

#¹\$²K³+⁴axCool Help

Properties

Methods

Events

axCool is a toolbar control containing a collection of Button objects used to create a toolbar that is associated with an application.

Typically, a toolbar contains buttons that correspond to items in an application's menu, providing a graphic interface for the user to access an application's most frequently used functions and commands.

File Name

axCool.ocx

Distribution

- Unzip the source code with directories intact.
- Load the sample Project1 with Visual Basic. This should register the OCX for use with other VB applications. If it does not load correctly, then from the DOS prompt, move to the install directory and type: **regsvr32 axCool.ocx** (you may need to put **c:\windows\system** in front of regsvr32 if you don't have a path.
- Whenever you want to add the control to a VB application, go to the Project/Components menu and select "ActiveX Coolbar".

Revisions

License

Tech Support

1Hlp_Contents

2Contents

3Contents

4Browse:0005

#⁵\$⁶K⁷+⁸Revisions

1.3

- Initial Release

1.4

- Removed AddItem method (cannot do this at runtime)
- Removed Remove method (cannot do this at runtime)
- Fixed errors when UpdateButtons was used at runtime

#⁹\$¹⁰K¹¹+¹²License

axCool is developed and published by Software Solution. You may use it freely for development with Microsoft Visual Basic 5.0. This product is freeware and includes source code which your may change to suit your purposes.

9Hlp_License

10License

11License

12Browse:0020

#¹³\$¹⁴K¹⁵+¹⁶Tech Support

If you have any problems installing or running axCool, please feel free to contact our technical support department at one of the following:

Internet: kirkq@execpc.com
Telephone: 414-251-0915
Snail Mail: N92W17053 Roger Ave
 Menomonee Falls, WI 53051

HEY! Check out our world wide web page at:

[HTTP://www.execpc.com/~kirkq](http://www.execpc.com/~kirkq)

#¹⁷\$¹⁸K¹⁹+²⁰**Properties**

All of the properties for this control are listed below:

Standard

- Align
- DragItem
- DragMode
- Enabled
- Height
- Index
- Left
- Name
- Tag
- ToolTipText
- Top
- Visible
- Width

Control Specific

- BorderStyle
- Count
- ShowFlatGrey
- TextLabel

#²¹\$²²K²³+²⁴**Methods**

All of the methods for this control are listed below:

Standard

- Drag
- Move
- ShowWhatsThis
- ZOrder

Control Specific

- Item
- ShowAboutBox
- UpdateButtons

#²⁵\$²⁶K²⁷+²⁸**Events**

All of the events for this control are listed below:

Standard
DragDrop
DragOver

Control Specific
Click
DropDownClick

#³³\$³⁴+³⁵ Depending on your host environment, this property/method/event may be referred to by a different name or may not apply to this control. Refer to your host environments documentation or help file for further information.

#³⁶\$³⁷K³⁸+³⁹

BorderStyle Property

Returns or sets the borderstyle for the control

Syntax:
object.**BorderStyle** [= integer]

The property syntax has these parts:

<u>Part</u>	<u>Description</u>
<i>object</i>	An object expression that evaluates to a control object
<i>integer</i>	An integer expression that evaluates to one of the items in the list below

Settings:

<u>Constant</u>	<u>Value</u>	<u>Description</u>
No Border	0	No border
Single	1	Single line border
Thin Raised	2	Thin raised border
Thick Raised	3	Thick raised border
Thin Inset	4	Thin inset border
Thick Inset	5	Thick inset border
Etched	6	Etched single line border
Bump	7	Raised single line border

#⁴⁰\$⁴¹K⁴²+⁴³

Count Property

Returns the number of buttons in the button collection

Syntax:
object.Count

The property syntax has these parts:

<u>Part</u>	<u>Description</u>
<i>object</i>	An object expression that evaluates to a control object

Remarks

#⁴⁴\$⁴⁵K⁴⁶+⁴⁷

ShowFlatGrey Property

Get/Sets a value to determine whether or not to display the picture in greyscale when the mouse is not over the button (Cool Button only)

Syntax:
object.ShowFlatGrey [= boolean]

The property syntax has these parts:

<u>Part</u>	<u>Description</u>
<i>object</i>	An object expression that evaluates to a control object
<i>boolean</i>	A boolean expression that evaluates to true/false

Remarks

#48\$49K50+51

TextLabels Property

Sets/gets a value to determine whether buttons will show the text labels

Syntax:

object.TextLabels [= boolean]

The property syntax has these parts:

<u>Part</u>	<u>Description</u>
<i>object</i>	An object expression that evaluates to a control object
<i>boolean</i>	A boolean expression that evaluates to true/false

Remarks

#⁵²\$⁵³K⁵⁴+⁵⁵

Click Event

Occurs when the button is pushed

Syntax:
object.**Click**(**Index**)

The event syntax has these parts:

<u>Part</u>	<u>Description</u>
<i>object</i>	An object expression that evaluates to a axGrid object
<i>index</i>	An integer expression that evalutes to the index of the button that was pushed

Remarks:

#⁵⁶\$⁵⁷K⁵⁸+⁵⁹

DropDownClick Event

Occurs when a dropdown button is clicked

Syntax:
object.**DropDownClick**

The event syntax has these parts:

<u>Part</u>	<u>Description</u>
<i>object</i>	An object expression that evaluates to a axGrid object
<i>index</i>	An integer expression that evaluates to the index of the dropdown button that was pushed

Remarks:

#⁶⁰\$⁶¹K⁶²+⁶³

Item Method

This can be used by the programmer to set/get information on a specific button, but most of the time the programmer can just use the Custom property sheet button to edit the buttons at design time.

Syntax:
object.Item(Key)

The method syntax has these parts:

<u>Part</u>	<u>Description</u>
object	An object expression that evaluates to a control object
Key	A variant expression that evaluates to an integer representing the index of the button or a string indicating the key of the button

Information:
This item collection contains the following fields:

<u>Name</u>	<u>Description</u>
Bitmap	Object: Picture to display on button
Caption	String: Caption of button
Enabled	Boolean: Determines if button is enabled or not Default=1
Style	Integer: Determines style of button as listed below Default=0
Tag	String: Tag of button
ToolTipText	String: ToolTip text to display when mouse is over button
Visible	Boolean: Determines if button is visible Default=1
Left	Integer: Returns the distance between the internal left edge of an object and the left edge of its container <i>This is autocalculated at design time</i>

Constants

<u>Constant</u>	<u>Value</u>	<u>Description</u>
Cool Button	0	Flat button that highlights upon mouseover
Toolbar Button	1	Button with thin border
Seperator	2	Toolbar seperator (single raised line)
Toolbar Handle	3	Toolbar handle (double raised line)
Standard Button	4	Standard button similar to intrinsic button

Example:

```
axCool.Item(1).Caption="New"  
set axCool.Item(1).Bitmap=loadpicture("new.bmp")  
axCool.Item(1).ToolTipText="New"  
axCool.UpdateButtons
```


#⁶⁴\$⁶⁵K⁶⁶+⁶⁷

ShowAboutBox Method

Show the about box

Syntax:
object.ShowAboutBox

The method syntax has these parts:

<u>Part</u>	<u>Description</u>
object	An object expression that evaluates to a control object

Example:

#⁶⁸\$⁶⁹K⁷⁰+⁷¹

UpdateButtons Method

Redraws all the buttons from the button collection. The user can utilize this method if it is necessary to change something about a button from code.

Syntax:
object.UpdateButtons

The method syntax has these parts:

<u>Part</u>	<u>Description</u>
object	An object expression that evaluates to a control object

Example:

```
axCool.Item(1).Caption="New caption"  
axCool.UpdateButtons
```

