

# Delphi Component "TAlphaPanel" in Unit "ALPHABAR"

TObject  
TPersistent  
TControl  
TWinControl  
TCustomPanel  
**TAlphaPanel**



The component TAlphaPanel is a panel component including speedbuttons. TAlphaPanel was designed as an index panel: The user should arrive the next index point with a simple mouse click on a letter button. Such an index point should be the next section of an indexed database table. If so the first record begins with a 'K' must be displayed, if the user clicked the 'K' button. You can use a TAlphaPanel as a "virtual keyboard" - for example for quick input of special characters.

When clicked, TAlphaPanel only raises an event "OnValueChange" wich the application have to respond. The layout, sizes and titles of the buttons the panel includes can be configured by numerous component properties: ⇒"AllowAllUp", "AlphaButtons", "BlankXSize", "BlankYSize", "ButtonFont", "ButtonHeight", "ButtonLeftMargin", "ButtonTopMargin", "ButtonWidth", "ButtonXSpacing", "ButtonYSpacing", "CatchButtons".

## Properties

### inherited

**The following properties correspond to those of TPanel:**

Align, BevelInner, BevelOuter, BevelWidth, BorderStyle, Color, Ctl3D, Cursor, Enabled, Locked, ParentColor, ParentCtl3D, ParentShowHint, PopupMenu, ShowHint, Visible.

### ActiveButton

**property ActiveButton :char default #0**

Contains the value of last pressed button. It's <#0>, if no button in the actual layout were pressed before. If the propertie ⇒"CatchButtons" is <TRUE>, you can set the button wich is pressed down. Writing the "ActiveButton" property raises the event ⇒"OnValueChange".

### AllowAllUp

**property AllowAllUp :boolean default false**

Defines the buttons behavior, if ⇒"CatchButtons" is set to <TRUE> and the active button is pressed again. If "AllowAllUp" is set to <FALSE>, the active button will remain caught, when it was klicked once again; an event ⇒"OnValueChange" will be raised once again. If "AllowAllUp" was set to <TRUE> in the same case, the button switches to "up" position and ⇒"ActiveButton" will be set to <#0>. A "OnValueChange" event occurs too.

### AlphaButtons

**property AlphaButtons :tStringList**

This property reads and writes the button layout. By default it's an empty panel. A new line in the string list correspond to a new line in the panel. A blank character creates a horizontal space (⇒"BlankXSize"). Each single character is the title of a single button! ⇒"ActiveButton" will be set to <#0>, if you define a new layout. An empty line creates a vertical space (⇒"BlankYSize").

## Delphi Component "TAlphaPanel" in Unit "ALPHABAR"

<b>BlankXSize</b>	<b>property BlankXSize :word default 9</b> This property defines the size of a horizontal space (⇒"AlphaButtons", "ButtonXSpacing").
<b>BlankYSize</b>	<b>property BlankYSize :word default 9</b> This property defines the size of a vertical space (⇒"AlphaButtons", "ButtonYSpacing").
<b>ButtonFont</b>	<b>property ButtonFont :tFont</b> This is the font used for the buttons titles.
<b>ButtonHeight</b>	<b>property ButtonHeight :word default 18</b> This is the height of each button.
<b>ButtonLeftMargin</b>	<b>property ButtonLeftMargin :word default 2</b> Defines the left margin for all rows.
<b>ButtonTopMargin</b>	<b>property ButtonTopMargin :word default 2</b> Defines the top margin for the first row.
<b>ButtonWidth</b>	<b>property ButtonWidth :word default 18</b> Defines the width of each button. For the width of a manual space between to buttons refer to ⇒"BlankXSize".
<b>ButtonXSpacing</b>	<b>property ButtonXSpacing :integer default -1</b> By default buttons will be arranged side by side respectively on top of each other. This property defines how to indent a new buttons in the row: left (< 0) or right (> 0).
<b>ButtonYSpacing</b>	<b>property ButtonYSpacing :integer default -1</b> By default buttons will be arranged side by side respectively on top of each other. This property defines how to indent the buttons of a new row: up (< 0) or down (> 0).
<b>CatchButtons</b>	<b>property CatchButtons :boolean</b> This property controls wether a button catches (<TRUE>) or not (<FALSE>, Standard), if it was clicked. If set to <TRUE>, the property ⇒"AllowAllUp" will be interpreted too.
<b>Height</b>	<b>property Height default 22</b> Diese geerbte Eigenschaft wird per Voreinstellung auf 22 gesetzt und ist damit hoch genug für eine Zeile Schalter in der Standardhöhe.
<b>Width</b>	<b>property Width default 448</b> By default this inherited property is set to 448 - big enough for a panel containing all letters from 'A' ...'Z' in the standard button width.

# Delphi Component "TAlphaPanel" in Unit "ALPHABAR"

## Methods - public

- Create**                                    **constructor Create(aOwner :tComponent); override**  
Initializes the component. By default an empty panel will be created.
- Destroy**                                   **destructor Destroy; virtual**  
Releases the component.
- GetButton**                                **function GetButton(value :char) :TSpeedButton**  
This function returns a pointer to the first button with the <tag> value <Ord(value)> or <NIL>, if no such button exists. With this function you can directly access each button. ⇒"AddLetterButtons"

## Methods - protected

- AddLetterButtons**                      **procedure AddLetterButtons; virtual**  
Diese Methode fügt in das leere TAlphaPanel die Schalter gemäß der Definitionen in ⇒"AlphaButtons" ein. Die Schalter werden von Delphi automatisch benannt. Der Ordinalwert der Schalteraufschrift wird in das Feld <tag> des tSpeedButtons eingetragen. Nur der Wert in <tag> dient zur weiteren Identifikation der Schalter! ⇒"GetButton"
- DestroyButtons**                         **procedure DestroyButtons**  
This procedure releases all button of the TAlphaPanel. For internal use only.
- Loaded**                                    **procedure Loaded; override**  
The buttons won't be written into the stream, but the string list ⇒"AlphaButtons". So this overwritten method creates the buttons after reading the component properties.

## Events

- OnValueChange**                         **property OnValueChange :tNotifyEvent**  
This event will be released, after a button in the TAlphaPanel was klicked. Wich button can be read from ⇒"ActiveButton". See also: ⇒"AllowAllUp", "AlphaButtons", "Create".

**VCL component "TAlphaPanel" - version 1.0**

**© 1995 by**

**Ingo Humann**

**Muehlstrasse 3**

**67105 Schifferstadt**

**GERMANY**

**Delphi Component "TAlphaPanel" in Unit "ALPHABAR"**

**CIS: 100116,3354**

**Internet: 100116.3354@compuserve.com**