

RtCheckList Component Ver. 1.12

Copyright 1995 Tomasz Stanczak

Compuserve: 100735,3273

CompuServe Information Manager, Microsoft Word, MSOffice Setup all have a very interesting control. Every time I see it I wish I could use it. In Word select Extras|Optionen|Kompatibilität (german Version - in english it would probably be Extra|Options|Compatibility) and you will see it. There is a listbox with checkboxes in it. Compared with them the standard multiselect listbox seems boring. The selection is also somehow awkward. If you forget to press Shift or Ctrl key all your selections are gone. So I thought I can make it better. And there is a RtCheckList: a listbox full of checkboxes where your selections stay permanent, and because it uses bitmaps to show them, you can make it look as you want it to be. Properties like TextAlign, BmpVAlign, BmpHAlign and WordWrap allow for additional control.

[TRtCheckList](#)

[Registration](#)

[Licence](#)

[Warranty](#)

[Support](#)

[Technical Notes](#)

[Revision History](#)

TRtCheckList

Properties

Methods

Events

TRtCheckList is a descendent of TCustomListBox, so it looks pretty the same and behaves similar. It doesn't have some of its properties which don't make much sense here, like ExtendedSelect or events like OnDrawItem. Some have been added. PictureChecked and PictureUnchecked properties contain bitmaps to show selections, checked and unchecked. PictureControlColor and PictureWindowColor define transparent colors. Values property lets set checkboxes with values independent from Items - which can be used only to presentation purposes then. Checked (selected) values are stored in Checked property. You can set all of your selection at once with the same property or every selection separately with an array property Value.

To control the text and bitmap look and placement there are three alignment properties TextAlign, BmpVAlign and BmpHAlign, as well as WordWrap and ExpandTabs which control text drawing.

The CheckMode property decides about which event will cause check status change. The OnCheck event lets you decide if a check status change should take place and the OnDrawItem event lets you draw the items by yourself.

Checked

Declaration

property Checked: TStrings;

Description

This property contains names or values (if defined in the Values property) of all checked items as TStringList objects.

CheckMode

Declaration

property CheckMode: TRtCheckMode;

Description

This property defines in which way a checkbox can be toggled.

PictureUnchecked

Declaration

property PictureUnchecked: TBitmap;

Description

This property defines the look of an unchecked item.

PictureChecked

Declaration

property PictureChecked: TBitmap;

Description

This property defines the look of a checked item.

PictureControlColor

Declaration

property PictureControlColor: TColor;

Description

Because bitmaps dont have transparent colors (or I dont know how to define one in AppStudio) I had to define a color to be transparent. It will be translated into the controls background color before it will be shown. Defaults to clOlive.

PictureWindowColor

Declaration

property PictureWindowColor: TColor;

Description

This is the second transparent color, because to show a 3D checkbox I had to translate ist background into clWindow color. Defaults to clFuchsia

Value

Declaration

property Value[const Value: string]: Boolean;

Description

Through this property you can set the values of every item separately.

Values

Example

Declaration

property Values: TStrings;

Description

This property lets you check items with text values other than those shown in the listbox, in this way you can make presentation (Items property) independent of programm logic.

BmpHAlign

Declaration

property BmpHAlign: TRtLeftRight;

Description

This property controls the horizontal placement of checkboxes. Two positions are available: rtaLeft and rtaRight.

BmpVAlign

Declaration

property BmpVAlign: TRtVAlignment;

Description

This property controls the vertical placement of checkboxes. Three positions are available: rtaCenter, rtaTop and rtaBottom.

TextAlign

Declaration

property TextAlign: TRtHAlignment;

Description

This property controls text alignment. Three positions are available: rtaLeft, rtaRight and rtaCenter.

ExpandTabs

Declaration

property ExpandTabs: Boolean;

Description

This property's purpose is to control if tabs in text should be expanded.

WordWrap

Declaration

property WordWrap: Boolean;

Description

This property controls the drawing of lines that are longer than the width of TRtCheckGroup control. If set to True, those lines will be broken between words if a word would extend past the edge of the control.

OnCheck

Declaration

property OnCheck: TRtCheckEvent;

Description

This event will be fired on checking/unchecking of every item either with the mouse, keyboard or programatically with Value property.

OnDrawItem

Declaration

property OnDrawItem: TRtDrawEvent;

Description

This event happens if a checkbox element should be drawn.

TRtCheckEvent

Unit

RtChkLst

Declaration

TRtCheckEvent = **procedure** (Sender: TObject; Index: integer; var Value: Boolean) **of object**;

Description

This procedure is called by OnCheck event. Parameter Index points the item to be checked/unchecked and Value is the new check status. To changes simply do Value:=not Value;

TRtDrawEvent

Unit

RtChkLst

Declaration

TRtDrawEvent = **procedure** (Sender: TObject; Index: integer; DrawRect: TRect; State: TOwnerDrawState;
var Handled: Boolean) **of object**;

Description

This procedure is called by OnDrawItem event. Parameter Index points to the item to be drawn. DrawRect defines the rectangle where the drawing should take place. In order to suppress the default drawing Handled must be set to True.

TRtCheckMode

Declaration

```
type TRtCheckMode = (cmNone, cmClick, cmDbClick);
```

Description

This type defines checkboxes possible behaviour:

cmNone	checkboxes toggle only programatically
cmClick	with a single mouse click and keyboard
cmDbClick	with a double click and keyboard

TRtHAlignment

Declaration

```
type TRtHAlignment = rtaLeft..rtaCenter;
```

Description

This is a subtype of TRtAlignment for horizontal alignment of text.

TRtVAlignment

Declaration

```
type TRtVAlignment = rtaCenter..rtaBottom;
```

Description

This is a subtype of TRtAlignment for vertical alignment of bitmaps.

TRtLeftRight

Declaration

```
type TRtLeftRight = rtaLeft..rtaRight;
```

Description

This is a subtype of TRtAlignment for horizontal alignment of bitmaps.

TRtAlignment

Declaration

```
type TRtAlignment = (rtaLeft, rtaRight, rtaCenter, rtaTop, rtaBottom);
```

Description

This type defines all possible alignment values.

Example

```
Items.Clear;  
Items.Add(Item1);  
Values.Clear;  
Values.Add(1);  
Value[1] := True;
```

Value[Item 1] := True; would have also function, but then it would be dependent on what you want to show to the user.

Registration

<i>Product</i>	<i>Price</i>	<i>s&h</i>
RtCheckList	\$15 (22 DM)	2\$ (3 DM)
RtControls	\$39 (60 DM)	2\$ (3 DM)

Only following payment methods will be accepted:

Compuserve: GO SWREG

ID# 7191 RtCheckList alone

ID# 7197 RtControls

Checks: Only checks drawn on a US bank valued in US\$ or drawn on a German bank and valued in DM will be accepted. The checks should be made payable to **Tomasz Stanczak**. Please let us time after sending check, we cannot send you our product before the money is transferred.

Please fill in the Order Form and send with a payment to:

Tomasz Stanczak
Hardenbergstr. 8
31275 Lehrte
Germany

Credit Card Orders: You can order with MC, Visa, Amex, or Discover from Public (software) Library by calling 800-2424-PsL or 713-524-6394 or by FAX to 713-524-6398 or by CIS Email to 71355,470. You can also mail credit card orders to PsL at P.O.Box 35705, Houston, TX 77235-5705.

THE ABOVE NUMBERS ARE FOR CREDIT CARD ORDERS ONLY.
THE AUTHOR OF THIS PROGRAM CANNOT BE REACHED AT THESE NUMBERS.

Any questions about the status of the shipment of the order, refunds, registration options, product details, technical support, volume discounts, dealer pricing, site licenses, non-credit card orders, etc, must be directed to Tomasz Stanczak:

CIS: 100735,3273
Internet: 100735.3273@compuserve.com
WWW: <http://ourworld.compuserve.com/homepages/tomasz>

To order please specify Item# 14429, RtControls Delphi VCL Pack, price \$39 plus s&h \$2 (e-mail) or \$4 (Europe) or \$6 overseas.

To insure that you get the latest version, PsL will notify us the day of your order and we will ship the product directly to you.

Shipping:

Currently only e-mail shipping is supported, so please always state your e-mail address!. In the future post shipping including printed documentation will be possible (no promise!).

Registered users will receive components full source code. Contact the author for multi-user discounts and site licenses.

Order Form

Name (block letters please): _____
Company Name: _____
Street Address: _____
City, State, Zip Code: _____
Country: _____
Phone Number: _____
E-Mail Address: _____

Product Name	Quantity	Price

Shipping	_____	
Total Fee	_____	

Technical Notes

RtCheckList doesnt use real checkboxes, it only draws them on the screen, so that the resources are preserved.

The shareware version of this component runs only if Delphi will be found in memory.

Licence

Program and documentation Copyright 1995 Tomasz Stanczak (RiT). Source code and design time use licensed for a single registered developer. Unlimited royalty-free distribution permitted as part of an EXE file, provided that internal copyright notice is not altered or removed.

Non-registered users may use the software to test it without any time frame, as long as they use the shareware version of RtCheckList. A limited license is granted to copy and distribute shareware version of RtCheckList only for the trial use of others.

Warranty

LIMITED WARRANTY

EXCEPT AS PROVIDED ABOVE, AUTHOR DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE PRODUCT. SHOULD THE PROGRAM PROVE DEFECTIVE, THE PURCHASER ASSUMES THE RISK OF PAYING THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR, OR CORRECTION AND ANY INCIDENTAL OR CONSEQUENTIAL DAMAGES. IN NO EVENT WILL AUTHOR BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING WITHOUT LIMITATION DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION AND THE LIKE) ARISING OUT OF THE USE OR THE INABILITY TO USE THIS PRODUCT EVEN IF AUTHOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Support

Support is available from the author via Compuserve Mail to 100735,3273 , Internet Mail to 100735.3273@compuserve.com or <http://ourworld.compuserve.com/homepages/tomasz>. Currently there is no postal, telephone, or fax support. All support is on a "best effort" basis, and no promise is made that all questions can be answered, problems solved, features added nor bugs exterminated.

Revision History

- 1.12** 04.03.96 New: Items react on accelerator keys.
- 1.11** 10.01.96 a conflict between OwnerDrawVariable TCustomListBox style (necessarey for WordWrap) and Columns>0 has arised. Resolved through setting Columns:=0 if WordWrap=True.

if a RtCheckList.Color=clWhite and Ctl3D=False then Items were painted under bitmaps.
- 1.10** 11.12.95 TextAlign, BmpVAlign, BmpHAlign, WordWrap and ExpandTabs properties have been added.
- 1.04** 22.09.95 OnDrawItem event was added
- 1.03** 15.09.95 property CheckMode added
- 1.02** 12.09.95 error in RtCheckList and RtCheckGroup when checkboxes are not being displayed when clWindow=clWhite corrected.
- 1.01** 05.09.95 OnCheck event was added
- 1.00** 24.08.95 this is the intial version.

Methods

Standard TListBox methods apply.

Events

OnChange

OnCheck

OnClick

OnDbClick

OnDragDrop

OnDragOver

OnDrawItem

OnEndDrag

OnEnter

OnExit

OnKeyDown

OnKeyPress

OnKeyUp

OnMouseDown

OnMouseMove

OnMouseUp

Properties

Align

BmpHAlign

BmpVAlign

BorderStyle

Checked

CheckMode

Color

Columns

Ctl3D

DragCursor

DragMode

Enabled

ExpandTabs

Font

ItemHeight

Items

MultiSelect

ParentColor

ParentCtl3D

ParentFont

ParentShowHint

PictureUnchecked

PictureChecked

PictureControlColor

PictureWindowColor

PopupMenu

ShowHint

Sorted

TabOrder

TabStop

TextAlign

Value

Values

Visible

WordWrap

