da seguinte maneira :

File → New Form

ou então carregar no ícone assinalado na figura seguinte



1.1. Propriedades

ActiveControl — seleccionar a componente que recebe o foco, quando a janela é criada

AutoScroll — indicar se as barras de deslocamento aparecem automaticamente (True) quando são necessárias, devido ao facto de a informação ser demasiado para a totalidade da janela

BorderIcons — seleccionar o tipo de ícones que aparecem na barra de título da janela (no canto superior direito). Os valores possíveis são :

Valor	Significado
biSystemMenu	A forma tem um menu de Controlo (conhecido por menu de Sistema)
biMinimize	A forma tem um botão de Minimizar
biMaximize	A forma tem um botão de Maximizar
biHelp	Se BorderStyle é bsDialog ou são excluídos biMinimize e biMaximize, aparece um ponto de interrogação (?) na barra de título da forma, que quando clicada, o cursor altera-se para crHelp; caso contrário, não aparece qualquer ponto de interrogação

BorderStyle — seleccionar o estilo da janela, que é um dos seguintes :

Valor	Significado
bsNone	Sem bordo visível (não é reajustável)
bsSingle	Bordo com uma linha simples (não é reajustável)
bsDialog	Bordo de uma caixa de diálogo standard (não é reajustável)
bsSizeable	Bordo standard reajustável.
bsToolWindow	Como bsSingle mas com um pequeno título
bsSizeToolWin	Como bsSizeable mas com um pequeno título

Caption — especificar o título da janela

ClientHeight — indicar a altura interior da janela (altura da área usada)

ClientWidth — indicar a largura interior da janela (largura da área usada)

Color — seleccionar a cor da área usada da janela

Ctl3D — indicar se a janela tem o aspecto a 3D (True) ou a 2D (False)

Enabled — inutilizar a janela (False), não respondendo a nenhum dos controlos (rato, teclado e relógio do sistema) ou utilizá-la (True)

Font — seleccionar o tipo de fonte (tipo, estilo, tamanho e outros efeitos dos caracteres) utilizada na escrita de texto inserido na janela

FormStyle — seleccionar o estilo da forma, que pode ser um dos seguintes :

Valor	Significado
fsNormal	A forma não é uma janela pai MDI nem uma janela filho MDI
fsMDIChild	A forma é uma janela filho MDI
fsMDIForm	A forma é uma janela pai MDI
fsStayOnTop	Esta forma permanece no topo do “desktop” e das outras formas no projecto, excepto outras que tenham também a FormStyle especificada por fsStayOnTop

Height — indicar a altura total da janela (parte exterior), incluindo a barra de título

HorzScrollBar — indicar se a barra de deslocamento horizontal é visível (True) ou escondida (False)

Left — indicar a distância, em pixels, entre a janela e o seu pai, relativamente às partes laterais esquerdas de ambas (quando não tem pai, considera-se o écran)

Menu — seleccionar o menu principal a associar à forma (se existir algum menu)

Name — nome interno da janela (identificador), pelo qual é referido no código da aplicação

ParentFont — indicar se assume a mesma fonte do seu pai (True) ou tem uma outra (false)

PopupMenu — seleccionar o menu *pop-up* (saltitante) que estará associado à janela (se existe), isto é, que aparece quando o botão direito do rato é clicado num ponto qualquer do interior da janela

Position — seleccionar o tamanho e o local onde vai aparecer a janela quando é criada, que é uma das seguintes combinações :

Valor	Significado
poDesigned	Aparece posicionada no écran e com o tamanho de como foi desenhada
poDefault	Aparece na posição do écran onde foi desenhada e com o tamanho determinado pelo Windows
poDefaultPosOnly	É mostrada com o tamanho que foi criada quando foi desenhada, mas o Windows escolhe a sua posição no écran
poDefaultSizeOnly	Aparece na posição onde foi desenhada, mas o Windows escolhe o seu tamanho
poScreenCenter	É desenhada com o tamanho original e posicionada ao centro do écran

Top — indicar a distância, em pixels, entre a janela e o seu pai, relativamente aos topos de ambos (quando não tem pai, considera-se o écran)

VertScrollBar — indicar se a barra de deslocamento vertical é visível (True) ou escondida (False)

Visible — indicar se a janela fica visível (True) ou escondido (False) quando é criada

Width — indicar a largura total da janela (parte exterior)

WindowState — indicar o estado em que a janela aparece no écran, quando é criada, que pode ser um dos seguintes :

Valor	Significado
wsNormal	A forma aparece no seu estado normal (isto é, como foi desenhada)
wsMinimized	A forma aparece no seu estado minimizado
wsMaximized	A forma aparece no seu estado maximizado

1.2. Eventos

OnClick — escrever código para efectuar um processamento especial quando se clica com o botão esquerdo do rato sobre a janela

OnClose — escrever código para efectuar um processamento especial quando a janela é fechada

OnCreate — escrever código para efectuar um processamento especial quando a janela é criada

OnDbClick — escrever código para efectuar um processamento especial quando se faz o duplo clique com o botão esquerdo do rato sobre a janela

OnMouseMove — escrever código para efectuar um processamento especial quando se move o rato sobre a janela

OnShow — escrever código para efectuar um processamento especial quando a janela é mostrada (isto é, quando a propriedade *Visible* é colocada a True)

2. TMenuItem

TMenuItem é uma componente pertencente à unidade '**menus**' e que está associada à construção de menus, através das componentes TMainMenu e TPopupMenu.

TMenuItem descreve as propriedades de um item de menu. Usa-se **TMenuItem** para especificar a aparência e comportamento de um item de menu. Cada componente TMainMenu ou TPopupMenu pode conter vários itens de menu. Quando um menu é definido em *design time*, o Menu Designer cria automaticamente um item de menu para cada comando do menu.

2.1. Propriedades

Break — indicar se o menu é dividido em colunas a partir deste item. Os valores possíveis são os seguintes :

Valor	Significado
mbNone	Não ocorre qualquer separação no menu. Este é o valor por defeito
mbBarBreak	O menu Pai ocupa uma outra coluna com um item que aparece no topo da nova coluna. Existe uma barra a separar as colunas
mbBreak	O menu ocupa uma outra coluna com um item que aparece no topo da nova coluna. Apenas existem espaços a separar a coluna nova da anterior

Caption — especificar o título do item do menu. Para associar um carácter acelerador (Alt + carácter), colocar o símbolo & imediatamente antes do carácter desejado para esse efeito

Checked — indicar se aparece uma marca (✓) junto ao título, permitindo que o item funcione como uma caixa de marcação (True) ou não apareça nada (False)

Enabled — inutilizar o item do menu (False), não respondendo a nenhum dos controlos (rato, teclado e relógio do sistema) e aparecendo esbatido; para ser utilizado colocar True

Name — nome interno do item do menu (identificador), pelo qual é referido no código da aplicação

ShortCut — indicar a combinação de teclas que os utilizadores podem digitar para rapidamente aceder ao item (atalho), em vez de se usar o rato; esta combinação de teclas aparece à direita do item no menu

Visible — indicar se o item do menu fica visível (True) ou escondido (False) quando é criado o menu

2.2. Eventos

OnClick — escrever código para efectuar um processamento especial quando se clica com o botão esquerdo do rato sobre a janela

3. TMainMenu

TMainMenu é uma componente que pertence ao conjunto “STANDARD” (figura seguinte).



A componente **TMainMenu** encapsula uma barra de menu e acompanha os menus *drop-down* de uma forma. Para começar a construção de um menu, adiciona-se um menu principal à forma e clica-se duas vezes seguidas (*double click*) a componente.

3.1. Propriedades

Items — é um objecto do tipo *TMenuItem* que descreve os itens do menu; é através desta propriedade que se acede à informação associada aos itens do menu

Name — nome interno do menu (identificador), pelo qual é referido no código da aplicação

3.2. Eventos

Não tem.

4. TPopupMenu

TPopupMenu é uma componente que pertence ao conjunto “STANDARD” (figura seguinte).



A componente **TPopupMenu** encapsula as propriedades, métodos e eventos de um menu *pop-up* (saltitante). Utilizar a **TPopupMenu** para definir menus *pop-up* que aparecem quando o utilizador clica num controlo com o botão direito do rato. Para tornar um menu *pop-up* válido, atribuir o objecto TPopupMenu à propriedade PopupMenu do controlo.

4.1. Propriedades

Alignment — indicar o local onde aparece o menu *pop-up* (saltitante) quando o utilizador clicar com o botão direito do rato num controlo que tenha o menu *pop-up* especificado na sua propriedade *PopupMenu*; os valores possíveis são os seguintes :

Valor	Significado
paLeft	O canto superior esquerdo do menu <i>pop-up</i> aparece sob o indicador do rato
paCenter	O menu <i>pop-up</i> aparece com o centro do topo sob o indicador do rato
paRight	O menu <i>pop-up</i> aparece com o seu canto superior direito sob o indicador do rato

AutoPopup — indicar se o menu *pop-up* aparece automaticamente (True), quando o utilizador clicar com o botão direito do rato num controlo que tenha o menu *pop-up* especificado na sua propriedade *PopupMenu*; se **AutoPopup** é False o aparecimento do menu *pop-up* é controlado através de instruções em Delphi

Items — é um objecto do tipo *TMenuItem* que descreve os itens do menu; é através desta propriedade que se acede à informação associada aos itens do menu

Name — nome interno do menu (identificador), pelo qual é referido no código da aplicação

4.2. Eventos

OnPopup — ocorre exactamente antes de aparecer o menu *pop-up*

OnClick e **OnDblClick** — ver componentes anteriores.

5. TLabel



A componente **TLabel** é um controlo que apresenta texto dentro duma forma. Este texto pode ser usado para etiquetar outra componente, podendo colocar o foco neste controlo quando o utilizador prime uma tecla aceleradora.

5.1. Propriedades

Align — indica o alinhamento da componente dentro do pai; os valores possíveis são os seguintes :

Valor	Significado
alNone	A componente permanece onde foi colocada na forma ou painel (é o valor por defeito)
alTop	A componente move-se para a topo da forma e redimensiona-se para ocupar o comprimento da forma. A altura da componente não é afectada
alBottom	A componente move-se para o fundo da forma e redimensiona-se para ocupar o comprimento da forma. A altura da componente não é afectada
alLeft	A componente move-se para o lado esquerdo da forma e redimensiona-se para ocupar a altura da forma. O comprimento da componente não é afectada
alRight	A componente move-se para o lado direito da forma e redimensiona-se para ocupar a altura da forma. O comprimento da componente não é afectada
alClient	A componente redimensiona-se ocupando a área usada duma forma. Se a componente já ocupa parte da área usada, a componente redimensiona-se para se ajustar à restante

Alignment — controla a posição horizontal do texto dentro do rótulo (label); os valores possíveis são os seguintes:

Valor	Significado
taLeftJustify	O texto aparece alinhado ao longo da aresta esquerda do rótulo
taRightJustify	O texto aparece alinhado ao longo da aresta direita do rótulo
taCenter	O texto é centrado horizontalmente no rótulo

AutoSize — quando True faz com que o rótulo ajuste o seu tamanho automaticamente, e assim a área usada se adapte à altura e ao comprimento do texto; quando False, o rótulo tem um tamanho fixo.

FocusControl — indicar o controlo que receberá o foco quando o utilizador premir a tecla aceleradora especificada no rótulo (especificar uma tecla aceleradora e atribuir a AccelChar o valor True)

Layout — indicar a posição vertical do texto no interior do rótulo; pode ter um dos seguintes valores :

Valor	Significado
tlTop	O texto aparece no topo do rótulo
tlCenter	O texto está centrado verticalmente no rótulo
tlBottom	O texto aparece ao longo do fundo do rótulo

ShowAccelChar — quando True permite que o rótulo mostre o valor da tecla aceleradora sublinhado

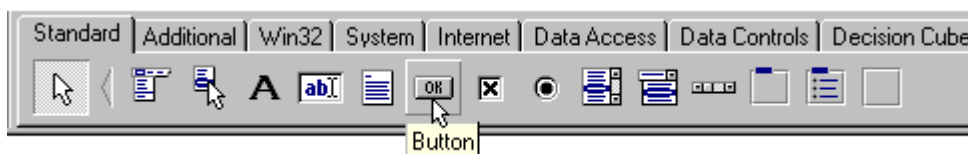
WordWrap — quando True permite que o rótulo mostre várias linhas de texto (estende-se até à margem direita e continua nas linhas seguintes); quando False limita o rótulo a uma única linha.

Caption, Color, Enabled, Font, Height, Left, Name, ParentColor, ParentFont, PopupMenu, Top, Visible e Width — ver componentes anteriores.

5.2. Eventos

OnClick, OnDblClick e OnMouseMove — ver componentes anteriores.

6. TButton



Utilizar a componente **TButton** para colocar numa forma um botão de comando standard do Windows. **TButton** apresenta várias propriedades para controlar o comportamento duma caixa de diálogo em construção. Os utilizadores escolhem os controlos através de botões para iniciar acções.

6.1. Propriedades

Cancel — quando é True o evento **OnClick** do botão é executado quando o utilizador prime *Esc*. Se existirem mais do que um botão com **Cancel**, apenas é chamado evento **OnClick** o botão visível com menor ordem do Tab.

Default — quando True o evento **OnClick** do botão é executado quando o utilizador prime *Enter*. Se existirem mais do que um botão com **Default**, apenas é chamado evento **OnClick** o botão visível com menor ordem do Tab.

ModalResult — indica, através dos valores que lhe são atribuídos, como e qual o botão que fecha a sua forma (modal) pai. Quando o botão é clicado, a propriedade **ModalResult** de seu pai assume o mesmo valor da propriedade **ModalResult** do botão.

TabOrder — indicar a ordem pela qual os controlos filhos são visitadas quando o utilizador pressiona a tecla Tab. O controlo cujo valor de **TabOrder** é igual a 0 é aquele que fica com o foco quando a forma aparece pela primeira vez. Inicialmente, a ordem do Tab é sempre a ordem pela qual os controlos foram adicionados à forma : o primeiro tem valor 0, o segundo 1, o terceiro 2, etc..

TabStop — quando True indica que o utilizador pode usar o Tab para alcançar o controlo (ou seja, o controlo está na ordem do Tab).

Caption, Enabled, Font, Height, Left, Name, ParentFont, PopupMenu, Top, Visible e Width — ver componentes anteriores.

6.2. Eventos

OnEnter — ocorre quando uma componente recebe o foco de entrada (a componente se torna activa)

OnExit — ocorre quando o foco de entrada transfere-se de uma componente para outra (uma componente deixa de estar activa)

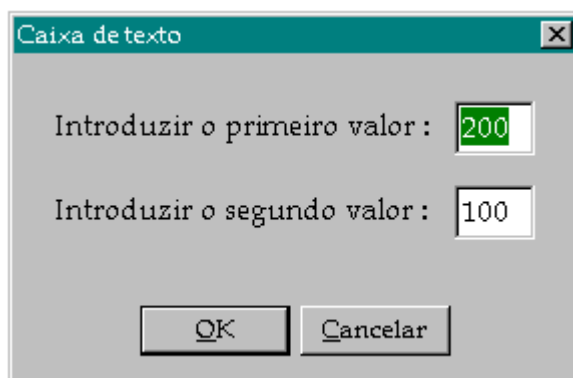
OnClick e OnMouseMove — ver componentes anteriores.

7. TEdit



Usa-se um objecto **TEdit** para apresentar um controlo de edição padrão do Windows numa forma. Os controlos de adição são usados para ir buscar ou mostrar texto aos utilizadores. Quando o texto apenas é mostrado ao utilizador, escolhe-se um controlo de edição para permitir aos utilizadores seleccionar texto e copiá-lo para o Clipboard. Se não for necessário ter capacidade de selecção, deve-se utilizar um rótulo (*label*).

Este tipo de objecto utiliza-se para criar caixas de texto (*text box*), quando combinadas com rótulos e uma forma (figura seguinte). As caixas de texto são fundamentalmente controlos de edição que exigem do utilizador a inserção de informação. O utilizador pode aceitar o texto corrente (se existir), introduzir texto novo, proceder à sua edição, eliminação ou modificação. Apesar da maioria das caixas de texto possuírem apenas uma linha, existem muitas que contêm várias. Uma caixa de texto suporta o deslocamento da informação através do teclado ou do rato.



7.1. Propriedades

AutoSelect — quando True indica que todo o texto no editor é automaticamente seleccionado quando o controlo recebe o foco. *AutoSelect* apenas se aplica a editores de uma única linha. Usa-se *AutoSelect* quando é mais provável o utilizador substituir o texto do editor do que acrescentar-lhe mais algum.

AutoSize — quando True o editor ajusta automaticamente o seu tamanho para que a área usada se adapte à altura do texto (o tamanho do editor é reajustado sempre que ocorra uma alteração que possa afectar a altura do editor, tal como uma alteração da fonte ou do estilo do bordo); quando False o editor tem uma altura fixa.

BorderStyle — os valores indicam a intensidade com que a área usada do editor é realçada : bsSingle (o editor tem uma única linha no bordo à volta da área usada) e bsNone (não existe qualquer bordo).

MaxLength — limitar o número de caracteres que podem ser introduzidos no editor : o valor 0 indica que não existe limite definido da aplicação para o comprimento. No entanto, especificar **MaxLength** não cortará o texto existente, apenas previne o utilizador a partir da adição de mais texto após se atingir o limite de **MaxLength** caracteres.

ParentCtl3D — para garantir que todas as componentes numa forma tenham uma aparência uniforme. Por exemplo, para conseguir que todas as componentes numa forma apareçam a 3D, atribuir à propriedade Ctl3D da forma (pai) o valor True e à propriedade **ParentCtl3D** de cada componente filho o valor True. Não só todas as componentes terão uma aparência a 3D, mas para alterar-se para uma aparência uniforme a 2D, apenas tem-se que alterar a propriedade Ctl3D da forma (colocar a False) e todas as componentes ficarão com uma aparência a 2D.

ReadOnly — quando True indica que o utilizador não pode alterar o texto do editor (embora permita que o utilizador o seleccione, o qual pode então ser manipulado pela aplicação ou copiado para o Clipboard); quando False permite ser editado e alterado o conteúdo do editor. No entanto,

Text — contém a string que foi introduzida pelo utilizador, que servirá como informação de entrada.

Color, Ctl3D, Enabled, Font, Height, Left, Name, ParentColor, ParentFont, PopupMenu, TabOrder, TabStop, Top, Visible e Width — ver componentes anteriores.

Modified — determina se a propriedade Text da caixa de edição foi alterada. Se alguma aplicação altera directamente a propriedade Text de uma caixa de edição, é atribuída a Modified o valor True.

SelLength — determina o comprimento, em caracteres, do texto seleccionado.

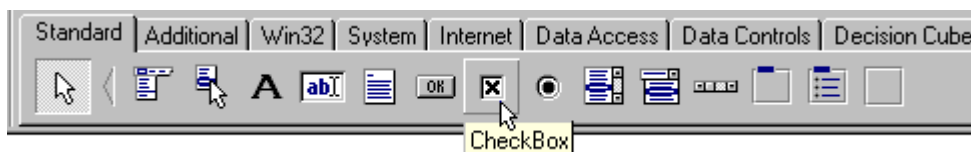
SelStart — determina a posição do primeiro carácter seleccionado, onde 0 está associado ao primeiro carácter. Se não existir texto seleccionado, SelStart indica a posição do cursor.

7.2. Eventos

OnChange — especificar a acção a ser processada sempre que o texto do editor é alterado. Usar a propriedade Modified para verificar se ocorreu alguma alteração. A propriedade Text do editor estará já actualizada para reflectir quaisquer trocas.

OnClick, OnDblClick, OnEnter, OnExit e OnMouseMove — ver componentes anteriores.

8. TCheckBox



A componente **TCheckBox** representa as caixas de marcação (*check box*) do Windows. Uma componente **TCheckBox** apresenta uma opção para o utilizador. O utilizador pode marcar a caixa para seleccionar a opção, ou desmarcá-la para não seleccionar a opção.

8.1. Propriedades

Alignment — indica a posição do título em relação à caixa de marcação (*check box*); os valores possíveis são os seguintes : taRightJustify (o título aparece à direita da caixa de marcação) e taLeftJustify (o título aparece à esquerda da caixa de marcação)

AllowGrayed — quando True indica que a caixa de marcação tem 3 estados possíveis : marcado (*checked*), não marcado (*unchecked*) e marcado acinzentado (*grayed*). Quando False a caixa de marcação tem apenas 2 estados possíveis : marcado e não marcado.

Checked — indica se a caixa de marcação está ou não marcada. Quando True, aparece uma marca (✓) na caixa de marcação (State = cbChecked); quando False, a caixa de marcação não tem a marca (State = cbUnchecked) ou tem a marca acinzentada (State = cbGrayed).

State — indicar se a caixa de marcação está marcada, não marcada ou acinzentada. Os valores possíveis são os seguintes :

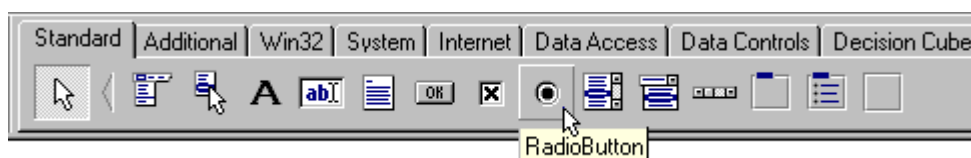
Valor	Significado
cbUnchecked	A caixa de marcação não tem marca, indicando que a opção não foi seleccionada
cbChecked	A caixa de marcação tem uma marca, indicando que a opção foi seleccionada
cbGrayed	A caixa de marcação tem uma marca, mas está acinzentada

Caption, Color, Ctl3D, Enabled, Font, Height, Left, Name, ParentColor, ParentCtl3D, ParentFont, PopupMenu, TabOrder, TabStop, Top, Visible e Width — ver componentes anteriores.

8.2. Eventos

OnClick, OnEnter, OnExit e OnMouseMove — ver componentes anteriores.

9. TRadioButton



Utilizar o **TRadioButton** para adicionar um botão de rádio a uma forma. Usar botões de rádio para apresentar um conjunto de opções mutuamente exclusivas ao utilizador — isto é, apenas um botão de rádio do conjunto pode ser seleccionado de cada vez. Quando o utilizador selecciona um botão de rádio, o botão de rádio seleccionado antes torna-se não seleccionado. Os botões de rádio são frequentemente agrupados numa caixa de grupo de rádio. Primeiro adiciona-se a caixa de grupo de rádio à forma, e depois escolhe-se os botões de rádio.

Por defeito, todos os botões de rádio que estão directamente controlados pela mesma componente, tal como um TRadioGroup, são agrupados. Por exemplo, dois botões de rádio numa forma podem estar marcados num certo momento apenas se forem controlados por componentes diferentes, tal como duas caixas de grupo diferentes.

9.1. Propriedades

Alignment — indicar a posição do título em relação ao botão de rádio (diferente do alinhamento do título na área do texto do botão, que é sempre à esquerda); os valores possíveis são os seguintes :

Valor	Significado
taLeftJustify	O título aparece à esquerda do botão de rádio
taRightJustify	O título aparece à direita do botão de rádio

Checked — quando True indica que a opção representada pelo botão de rádio está seleccionada, quando False, indica que não está seleccionada.

Caption, Color, Ctl3D, Enabled, Font, Height, Left, Name, ParentColor, ParentCtl3D, ParentFont, PopupMenu, TabOrder, TabStop, Top, Visible e Width — ver componentes anteriores.

9.2. Eventos

OnClick, OnDblClick, OnEnter, OnExit e OnMouseMove — ver componentes anteriores.

10. TListBox



A componente **TListBox** é um invólucro para o controlo da caixa de listagem do Windows. Usar o **TListBox** para mostrar a lista descendente com capacidade de deslocamento de itens que o utilizador pode seleccionar, adicionar ou apagar.

10.1. Propriedades

Columns — indicar o número de colunas que, numa caixa de listagem com várias colunas, são visíveis sem ter que utilizar uma barra de deslocamento horizontal.

ExtendedSelect — trabalha em conjunto com MultiSelect : quando True (e MultiSelect True), o utilizador pode seleccionar um item e, mantendo em baixo a tecla Shift, seleccionar uma outra e todos os itens entre aqueles dois itens tornam-se seleccionados.

IntegralHeight — controlar a altura da caixa de listagem : quando True (e Style é *lsOwnerDrawFixed*) a caixa de listagem será sempre um múltiplo do valor de *ItemHeight* (os itens nunca serão parcialmente mostrados, mas sim totalmente); quando False, a altura da caixa de listagem pode ser atribuído qualquer valor, mas o último item pode não ser totalmente mostrado).

ItemHeight — quando Style é *lsOwnerDrawFixed* controla a altura de um item na caixa de listagem. Pode também afectar a altura da própria caixa de listagem : quando IntegralHeight é True, a altura da caixa de listagem é sempre um incremento do valor de *ItemHeight*.

Se o valor de Style é *lsStandard* ou *lsOwnerDrawVariable*, o valor de *ItemHeight* é ignorado.

Items — contém as opções (strings) que aparecem na caixa de listagem. Usar **Items** para adicionar, inserir, apagar e mover itens. Por defeito, os itens são do tipo TStrings. Para se manipular estes itens têm que se utilizar as propriedades e métodos de TStrings, como por exemplo a seguinte linha de código para adicionar o texto 'Nova opção' à caixa de listagem : `ListBox1.Items.Add ('Nova opção');`

MultiSelect — quando True permite que o utilizador selecione vários itens ao mesmo tempo da caixa de listagem.

Sorted — quando True indica que os itens estão ordenados alfabeticamente na caixa de listagem

Style — especificar se a caixa de listagem é standard (mostrar texto — valor por defeito) ou *owner-draw* (pode mostrar imagens gráficas). Os valores possíveis são os seguintes :

Valor	Significado
lbStandard	Todos os itens são strings, com cada item com a mesma altura
lbOwnerDrawFixed	Cada item tem a altura especificada na propriedade ItemHeight
lbOwnerDrawVariable	Os itens podem ser de alturas variadas

TabWidth — especifica o tamanho dos Tabs na caixa de listagem (o número de unidades base de diálogo mostradas por cada carácter Tab que aparece nas strings duma caixa de listagem). Se tem o valor 0 (valor por defeito), os Tabs assumem por defeito o comprimento especificado pelo Windows, que é 2 unidade base de diálogo.

Align, BorderStyle, Color, Ctl3D, Enabled, Font, Height, Left, Name, ParentColor, ParentCtl3D, ParentFont, PopupMenu, TabOrder, TabStop, Top, Visible e Width — ver componentes anteriores.

ItemIndex — especifica o índice do item seleccionado da lista da caixa de listagem. Quando a aplicação está em execução (runtime), pode-se atribuir a *ItemIndex* o índice do item a ser seleccionado. O *ItemIndex* do primeiro item da caixa de listagem é 0. Se não estiver seleccionado qualquer item, o valor é -1, que é o valor por defeito, a menos que MultiSelect seja True.

SelCount — indica o número de itens seleccionados numa caixa de listagem, quando é permitido a selecção múltipla (MultiSelect tem o valor True).

Selected — indica que o estado de um item específico duma caixa de listagem. Se o item especificado pelo parâmetro Index está seleccionado na caixa de listagem, o valor de *Selected* é True. Se o item não está seleccionado, *Selected* é False : `Selected [Index : Integer] : Boolean;`

TopIndex — especifica o índice do item que aparece no topo da caixa de listagem. Esta propriedade pode ser usada, por exemplo, para alterar o item mais alto para um item diferente do primeiro da lista.

10.2. Eventos

OnClick, OnDblClick, OnEnter, OnExit e OnMouseMove — ver componentes anteriores.

11. TComboBox



Uma componente do tipo **TComboBox** cria a uma caixa listagem combinada do Windows. Os utilizadores podem seleccionar um item a partir da lista ou escrever directamente na caixa de edição.

11.1. Propriedades

DropDownCount — indica o número máximo de itens visíveis ao mesmo tempo na lista *drop-down* (por defeito 8). Se *DropDownCount* é maior do que o número total de itens da lista *drop-down* então todos eles são visíveis ao mesmo tempo; se for menor aparecerá uma barra de deslocamento.

Style — indicar como a lista e a região de edição da caixa de listagem combinada deve aparecer. Os valores possíveis são :

Valor	Significado
csSimple	Cria uma caixa de edição com uma lista fixa (<i>list box</i>) por baixo; o tamanho da lista é determinada pelo <i>Height</i> da caixa de listagem combinada
csDropDown	Cria uma lista <i>drop-down</i> com uma caixa de edição com entrada manual de texto; todos os itens são strings da mesma altura
csDropDownList	Cria uma lista <i>drop-down</i> sem caixa de edição — não se pode introduzir texto manualmente; todos os itens são strings da mesma altura
csOwnerDrawFixed	Cria uma lista <i>drop-down</i> traçada pessoalmente com uma caixa de edição para entrada manual de texto; cada item na lista tem uma altura fixada pela propriedade <i>ItemHeight</i>
csOwnerDrawVariable	Cria uma lista <i>drop-down</i> traçada pessoalmente com uma caixa de edição para entrada manual de texto; a lista de itens pode ter várias alturas

Color, Ctl3D, Enabled, Font, Height, ItemHeight, Items, Left, MaxLength, Name, ParentColor, ParentCtl3D, ParentFont, PopupMenu, Sorted, TabOrder, TabStop, Text, Top, Visible e Width — ver componentes anteriores.

ItemIndex, SelLength, SelStart e SelText — ver componentes anteriores.

11.2. Eventos

OnChange — especificar uma acção a ser processada imediatamente após o utilizador editar texto na região de edição ou seleccionar um item a partir da lista. A propriedade *Text* fornece o novo valor da região de edição.

OnClick, OnDblClick, OnEnter e OnExit — ver componentes anteriores.

12. TScrollBar



TScrollBar é uma barra de deslocamento (*scroll bar*) do Windows, que é utilizada para deslocar o conteúdo de uma janela, forma ou controlo. Usa-se o **TScrollBar** para adicionar uma barra de deslocamento em qualquer local de uma forma. Muitos controlos têm propriedades que adicionam barras de deslocamento, as quais são uma parte integrante do controlo.

12.1. Propriedades

Kind — indica se a barra de deslocamento é horizontal (sbHorizontal) ou vertical (sbVertical).

LargeChange — indica qual a alteração do valor de Position quando o utilizador clica sobre a barra de deslocamento ou prime as teclas PgUp ou PgDn. Paralelamente, a caixa de deslocamento desloca-se aquele número de posições sobre a barra. Por exemplo, se Max é 300 e Min é 1, a barra de deslocamento pode assumir 300 posições. Se a propriedade **LargeChange** tem o valor de 100 e a propriedade Position é 1, o utilizador pode clicar na barra de deslocamento 3 vezes seguidas para que a caixa de deslocamento se mova até ao final da barra de deslocamento ($300/100 = 3$).

Max — é a posição máxima representada pela barra de deslocamento (valor máximo que a propriedade Position pode tomar)

Min — é a posição mínima representada pela barra de deslocamento (valor mínimo que a propriedade Position pode tomar)

Position — é a posição corrente da barra de deslocamento. Este valor pode ser usado para indicar como deslocar uma qualquer das componentes controladas pela barra de deslocamento. Quando o utilizador desloca a barra de deslocamento, o valor de *Position* altera-se. Pode-se atribuir valores a Position utilizando instruções de programação para mover a caixa de deslocamento da barra de deslocamento

SmallChange — especifica qual a alteração no valor de Position quando o utilizador clica nas setas de deslocamento da barra de deslocamento ou prime as teclas com setas do teclado. Paralelamente, a caixa de deslocamento desloca-se aquele número de posições sobre a barra. Por exemplo, se Max é 100 e Min é 1, a barra de deslocamento pode assumir 100 posições. Se o valor da propriedade *SmallChange* é 5 e a propriedade Position é 1, o utilizador pode clicar na setas de deslocamento para baixo ou para a direita 20 vezes seguidas para que a caixa de deslocamento se mova para o final da barra de deslocamento.

Ctl3D, Enabled, Height, Left, Name, ParentCtl3D, PopupMenu, TabOrder, TabStop, Top, Visible e Width — ver componentes anteriores.

12.2. Eventos

OnChange — ocorre imediatamente após se alterar o valor de Position. Utilizado para controlar outro objecto ou grupo de objectos usando TScrollBar. Adaptar os objectos controlados pela barra de deslocamento para reflectir o novo valor de Position. *OnChange* ocorre sempre que o valor de Position é alterado pelo utilizador ou através de instruções em Delphi. Se o valor de Position é alterado pelo utilizador, OnChange ocorre imediatamente após o evento OnScroll.

OnScroll — ocorre quando o utilizador desloca a barra de deslocamento com o rato ou o teclado. Utilizado para adaptar o valor de Position quando o utilizador desloca a barra de deslocamento.

procedure (Sender : TObject ; ScrollCode : TScrollCode ; var ScrollPos : Integer) of Object ;
O parâmetro ScrollPos contém o novo valor que está prestes a ser atribuído à propriedade Position como resultado do deslocamento. Modificar ScrollPos para alterar o resultado do deslocamento.

O parâmetro ScrollCode indica o tipo de acção do utilizador que apenas deslocou a barra de deslocamento. O parâmetro ScrollCode tem um dos seguintes valores :

Valor	Significado
scLineUp	Clicou-se na seta de deslocamento do topo ou da esquerda, ou premiu-se a tecla de seta para cima (Up) ou para a esquerda (Left)
scLineDown	Clicou-se na seta de deslocamento do fundo ou da direita, ou premiu-se a tecla de seta para baixo (Down) ou para a direita (Right)
scPageUp	Clicou-se na área à esquerda da caixa de deslocamento ou premiu-se a tecla PgUp
scPageDown	Clicou-se na área à direita da caixa de deslocamento ou premiu-se a tecla PgDn
scPosition	Posicionou-se a caixa de deslocamento e soltou-se
scTrack	Está-se a mover a caixa de deslocamento
scTop	A caixa de deslocamento foi movida para o topo ou para a extremidade esquerda da barra de deslocamento
scBottom	A caixa de deslocamento foi movida para o fundo ou para a extremidade direita na barra de deslocamento
scEndScroll	Terminou-se o movimento da caixa de deslocamento na barra de deslocamento

OnEnter e **OnExit** — ver componentes anteriores.

13. TGroupBox



A componente **TGroupBox** cria uma caixa de grupo (*group box*) standard do Windows, sendo utilizada para agrupar controlos relacionados entre si numa forma. Quando uma componente é colocada dentro de uma caixa de grupo, a caixa de grupo torna-se pai daquela componente.

13.1. Propriedades

Align, **Caption**, **Color**, **Ctl3D**, **Enabled**, **Font**, **Left**, **Name**, **ParentColor**, **ParentCtl3D**, **ParentFont**, **PopupMenu**, **TabOrder**, **TabStop**, **Top**, **Visible** e **Width** — ver componentes anteriores.

13.2. Eventos

OnClick, **OnDblClick**, **OnEnter**, **OnExit** e **OnMouseMove** — ver componentes anteriores.

14. TRadioGroup



A componente **TRadioGroup** representa um grupo de botões de rádio que funcionam em conjunto. Um objecto **TRadioGroup** é uma caixa de grupo especial que contém apenas botões de rádio. Assim, dois botões de rádio numa forma podem estar marcados ao mesmo tempo apenas se eles estão colocados em controlos separados, tais como caixas de grupo.

Para adicionar botões de rádio a um **TRadioGroup**, editar a propriedade **Items** no Object Inspector. Cada opção (string) do **Items** produz um botão de rádio que aparece na caixa de grupo com a opção como seu título. O valor da propriedade **ItemIndex** determina qual o botão de rádio que está actualmente seleccionado.

Para mostrar os botões de rádio numa única coluna ou em várias colunas, atribuir valores à propriedade **Columns** (este valor corresponde ao número de colunas).

14.1. Propriedades

Columns — especifica o número de colunas do grupo de botões de rádio (*radio group*). Os seus valores podem variar entre 1 e 16 (o valor por defeito é 1, o que significa que os botões de rádio são agrupados numa simples linha vertical)

ItemIndex — indica qual o botão de rádio do grupo que está actualmente seleccionado (o primeiro tem 0). O valor de **ItemIndex** modifica-se em *runtime* quando o utilizador selecciona um dos botões de rádio. Se precisar que um dos botões apareça seleccionado quando a aplicação arranca, atribuir o índice daquele botão a **ItemIndex** em *design time*; caso contrário, deixar o **ItemIndex** com o valor por defeito (-1), que significa que nenhum botão está seleccionado.

Align, Caption, Color, Ctl3D, Enabled, Font, Height, Items, Left, Name, ParentColor, ParentCtl3D, ParentFont, PopupMenu, TabOrder, TabStop, Top, Visible e Width — ver componentes anteriores.

14.2. Eventos

OnClick, OnDblClick e OnEnter — ver componentes anteriores.

15. TOpenDialog



A componente **TOpenDialog** mostra a caixa de diálogo (modal) comum *Open* do Windows para seleccionar e abrir ficheiros. A caixa de diálogo não aparece em *runtime* até que seja activada por uma chamada ao método **Execute**. Quando o utilizador clica *Open*, a caixa de diálogo fecha-se e o ficheiro ou ficheiros seleccionados são armazenados na propriedade **Files**.

15.1. Propriedades

Ctl3D — quando **True**, a caixa de diálogo e os controlos dela aparecem a 3D; quando **False**, o diálogo e os seus controlos aparecem lisos.

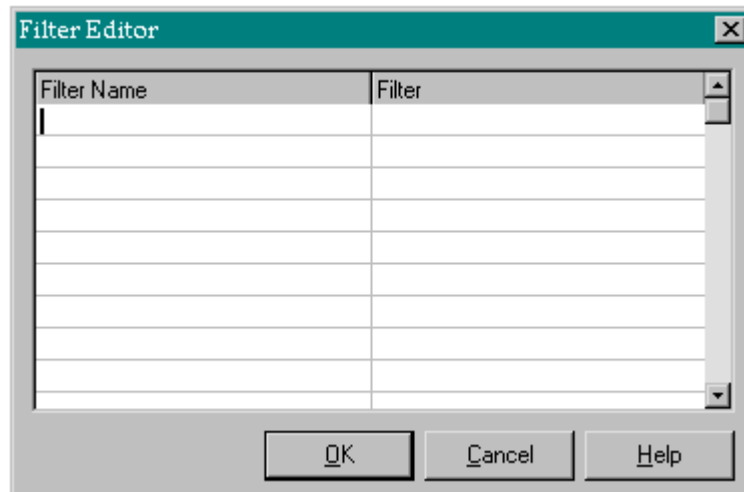
DefaultExt — especifica a extensão de ficheiro por defeito, a qual é acrescentada automaticamente ao nome do ficheiro seleccionado, excepto se o nome do ficheiro já incluía uma extensão registada. Se o utilizador seleccionar um nome de ficheiro com uma extensão não registada, **DefaultExt** é acrescentado às extensões. As extensões com mais de 3 caracteres não são suportadas. Não se inclui o ponto (.) que divide o nome do ficheiro da sua extensão.

FileName — devolve o nome e o caminho completos da directoria do ficheiro seleccionado (é o mesmo do primeiro item na propriedade **Files**). Para fazer aparecer um nome de ficheiro por defeito na caixa de edição da caixa de diálogo, atribui-se um valor a **FileName** no Object Inspector ou através

de código em Pascal. Com instruções em Pascal a alteração de *FileName* não tem qualquer efeito enquanto a caixa de diálogo estiver activa.

Filter — determina as máscaras de ficheiros (filtros) disponíveis na caixa de diálogo. A caixa de diálogo para seleccionar ficheiro inclui uma caixa de listagem combinada. Quando o utilizador escolhe um tipo de ficheiro da lista, apenas os ficheiros do tipo seleccionado são mostrados na caixa de diálogo.

Para configurar as máscaras dos ficheiros em *design time*, clicar nas reticências (...) à direita da propriedade *Filter* no Object Inspector, para abrir o editor de filtros (ver figura seguinte – *Filter editor*).



Na coluna da esquerda do editor, por baixo de *Filter Name*, escrever uma breve descrição de cada tipo de ficheiro que estará disponível em runtime. Na coluna da direita, por baixo de *Filter*, escrever a máscara do ficheiro correspondente a cada descrição. Por exemplo, a descrição "Text files" pode aparecer à esquerda da máscara "*.TXT", e a descrição "Pascal source files" pode aparecer à esquerda da máscara "*.PAS". Uma vez que a descrição aparece na lista *drop-down* em *runtime*, é muitas vezes útil mostrar a máscara explicitamente na descrição (por exemplo, "Text files (*.txt)").

Para criar máscaras de ficheiros utilizando código em Pascal, atribuir um valor à propriedade **Filter** que consista de uma descrição e uma máscara separada por uma barra vertical (|). Não incluir espaços junto da barra vertical. Por exemplo,

```
OpenDialog1.Filter := 'Text files (*.txt) | *.TXT'
```

Os vários filtros devem estar separados por barras verticais. Por exemplo,

```
OpenDialog1.Filter := 'Text files (*.txt) | *.TXT | Pascal files (*.pas) | *.PAS'
```

Para incluir várias máscaras num único filtro, separar as máscaras com ponto e vírgulas (;). Isto funciona tanto no Object Inspector como utilizando código em Pascal. Por exemplo,

```
OpenDialog1.Filter := 'Pascal files | *.PAS;*.DPK;*.DPR'
```

Se nenhum valor é atribuído ao **Filter**, a caixa de diálogo mostra todos os tipos de ficheiros.

FilterIndex — determina que filtro é seleccionado por defeito quando a caixa de diálogo é aberta : atribuir o valor de 1 para que o tipo de ficheiro escolhido por defeito seja o primeiro da lista, ou o valor 2 para que o escolhido por defeito seja o segundo, etc.. Se o valor de *FilterIndex* está fora do intervalo, o primeiro tipo de ficheiro listado no *Filter* é o por defeito.

InitialDir — especifica a directoria corrente quando se abre a caixa de diálogo. Por exemplo, para colocar o diálogo na directoria "C:\Disciplinas\Algoritmos", atribuir a *InitialDir* o valor de C:\Disciplinas\Algoritmos. Se nenhum valor é atribuído a *InitialDir*, ou se a directoria especificada não existe, a caixa de diálogo abre na directoria corrente.

Options — determina a aparência e o comportamento da caixa de diálogo de selecção de ficheiro. Os valores possíveis são os seguintes :

Valor	Significado
ofAllowMultiSelect	Permite que o utilizador seleccione mais do que um ficheiro no diálogo
ofCreatePrompt	Provoca uma mensagem de aviso se o utilizador tentar seleccionar um ficheiro inexistente
ofExtensionDifferent	Esta flag é desligada em <i>runtime</i> se o nome do ficheiro seleccionado tem uma extensão que difere da DefaultExt
ofFileMustExist	Gera uma mensagem de erro se se seleccionar um ficheiro inexistente
ofHideReadOnly	Remove a caixa de marcação “Open As Read Only” do diálogo
ofNoChangeDir	Após o utilizador clicar OK, repõe a directoria corrente seja qual for aquela utilizada antes da abertura do diálogo de selecção de ficheiro
ofNoDereferenceLinks	Para o utilizador seleccionar um atalho, atribuir a FileName o caminho e o nome do ficheiro do próprio atalho (o ficheiro .LNK), em vez do ficheiro ligado ao atalho
ofNoLongNames	Mostra apenas nomes de ficheiros com 8 caracteres (formato 8.3)
ofNoNetworkButton	Remove o botão Network (que abre um diálogo “Map Network Drive”) do diálogo de selecção de ficheiros
ofNoReadOnlyReturn	Provoca uma mensagem de erro se o utilizador tentar seleccionar um ficheiro apenas de leitura (<i>read-only file</i>)
ofNoTestFileCreate	Aplica-se só quando o utilizador tenta guardar um ficheiro numa directoria partilhada da rede criada não modificada (<i>create-no-modify</i>)
ofNoValidate	Permite a selecção de nomes de ficheiros com caracteres inválidos
ofOldStyleDialog	Cria o diálogo de estilo antigos para selecção de ficheiros
ofOverwritePrompt	Provoca uma mensagem de aviso quando o utilizador tenta seleccionar um ficheiro que esteja a ser usado
ofPathMustExist	Provoca uma mensagem de erro quando o utilizador tenta seleccionar um ficheiro com um caminho de directoria inexistente
ofReadOnly	Selecção a caixa de marcação “Open As Read Only” por defeito quando a o diálogo é aberto.
ofShareAware	Ignora erros partilhados e permite que os ficheiros sejam seleccionados mesmo quando ocorre violações de partilha
ofShowHelp	Apresenta um botão de Ajuda (Help) no diálogo

Title — para especificar o texto que aparece na barra de título do diálogo para seleccionar ficheiro. Se não for atribuído qualquer valor, o diálogo assume o título “Open”.

Name — ver componentes anteriores.

15.2. Eventos

OnFolderChange — ocorre quando o utilizador muda de directoria, cujo conteúdo é mostrado na caixa de diálogo. Isto pode acontecer quando o utilizador clica duas vezes (double-clicks) numa directoria, clica a seta para cima (Up arrow), ou utiliza a caixa de listagem que está no topo do diálogo para navegar através da estrutura de directorias.

OnSelectionChange — ocorre quando o utilizador faz algo para alterar a lista apresentada na caixa de diálogo. Isto pode incluir a abertura do diálogo para seleccionar ficheiro, obscurecer um ficheiro ou directoria, seleccionar um novo filtro, seleccionar uma nova directoria, ou criar uma nova pasta.

OnTypeChange — ocorre quando o utilizador selecciona um novo filtro a partir da caixa de listagem do tipo de ficheiros (“Files of Type”), que está no fundo do diálogo.

OnClose e **OnShow** — ver componentes anteriores.

16. TSaveDialog



TSaveDialog apresenta um diálogo *Save As* para guardar ficheiros. **TSaveDialog** apresenta uma caixa de diálogo modal do Windows para seleccionar e guardar ficheiros. O diálogo não aparece em *runtime* até que seja activado através de uma chamada ao método `Execute`. Quando o utilizador clica `Save`, o diálogo fecha-se e o nome de ficheiro seleccionado é armazenado na propriedade `FileName`.

16.1. Propriedades

`Ctl3D`, `DefaultExt`, `FileName`, `Filter`, `FilterIndex`, `InitialDir`, `Name`, `Options` e `Title` — ver componente `TOpenDialog`.

16.2. Eventos

`OnClose`, `OnFolderChange`, `OnSelectionFolder`, `OnShow` e `OnTypeChange` — ver componente `TOpenDialog`.

Índice

1. TForm	1
1.1. Propiedades -----	1
1.2. Eventos-----	3
2. TMenuItem	3
2.1. Propiedades -----	3
2.2. Eventos -----	4
3. TMainMenu	4
3.1. Propiedades -----	4
3.2. Eventos -----	4
4. TPopupMenu	4
4.1. Propiedades -----	5
4.2. Eventos -----	5
5. TLabel	5
5.1. Propiedades -----	5
5.2. Eventos -----	6
6. TButton	6
6.1. Propiedades -----	6
6.2. Eventos -----	7
7. TEdit	7
7.1. Propiedades -----	8
7.2. Eventos -----	8

8. TCheckBox	9
8.1. Propriedades -----	9
8.2. Eventos -----	9
9. TRadioButton	9
9.1. Propriedades -----	10
9.2. Eventos -----	10
10. TListBox	10
10.1. Propriedades -----	10
10.2. Eventos -----	11
11. TComboBox	11
11.1. Propriedades -----	12
11.2. Eventos -----	12
12. TScrollBar	12
12.1. Propriedades -----	13
12.2. Eventos -----	13
13. TGroupBox	14
13.1. Propriedades -----	14
13.2. Eventos -----	14
14. TRadioGroup	14
14.1. Propriedades -----	15
14.2. Eventos -----	15
15. TOpenDialog	15
15.1. Propriedades -----	15
15.2. Eventos -----	17
16. TSaveDialog	18
16.1. Propriedades -----	18
16.2. Eventos -----	18
