Component Reference

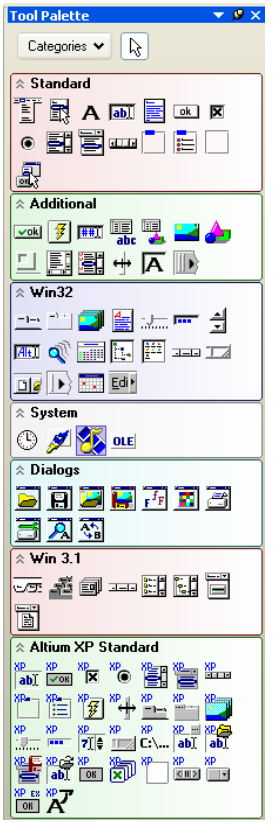
Summary

This reference outlines the supported components and their main properties and methods.

Technical Reference
TR0119 (v1.3) May 2, 2007

Component Tool Palette Reference

This reference outlines the supported graphical components and their main properties and methods from the **Tool Palette** panel. The components are used on a form (the form has an associated file with a DFM extension that has details of the form's components locations and other attributes including the form itself). These components can be used by DelphiScript, VBScript or JavaScript when designing script forms in Altium Designer.



The **Tool Palette** panel (accessed from the **Script** button on the bottom of the Altium Designer status bar) contains a palette of components that can be placed on a script form. A component has properties, events and methods that can be manipulated by the coder / user. The components are categorised according to the **Tool Palette** panel's tabs they are on: Standard, Additional, Win32, Dialogs, Win31, Altium XP Standard and System tabs.

The controls are visual components, meaning the user can see them and possibly interact with them at runtime. All controls have properties, methods, and events that describe aspects of their appearance, such as the position of the control, the cursor or hint associated with the control, methods to paint or move the control, and events that respond to user actions.

The properties and events are outlined in the **Object Inspector** panel for the currently focused component including the form itself. The methods are simply procedures and functions of the component's object itself.

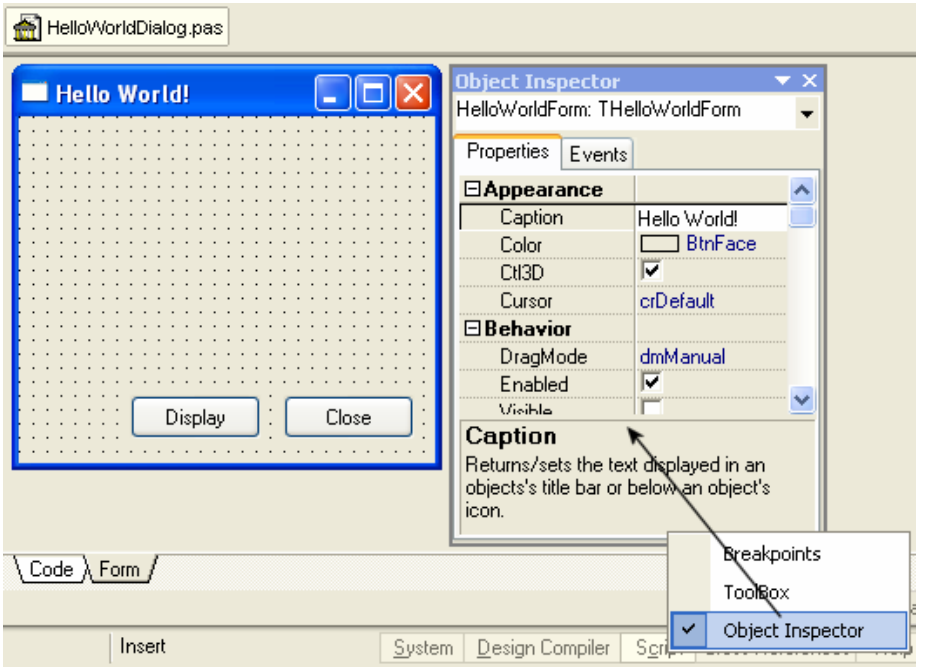
Check out the **Object Inspector** panel to check the properties, methods and events supported by the currently focussed component on a script form.

Script Forms Overview

Script Forms

A script form, which is the primary user interface, and other forms such as dialog boxes, secondary windows and so on. You can begin your form design by creating a new script form. With a project open in Altium Designer, right click on a project in the **Projects** panel, and a pop up menu appears, click on the **Add New to Project** item, and choose **Script Form** item.

A new script form appears with the Form1 name as the default name. You can always change the name for the script form by going to the **Object Inspector** panel. Remember to keep script form names unique in a project.



The form is a window, and therefore by default includes standard window functionality such as:

- Control menu
- Minimize and Maximize buttons
- Title bar
- Resizeable borders

You can change these features, as well as any other property of the form, at design time using the **Object Inspector**. Since a script form is also a component, and thus components from the **Toolbox** panel have the three following items:

- Properties
- Events
- Methods

List of properties for a component

To see a list of properties for a component, select a component and in the **Object Inspector,** activate the **Properties** tab.

List of events for a component

To see a list of events a component can react on, select a component, and in the **Object Inspector** activate the **Events** tab. To create an event handling procedure, decide on what event you want your component to react, and double click the event name.

List of methods for a component

To see a list of methods for a component, see the Tool PaletteReference.

Important Notes

Keep your script form names unique in a project they are stored, so script forms can be referenced correctly in scripts since script forms are global variables in a project.

Script Form Tasks

To make the form stay on top of other open panels for example, set the **FormStyle** property to **fsStayOnTop**.

To define the default behaviour of a form, set the FormKind to one of the following values; fkNone, fkNormal, fkServerPanel or fkModal. If fkModal is chosed, then the form will be a modal form ie waiting for user input before proceeding such as closing the form. If fkServerPanel then the form will be shown as a Server panel. If fkNormal then the form acts as a normal non-modal form.

To remove the form's default scroll bars, change the value of the HorzScrollBar and VertScrollBar properties.

To make the form a MDI frame or MDI child, use the FormStyle property.

To change the form's border style, use the BorderIcons and BorderStyle properties. (The results are visible at runtime.)

To change the icon for the minimized form, use the Icon property.

To specify the initial position of a form in the application window, use the Position property.

To specify the initial state of the form, (e.g., minimized, maximized or normal) use the WindowState property.

To define the working area of the form at runtime, use the ClientHeight and ClientWidth properties. (Note that ClientHeight and ClientWidth represent the area within the form's border; Height and Width represent the entire area of the form.)

To specify which control has initial focus in the form at runtime, use the ActiveControl property.

To pass all keyboard events to form, regardless of the selected control, use the KeyPreview property.

To specify a particular menu, if your form contains more than one menu, use the Menu property.

Script Form methods and properties

The **Component Tool Palette** panel's controls are based on Borland Delphi's Visual Component Library and you can refer to Borland Delphi's documentation for more details on methods, properties and events for the **TForm** component.

TForm Properties	TForm Methods	TForm Events
• Derived from TCustomForm	**In TForm**	OnActivate
Active	ArrangeIcons	OnCanResize
ActiveControl	Cascade	OnClick
ActiveMDIChild	Next	OnClose
ActiveOleControl	Previous	OnCloseQuery
AlphaBlend	Tile	OnConstrainedResize
AlphaBlendValue	**Derived from TCustomForm**	OnContextPopup
BorderIcons	AfterConstruction	OnCreate
BorderStyle	BeforeDestruction	OnDblClick
Canvas	Close	OnDeactivate
ClientHandle	CloseQuery	OnDestroy
ClientHeight	Create	OnDockDrop
ClientRect	CreateNew	OnDockOver
ClientWidth	DefocusControl	OnDragDrop
DefaultMonitor	Destroy	OnDragOver
Designer	FocusControl	OnEndDock
DropTarget	GetFormImage	OnGetSiteInfo
Floating	Hide	OnHelp
FormState	IsShortCut	OnHide
FormStyle	MakeFullyVisible	OnKeyDown
HelpFile	MouseWheelHandler	OnKeyPress
Icon	Print	OnKeyUp
KeyPreview	Release	OnMouseDown
MDIChildCount	SendCancelMode	OnMouseMove
MDIChildren	SetFocus	OnMouseUp
Menu	SetFocusedControl	OnMouseWheel
ModalResult	Show	OnMouseWheelDown
Monitor	ShowModal	OnMouseWheelUp
ObjectMenuItem	WantChildKey	OnPaint
	Derived from TScrollingWinControl	OnResize

OldCreateOrder
OleFormObject
Parent
ParentBiDiMode
PixelsPerInch
Position
PrintScale
Scaled
TileMode
TransparentColor
TransparentColorValue
Visible
WindowMenu
WindowState

- Derived from TScrollingWinControl

AutoScroll
HorzScrollBar
VertScrollBar

Derived from TWinControl

AlignDisabled
BorderWidth
Brush
ClientOrigin
ControlCount
Controls
Ctl3D
DockClientCount
DockClients
DockManager
DockSite
DoubleBuffered
Handle
ParentWindow
Showing

DisableAutoRange
EnableAutoRange
ScrollInView

Derived from TWinControl

Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
Update
UpdateControlState

Derived from TControl

BeginDrag
BringToFront
ClientToParent
ClientToScreen

OnShortCut
OnShow
OnStartDock
OnUnDock

TabOrder	Dock	
TabStop	DragDrop	
UseDockManager	Dragging	
VisibleDockClientCount	DrawTextBiDiModeFlags	
• Derived from TControl	DrawTextBiDiModeFlagsReadingOnly	
Action	EndDrag	
Align	GetControlsAlignment	
Anchors	GetTextBuf	
AutoSize	GetTextLen	
BiDiMode	InitiateAction	
BoundsRect	IsRightToLeft	
Caption	ManualDock	
Color	ManualFloat	
Constraints	ParentToClient	
ControlState	Perform	
ControlStyle	Refresh	
Cursor	ReplaceDockedControl	
DockOrientation	ScreenToClient	
DragKind	SendToBack	
DragMode	SetTextBuf	
Enabled	UseRightToLeftAlignment	
FloatingDockSiteClass	UseRightToLeftReading	
Font	UseRightToLeftScrollBar	
Height	**Derived from TComponent**	
HelpContext	DestroyComponents	
HelpKeyword	Destroying	
HelpType	ExecuteAction	
Hint	FindComponent	
HostDockSite	FreeNotification	
Left	FreeOnRelease	
LRDockWidth	GetNamePath	
Name	GetParentComponent	
ParentFont	HasParent	
PopupMenu	InsertComponent	
ShowHint	IsImplementorOf	

TBDockHeight Top UndockHeight UndockWidth Width WindowProc • Derived from TComponent ComObject ComponentCount ComponentIndex Components ComponentState ComponentStyle DesignInfo Owner Tag VCLComObject	ReferenceInterface RemoveComponent RemoveFreeNotification SafeCallException SetSubComponent UpdateAction **Derived from TPersistent** Assign **Derived from TObject** ClassInfo ClassName ClassNameIs ClassParent ClassType CleanupInstance DefaultHandler Dispatch FieldAddress Free FreeInstance GetInterface GetInterfaceEntry GetInterfaceTable InheritsFrom InitInstance InstanceSize MethodAddress MethodName NewInstance	

Standard Tab

Standard Tab

The standard tab deals with mainmenu, button, popup menu, label, edit, memo, checkbox, radiobutton, listbox, combobox, scrollbar, combobox, groupbox, panel, radiogroup and action list components.

See also

ActionList component

Button component

Checkbox component

Combobox component

Edit component

Groupbox component

Label component

Listbox component

MainMenu component

Memo component

Popup menu component

RadioGroup component

Scrollbar component

ActionList component

Use Action lists to centralize the response to user commands (actions). Action list components maintain a list of actions that are available to the client controls in an application. Add action list components to your form or data module from the standard page of the component palette. Double-click the action list to display the action list editor, from which you can add, delete, and rearrange actions.

Main properties are: ActionCount, Actions, Images, State.

Methods

Derived from TCustomActionList

Create

Destroy

ExecuteAction

IsShortCut

UpdateAction

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Button component

Users click button controls on scripts to initiate actions. You can assign an action to a TButton component by creating an OnClick event handler for it. Double-clicking a button at design time takes you to the button's OnClick event handler in the Code editor.

Set Cancel to True if you want the button to trigger its OnClick event when the user presses Esc.

Set Default to True if you want the Enter key to trigger the button's OnClick event.

Use TButton to put a standard push button on a form. TButton introduces several properties to control its behavior in a dialog box setting. To use a button that displays a bitmap instead of a label, use TBitBtn. To use a button that can remain in a depressed position, use TSpeedButton.

Main properties are : Cancel, Default and ModalResult.

Methods

In TButton

Click

Create

UseRightToLeftAlignment

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler

ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Checkbox component

A TCheckBox component presents an option for the user. The user can check the box to select the option, or uncheck it to deselect the option.

Main properties are AllowGrayed, Checked and State.

Methods

Derived from TCustomCheckBox

Create

GetControlsAlignment

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy

DisableAlign

DockDrop

EnableAlign

FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction

IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Combobox component

A TComboBox component is an edit box with a scrollable drop-down list attached to it. Users can select an item from the list or type directly into the edit box. Main properties are ItemIndex, Items, Sorted.

Methods

Derived from TCustomComboBox

Create

Destroy

Derived from TCustomCombo

AddItem

Clear

ClearSelection

CopySelection

DeleteSelected

Focused

SelectAll

Derived from TCustomListControl

MoveSelection

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds

SetFocus

Update

UpdateControlState

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent

FreeNotification

FreeOnRelease

GetNamePath

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Edit component

Use a TEdit object to put a standard Windows edit control on a form. Edit controls are used to retrieve text that users type. Edit controls can also display text to the user.

When only displaying text to the user, choose an edit control to allow users to select text and copy it to the Clipboard. Choose a label object if the selection capabilities of an edit control are not needed.

Main properties are : Readonly, CharCase, Modified, MaxLength, PasswordChar.

Methods

Derived from TCustomEdit

Clear

ClearSelection

ClearUndo

CopyToClipboard

Create

CutToClipboard

DefaultHandler

GetSelTextBuf

PasteFromClipboard

SelectAll

SetSelTextBuf

Undo

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

Destroy

CreateParentedControl

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat

MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Groupbox component

A group box (TGroupBox) arranges related controls on a form. The most commonly grouped controls are radio buttons. After placing a group box on a form, select components from the Toolbox panel and place them in the group box. The Caption property contains text that labels the group box at runtime.

Main properties are TabOrder, TabStop.

Methods

Derived from TCustomGroupBox

Create

Derived from TCustomControl

Destroy

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf

GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Label component

Use TLabel to add text that the user can't edit to a form. This text can be used to label another control, and can set focus to that control when the user types an accelerator key.

To add an object to a form that can respond to keyboard input (other than setting focus to another object when an accelerator key is typed) in addition to displaying text, use TStaticText. To add an object to a form that displays text that a user can scroll or edit, use TEdit.

Methods

Derived from TCustomLabel

Create

Derived from TGraphicControl

Destroy

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

DefaultHandler

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen

HasParent

Hide

InitiateAction

Invalidate

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

Repaint

ReplaceDockedControl

ScreenToClient

SendToBack

SetBounds

SetTextBuf

Show

Update

UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Listbox component

Use TListBox to display a scrollable list of items that users can select, add or delete. TListBox is a wrapper for the Windows listbox control. For specialized list boxes use other descendant classes of TCustomListBox or derive your own class from TCustomListBox.

Main properties are : Count, ExtendedSelect, ItemIndex, Items, MultiSelect, SelCount, Selected, Sorted.

Methods

Derived from TCustomListBox

AddItem

Clear

ClearSelection

CopySelection

Create

DeleteSelected

Destroy

ItemAtPos

ItemRect

SelectAll

Derived from TCustomListControl

MoveSelection

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent

GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Mainmenu component

Menus provide an easy way for your users to execute logically grouped commands. The Menu Designer enables you to easily add a menu—either predesigned or custom tailored—to your form. You add a menu component to the form, open the Menu Designer, and type menu items directly into the Menu Designer window. You can add or delete menu items, or drag and drop them to rearrange them during design time.

Methods

In TMainMenu

GetOle2AcceleratorTable

Merge

PopulateOle2Menu

SetOle2MenuHandle

Unmerge

Derived from TMenu

Create

Destroy

DispatchCommand

DispatchPopup

FindItem

GetHelpContext

IsRightToLeft

IsShortCut

ParentBiDiModeChanged

ProcessMenuChar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Memo component

Use TMemo to put a standard Windows multi-line edit control on a form. Multi-line edit boxes allow the user to enter more than one line of text. They are appropriate for representing lengthy information.

Main properties are: Lines, WordWrap, WantReturns, WantTabs, MaxLength, ReadOnly.

Methods

Derived from TCustomMemo

Create

Destroy

GetControlsAlignment

Derived from TCustomEdit

Clear

ClearSelection

ClearUndo

CopyToClipboard

CutToClipboard

DefaultHandler

GetSelTextBuf

PasteFromClipboard

SelectAll

SetSelTextBuf

Undo

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds

SetFocus

Update

UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction

DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize

MethodAddress

MethodName

NewInstance

Panel component

The TPanel component provides a generic container for other window controls. Panels are typically used to visually group components together on a script form. Panels can be aligned with the form to maintain the same relative position when the form is resized. The BorderWidth property determines the width, in pixels, of the border around a panel.

You can also place other controls onto a panel and use the Align property to ensure proper positioning of all the controls in the group on the form. You can make a panel alTop aligned so that its position will remain in place even if the form is resized.

The look of the panel can be changed to a raised or lowered look by using the BevelOuter and BevelInner properties.

Main properties are : Alignment, BevelInner, BevelOuter, BevelWidth.

Methods

Derived from TCustomPanel

Create

Derived from TCustomControl

Destroy

GetControlsAlignment

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetTextBuf
GetTextLen
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh

ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Popup menu component

Pop-up, or local, menus are a common ease-of-use feature for any application. They enable users to minimize mouse movement by clicking the right mouse button in the application workspace to access a list of frequently used commands. Use TPopupMenu to define the pop-up menu that appears when the user clicks on a control with the right mouse button. To make a pop-up menu available, assign the TPopupMenu object to the control's PopupMenu property.

Note: If the popup menu's ParentBiDiMode is True, the popup menu's BiDiMode is set to the BiDiMode of the control that activates it. If a control cannot be found, the BiDiMode is set to Application.BiDiMode. The popup menu's BiDiMode affects all of its menu items.

Methods

Create

Destroy

DoPopup

Popup

UseRightToLeftAlignment

Derived from TMenu

DispatchCommand

DispatchPopup

FindItem

GetHelpContext

IsRightToLeft

IsShortCut
ParentBiDiModeChanged
ProcessMenuChar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstrucion
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch
FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Radiobutton component

Use TRadioButton to add a radio button to a form. Radio buttons present a set of mutually exclusive options to the user- that is, only one radio button in a set can be selected at a time. When the user selects a radio button, the previously selected radio button becomes unselected. The main property is Checked.

Methods

In TRadioButton

Create

GetControlsAlignment

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat

MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

RadioGroup component

A TRadioGroup object is a special group box that contains only radio buttons. Radio buttons that are placed directly in the same control component are grouped. When the user checks a radio button, all other radio buttons in its group become unchecked. Therefore, two radio buttons on a script form can be checked at the same time only if they are placed in separate containers, such as group boxes.

Main properties are : Columns, ItemIndex, Items.

Methods

Derived from TCustomRadioGroup

Create

Destroy

FlipChildren

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds

SetFocus

Update

UpdateControlState

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Scrollbar component

Use TScrollBar to add a free-standing scroll bar to a form. Many controls have properties that add scroll bars which are an integral part of the control (TControlScrollBar objects). TScrollBar allows controls that do not have integrated scroll bars or groupings of controls to be scrolled when the user manipulates the TScrollBar object.

Main properties are : Position, Min, Max.

Methods

In TScrollBar

Create

SetParams

Derived from TWinControl

Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DefaultHandler
Destroy
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen

HasParent

Hide

InitiateAction

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Additional Tab

Additional Tab

The Additional tab deals with Bitbtn, SpeedButton, Maskedit, Stringgrid, Drawgrid, Image, Shape, Bevel, Scrollbox, Checklistbox, SPlitter, StaticText and Controlbar components.

See also

Bevel component

Bitbtn component

Checklistbox component

Controlbar component

Drawgrid component

Image component

Maskedit component

Scrollbox component

Shape component

SpeedButton component

Splitter component

StaticText component

Stringgrid component

Bevel component

The TBevel component represents a beveled outline. The bevel can appear raised or lowered.

Main properties are : Shape, Style.

Methods

In TBevel

Create

Derived from TGraphicControl

Destroy

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

DefaultHandler

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen

HasParent

Hide

InitiateAction

Invalidate

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

Repaint

ReplaceDockedControl

ScreenToClient

SendToBack

SetBounds

SetTextBuf

Show

Update

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Bitbtn component

Bitmap buttons exhibit the same behavior as button controls and you use them to initiate actions from forms and dialog boxes.

These Bitmap buttons implement properties that specify the bitmap images, along with their appearance and placement on the button. You can choose from predefined bitmap buttons styles or use your own customized bitmap for the button. Although the button can be associated with only one bitmap, the bitmap (glyph property) can be subdivided into four equal-sized portions, which are displayed based on the state of the button: up, down, disabled, and clicked.

The Kind property of TBitBtn provides commonly used buttons, such as OK, Cancel, Help, and so on. These predefined button types have corresponding graphical images and default behaviors, so you can easily add them to your application with little or no coding necessary.

Main properties are : Glyph, Kind, Layout, Spacing, Cancel, Default, ModalResult.

Methods

In TBitBtn

Click

Create

Destroy

Derived from TButton

UseRightToLeftAlignment

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DockDrop

DisableAlign

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient

Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Checklistbox component

The TCheckListBox component is similar to TListBox, except that each item has a check box next to it. Users can check or uncheck items in the list.

Main properties are : AllowedGrayed, Checked, Flat, ItemEnabled, State.

Methods

In TCheckListBox

Destroy

Derived from TCustomListBox

AddItem

Clear

ClearSelection

CopySelection

Create

DeleteSelected

ItemAtPos

ItemRect

SelectAll

Derived from TCustomListControl

MoveSelection

Derived from TWinControl

Broadcast

CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DefaultHandler
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging

DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Controlbar component

The TControlBar component manages the layout of toolbar components. Use TControlBar as a docking site for toolbar components. Control bars contain child controls (usually TToolBar objects) that can be moved and resized independently.

Main properties : AutoDock, AutoDrag, Picture, RowSize, RowSnap

Methods

Derived from TCustomControlBar

Create

Destroy

StickControls

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds

SetFocus

Update

UpdateControlState

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen

HasParent

Hide

InitiateAction

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress

MethodName

NewInstance

Drawgrid component

The TDrawGrid represents a grid control that displays information in column and row format. Add a TDrawGrid object to a script form to present arbitrary information in a tabular format. TDrawGrid provides many properties to control the appearance of the grid, as well as events and methods that take advantage of the tabular organization of the grid in responding to user actions.

Main properties are : Col, ColCount, EditorMode, Row, RowCount, RowHeights.

Methods

Derived from TCustomDrawGrid

CellRect

MouseToCell

Derived from TCustomGrid

Create

Destroy

MouseCoord

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh

ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Image component

The image component displays a graphical image, like a bitmap, icon, or metafile. The Picture property determines the graphic to be displayed. Use Center, AutoSize, Stretch, and Transparent to set display options.

Use the TImage component to display a graphical image on a form. Use the TPicture object in the Picture property to specify the actual bitmap, icon, metafile, or other graphic object displayed by TImage. Properties and methods of TPicture can be used for such things as loading an image from file, clearing the image in the TImage, and assigning an image for another control. TImage introduces several properties to determine how the image is displayed within the boundaries of the TImage object.

Main properties are : Canvas, Center, Picture, Stretch, Transparent.

Methods

In TImage

Create

Destroy

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

DefaultHandler

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen

HasParent

Hide

InitiateAction

Invalidate

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

Repaint

ReplaceDockedControl

ScreenToClient

SendToBack

SetBounds

SetTextBuf

Show

Update

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Maskedit component

Use a TMaskEdit object to put a masked edit control on your form. Masked edit controls validate the text the user enters against a mask that encodes the valid forms the text can take. The mask can also format text that is displayed to the user.

Main properties are: EditMask, EditText, IsMasked, MaxLength and Text.

Methods

Derived from TCustomMaskEdit

Clear

Create

GetTextLen

ValidateEdit

Derived from TCustomEdit

ClearSelection

ClearUndo

CopyToClipboard

CutToClipboard

DefaultHandler

GetSelTextBuf

PasteFromClipboard

SelectAll

SetSelTextBuf

Undo

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

Destroy

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock

ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Scrollbox component

The ScrollBox component represents a scrolling area in a window on a form. Use this TScrollBox component to create a scrollbox or boxes to create multiple scrolling areas (views) in a window. A scroll box can contain objects such as buttons and checkboxes.

Main properties are : BorderStyle, AutoScroll, HorzScrollBar, VertScrollbar.

Methods

In TScrollBar

Create

SetParams

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent

GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Shape component

The TShape component represents a geometric shape that can be drawn on a script form. Add a TShape object to a form to draw a simple geometric shape on the form. TShape introduces properties to describe the pen used to outline the shape and the brush used to fill it.

Main properties are : Brush, Pen, Shape.

Methods

In TShape

Create

Destroy

Derived from TControl

BeginDrag
BringToFront
ClientToParent
ClientToScreen
DefaultHandler
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
Invalidate
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
Repaint
ReplaceDockedControl
ScreenToClient
SendToBack
SetBounds
SetTextBuf
Show
Update
UseRightToLeftAlignment

UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

SpeedButton component

Use TSpeedButton to add a button to a group of buttons in a form. TSpeedButton introduces properties that can be used to set graphical images that represent the different button states (selected, unselected, disabled and so on). Use other properties to specify multiple images or to rearrange the images and text on the button. TSpeedButton also introduces properties that allow speed buttons to work together as a group. Speed buttons are commonly grouped in panels to create specialized tool bars and tool palettes.

Main properties are: AllowAllUp, DOwn, Flat, Glyp, Spacing, Caption, GroupIndex.

Methods

In TSpeedButton

Click

Create

Destroy

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

DefaultHandler

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen
HasParent
Hide
InitiateAction
Invalidate
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
Repaint
ReplaceDockedControl
ScreenToClient
SendToBack
SetBounds
SetTextBuf
Show
Update
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface

RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Splitter component

Add a splitter to a form between two aligned controls to allow users to resize the controls at runtime. The splitter sits between a control aligned to one edge of the form and the controls that fill up the rest of the client area. Give the splitter the same alignment as the control that is anchored to the edge of the form. When the user moves the splitter, it resizes the anchored control. This, in turn, changes the client area of the form, and the controls that fill up the rest of the client area resize accordingly.

Main properties are : AutoSnap, Beveled, MinSize, ResizeStyle.

Methods

In TSplitter

CanResize

Create

Destroy

DoCanResize

MouseDown

MouseMove

MouseUp

Paint

RequestAlign

StopSizing

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

DefaultHandler

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen

HasParent

Hide

InitiateAction

Invalidate

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler
ParentToClient
Perform
Refresh
Repaint
ReplaceDockedControl
ScreenToClient
SendToBack
SetBounds
SetTextBuf
Show
Update
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

StaticText component

The TStaticText component functions like TLabel, except that it has a window handle. Use TStaticText instead of TLabel when the component's accelerator key must belong to a windowed control.

Main properties are : FocusControl, ShowAccelChar, Caption

Methods

Derived from TCustomStaticText

Create

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

Destroy

DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetTextBuf
GetTextLen

Hide

InitiateAction

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Stringgrid component

The TStringGrid object when dropped on a script form can be used to present textual data in a tabular format. TStringGrid provides many properties to control the appearance of the grid, as well as events and methods that take advantage of the tabular organization of the grid in responding to user actions.

Main properties are : Cells, Cols, Objects, Rows.

Methods

In TStringGrid

Create

Destroy

Derived from TCustomDrawGrid

CellRect

MouseToCell

Derived from TCustomGrid

MouseCoord

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds

SetFocus

Update

UpdateControlState

Derived from TControl

BeginDrag

BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying

ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName

Additional Tab

NewInstance

Win32 Tab

Win32 Tab

The Win32 tab deals with tab control, page control, image list, richedit, trackbar, progress bar, updown, hot key, animate, datetimepicker, treeview, headercontrol, statusbar, toolbar, coolbar, monthcalender and page scroller tabs.

See also

Coolbar component

Datetimepicker component

Headercontrol component

Hotkey component

ImageList component

Monthcalender component

PageControl component

PageScroller component

Progressbar component

Richedit component

Statusbar component

Tabcontrol component

Trackbar component

Treeview component

Updown component

Animate component

A TAnimate control displays an animation clip, consisting of a series of image frames. The control can load an animation from a file, a resource, or a set of standard animations. To specify an animation, set the FileName, CommonAVI, ResName, or ResID properties. At design time, you can browse through the frames of the animation using the context menu.

Methods

In TAnimate

Create

Play

Reset

Seek

Stop

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds

SetFocus

Update

UpdateControlState

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

Dock

DragDrop

Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath

InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Coolbar component

The TCoolBar component is a wrapper class for a Windows rebar control, more commonly known as a CoolBar. A CoolBar contains child controls that can be moved and resized independently. Each control

resides on an individual band, represented by a TCoolBand object listed in the Bands property. The user positions the controls by dragging the sizing grip to the left of each band.

Main properties : Align, BandBorderStyle, BandMaximize, Bands, Bitmap, Constraints, FixedOrder, FixedSize, Images,ShowText, Vertical.

Methods

In TCoolBar

Create

Destroy

FlipChildren

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DisableAlign

DockDrop

EnableAlign

FindChildControl

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds

SetFocus

Update

UpdateControlState

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetTextBuf

GetTextLen

Hide

InitiateAction

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Datetimepicker component

The TDateTimePicker component is designed specifically for entering dates or times. In dmComboBox date mode, it resembles a list box or combo box, except that the drop-down list is replaced with a calendar illustration; users can select a date from the calendar. Dates or times can also be selected by scrolling with Up and Down arrows and by typing.

Main properties : Checked, DateFormat, DateMode, Format, Kind, ParseInput, ShowCheckBox, Time.

Methods

In TDateTimePicker

Create

Derived from TCommonCalendar

BoldDays

Destroy

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Headercontrol component

The THeaderControl component is a container for THeaderSection objects. It provides a set of resizable column headers. The header sections can be positioned above columns or fields of information. For example, header sections could be placed over a list box (TListBox).

Main properties : DragReorder, FullDrag, HotTrack, Images, Sections, Style.

Methods

Derived from TCustomHeaderControl

Create

Destroy

FlipChildren

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction

IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Hotkey component

Use the THotKey component to create key combinations that can be used as a shortcut. Main properties : Autosize, Hotkey, InvalidKeys, Modifiers.

Methods

Derived from TCustomHotKey

Create

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy

DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf

GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

ImageList component

The TImageList component represents a collection of same-sized images, each of which can be referred to by its index. Image lists are used to efficiently manage large sets of icons or bitmaps. All images in an image list are contained in a single, wide bitmap in screen device format. An image list may also include a monochrome bitmap that contains masks used to draw images transparently (icon style). The image list is capable of holding a large number of same sized images and retrieving them via their index within the range 0 to n - 1. The image list also has methods to facilitate storing, retrieving, and drawing of the stored images.

Main properties : Count, Height, DrawingStyle, ShareImages, Width, Maksed, ImageType.

Methods

Derived from TDragImageList

BeginDrag

DragLock
DragMove
DragUnlock
EndDrag
GetHotSpot
HideDragImage
SetDragImage
ShowDragImage
Derived from TCustomImageList
Add
AddIcon
AddImages
AddMasked
Assign
Clear
Create
CreateSize
Delete
Destroy
Draw
DrawOverlay
FileLoad
GetBitmap
GetIcon
GetImageBitmap
GetInstRes
GetMaskBitmap
GetResource
HandleAllocated
Insert
InsertIcon
InsertMasked
Move
Overlay
RegisterChanges

Replace
ReplaceIcon
ReplaceMasked
ResInstLoad
ResourceLoad
UnRegisterChanges
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

ListView component

The TListView component manages and displays a list of items in a form. The items can be displayed in columns with column headers and sub-items, or vertically or horizontally, with small or large icons.

TListView has many same properties, events, and methods as TCustomListView.

Methods

In TCustomListView

ActionChange

AddItem

AlphaSort

Arrange

CanChange

CanEdit

Change

ChangeScale

Clear

ClearSelection

ColClick

ColRightClick

ColumnsShowing

CopySelection

Create

CreateListItem

CreateListItems

CreateParams
CreateWnd
CustomDraw
CustomDrawItem
CustomDrawSubItem
CustomSort
Delete
DeleteSelected
Destroy
DestroyWnd
DoEndDrag
DoInfoTip
DoStartDrag
DrawItem
Edit
FindCaption
FindData
GetActionLinkClass
GetCount
GetDragImages
GetHitTestInfoAt
GetItemAt
GetItemIndex
GetNearestItem
GetNextItem
GetSearchString
GetSelCount
InsertItem
IsCustomDrawn
IsEditing
MouseUp
Notification
OwnerDataFetch
OwnerDataFind
OwnerDataHint

OwnerDataStateChange
Scroll
SelectAll
SetItemIndex
SetMultiSelect
SetViewStyle
StringWidth
UpdateColumn
UpdateColumns
UpdateItems
WndProc
Derived from TCustomListControl
MoveSelection
Derived from TWinControl
Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DefaultHandler
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl

Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack

SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Monthcalender component

The TMonthCalendar component is stand-alone calendar in which a user can select a date or range of dates.

Main properties : Date, EndDate, FirstDayOfWeek, MaxDate, MaxSelectRange, MinDate, MultiSelect, ShowToday, ShowTodayCircle, WeekNumbers.

Methods

In TMonthCalendar

CanAutoSize

ConstrainedResize

Create

CreateParams

GetCalendarHandle

MsgSetCalColors

MsgSetDateTime

MsgSetRange

Derived from TCommonCalendar

BoldDays

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

Destroy

DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetTextBuf
GetTextLen

Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
ParentToClient
Perform
ReferenceInterface
Refresh
RemoveComponent
RemoveFreeNotification
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
RemoveComponent
RemoveFreeNotification
SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

PageControl component

Use TPageControl to create a multiple page dialog or tabbed notebook. TPageControl displays multiple overlapping pages that are TTabSheet objects. The user selects a page by clicking the page's tab that appears at the top of the control. To add a new page to a TPageControl object at design time, right-click the TPageControl object and choose New Page. To create a tabbed control that uses only a single body portion (page), use TTabControl instead.

Main Properties : ActivePage, ActivePageIndex, PageCount, Pages

Methods

In TPageControl

CanShowTab
Change
Create
Destroy
DoAddDockClient
DockOver
DoRemoveDockClient
FindNextPage
GetChildren
GetImageIndex
GetPageFromDockClient
GetSiteInfo
Loaded
SelectNextPage
SetChildOrder
ShowControl
UpdateActivePage
Derived from TCustomTabControl
GetHitTestInfoAt
IndexOfTabAt
RowCount
ScrollTabs
TabRect
Derived from TWinControl
Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DefaultHandler
DisableAlign
DockDrop
EnableAlign
FindChildControl

FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction

IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

PageScroller component

Use TPageScroller to define a display area for a narrow window, such as a toolbar. If the window is larger than the display area, TPageScroller attaches arrows to the edges of the window, allowing it to be scrolled within the display area.

The page scroller control is similar to the scroll box control. However, the page scroller provides arrows that extend over the edges of the display area, whereas the scroll box provides scroll bars. Also, the page scroller supports scrolling for only one orientation (horizontal or vertical), so the constrained window must fit either the width or height of the display area.

Methods

In TPageScroller

AlignControls

Create

CreateParams

CreateWnd

GetButtonState

Scroll
Derived from TWinControl
Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
Destroy
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetTextBuf

GetTextLen

Hide

InitiateAction

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SetSubComponent
SafeCallException
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Progressbar component

Use a TProgressBar component to add a progress bar to a script form. Progress bars provide users with visual feedback about the progress of a procedure within an application. As the procedure progresses, the rectangular progress bar gradually fills from left to right with the system highlight color.

Main properties : Min, Max, Orientation, Position, Smooth, Step

Methods

In TProgressBar

Create

StepBy

StepIt

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show

UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Richedit component

Use a TRichEdit object to put a standard Windows rich text edit control on a form. Rich text edit controls let the user enter text that includes variation in font attributes and paragraph formatting information.

Main properties : Lines, Paragraph, PlainText, DefAttributes, HideSelection, HideScrollBars.

Methods

Derived from TCustomRichEdit

Clear

Create

Destroy

FindText

GetSelTextBuf

Print

RegisterConversionFormat

Derived from TCustomMemo

GetControlsAlignment

Derived from TCustomEdit

ClearSelection

ClearUndo

CopyToClipboard

CutToClipboard

PasteFromClipboard

SelectAll

SetSelTextBuf

Undo

Derived from TWinControl

Broadcast

CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetTextBuf
GetTextLen
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
ReplaceDockedControl
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
IsImplementorOf
InsertComponent

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Statusbar component

The TStatusBar component is a row of panels, usually aligned at the bottom of a script form, that display information about an application as it runs. Each panel is represented by a TStatusPanel object listed in the Panels property. The SimplePanel property can be used to toggle the status bar at runtime between a single- and multiple-panel display.

Methods

Derived from TCustomStatusBar
Create
Destroy
ExecuteAction
FlipChildren
SetBounds
Derived from TWinControl
Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DefaultHandler
DisableAlign
DockDrop
EnableAlign
FindChildControl
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag

BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying

FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Tabcontrol component

Use TTabControl to add a control with multiple tab settings to a form. Unlike a page control, TTabControl is not made up of several pages that contain different controls. Instead, TTabControl is a single object. When the current tab changes, the tab control must directly update its contents to reflect the change in an OnChange event handler.

Main properties : Iamges, MultiLine, MultiSelect, Style, Tabs, TabIndex.

Methods

Derived from TCustomTabControl

Create

Destroy

GetHitTestInfoAt

IndexOfTabAt

RowCount

ScrollTabs

TabRect

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient

SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Toolbar component

A TToolBar component manages tool buttons and other controls, arranging them in rows and automatically adjusting their sizes and positions.

Main properties : ButtonCount, ButtonHeight, Buttons,ButtonWidth, Canvas, Customizable, CustomizeKeyName,CustomizeValueName, DisabledImages, Flat, HotImages, Images, Indent, List, Menu, RowCount, ShowCaptions, Transparent,Wrapable.

Methods

In TToolBar

AlignControls

CanAutoSize

CancelMenu

ChangeScale

CheckMenuDropdown

ClickButton

Create

CreateParams

CreateWnd

CustomDraw

CustomDrawButton

Destroy

DoQueryDelete

DoQueryInsert

FindButtonFromAccel

FlipChildren

GetChildren

InitMenu

IsCustomDrawn

Loaded

Notification

RepositionButton

RepositionButtons

TrackMenu

WndProc

WrapButtons

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment

UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Trackbar component

Use a TTrackBar component to put a track bar on a script form. A track bar represents a position along a continuum using a slider and, optionally, tick marks. A track bar can also display a selected range marked by triangular ticks at the starting and ending positions of the selection.

Main properties : Min, Max, Position, SliderVisible, ThumbLength, Orientation, LineSize, Frequency

Methods

In TTrackBar

Create

SetTick

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform

Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Treeview component

Use TTreeView to add an expanding and contracting outline to a form. Each node in a tree view control consists of a label and a number of optional bitmapped images. Each node can have a list of subnodes associated with it. By clicking on a node, the user can expand or collapse the associated list of subnodes.

Main properties :
AutoExpand,BorderStyle,Canvas,ChangeDelay,DropTarget,HideSelection,HotTrack,Images,Indent,Items,MultiSelect,MultiSelectStyle,ReadOnly,RightClickSelect,RowSelect,Selected

SelectionCount,Selections,ShowButtons,ShowLines,ShowRoot,,SortType,StateImages,ToolTips, TopItem.

Methods

Derived from TCustomTreeView

AlphaSort

ClearSelection

Create

CustomSort

Deselect

Destroy

FindNextToSelect

FullCollapse

FullExpand

GetHitTestInfoAt

GetNodeAt
GetSelections
IsEditing
LoadFromFile
LoadFromStream
SaveToFile
SaveToStream
Select
Subselect
Derived from TWinControl
Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DefaultHandler
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds

SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading

UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Updown component

Use a TUpDown component to add an up-down control to a form. Up-down controls consist of a pair of arrow buttons, such as the arrows that appear in a spin box. Up-down controls allow users to change the size of a numerical value by clicking on arrow buttons.

Main properties : AlignButton, ArrowKeys, Increment, Associate, Min, Max, Orientation, Position, Thousands, Wrap.

Methods

Derived from TCustomUpDown

Create

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

Destroy

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh

ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance

Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Dialogs Tab

Dialogs tab

The Dialogs tab deals with OpenDialog, SaveDialog, OpenPictureDialog, SavePictureDialog, Font dialog, Color dialog, Print dialog, Printer Setup dialog, Find dialog, and Replace dialog components.

See also

Color dialog component

Find dialog component

Font dialog component

Open dialog component

OpenPicture component

Print dialog component

Printer Setup dialog component

Replace dialog component

Save dialog component

SavePicture dialog component

Color dialog component

The TColorDialog component displays a Windows dialog box for selecting colors. The dialog does not appear at runtime until it is activated by a call to the Execute method. When the user selects a color and clicks OK, the dialog closes and the selected color is stored in the Color property.

Methods

In TColorDialog

Create

Destroy

Execute

Derived from TCommonDialog

DefaultHandler

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Find dialog component

The TFindDialog component displays a modeless Windows dialog box that prompts the user for a search string. The dialog does not appear at runtime until it is activated by a call to the Execute method. MainProperty : FindText.

Methods

In TFindDialog

CloseDialog

Create

Destroy

Execute

Derived from TCommonDialog

DefaultHandler

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Font dialog component

The TFontDialog component displays a modal Windows dialog box for selecting fonts. The dialog does not appear at runtime until it is activated by a call to the Execute method. When the user selects a font and clicks OK, the dialog closes and the selected font is stored in the Font property.

Methods

In TFont

Assign

Create

Destroy

Derived from TGraphicsObject

HandleAllocated

Derived from TPersistent

GetNamePath

Derived from TObject

AfterConstruction

BeforeDestruction

ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance
SafeCallException

Open dialog component

The TOpenDialog component displays a modal Windows dialog box for selecting and opening files. The dialog does not appear at runtime until it is activated by a call to the Execute method. When the user clicks Open, the dialog closes and the selected file or files are stored in the Files property.

Methods

In TOpenDialog
Create
Destroy
Execute
GetStaticRect
Derived from TCommonDialog
DefaultHandler
Derived from TComponent
BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

OpenPicture dialog component

Add a TOpenPicture component to an action list to add a graphics file selection dialog to your application. Controls such as menu items and tool buttons linked to this action cause the application to display the open picture dialog when invoked.

Use the OnAccept event to respond after the user selects a file name in the dialog or the OnCancel event to respond when the user cancels from the dialog. In the event handler, you can read the file name from the dialog specified by the Dialog property.

Methods

Derived from TCommonDialogAction

Create

ExecuteTarget

HandlesTarget

Derived from TCustomAction

Destroy

DoHint

Execute

Derived from TContainedAction

GetParentComponent

Update

Derived from TBasicAction

RegisterChanges

UnRegisterChanges

UpdateTarget

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease
GetNamePath
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Print dialog component

The TPrintDialog component displays a standard Windows dialog box for sending jobs to a printer. The dialog is modal and does not appear at runtime until it is activated by a call to the Execute method.

Methods

In TPrintDialog

Execute

Derived from TCommonDialog

Create

DefaultHandler

Destroy

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Printer Setup dialog component

The TPrinterSetupDialog component displays a modal Windows dialog box for configuring printers. The contents of the dialog vary depending on the printer driver selected.

The dialog does not appear at runtime until it is activated by a call to the Execute method.

Methods

In TPrinterSetupDialog

Execute

Derived from TCommonDialog

Create

DefaultHandler

Destroy

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Replace dialog component

The TReplaceDialog component is a special version of TFindDialog that prompts the user for both a search string and a replace string. Like the Find dialog, the Replace dialog is modeless and does not appear at runtime until activated by a call to the Execute method.

Methods

In TReplaceDialog

Create

Derived from TFindDialog

CloseDialog

Destroy

Execute

Derived from TCommonDialog

DefaultHandler

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Save dialog component

The TSaveDialog dialog displays a modal Windows dialog box for selecting file names and saving files. The dialog does not appear at runtime until it is activated by a call to the Execute method. When the user clicks Save, the dialog closes and the selected file name is stored in the FileName property.

Methods

In TSaveDialog

Execute

Derived from TOpenDialog

Create

Destroy

Derived from TCommonDialog

DefaultHandler

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize

MethodAddress

MethodName

NewInstance

SavePicture dialog component

Add a TSavePicture component to an action list to add a graphics “save as” dialog to your application. Controls such as menu items and tool buttons linked to this action cause the application to display the save picture dialog when invoked.

Use the OnAccept event to respond after the user selects a file name in the dialog or the OnCancel event to respond when the user cancels from the dialog. In the event handler, you can read the file name from the dialog specified by the Dialog property.

Methods

Derived from TCommonDialogAction

Create

ExecuteTarget

HandlesTarget

Derived from TCustomAction

Destroy

DoHint

Execute

Derived from TContainedAction

GetParentComponent

Update

Derived from TBasicAction

RegisterChanges

UnRegisterChanges

UpdateTarget

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Win3.1 Tab

Win3.1 Tab

The Win3.1 tab deals with tabset, tabbed notebook, notebook, header, filelistbox, directorylistbox, drive combobox, filter combobox components.

See also

DirectoryListbox component

Drive combobox component

FileListbox component

Filter combobox component

Header component

Notebook component

Tabbed notebook component

Tabset component

DirectoryListbox component

The TDirectoryListBox component represents a list box control that is aware of the directory structure of the current drive. Main properties : CaseSensitive, Directory, DirLabel, Drive, FileList, PreserveCase.

Methods

In TDirectoryListBox

Create

Destroy

DisplayCase

FileCompareText

GetItemPath

OpenCurrent

Update

Derived from TCustomListBox

AddItem

Clear

ClearSelection

CopySelection

DeleteSelected

SelectAll

ItemAtPos

ItemRect

Derived from TCustomListControl

MoveSelection
Derived from TWinControl
Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DefaultHandler
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock

DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease

GetNamePath

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Drive combobox component

The old TDriveComboBox component implements a specialized combo box that displays all the drives available when the script is executed. Add TDriveComboBox component to a script form to allow users

to select a drive. Use TDriveComboBox along with TFileListBox, TFilterComboBox, and TDirectoryListBox components to add full file selection capabilities to a form. However to add a standard Windows file open or save dialog to a script form, use TOpenDialog or TSaveDialog components instead.

Methods

In TDriveComboBox

Create

Destroy

Derived from TCustomCombo

AddItem

Clear

ClearSelection

CopySelection

DeleteSelected

Focused

SelectAll

Derived from TCustomListControl

MoveSelection

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform

Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

FileListbox component

A old style file list box component. Add TFileListBox to a form to allow users to select a file. Use TFileListBox along with TDriveComboBox, TFilterComboBox, and TDirectoryListBox components to add full file selection capabilities to a form.

Note: To add a standard Windows file open or save dialog to a script form, use TOpenDialog or TSaveDialog component instead.

Methods

In TFileListBox

ApplyFilePath

Create

Destroy

Update

Derived from TCustomListBox

AddItem

Clear

ClearSelection

CopySelection

DeleteSelected

SelectAll

ItemAtPos

ItemRect

Derived from TCustomListControl
MoveSelection
Derived from TWinControl
Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DefaultHandler
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen

HasParent

Hide

InitiateAction

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Filter combobox component

The TFilterComboBox component is a specialized combo box that presents the user with a choice of file filters. Add a filter combo box to a form that includes a file selection control to provide a set of predefined file filters. Use TFilterComboBox along with TDriveComboBox, TDirectoryListBox, and TFileListBox components to add full file selection capabilities to a script form. To add a standard Windows file open or save dialog to a script form, use TOpenDialog or TSaveDialog components instead.

Methods

In TFilterComboBox

Create

Destroy

Derived from TCustomCombo

AddItem

Clear

ClearSelection

CopySelection

DeleteSelected

Focused

SelectAll

Derived from TCustomListControl

MoveSelection

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

GetTabOrderList

HandleAllocated

HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler

ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Header component

The THeader component is provided for backward compatibility. Scripts should use the THeaderControl component instead.

Methods

In THeader

Create

Destroy

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetTextBuf
GetTextLen
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat

MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Notebook component

The Notebook components are frequently used with tab set controls (TTabSet) to let the user select pages in the notebook by clicking a tab. The old style TNotebook component is provided for backward compatibility. Scripts should use TPageControl instead.

Methods

In TNotebook

Create

Destroy

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetTextBuf
GetTextLen

Hide

InitiateAction

IsRightToLeft

ManualDock

ManualFloat

MouseWheelHandler

ParentToClient

Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Tabbed notebook component

The TTabbedNotebook component is an old type of component is provided for backwards compatibility. Scripts should use TPageControl instead.

Methods

In TTabbedNotebook

Create

Destroy

GetIndexForPage

TabFontChanged

Derived from TCustomTabControl

GetHitTestInfoAt
IndexOfTabAt
RowCount
ScrollTabs
TabRect
Derived from TWinControl
Broadcast
CanFocus
ContainsControl
ControlAtPos
CreateParented
CreateParentedControl
DisableAlign
DockDrop
EnableAlign
FindChildControl
FlipChildren
Focused
GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag

BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetTextBuf
GetTextLen
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent

FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
DefaultHandler
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress

MethodName

NewInstance

Tabset component

Tab set controls are commonly used to display tabbed pages within a dialog box. The TTabSet component is an old type and is provided for backward compatibility. Use TTabControl component in 32-bit Windows applications.

You create a set of tabs for the tab set control when you specify a list of strings as the value of the Tabs property. One tab is created for each string.

To determine which tab is currently selected or to use code to select a tab, use the TabIndex property. To find out which tab is the first visible tab in the tab set control or to make a tab the first visible tab, use the FirstIndex property.

Main properties : TabIndex, FirstIndex.

Methods

In TTabSet

Create

Destroy

ItemAtPos

ItemRect

ItemWidth

MinClientRect

SelectNext

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList
HandleAllocated
HandleNeeded
InsertControl
Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetTextBuf
GetTextLen
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler

ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
GetParentComponent
HasParent
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

System Tab

System tab

The System tab deals with timer, paintbox, media player and ole container components.

See also

Media player component

OLE container

Paintbox component

Timer component

Media player component

Use the MediaPlayer component to enable your script to control a media playing or recording device such as a CD-ROM player, video player/recorder, or MIDI sequencer. The TMediaPlayer component includes a set of buttons (Play, Stop, Eject, and so on) that control a multimedia device such as a CD-ROM drive, MIDI sequencer, or VCR. A multimedia device may be hardware or software.

The media player component contains of multiple buttons. These buttons can be clicked with the mouse, but are not separate objects or button components. The multimedia device is played, paused, stopped, and so on when the user clicks the corresponding button on the TMediaPlayer component. The device can also be controlled by the control methods that correspond to the buttons (Play, Pause, Stop, Next, Previous, Step, Back, StartRecording, and Eject).

The type of multimedia device (such as dtWaveAudio or dtVideodisc) is specified by the DeviceType property. If the device stores its media in a file, the name of the media file is specified by the FileName property. If DeviceType is dtAutoSelect, the media player attempts to determine the type of device from the extension of the file specified by FileName.

To have the media player attempt to open the device specified by DeviceType automatically when the media player component is created at runtime, set the AutoOpen property to True.

Methods

In TMediaPlayer

AutoButtonSet

Back

Click

Close

Create

Destroy

DoNotify

Eject

KeyDown

Loaded

MMNotify

Next

Notification

Open

Paint

Pause

PauseOnly

Play

PostClick

Previous

Resume

Rewind

Save

StartRecording

Step

Stop

Updated

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate
PaintTo
Realign
RemoveControl
Repaint
ScaleBy
ScrollBy
SetBounds
SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform

Refresh

ReplaceDockedControl

ScreenToClient

SendToBack

SetTextBuf

Show

UseRightToLeftAlignment

UseRightToLeftReading

UseRightToLeftScrollBar

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

OLE container

Use TOleContainer to handle many of the complexities of OLE 2.0. TOleContainer lets the user choose an OLE object to insert by simply calling the InsertObjectDialog method. TOleContainer can create either an embedded OLE object or a linked OLE object.

Methods

In TOleContainer

ChangeIconDialog

Close

Copy

Create

CreateLinkToFile

CreateObject

CreateObjectFromFile

CreateObjectFromInfo

Destroy

DestroyObject

DoVerb

GetIconMetaPict

InsertObjectDialog

LoadFromFile

LoadFromStream

ObjectPropertiesDialog

Paste

PasteSpecialDialog

Run

SaveAsDocument

SaveToFile

SaveToStream

UpdateObject

UpdateVerbs

Derived from TWinControl

Broadcast

CanFocus

ContainsControl

ControlAtPos

CreateParented

CreateParentedControl

DefaultHandler

DisableAlign

DockDrop

EnableAlign

FindChildControl

FlipChildren

Focused

GetTabOrderList

HandleAllocated

HandleNeeded

InsertControl

Invalidate

PaintTo

Realign

RemoveControl

Repaint

ScaleBy

ScrollBy

SetBounds

SetFocus
Update
UpdateControlState
Derived from TControl
BeginDrag
BringToFront
ClientToParent
ClientToScreen
Dock
DragDrop
Dragging
DrawTextBiDiModeFlags
DrawTextBiDiModeFlagsReadingOnly
EndDrag
GetControlsAlignment
GetParentComponent
GetTextBuf
GetTextLen
HasParent
Hide
InitiateAction
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
ReplaceDockedControl
ScreenToClient
SendToBack
SetTextBuf
Show
UseRightToLeftAlignment
UseRightToLeftReading

UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException
SetSubComponent
UpdateAction
Derived from TPersistent
Assign
Derived from TObject
AfterConstruction
ClassInfo
ClassName
ClassNameIs
ClassParent
ClassType
CleanupInstance
Dispatch
FieldAddress
Free
FreeInstance
GetInterface
GetInterfaceEntry
GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Paintbox Component

The TPaintBox component allows your script to draw on a script form. Write an OnPaint event handler to render an image directly on the paint box's Canvas. Drawing outside the boundaries of the paint box is prevented. Use TPaintBox to add custom images to a form. Unlike TImage, which displays an image that is stored in a bitmap, icon, or metafile, TPaintBox requires a script to draw the image directly on a canvas.

Methods

In TPaintBox

Create

Paint

Derived from TGraphicControl

Destroy

Derived from TControl

BeginDrag

BringToFront

ClientToParent

ClientToScreen

Dock

DragDrop

Dragging

DrawTextBiDiModeFlags

DrawTextBiDiModeFlagsReadingOnly

EndDrag

GetControlsAlignment

GetParentComponent

GetTextBuf

GetTextLen

HasParent

Hide

InitiateAction
Invalidate
IsRightToLeft
ManualDock
ManualFloat
MouseWheelHandler
ParentToClient
Perform
Refresh
Repaint
ReplaceDockedControl
ScreenToClient
SendToBack
SetBounds
SetTextBuf
Show
Update
UseRightToLeftAlignment
UseRightToLeftReading
UseRightToLeftScrollBar
Derived from TComponent
BeforeDestruction
DestroyComponents
Destroying
ExecuteAction
FindComponent
FreeNotification
FreeOnRelease
GetNamePath
InsertComponent
IsImplementorOf
ReferenceInterface
RemoveComponent
RemoveFreeNotification
SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry

GetInterfaceTable

InheritsFrom

InitInstance

InstanceSize

MethodAddress

MethodName

NewInstance

Timer component

Use the Timer component to trigger an event, either one time or repeatedly, after a measured interval. Write the code that you want to occur at the specified time inside the timer component's OnTimer event. The Timer component has an Interval property that determines how often the timer’s OnTimer event occurs. Use one timer component for each timer in the script.

Methods

In TTimer

Create

Destroy

Derived from TComponent

BeforeDestruction

DestroyComponents

Destroying

ExecuteAction

FindComponent

FreeNotification

FreeOnRelease

GetNamePath

GetParentComponent

HasParent

InsertComponent

IsImplementorOf

ReferenceInterface

RemoveComponent

RemoveFreeNotification

SafeCallException

SetSubComponent

UpdateAction

Derived from TPersistent

Assign

Derived from TObject

AfterConstruction

ClassInfo

ClassName

ClassNameIs

ClassParent

ClassType

CleanupInstance

DefaultHandler

Dispatch

FieldAddress

Free

FreeInstance

GetInterface

GetInterfaceEntry
GetInterfaceTable
InheritsFrom
InitInstance
InstanceSize
MethodAddress
MethodName
NewInstance

Altium Standard Tab

Altium Standard Tab

The Altium Standard tab deals with Standard components but with Altium Theme applied.

See also

TXPEdit component

TXPBitBtn component

TXPCheckBox component

TXPRadioButton component

TXPListBox component

TXPComboBox component

TXPScrollbar component

TXPGroupBox component

TXPRadioGroup component

TXPSpeedButton component

TXPSplitter component

TWinXPTabControl component

TWinXPPageControl component

TXPImageList component

TXPTrackBar component

TXPProgressBar component

TXPSpinEdit component

TXPStatusbar component

TPathLabel component

TXPButtonEdit component

TXPDirectoryEdit component

TXPHistoryEdit component

TXPFIleNameEdit component

TXPButton component

TXPCheckBoxesControl component

TXPExtPanel component

TXPMultiButton component

TXPSplitButton component

TXPButtonEx component

TXPEdit component

This TXPEdit component is based on standard Borland Edit component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Function GetCursorPos : TPoint;

Procedure Clear;

Procedure ClearSelection;

Procedure ClearUndo;

Procedure CopyToClipboard;

Procedure CutToClipboard;

Procedure PasteFromClipboard;

Procedure SelectAll;

Procedure Undo;

Properties

Property CanUndo : Boolean;

Property Color;

Property Modified : Boolean;

Property SelStart : Integer;

Property SelLength : Integer;

Property SelText : TXPString;

Property Text : TCaption;

Property UnicodeText : WideString;

Property UseColor : Boolean;

Property OnChange : TNotifyEvent;

Property Anchors;

Property AutoSelect;

Property AutoSize;

Property BorderStyle;

Property CharCase;

Property Constraints;

Property Ctl3D;

Property Cursor;
Property DragCursor;
Property DragKind;
Property DragMode;
Property Enabled;
Property Font;
Property HideSelection;
Property HotTrack;
Property ImeMode;
Property ImeName;
Property MaxLength;
Property OEMConvert;
Property ParentColor;
Property ParentFont;
Property ParentShowHint;
Property PasswordChar;
Property PopupMenu;
Property ReadOnly;
Property ShowHint;
Property TabOrder;
Property TabStop;
Property Text;
Property Transparent;
Property Visible;

Event Handlers

Property OnChange;
Property OnClick;
Property OnContextPopup;
Property OnDblClick;
Property OnDragDrop;
Property OnDragOver;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDock;

Property OnStartDrag;

TXPBitBtn

This TXPBitBtn component is based on the Borland Bitbtn component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Procedure Click;

Properties

Property Action;

Property Anchors;

Property Caption;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;

Property ParentFont;

Property ParentShowHint;

Property PopupMenu;

Property ShowHint;

Property TabOrder;

Property TabStop;

Property Visible;

Property Cancel;
Property Default;
Property Glyph;
Property Kind;
Property Layout;
Property Margin;
Property ModalResult;
Property MultilineCaption;
Property NumGlyphs;
Property SmoothDraw;
Property Spacing;
Property Style;
Property Color;
Property HotColor;
Property HotFrameColor;
Property PressedColor;

Event Handlers

Property OnMouseEnter;
Property OnMouseLeave;
Property OnClick;
Property OnContextPopup;
Property OnDragDrop;
Property OnDragOver;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;
Property OnExit;
Property OnKeyDown;
Property OnKeyPress;
Property OnKeyUp;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;
Property OnStartDock;

Property OnStartDrag;

TXPCheckBox

This TXPCheckBox component is based on the Borland Checkbox component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Action;

Property Anchors;

Property Caption;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;

Property ParentFont;

Property ParentShowHint;

Property PopupMenu;

Property ShowHint;

Property TabOrder;

Property TabStop;

Property Visible;

Property Alignment : TLeftRight;

Property Checked : Boolean;

Property Color;

Property ParentColor;

Property AllowGrayed : Boolean;

Property State : TCheckBoxState;

Property SmoothDraw;

Event Handlers

Property OnClick;

Property OnContextPopup;

Property OnDragDrop;

Property OnDragOver;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDock;

Property OnStartDrag;

Property OnDblClick;

TXPRadioButton

This TXPRadioButton component is based on the Borland RadioButton component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Action;

Property Anchors;

Property Caption;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;
Property Font;
Property ParentFont;
Property ParentShowHint;
Property PopupMenu;
Property ShowHint;
Property TabOrder;
Property TabStop;
Property Visible;
Property GroupIndex : Integer;
Property SmoothDraw;
Property Alignment : TLeftRight;
Property Checked : Boolean;
Property Color;
Property ParentColor;

Event Handlers

Property OnClick;
Property OnContextPopup;
Property OnDragDrop;
Property OnDragOver;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;
Property OnExit;
Property OnKeyDown;
Property OnKeyPress;
Property OnKeyUp;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;
Property OnStartDock;
Property OnStartDrag;
Property OnDblClick;

TXPListBox

This TXPListbox component is based on the Borland ListBox component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Procedure Clear;

Procedure ClearSelection;

Function ItemAtPos(Pos : TPoint; Existing : Boolean) : Integer;

Function ItemRect(Index : Integer) : TRect;

Procedure SelectAll;

Properties

Property Count : Integer;

Property ItemIndex : Integer;

Property TopIndex : Integer;

Property SelCount : Integer;

Property Selected[Index : Integer] : Boolean;

Property Checked [Index : Integer] : Boolean;

Property State [Index : Integer] : TCheckBoxState;

Property Canvas;

Property Align;

Property AllowGrayed : Boolean;

Property AlphaNumericSorting : Boolean;

Property Anchors;

Property AutoComplete : Boolean;

Property Columns : Cardinal;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property ExtendedSelect : Boolean;

Property Font;

Property ItemHeight : Integer;
Property Items : TStrings;
Property MultiSelect : Boolean;
Property ParentColor;
Property ParentFont;
Property ParentShowHint;
Property PopupMenu;
Property ScrollBars : TXPScrollStyle;
Property ShowHint;
Property Sorted : Boolean;
Property Style : TXPListBoxStyle;
Property TabOrder;
Property TabStop;
Property UseCheckBoxes : Boolean;
Property Visible;

Event Handlers

Property OnClick;
Property OnClickCheck : TNotifyEvent;
Property OnContextPopup;
Property OnDblClick;
Property OnDragDrop;
Property OnDragOver;
Property OnDrawItem : TDrawItemEvent;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;
Property OnExit;
Property OnKeyDown;
Property OnKeyPress;
Property OnKeyUp;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;
Property OnStartDock;

Property OnStartDrag;

TXPComboBox

This TXPComboBox component is based on the Borland ComboBox component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Procedure Clear;

Procedure ClearSelection;

Procedure SelectAll;

Properties

Property DroppedDown : Boolean;

Property AlphaNumericSorting;

Property Anchors;

Property AutoComplete;

Property AutoDropDown;

Property AutoDropWidth;

Property AutoSelect;

Property AutoSize;

Property BorderStyle;

Property CaseSensitive;

Property CharCase;

Property Constraints;

Property Ctl3D;

Property DragCursor;

Property DragKind;

Property DragMode;

Property DropDownCount;

Property Enabled;

Property Font;

Property HideSelection;

Property HotTrack;

Property ImeMode;
Property ImeName;
Property ItemHeight;
Property Items;
Property ItemIndex;
Property MaxLength;
Property OEMConvert;
Property ParentColor;
Property ParentCtl3D;
Property ParentFont;
Property ParentShowHint;
Property PopupMenu;
Property ReadOnly;
Property ShowButton;
Property ShowButton2;
Property ShowHint;
Property Sorted;
Property Style;
Property TabOrder;
Property TabStop;
Property Text;
Property Transparent;
Property Visible;

Event Handlers

Property OnButtonClick;
Property OnButton2Click;
Property OnChange;
Property OnClick;
Property OnCloseUp;
Property OnContextPopup;
Property OnDblClick;
Property OnDragDrop;
Property OnDragOver;
Property OnDropDown;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnSelect;

Property OnStartDock;

Property OnStartDrag;

TXPScrollbar

This TXPScrollBar component is based on the Borland Scrollbar component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Procedure SetParams(APosition, AMin, AMax : Integer);

Properties

Property Align;

Property Anchors;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Kind : TScrollBarKind;

Property LargeChange : TScrollBarInc;

Property Max : Integer;

Property Min : Integer;

Property PageSize : Integer;

Property ParentShowHint;

Property PopupMenu;

Property Position : Integer;

Property ShowHint;

Property SmallChange : TScrollBarInc;

Property TabOrder;

Property TabStop;

Property Visible;

Event Handlers

Property OnContextPopup;

Property OnChange : TNotifyEvent;

Property OnDragDrop;

Property OnDragOver;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnScroll : TScrollEvent;

Property OnStartDock;

Property OnStartDrag;

TXPGroupBox

This TXPGroupBox component is based on the Borland GroupBox component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Align;
Property Anchors;
Property Caption;
Property Constraints;
Property DockSite;
Property DragCursor;
Property DragKind;
Property DragMode;
Property Enabled;
Property Font;
Property ParentColor;
Property ParentFont;
Property ParentShowHint;
Property PopupMenu;
Property ShowHint;
Property SmoothDraw;
Property TabOrder;
Property TabStop;
Property Visible;

Event Handlers

Property OnClick;
Property OnContextPopup;
Property OnDblClick;
Property OnDragDrop;
Property OnDockDrop;
Property OnDockOver;
Property OnDragOver;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;
Property OnExit;
Property OnGetSiteInfo;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDock;

Property OnStartDrag;

Property OnUnDock;

TXPRadioGroup

This TXPRadioGroup component is based on the Borland RadioGroup component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Align;

Property Anchors;

Property Caption;

Property Constraints;

Property DockSite;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;

Property ParentColor;

Property ParentFont;

Property ParentShowHint;

Property PopupMenu;

Property ShowHint;

Property SmoothDraw;

Property TabOrder;

Property TabStop;

Property Visible;

Property Columns : Integer;

Property ItemIndex : Integer;

Property Items : TStrings;

Event Handlers

Property OnClick;

Property OnContextPopup;

Property OnDblClick;

Property OnDragDrop;

Property OnDockDrop;

Property OnDockOver;

Property OnDragOver;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnGetSiteInfo;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDock;

Property OnStartDrag;

Property OnUnDock

TXPSpeedButton

This TXPSpeedButton component is based on the Borland SpeedButton component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Action;

Property AllowAllUp;
Property Anchors;
Property BiDiMode;
Property Color;
Property Constraints;
Property GroupIndex;
Property Down;
Property DropdownMenu;
Property Caption;
Property Enabled;
Property Flat;
Property Font;
Property Glyph;
Property Kind;
Property Layout;
Property Margin;
Property NumGlyphs;
Property ParentColor;
Property ParentFont;
Property ParentShowHint;
Property ParentBiDiMode;
Property PopupMenu;
Property ShowHint;
Property SmoothDraw;
Property Spacing;
Property Style;
Property Transparent;
Property Visible;

Event Handlers

Property OnClick;
Property OnDblClick;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;

TXPSplitter

This TXPSplitter component is based on the Borland Splitter component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

property OnCanResize: TCanResizeEvent;

property OnMoved: TNotifyEvent;

property OnPaint: TNotifyEvent;

Properties

Property DrawNotch : Boolean;

Property Transparent : Boolean;

Property UseDefaultColor : Boolean;

property Canvas;

property Align;

property AutoSnap: Boolean;

property Beveled: Boolean;

property Color;

property Constraints;

property MinSize: NaturalNumber;

property ParentColor;

property ResizeStyle: TResizeStyle;

property Visible;

Event Handlers

TWinXPTabControl

This TXPTabControl component is based on the Borland TabControl component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more

details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Function IndexOfTabAt(X, Y : Integer) : Integer;

Function TabRect(Index : Integer) : TRect;

Procedure ScrollTabs(Delta : Integer);

Procedure SelectNextPage(GoForward : Boolean);

Properties

Property Tabs : TStrings;

Property TabIndex : Integer;

Property Align;

Property Anchors;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;

Property HelpContext;

Property Images : TCustomImageList;

Property ParentFont;

Property ParentShowHint;

Property PopupMenu;

Property Visible;

Property ShowHint;

Property TabOrder;

Property TabPosition : TWinXPTabPosition;

Property TabStop;

Property PassFocusOnChange : Boolean;

Property SmoothDraw;

Property Tabs;

Property TabIndex

Event Handlers

Property OnChange : TNotifyEvent;

Property OnChanging : TTabChangingEvent;

Property OnContextPopup;

Property OnDockDrop;

Property OnDockOver;

Property OnDragDrop;

Property OnDragOver;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnResize;

Property OnStartDock;

Property OnStartDrag;

Property OnUnDock;

TWinXPPageControl

This TXPPageControl component is based on the Borland PageControl component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Function IndexOfTabAt(X, Y : Integer) : Integer;

Function TabRect(Index : Integer) : TRect;

Procedure ScrollTabs(Delta : Integer);

Procedure SelectNextPage(GoForward : Boolean);

Procedure InsertPage(Page : TWinXPTabSheet);

Procedure InsertPageAt(Index : Integer; Page : TWinXPTabSheet);

Procedure RemovePage(Page : TWinXPTabSheet);

Function FindNextPage(CurPage : TWinXPTabSheet; GoForward, CheckTabVisible : Boolean) : TWinXPTabSheet;

Properties

Property Tabs : TStrings;

Property TabIndex : Integer;

Property Align;

Property Anchors;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;

Property HelpContext;

Property Images : TCustomImageList;

Property ParentFont;

Property ParentShowHint;

Property PopupMenu;

Property Visible;

Property ShowHint;

Property TabOrder;

Property TabPosition : TWinXPTabPosition;

Property TabStop;

Property PassFocusOnChange : Boolean;

Property ActivePageIndex : Integer;

Property PageCount : integer; //read-only

Property Pages[Index : integer] : TWinXPTabSheet; //read-only

Property ActivePage : TWinXPTabSheet;

Property SmoothDraw;

Property TabIndex;

Event Handlers

Property OnChange : TNotifyEvent;

Property OnChanging : TTabChangingEvent;

Property OnContextPopup;
Property OnDockDrop;
Property OnDockOver;
Property OnDragDrop;
Property OnDragOver;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;
Property OnExit;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;
Property OnResize;
Property OnStartDock;
Property OnStartDrag;
Property OnUnDock;

TXPImageList

This TXPImageList component is based on the Borland ImageList component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Same as TImageList methods

Properties

Same as TImageList properties

Event Handlers

Same as TImageList event handlers

TXPTrackBar

This TXPTrackbar component is based on the Borland TrackBar component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Procedure SetTick(Value : Integer);

Properties

Property Align;

Property Anchors;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Frequency : Integer;

Property LineSize : Integer;

Property Max : Integer;

Property Min : Integer;

Property Orientation : TTrackBarOrientation;

Property PageSize : Integer;

Property ParentShowHint;

Property PopupMenu;

Property Position : Integer;

Property SliderVisible : Boolean;

Property SelEnd : Integer;

Property SelStart : Integer;

Property ShowHint;

Property TabOrder;

Property TabStop;

Property ThumbLength : Integer;

Property TickMarks : TTickMark;

Property TickStyle : TTickStyle;

Property Visible;

Event Handlers

Property OnContextPopup;

Property OnChange : TNotifyEvent;

Property OnDragDrop;

Property OnDragOver;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnStartDock;

Property OnStartDrag;

TXPProgressbar

This TXPProgressbar component is based on the Borland Progressbar component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Procedure StepIt;

Procedure StepBy(Delta : Integer);

Properties

Property Max : Integer;

Property Min : Integer;

Property Orientation : TProgressBarOrientation;

Property Position : Integer;

Property Smooth : Boolean;

Property Step : Integer;

Property Align;

Property Anchors;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property ParentColor;

Property ParentShowHint;

Property PopupMenu;

Property ShowHint;

Property TabStop;

Property Visible;

Event Handlers

Property OnContextPopup;

Property OnDragDrop;

Property OnDragOver;

Property OnEndDock;

Property OnEndDrag;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDock;

Property OnStartDrag;

TXPSpinEdit

This TXPSpinEdit component is based on the Borland SpinEdit component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Button : TXPUpDown; //read-only
Property Anchors;
Property AutoSelect;
Property AutoSize;
Property Constraints;
Property Cursor;
Property DragCursor;
Property DragKind;
Property DragMode;
Property EditorEnabled : Boolean;
Property Enabled;
Property Font;
Property HotTrack;
Property Increment : Longint;
Property MaxLength;
Property MaxValue : Longint;
Property MinValue : Longint;
Property ParentColor;
Property ParentFont;
Property ParentShowHint;
Property PopupMenu;
Property ReadOnly;
Property ShowHint;
Property TabOrder;
Property TabStop;
Property Transparent;
Property Value : Longint;
Property Visible;

Event Handlers

Property OnChange;
Property OnClick;
Property OnContextPopup;

Property OnDblClick;

Property OnDragDrop;

Property OnDragOver;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDrag;

TXPStatusBar

This TXPStatusBar component is based on the Borland StatusBar component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

This component allows controls to be inserted inside its panels.

Methods

Same as TStatusBar methods

Properties

Same as TStatusBar properties

Event Handlers

Same as TStatusBar event handlers

TPathLabel

This **TPathLabel** component is a label that displays the file path and has extra features such as the XP look and feel. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Same as TLabel methods

Properties

Same as TLabel properties

Event Handlers

Same as TLabel event handlers

TXPButtonEdit

This TXPButtonEdit component is based on the Borland Button component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Anchors;

Property AutoSelect;

Property AutoSize;

Property CharCase;

Property Constraints;

Property Cursor;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;

Property HideSelection;

Property HotTrack;

Property ImeMode;

Property ImeName;
Property MaxLength;
Property OEMConvert;
Property ParentColor;
Property ParentFont;
Property ParentShowHint;
Property PopupMenu;
Property ReadOnly;
Property ShowHint;
Property TabOrder;
Property TabStop;
Property Text;
Property Transparent;
Property Visible;

Event Handlers

Property OnButtonClick;
Property OnChange;
Property OnClick;
Property OnContextPopup;
Property OnDblClick;
Property OnDragDrop;
Property OnDragOver;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;
Property OnExit;
Property OnKeyDown;
Property OnKeyPress;
Property OnKeyUp;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;
Property OnStartDock;
Property OnStartDrag

TXPDirectoryEdit

This **TXPDirectoryEdit** component is based on the Edit Borland component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the **Object Inspector** panel.

Methods

Properties

Property AcceptFiles : Boolean;

Property Anchors;

Property AutoSelect;

Property AutoSize;

Property CharCase;

Property Constraints;

Property Cursor;

Property DialogText : String;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;

Property HideSelection;

Property HotTrack;

Property ImeMode;

Property ImeName;

Property InitialDir : String;

Property MaxLength;

Property MultipleDirs : Boolean;

Property OEMConvert;

Property ParentColor;

Property ParentFont;

Property ParentShowHint;

Property PopupMenu;

Property ReadOnly;

Property ShowHint;

Property TabOrder;

Property TabStop;

Property Text;

Property Transparent;

Property Visible;

Event Handlers

Property OnButtonClick;

Property OnChange;

Property OnClick;

Property OnContextPopup;

Property OnDblClick;

Property OnDragDrop;

Property OnDragOver;

Property OnDropFiles : TNotifyEvent;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDock;

Property OnStartDrag;

TXPHistoryEdit

This **TXPHistoryEdit** component is based on the Edit Borland component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods

and properties. You can also check out the methods and properties for this component in the **Object Inspector** panel.

Methods

Procedure SaveHistory;

Procedure LoadHistory;

Procedure EmptyHistory;

Properties

Property AutoSize;

Property CaseSensitive;

Property CharCase;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property DropDownCount;

Property Enabled;

Property Font;

Property HideSelection;

Property HotTrack;

Property ImeMode;

Property ImeName;

Property ItemHeight;

Property Items;

Property MaxLength;

Property OEMConvert;

Property ParentColor;

Property ParentFont;

Property ParentShowHint;

Property PopupMenu;

Property ReadOnly;

Property RegKey : String;

Property ShowButton2;

Property ShowHint;
Property Sorted;
Property Style;
Property TabOrder;
Property TabStop;
Property Text;
Property Transparent;
Property Visible;

Event Handlers

Property OnButtonClick;
Property OnButton2Click;
Property OnChange;
Property OnClick;
Property OnCloseUp;
Property OnContextPopup;
Property OnDblClick;
Property OnDragDrop;
Property OnDragOver;
Property OnDropDown;
Property OnEndDock;
Property OnEndDrag;
Property OnEnterHit : TNotifyEvent;
Property OnEnter;
Property OnExit;
Property OnKeyDown;
Property OnKeyPress;
Property OnKeyUp;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;
Property OnSelect;
Property OnStartDock;
Property OnStartDrag;

TXPFileNameEdit

This **TXPFileNameEdit** component is based on the Edit Borland component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the **Object Inspector** panel.

Methods

Properties

Property AcceptFiles : Boolean;

Property Anchors;

Property AutoSelect;

Property AutoSize;

Property CharCase;

Property Constraints;

Property Cursor;

Property DefaultExt : String;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Filter : String;

Property FilterIndex : Integer;

Property Font;

Property HideSelection;

Property HotTrack;

Property ImeMode;

Property ImeName;

Property InitialDir : String;

Property MaxLength;

Property OEMConvert;

Property Options : TOpenOptions;

Property ParentFont;

Property ParentShowHint;
Property PopupMenu;
Property ReadOnly;
Property ShowHint;
Property TabOrder;
Property TabStop;
Property Text;
Property Title : String;
Property Transparent;
Property Visible;

Event Handlers

Property OnButtonClick;
Property OnChange;
Property OnClick;
Property OnContextPopup;
Property OnDblClick;
Property OnDragDrop;
Property OnDragOver;
Property OnDropFiles : TNotifyEvent;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;
Property OnExit;
Property OnKeyDown;
Property OnKeyPress;
Property OnKeyUp;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;
Property OnStartDock;
Property OnStartDrag

TXPButton

This TXPButton component is based on the Borland Button component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Align;

Property Anchors;

Property Color;

Property Enabled;

Property Hint;

Property ShowHint;

Property ParentColor;

Property Visible;

Property Caption;

Property Font;

Property MenuDropped : Boolean;

Property State : TXPButtonState;

Property Bitmap : TBitmap;

Property DownBitmap : TBitmap;

Property HotBitmap : TBitmap;

Property PressedBitmap : TBitmap;

Property DarkBitmap : TBitmap;

Property LightBitmap : TBitmap;

Property Alignment;

Property CanBeDown;

Property Cancel;

Property Default;

Property Down;

Property DownColor;

Property FrameColor;

Property HotColor;

Property HotFrameColor;

Property ModalResult;

Property MultilineCaption;

Property OnlyDrawBitmap;

Property PressedColor;

Property SmoothDraw;

Event Handlers

Property OnClick;

Property OnDblClick;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnMouseHover;

TXPCheckBoxesControl

This TXPCheckBoxesControl component is based on the Borland CheckBox component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Procedure BeginUpdate;

Procedure EndUpdate

Properties

Property Checked [Index : Integer] : Boolean;

Property CBEnabled[Index : Integer] : Boolean;

Property Align;

Property Alignment : TLeftRight;

Property Color;

Property ParentColor;

Property CheckBoxes : TXPCheckBoxItems;

Property Enabled;

Property EqualColumns : Boolean;

Property Font;

Property HorizontalGap : Integer;

Property ParentFont;

Property SmoothDraw;

Property TabOrder;

Property VerticalGap : Integer;

Property AutoSize : Boolean;

Event Handlers

TXPExtPanel

This TXPExtPanel component is based on the Borland Panel component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Alignment : TAlignment;

Property BeginColor : TColor;

Property Bitmap : TBitmap;

Property BorderColor : TColor;

Property BorderWidth : TBorderWidth;

Property EndColor : TColor;

Property GradientStyle : TGradientPanelSty

Property Kind : TXPExtPanelKind;

Property Layout : TTextLayout;

Property ParentColor;

Property ShowBorder : Boolean;

Property ShowBottom : Boolean;

Property ShowLeft : Boolean;

Property ShowRight : Boolean;
Property ShowTop : Boolean;
Property Stretch : Boolean;
Property BitmapTransparent : Boolean;
Property TabStop;
Property Transparent : Boolean;
Property UseDefaultColor : Boolean;
Property UseDockManager;
Property Canvas;
Property Align;
Property Alignment;
Property Anchors;
Property AutoSize;
Property BeginColor;
Property Bitmap;
Property BitmapTransparent;
Property BorderColor;
Property BorderWidth;
Property Caption;
Property Color;
Property Constraints;
Property DockSite;
Property DragCursor;
Property DragKind;
Property DragMode;
Property Enabled;
Property EndColor;
Property Font;
Property GradientStyle;
Property Kind;
Property Layout;
Property ParentColor;
Property ParentCtl3D;
Property ParentFont;
Property ParentShowHint;

Property PopupMenu;
Property ShowBorder;
Property ShowBottom;
Property ShowHint;
Property ShowLeft;
Property ShowRight;
Property ShowTop;
Property Stretch;
Property TabOrder;
Property Transparent;
Property UseDefaultColor;
Property UseDockManager;
Property Visible;

Event Handlers

Property OnPaint : TNotifyEvent;
Property OnCanResize;
Property OnClick;
Property OnConstrainedResize;
Property OnContextPopup;
Property OnDblClick;
Property OnDockDrop;
Property OnDockOver;
Property OnDragDrop;
Property OnDragOver;
Property OnEndDock;
Property OnEndDrag;
Property OnEnter;
Property OnExit;
Property OnGetSiteInfo;
Property OnMouseDown;
Property OnMouseMove;
Property OnMouseUp;
Property OnPaint;
Property OnResize;

Property OnStartDock;

Property OnStartDrag;

Property OnUnDock;

TXPMultiButton

This TXPMultiButton component is based on the Borland Button component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Anchors;

Property Captions : TStrings;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;

Property MultilineCaption : Boolean;

Property ParentFont;

Property ParentShowHint;

Property PopupMenu;

Property ShowHint;

Property SmoothDraw;

Property TabOrder;

Property TabStop;

Property Visible;

Event Handlers

Property OnButtonClick : TButtonClickEvent;

Property OnContextPopup;

Property OnDragDrop;

Property OnDragOver;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnMouseDown;

Property OnMouseEnter : TNotifyEvent;

Property OnMouseLeave : TNotifyEvent;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDock;

Property OnStartDrag;

TXPSplitButton

This TXPSplitButton component is based on the Borland Button component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Procedure Click;

Properties

Property Action;

Property Anchors;

Property Caption;

Property Constraints;

Property DragCursor;

Property DragKind;

Property DragMode;

Property Enabled;

Property Font;
Property ParentFont;
Property ParentShowHint;
Property PopupMenu;
Property ShowHint;
Property TabOrder;
Property TabStop;
Property Visible;
Property Cancel;
Property Default;
Property Glyph;
Property Kind;
Property Layout;
Property Margin;
Property ModalResult;
Property MultilineCaption;
Property NumGlyphs;
Property SmoothDraw;
Property Spacing;
Property Style;
Property Color;
Property HotColor;
Property HotFrameColor;
Property PressedColor;
Property DropDownMenu : TPopupMenu;
Property PopupControl : TWinControl;
Property AutoAddCloseButton : Boolean;

Event Handlers

Property OnMouseEnter;
Property OnMouseLeave;
Property OnClick;
Property OnContextPopup;
Property OnDragDrop;
Property OnDragOver;

Property OnEndDock;

Property OnEndDrag;

Property OnEnter;

Property OnExit;

Property OnKeyDown;

Property OnKeyPress;

Property OnKeyUp;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnStartDock;

Property OnStartDrag;

Property OnAfterHidePopup : TNotifyEvent;

Property OnBeforeShowPopup : TNotifyEvent;

TXPButtonEx

This TXPButtonEx component is based on the Borland Button component and has extra features such as the XP look and feel. Consult the Borland Delphi Component reference for more details on methods and properties. You can also check out the methods and properties for this component in the Object Inspector panel.

Methods

Properties

Property Align;

Property Anchors;

Property Color;

Property Enabled;

Property Hint;

Property ShowHint;

Property ParentColor;

Property Visible;

Property Caption;

Property Font;

Property MenuDropped : Boolean;

Property State : TXPButtonState;

Property Images : TCustomImageList;

Property BitmapIndex : Integer;

Property DownBitmapIndex : Integer;

Property HotBitmapIndex : Integer;

Property PressedBitmapIndex : Integer;

Property DarkBitmapIndex : Integer;

Property LightBitmapIndex : Integer;

Property Alignment;

Property CanBeDown;

Property Down;

Property DownColor;

Property FrameColor;

Property HotColor;

Property HotFrameColor;

Property MultilineCaption;

Property OnlyDrawBitmap;

Property PressedColor;

Property SmoothDraw;

Event Handlers

Property OnClick;

Property OnDblClick;

Property OnMouseDown;

Property OnMouseMove;

Property OnMouseUp;

Property OnMouseHover;

Virtual Key Code

This table is a Virtual Key Code Corresponding keys which can be used such as tracking which keys have been pressed in an Edit component on a script form.

VK_LBUTTON Left mouse button

VK_RBUTTON Right mouse button

VK_CANCEL Control+Break

VK_MBUTTON Middle mouse button

VK_BACK Backspace key
VK_TAB Tab key
VK_CLEAR Clear key
VK_RETURN Enter key
VK_SHIFT Shift key
VK_CONTROL Ctrl key
VK_MENU Alt key
VK_PAUSE Pause key
VK_CAPITAL Caps Lock key
VK_KANA Used with IME
VK_HANGUL Used with IME
VK_JUNJA Used with IME
VK_FINAL Used with IME
VK_HANJA Used with IME
VK_KANJI Used with IME
VK_CONVERT Used with IME
VK_NONCONVERT Used with IME
VK_ACCEPT Used with IME
VK_MODECHANGE Used with IME
VK_ESCAPE Esc key
VK_SPACE Space bar
VK_PRIOR Page Up key
VK_NEXT Page Down key
VK_END End key
VK_HOME Home key
VK_LEFT Left Arrow key
VK_UP Up Arrow key
VK_RIGHT Right Arrow key
VK_DOWN Down Arrow key
VK_SELECT Select key
VK_PRINT Print key (keyboard-specific)
VK_EXECUTE Execute key
VK_SNAPSHOT Print Screen key
VK_INSERT Insert key
VK_DELETE Delete key

VK_HELP Help key

VK_LWIN Left Windows key (Microsoft keyboard)

VK_RWIN Right Windows key (Microsoft keyboard)

VK_APPS Applications key (Microsoft keyboard)

VK_NUMPAD0 0 key (numeric keypad)

VK_NUMPAD1 1 key (numeric keypad)

VK_NUMPAD2 2 key (numeric keypad)

VK_NUMPAD3 3 key (numeric keypad)

VK_NUMPAD4 4 key (numeric keypad)

VK_NUMPAD5 5 key (numeric keypad)

VK_NUMPAD6 6 key (numeric keypad)

VK_NUMPAD7 7 key (numeric keypad)

VK_NUMPAD8 8 key (numeric keypad)

VK_NUMPAD9 9 key (numeric keypad)

VK_MULTIPLY Multiply key (numeric keypad)

VK_ADD Add key (numeric keypad)

VK_SEPARATOR Separator key (numeric keypad)

VK_SUBTRACT Subtract key (numeric keypad)

VK_DECIMAL Decimal key (numeric keypad)

VK_DIVIDE Divide key (numeric keypad)

VK_F1 F1 key

VK_F2 F2 key

VK_F3 F3 key

VK_F4 F4 key

VK_F5 F5 key

VK_F6 F6 key

VK_F7 F7 key

VK_F8 F8 key

VK_F9 F9 key

VK_F10 F10 key

VK_F11 F11 key

VK_F12 F12 key

VK_F13 F13 key

VK_F14 F14 key

VK_F15 F15 key

VK_F16 F16 key
VK_F17 F17 key
VK_F18 F18 key
VK_F19 F19 key
VK_F20 F20 key
VK_F21 F21 key
VK_F22 F22 key
VK_F23 F23 key
VK_F24 F24 key
VK_NUMLOCK Num Lock key
VK_SCROLL Scroll Lock key
VK_LSHIFT Left Shift key (only used with GetAsyncKeyState and GetKeyState)
VK_RSHIFT Right Shift key (only used with GetAsyncKeyState and GetKeyState)
VK_LCONTROL Left Ctrl key (only used with GetAsyncKeyState and GetKeyState)
VK_RCONTROL Right Ctrl key (only used with GetAsyncKeyState and GetKeyState)
VK_LMENU Left Alt key (only used with GetAsyncKeyState and GetKeyState)
VK_RMENU Right Alt key (only used with GetAsyncKeyState and GetKeyState)
VK_PROCESSKEY Process key
VK_ATTN Attn key
VK_CRSEL CrSel key
VK_EXSEL ExSel key
VK_EREOF Erase EOF key
VK_PLAY Play key
VK_ZOOM Zoom key
VK_NONAME Reserved for future use
VK_PA1 PA1 key
VK_OEM_CLEAR Clear key

Index

Revision History

Date	Version No.	Revision
01-Dec-2004	1.0	New product release
26-Apr-2005	1.1	Updated for Altium Designer 2004
9 Nov 2005	1.2	Updated for Altium Designer 6
2 May 2007	1.3	Text revision.