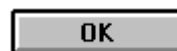


III. Delphi alapvezérlők

1. Gombok

Nyomógomb



Tulajdonságok

<i>Elhelyezkedés</i> pixel	Left Align	Top <i>RT</i> alTop...	Height BoundRect	Width <i>RT</i> TRect
<i>Bill. kezelés</i>	Default	Cancel		
<i>Engedély, láthatóság</i>	Enabled	Visible	Showing <i>RT</i>	
<i>Kurzor</i>	Cursor	TCursor	crXXX (egész)	
<i>Tabulátor</i>	TabOrder	TabStop		
<i>Nevek</i>	Name	Caption		
<i>Szülő</i>	Parent RT	Owner <i>RT</i> R/O		
<i>Spec. adatok</i>	Tag	longint		
<i>Vonzolás</i>	Dragmode Tdragmode dmAutomatic dmManual	DragCursor Tcursor crXXX		
<i>Fontok</i>	Font .Color .Name .Size .Style	Tfont Tcolor inch/72 TfontStyle fsBold...	ParentFont boolean	
<i>Információk</i>	Hint String	ShowHint boolean	Parent ShowHint	Boolean
<i>Menük</i>	PopupMenu TpopupMenu	HelpContext		
<i>Egyéb</i>	ModalResult Handle	<i>RT</i> HWND	Component Index	<i>RT</i> R/O

Események

OnClick TNotifyEvent = procedure (Sender: TObject) of object;
OnDragDrop TDragDropEvent = procedure(Sender, Source: TObject;
X, Y: Integer) of object;
OnDragOver TDragOverEvent = procedure(Sender, Source: TObject;
X, Y: Integer; State: TDragState;
var Accept: Boolean) of object;
dsXXX

<i>OnEndDrag</i>	TEndDragEvent = procedure(Sender, Target: TObject; X, Y: Integer) of object;
<i>OnEnter</i>	TNotifyEvent = procedure (Sender: TObject) of object;
<i>OnExit</i>	TNotifyEvent = procedure (Sender: TObject) of object;
<i>OnKeyDown</i>	TKeyEvent = procedure (Sender: TObject; var Key: Word; Shift: TShiftState) of object;
<i>OnKeyUp</i>	VK_XXX , ssXXX
<i>OnKeyPress</i>	Type TKeyPressEvent = procedure (Sender: TObject; var Key: Char) of object;
<i>OnMouseDown</i>	TMouseEvent = procedure (Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer) of object;
<i>OnMouseUp</i>	
<i>OnMouseMove</i>	MbXXX

Metódusok

<i>Create</i>	Constructor Create(AOwner: TComponent);
<i>Destroy</i>	Destructor Destroy;
<i>Free</i>	Procedure Free;
<i>Invalidate</i>	Procedure Invalidate;
<i>Repaint</i>	Procedure Repaint; <i>nem töröl</i>
<i>Refresh</i>	Procedure Refresh; <i>töröl</i>
<i>Update</i>	Procedure Update; <i>minden újrafestés</i>
<i>Hide</i>	Procedure Hide;
<i>Show</i>	Procedure Show
<i>SetBounds</i>	Procedure Setbounds(ALeft, ATop, AWidth, AHeight: Integer);
<i>ScaleBy</i>	Procedure ScaleBy(M, D: Integer); M/D
<i>ScrollBy</i>	Procedure ScrollBy(DeltaX, DeltaY: Integer); a tartalom
<i>ClientToScreen</i>	Function ClientToScreen(Point: TPoint): TPoint;
<i>ScreenToClient</i>	Function ScreenToClient(Point: TPoint): TPoint;
<i>BringToFront</i>	Procedure BringToFront;
<i>SendToBack</i>	Procedure SendToBack;
<i>CanFocus</i>	Function CanFocus: Boolean;
<i>SetFocus</i>	Procedure SetFocus;
<i>Focused</i>	Function Focused: Boolean;
<i>SetTextBuf</i>	Procedure SetTextBuf(Buffer: PChar);
<i>GetTextBuf</i>	Function GetTextBuf(Buffer: PChar; BufSize: Integer): Integer;
<i>GetTextLen</i>	Function GetTextLen: Integer;
<i>BeginDrag</i>	Procedure BeginDrag(Immediate: Boolean);
<i>Dragging</i>	Function Dragging: Boolean;
<i>EndDrag</i>	Procedure EndDrag(Drop: Boolean);
<i>FindComponent</i>	Function FindComponent(const AName: string): TComponent;

Elhelyezkedés

```
var x      : boolean;

procedure TForm1.FormActivate(Sender: TObject);
begin
  x:=false;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  If x Then
  begin
    Button1.Height := Button1.Height div 2;
    Button1.Width  := Button1.Width div 2;
  end
  else
  begin
    Button1.Height := Button1.Height*2;
    Button1.Width  := Button1.Width*2;
  end;
  x:=not(x);
end;
```

Mozgatás

```
procedure TForm1.FormActivate(Sender: TObject);
begin
  Button1.Left:=10;
  Button1.Top:=10;
  Button1.Width:=100;
  Button1.Height:=25;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  Button1.Left := Button1.Left+10;
  Button1.Top  := Button1.Top+10;
end;
```

Engedélyezés

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  Button2.Enabled := not (Button2.Enabled);
end;
```

Felirat

```
var x      : boolean;

procedure TForm1.FormActivate(Sender: TObject);
begin
  Button1.Caption := 'Egy';
  x:=false;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  If x Then
  begin
    Button1.Caption := 'Egy';
  end
  else
  begin
    Button1.Caption := 'Ketto';
  end;
  x:=not(x);
end;
```

A szülő háttérszíne

```
var x      : TColor;

procedure TForm1.FormActivate(Sender: TObject);
begin
  x:=clBlack;
  Form1.Color:=x;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  if (x-clBlack)<$02FFFFFF then
    x:=x+1
  else
    x:=clBlack;
  TForm1(Button1.Parent).Color:=x;
end;
```

A gomb egérmutatója

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  If Button1.Cursor > -17 Then
    Button1.Cursor:=Button1.Cursor-1
  Else
    Button1.Cursor:=0;
end;
```

A Tag tulajdonság

```
var x      : boolean=false;

procedure TForm1.FormActivate(Sender: TObject);
var s:string;
begin
  Button1.Tag:=100;
  Str(Button1.Tag,s);
  Button1.Caption:=s;
end;

procedure TForm1.Button1Click(Sender: TObject);
var s:string;
begin
  If x Then
    begin
      Str(Button1.Tag,s);
      Button1.Caption:=s;
    end
  Else
    Button1.Caption:='0';
  x:=not(x);
end;
```

Betűtípus-választás

```
var x      : integer;

procedure TForm1.FormActivate(Sender: TObject);
begin
  x:=0;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  If x >= Screen.Fonts.Count - 1 Then
    x:=0
  Else
    x:=x+1;
  Button1.Font.Name:= Screen.Fonts.strings[x];
  Label1.Caption:=Screen.Fonts.Strings[x];
end;
```

Betűméretek

```
var y : integer;

procedure TForm1.FormActivate(Sender: TObject);
begin
    y:=12;
    Label1.Font.Size:=y;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
    y:=y+2;
    Label1.Font.size:=y;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
    y:=y-2;
    Label1.Font.size:=y;
end;
```

Betűtípus módosítása

```
var x : boolean;

procedure TForm1.FormActivate(Sender: TObject);
begin
    x:=false;
    Button1.Font.style:=[];
    Button1.Font.style:=Button1.Font.style+[fsBold];
    Button1.Font.style:=Button1.Font.style+[fsItalic];
    Button1.Font.style:=Button1.Font.style+[fsUnderline];
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
    if x then
    begin
        Button1.Font.style:=Button1.Font.style+[fsBold];
        Button1.Font.style:=Button1.Font.style+[fsItalic];
        Button1.Font.style:=Button1.Font.style+[fsUnderline];
    end
    else
    begin
        Button1.Font.style:=Button1.Font.style-[fsBold];
        Button1.Font.style:=Button1.Font.style-[fsItalic];
        Button1.Font.style:=Button1.Font.style-[fsUnderline];
    end;
    x:=not(x);
end;
```

A SetBounds metódus használata

```
var x boolean;
procedure TForm1.Button1Click(Sender: TObject);
begin
    if x then
        Button1.SetBounds(10,10,100,25)
    else
        Button1.SetBounds(20,20,25,100);
    x:=not(x);
end;

procedure TForm1.FormActivate(Sender: TObject);
begin
    x:=true;
    Button1.SetBounds(20,20,25,100);
end;
```

DragDrop

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  Button1.DragMode:=dmAutomatic;
end;

procedure TForm1.Button2DragOver(Sender, Source: TObject; X, Y:
Integer;
  State: TDragState; var Accept: Boolean);
begin
  if source=Button1 then
    Button1.DragCursor:=crDrag;
end;

procedure TForm1.Button2DragDrop(Sender, Source: TObject; X, Y:
Integer);
begin
  Button1.Hide;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
  Button1.show;
  Button1.DragCursor:=crDefault;
end;
```

Jelölőnégyzet



A jelölőnégyzetek használatukban megegyeznek a nyomógombokkal, de

Nincs

Cancel
Default
ModalResult

Újdonság

<i>AlignMent</i>	<i>TLeftRight</i>	<i>taLeftJustified, taRightJustifies</i>
<i>AllowGrayed</i>	<i>Boolean</i>	
<i>Color</i>	<i>TColor</i>	
<i>ParentColor</i>	<i>Boolean</i>	
<i>Ctl3D</i>	<i>Boolean</i>	
<i>ParentCtl3D</i>	<i>Boolean</i>	
<i>Checked</i>	<i>Boolean</i>	
<i>State</i>	<i>TCheckBoxStata</i>	<i>cbUnChecked, cbChecked, cbGrayed</i>

Háttérszín-váltás jelölőnégyzet szerint

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  Color:=clBlack;
  CheckBox1.Checked:=true;
  CheckBox1.ParentColor:=true;
end;

procedure TForm1.CheckBox1Click(Sender: TObject);
begin
  If (CheckBox1.Checked) Then
  begin
    TForm(CheckBox1.Parent).Color:=clBlack;
    CheckBox1.Font.Color:=clWhite;
  end
  else
  begin
    TForm(CheckBox1.Parent).Color:=clWhite;
    CheckBox1.Font.Color:=clBlack;
  end;
end;
```

A választógombok használatukban megegyeznek a jelölőnégyzetekkel, de csoportban értelmezettek és csak egyetlen lehet kiválasztott a többi automatikusan kiugrik (TForm, TPanel, TGroupBox)

Nincs

*AllowGrayed
State*

Színváltás választógomb szerint

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  Color:=clRed;
  RadioButton1.Checked:=true;
  RadioButton1.Caption:='Piros';
  RadioButton2.Caption:='Sarga';
  RadioButton3.Caption:='Zold';
  RadioButton1.ParentColor:=true;
  RadioButton2.ParentColor:=true;
  RadioButton3.ParentColor:=true;
end;

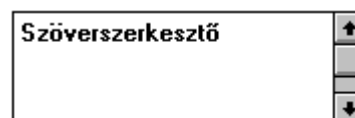
procedure TForm1.RadioButton1Click(Sender: TObject);
begin
  TForm(RadioButton1.Parent).Color:=clRed;
end;

procedure TForm1.RadioButton2Click(Sender: TObject);
begin
  TForm(RadioButton1.Parent).Color:=clYellow;
end;

procedure TForm1.RadioButton3Click(Sender: TObject);
begin
  TForm(RadioButton1.Parent).Color:=clGreen;
end;
```

2. Szövegszerkesztő vezérlők

Szövegmező



Ismert tulajdonságok

<i>Elhelyezkedés</i>	Left	Top	Height	Width
<i>Engedély, láthatóság</i>	Align	BoundRect	Showing	
<i>Szín</i>	Enabled	Visible		
<i>Kurzor</i>	Color	ParentColor		
<i>Tabulátor</i>	Cursor			
<i>Nevek</i>	TabOrder	TabStop		
<i>Szülő</i>	Name	Owner		
<i>Spec. adatok</i>	Parent			
<i>Mozgatás</i>	Tag	DragCursor		
<i>Betűtípus</i>	Dragmode	.Color	.Name	.Size
<i>Információk</i>	Font	ParentFont		
	.Style	ShowHint	Parent	
	Hint		ShowHint	
<i>Felbukkanó menü</i>	PopupMenu	HelpContext		
<i>Egyéb</i>	Component	Handle		
	Index			

Új tulajdonságok

<i>Megjelenés</i>	AutoSelect	boolean	AutoSize	boolean
	BorderStyle	TBorderStyle	bsNone	bsSingle
<i>Szöveg</i>	Text	PChar RO	MaxLength	
	Modified	boolean	ReadOnly	boolean
<i>Kiválasztás</i>	SelStart	SelLength	SelText	Hide
	integer	integer	PChar	Selection
				boolean
<i>BetűTípus</i>	CharCase	TEditCharCase	ecNormal	ecXXCase
	OEMConvert	boolean		
	Passwordchar	char		

Események

<i>OnChange</i>	<code>TNotifyEvent = procedure (Sender: TObject) of object;</code>
<i>OnDbleClick</i>	<code>TNotifyEvent = procedure (Sender: TObject) of object;</code>
<i>OnDragDrop</i>	<code>TDragDropEvent = procedure (Sender, Source: TObject; X, Y: Integer) of object;</code>
<i>OnDragOver</i>	<code>TDragOverEvent = procedure (Sender, Source: TObject; X, Y: Integer; State: TDragState; var Accept: Boolean) of object;</code>
	<code>dsXXX</code>
<i>OnEndDrag</i>	<code>TEndDragEvent = procedure (Sender, Target: TObject; X, Y: Integer) of object;</code>
<i>OnEnter</i>	<code>TNotifyEvent = procedure (Sender: TObject) of object;</code>
<i>OnExit</i>	<code>TNotifyEvent = procedure (Sender: TObject) of object;</code>
<i>OnKeyDown</i>	<code>TKeyEvent = procedure (Sender: TObject; var Key: Word; Shift: TShiftState) of object;</code>

OnKeyUp	VK_XXX
OnKeyPress	ssXXX
OnMouseDown	TMouseEvent = procedure (Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer) of object;
OnMouseUp	mbXXX
OnMouseMove	

Új metódusok

Clear	constructor Create(AOwner: TComponent);
ClearSelection	procedure ClearSelection;
CopyToClipboard;	procedure CopyToClipboard;
CutToClipboard	procedure CutToClipboard;
PasteFromClipboard	procedure PasteFromClipboard;
SetSelTextBuf	procedure SetSelTextBuf(Buffer: PChar);
GetSelTextBuf	function GetSelTextBuf(Buffer: PChar; BufSize: Integer): Integer;
SelectAll	procedure SelectAll;

Szöveg kiválasztása

```

procedure TForm1.FormClick(Sender: TObject);
begin
  edit1.SelStart:= 2;
  edit1.SelLength:= 3;
  edit1.Seltext:='Lajos';
end;

```

Változás érzékelése

```

procedure TForm1.Edit1Change(Sender: TObject);
begin
  Edit2.text:=Edit1.text;
end;

```

Kiterjesztett szövegszerkesztő (Memo)

Elmaradó tulajdonságok

AutoSelect
AutoSize

CharCase
OEMConvert
PasswordChar

Új tulajdonságok

Alignment
ScrollBars

WantReturn
WantTabs
WordWrap
Lines

TAlignment

TScrollStyle

ssNone, ssHorizontal,
ssVertical, ssBoth

boolean

boolean

boolean

TStrings

Add, AddString, Insert, Move,
Exchange, IndexOf

3 A TStringList osztályok

Tulajdonságok

Count	szám - Integer 0-tól számozva
Objects	bittérképre - property Objects[Index: Integer]: TObject;
Strings	sztriglistára - property Strings[Index: Integer]: string;
Values	property Values[const Name: string]: string;

Csak a TStringList

Duplicates	property Duplicates: TDuplicates; dupIgnore, dupAccept, dupError
Sorted	property Sorted: Boolean;

Fontosabb metódusok

```
function Add(const S: string): Integer;
function AddObject(const S: string; AObject: TObject): Integer;
procedure AddStrings(Strings: TStringList);
procedure Insert(Index: Integer; const S: string);
procedure InsertObject(Index: Integer; const S: string; AObject:
TObject);
procedure SetText(Text: PChar);
procedure LoadFromFile(const FileName: string);

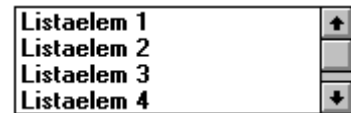
procedure Move(CurIndex, NewIndex: Integer);
procedure Exchange(Index1, Index2: Integer);

function GetText: PChar;
function IndexOf(const S: string): Integer;
function IndexOfObject(AObject: TObject): Integer;

Procedure Clear;
procedure Delete(Index: Integer);
procedure SaveToFile(const FileName: string);
```

4. Listaablakok

Listaablak



Ismert Tulajdonságok

<i>Elhelyezkedés</i>	Left Align	Top BoundRect	Height	Width
<i>Engedély, láthatós.</i>	Enabled	Visible	Showing	Ctl3D
<i>Szín</i>	Color	ParentColor		
<i>Kurzor</i>	Cursor			
<i>Tabulátor</i>	TabOrder	TabStop		
<i>Nevek</i>	Name			
<i>Szülő</i>	Parent	Owner		Parent Ctl3D
<i>Spec. adatok</i>	Tag			
<i>Vonzolás</i>	Dragmode	DragCursor		
<i>Fontok</i>	Font .Style	.Color ParentFont	.Name	.Size
<i>Információk</i>	Hint	ShowHint	Parent ShowHint	
<i>Menük</i>	PopupMenu	HelpContext		
<i>Egyéb</i>	Component Index	Handle		

Új tulajdonságok

<i>Megjelenés</i>	Columns IntegralHeight ItemHeight Sorted Style	<i>longint</i> <i>boolean</i> <i>integer</i> <i>boolean</i> <i>TListBoxStyle</i>	lbStandard lbOwnerDrawFixed lbOwnerDraw
<i>Választás</i>	Topindex ExtendedSelect ItemIndex Items MultiSelect SelCount Selected	<i>integer</i> <i>boolean</i> <i>integer</i> <i>TStrings</i> <i>boolean</i> <i>integer</i>	
<i>Egyéb</i>	Canvas	<i>property Selected[X: Integer]: Boolean;</i>	

Ismert események

OnClick	OnDbleClick	OnDragDrop	OnDragOver
OnEndDrag	OnEnter	OnExit	OnKeyDown
OnKeyUp	OnKeyPress	OnMouseDown	OnMouseUp
OnMouseMove			

Új események

OnDrawItem	<pre>TDrawItemEvent = procedure(ListBox: TListBox; Index: Integer; Rect: TRect; State: TOwnerDrawState) of object; Ez festhet ki (Canvas) TOwnerDrawState = set of (odSelected, odGrayed, odDisabled, odChecked, odFocused);</pre>
OnMeasureItem	<pre>TMeasureItemEvent = procedure(Control: TWinControl; var Height: Integer) of object;</pre>

Saját metódus

```
function ItemAtPos(Pos: TPoint; Existing: Boolean): Integer;
```

Átvitel

```
procedure TForm1.FormCreate(Sender: TObject);
var i:integer;
begin
    for i:=0 To 10 do
        ListBox1.Items.add('adat' + inttostr(i));
        Button2.Enabled := False;
    end;

    procedure TForm1.Button1Click(Sender: TObject);
    var i:integer;
    begin
        Button2.Enabled:=True;
        Button1.Enabled:=False;
        for i:=0 To ListBox1.Items.Count - 1 do
            begin
                ListBox2.Items.Add(ListBox1.Items[i]);
                ListBox1.Items.Delete(i);
            end;
        end;

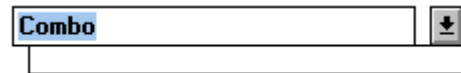
    procedure TForm1.Button2Click(Sender: TObject);
    var i:integer;
    begin
        Button1.Enabled:=True;
        Button2.Enabled:=False;
        for i:=0 To ListBox2.Items.Count - 1 do
            begin
                ListBox1.Items.Add(ListBox2.Items[i]);
                ListBox2.Items.Delete(i);
            end;
        end;

    procedure TForm1.ListBox1Click(Sender: TObject);
    begin
        ListBox2.Items.Add(ListBox1.Items[ListBox1.ItemIndex]);
        ListBox1.Items.Delete(ListBox1.ItemIndex);
    end;

    procedure TForm1.ListBox2Click(Sender: TObject);
    begin
        ListBox1.Items.Add(ListBox2.Items[ListBox2.ItemIndex]);
        ListBox2.Items.Delete(ListBox2.ItemIndex);
    end;
```

5. Kombinált (lista és editor) ablak

Kombinált listaablak



Új tulajdonságok

DropDownCount
Style

```
TComboBoxStyle = (csDropDown, csSimple,  
csDropDownList, csOwnerDrawFixed,  
csOwnerDrawVariable);
```

6. Csúszka

Csúszka



Új tulajdonságok

Típus	Kind	TScrollBarKind = (sbHorizontal, sbVertical);
Határok	Max	Integer
	Min	Integer
Változások	SmallChange	TScrollBarInc = 1..32767;
	LargeChange	TScrollBarInc = 1..32767;
Érték	Position	integer

Események

OnChange	OnDbleClick	OnDragDrop	OnDragOver
OnEndDrag	OnEnter	OnExit	OnKeyDown
OnKeyUp	OnKeyPress	OnMouseDown	OnMouseUp
OnMouseMove			

A szokásos metódusok

```
procedure SetParams(APosition, AMin, AMax: Integer);
```

Példa

```
var   r, g, b : Integer
procedure TForm1.FormCreate(Sender: TObject);
begin
  ScrollBar1.Min:=0;
  ScrollBar2.Min:=0;
  ScrollBar3.Min:=0;
  ScrollBar1.Max:=255;
  ScrollBar2.Max:=255;
  ScrollBar3.Max:=255;
  ScrollBar1.Position:=255;
  ScrollBar2.Position:=255;
  ScrollBar3.Position:=255;
end;

procedure TForm1.ScrollBarChange(Sender: TObject);
begin
  form1.Refresh
end;

procedure TForm1.FormPaint(Sender: TObject);
begin
  Color:=RGB(255-ScrollBar1.Position,
            255-ScrollBar2.Position,
            255-ScrollBar3.Position);
end;
```

7. Időzítő - Timer

Tulajdonságok

ComponentIndex	
Enabled	
Interval	word (sec/1000)
Name	
Owner	
Tag	

Események

OnTimer	TNotifyEvent
----------------	---------------------

Metódusok

Create	Destroy	Free
---------------	----------------	-------------

Példa

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  Timer1.Enabled:=true;
end;

procedure TForm1.Timer1Timer(Sender: TObject);
begin
  MessageBeep(0);
end;
```

8. Párbeszédablakok

```
procedure Show;

function ShowModal: Integer;
    - a visszatérési érték a Form/ShowModal

procedure ShowMessage(const Msg: string);

function MessageBox(Text, Caption: PChar; Flags: Word): Integer;

function MessageDlg(const Msg: string; AType: TMsgDlgType;
    AButtons: TMsgDlgButtons; HelpCtx: Longint): Word;

    TMsgDlgType = (mtWarning, mtError, mtInformation,
        mtConfirmation, mtCustom);

    TMsgDlgBtn = (mbYes, mbNo, mbOK, mbCancel, mbAbort,
        mbRetry, mbIgnore, mbAll, bHelp);

    TMsgDlgButtons = set of TMsgDlgBtn;

function MessageDlgPos(const Msg: string; AType: TMsgDlgType;
    AButtons: TMsgDlgButtons; HelpCtx: Longint;
    X, Y: Integer): Word;

function InputBox(const ACaption, APrompt, ADefault: string):
string;

function InputQuery(const ACaption, APrompt: string;
    var Value: string): Boolean;
```

8. Általános párbeszédablakok

Tulajdonságok

ComponentIndex	Ctl3D	HelpContext
Name	Owner	Tag

Metódusok

Create	Destroy	Free
Execute	boolean	true - OK false - Cancel

Open, Save

DefaultExtension	TFileExtension = string[3];
FileEditStyle	TFileEditStyle = (fsEdit, fsComboBox); névmegadás
FileName	TFileName = string[79];
Files	TStrings - a kiválasztottak
Filter	String - név filter név filter...
HistoryList	Tstrings
InitialDir	String
Options	TOpenOptions = (...ofAllowMultiSelect,...)
Title	String

```
procedure TForm1.FormClick(Sender: TObject);
begin
  OpenFileDialog1.Filter:='TXT|*.TXT';
  if OpenFileDialog1.Execute then
    Label1.Caption:=OpenDialog1.FileName;
end;
```

Font

Device	TFontDialogDevice = (fdScreen, fdPrinter, fdBoth);
Font	TFont
MaxFontSize	integer;
MinFontSize	integer;
Options	TFontDialogOptions = (...fdTrueTypeOnly...)

```
procedure TForm1.FormClick(Sender: TObject);
begin
  FontDialog1.Device:=fdBoth;
  if FontDialog1.Execute then
    Label1.Font:=FontDialog1.Font;
end;
```

10. Menük

A MainMenu főbb tulajdonságai

Items	Items[Index]:TMenuItem (*)
Name	string

A Popup menu főbb tulajdonságai

Alignment	TPopupAlignment = (paLeft, paRight, paCenter);
AutoPopup	boolean
Items	Items[Index]:TMenuItem (*)
PopupComponent	TComponent - az utolsónak választott
Name	string

(*) MenuElemek TMenuItem

Tulajdonságok

Break	TMenuItemBreak = (mbNone, mbBreak, mbBarBreak);
Caption	string
Checked	boolean
Command	word (WM_COMMAND sorszám)
Count	integer - elemszám
Enabled	boolean
GroupIndex	integer
HelpContext	
Hint	
Items	Items[Index]:TMenuItem - a menu elemei
Name	
Owner	
Parent	
Shortcut	TShortcut (programozható vagy megadható) function Shortcut(Key: Word; Shift: TShiftState): TShortcut;
Tag	
Visible	

Esemény

OnClick	TNotifyEvent
----------------	--------------

Metódusok

```
procedure Add(Item: TMenuItem);
procedure Click;
function IndexOf(Item: TMenuItem): Integer;
procedure Insert(Index: Integer; Item: TMenuItem);
function Remove(Item: Pointer): Integer;
procedure Popup(X, Y: Integer);
```