# ToolButton Help Contents

 **Tool Button**

 **Context Help**

# About Property

**Applies To**
 ContextHelp, ToolButton

**Description**
 Displays version and copyright information for the control; read-only at design time and not available at run time.

**Remarks**
 You can click the ellipsis ("...") button to the right of the settings box in the properties window to display the About dialog.   The dialog contains the toolbox icon for the control, the version number of the control, and copyright information.

**Data Type**
 None

# ButtonSource Property

**Applies To**
ToolButton

**Description**
Sets or returns the source of the displayed button face.

**Usage**
[*form*.]*toolbutton*.**ButtonSource**[ = *source%*]

**Remarks**
The ButtonSource property settings are:

| Setting | Description |
| --- | --- |
| 0 | (Default) Standard.  The button face will be taken from the StandardButton property. |
| 1 | Custom.  The button face will be taken from the CustomButton property. |

If ButtonSource is changed from 1 (custom) to 0 (standard), several other properties are reset to insure their validity:

- AutoSize is set to -1 (true)
- ButtonType is set to 0 (command)
- Value is set to 0 (up)

**Data Type**
Integer (Enumerated)

# ButtonType Property

**Applies To**
 ToolButton

**Description**
 Sets or returns the type of button.

**Usage**
 [*form*.]*toolbutton*.**ButtonType**[ = *type%*]

**Remarks**
 The ButtonType property settings are:

| Setting | Description |
| --- | --- |
| 0 | (Default) Command.   Remains in the up position after being clicked. |
| 1 | Attribute.   Alternates between up and down when clicked. |

 ButtonType may only be set to 1 (attribute) if the ButtonSource property is set to 1 (custom).   The standard buttons are all command buttons.   If ButtonType is set to 0 (command), Value is automatically reset to 0 (up).

**Data Type**
 Integer (Enumerated)

# ⌨? ContextHelp

**Description**
  ContextHelp is a control that you can use to permit a user to request context-sensitive help.

**Remarks**
  An application sets the Enabled property to True (-1) to enable context-sensitive help mode.   ContextHelp displays the mouse cursor shown in the icon above and clips it to its parent form, so that the user can only select controls belonging to that form.   The ControlClick event procedure is called when the user presses the left mouse button to select a control, or cancels.   The user may cancel by either pressing the right mouse button or the Esc key.

**Properties**

| | | |
|---|---|---|
| About | Index | Tag |
| Name | Left | Top |
| Enabled | Parent | |

**Events**
ControlClick


# ControlClick Event

**Applies To**
  ContextHelp

**Description**
  Occurs when the user presses and releases the left or right mouse button, after the Enabled property of the ContextHelp control has been set to True (-1).

**Syntax**
  **Sub** *ctlname*_**ControlClick (***HelpID* **As Long,** *Position* **As Long)**

**Remarks**
  ControlClick uses these arguments:

| Argument | Description |
|---|---|
| *HelpID* | The value of the ContextHelpID property of the control which the user clicked. If the clicked control has no ContextHelpID property, or its value is zero, ContextHelp tries its container, then its container's container, etc.   *HelpID* will be zero if no ContextHelpID was found, or if the user cancelled. |
| *Position* | A numeric value indicating what part of the control was clicked.   *Position* will be negative if no ContextHelpID was found, or if the user cancelled. |

If you need to test the *Position* argument, you can declare constants that define the possible values by loading the TOOLCONS.TXT file into the global module.   The *Position* constants have the following values:

| Constant | Value | Meaning |
|---|---|---|
| HTERROR | -2 | No HelpContextID was found, or user cancelled |
| HTCLIENT | 1 | In a client area |
| HTCAPTION | 2 | In a title bar |
| HTSYSMENU | 3 | In a System menu (also referred to as a Control menu) |
| HTSIZE | 4 | In a size box |

| | | |
|---|---|---|
| HTMENU | 5 | In a menu area |
| HTHSCROLL | 6 | In a horizontal scroll bar |
| HTVSCROLL | 7 | In a vertical scroll bar |
| HTMINBUTTON | 8 | In a Minimize button |
| HTMAXBUTTON | 9 | In a Maximize button |
| HTLEFT | 10 | In the left border of a window |
| HTRIGHT | 11 | In the right border of a window |
| HTTOP | 12 | In the upper horizontal border of a window |
| HTTOPLEFT | 13 | In the upper-left corner of a window border |
| HTTOPRIGHT | 14 | In the upper-right corner of a window border |
| HTBOTTOM | 15 | In the lower horizontal border of a window |
| HTBOTTOMLEFT | 16 | In the lower-left corner of a window border |
| HTBOTTOMRIGHT | 17 | In the lower-right corner of a window border |
| HTBORDER | 18 | In the border of a window that does not have a sizing border |

**Note** These constants are actually the return value of the WM_NCHITTEST message. Consult Windows API documentation for more details.

**Example**

The example simply prints the ContextHelpID of the selected control.   To try this example, create a form (set HelpContextID = 1) with a command button (set HelpContextID = 2) and a ContextHelp control.

## 'ControlClick Event Example

```
Sub ContextHelp1_ControlClick (HelpID As Long, Position As Long)

    If (Position >= 0) Then
        MsgBox "HelpContextID = " + HelpID, 48
    Else
        ' User cancelled
    End If

End Sub

Sub Command1_Click ()

    ' Enable context-sensitive help
    ContextHelp1.Enabled = -1

End Sub
```

# CustomButton Property

**Applies To**
ToolButton

**Description**
Sets or returns the custom button face to be displayed.

**Usage**
[*form*.]*toolbutton*.**CustomButton**[ = *button%*]

**Remarks**
The CustomButton property determines which button face to display from the custom set.

The valid range for *button* is 0 - CustomCount inclusive.   A value of 0 indicates that a blank button face will be used.

**Data Type**
Integer

# CustomCount Property

**Applies To**
ToolButton

**Description**
Sets or returns the number of button faces in the custom set.

**Usage**
[*form*.]*toolbutton*.**CustomCount**[ = *count%*]

**Remarks**
The CustomCount property defines the valid range of values for the CustomButton property.   It must evenly divide the Picture width.

**Data Type**
Integer

# HintMessage Property

**Applies To**
ToolButton

**Description**
Sets or returns the hint message associated with the button.

**Usage**
[*form*.]*toolbutton*.**HintMessage**[ = *message$*]

**Remarks**
The HintMessage property provides a place to store a one-line help message to be displayed in the status bar of an application when the MouseDown event occurs.

**Data Type**
String

**Example**
The example displays a one-line help message in a status bar while the ToolButton is pressed.   To try this example, create a form with a ToolButton (set HintMessage = "One-line help") and a Label control.

## 'HintMessage Property Example

```
Sub ToolButton1_MouseDown (Button As Integer, Shift As Integer, X As Single, Y
As Single)

        ' Display help message associated with this button
        Label1.Caption = ToolButton1.HintMessage

End Sub

Sub ToolButton1_MouseUp (Button As Integer, Shift As Integer, x As Single, Y
As Single)

        ' Clear the help message
        Label1.Caption = ""

End Sub
```

# Preserve Property

**Applies To**

ToolButton

**Description**

Sets or returns the standard colors to preserve when displaying the button.

**Usage**

[*form*.]*toolbutton*.**Preserve**[ = *mask%*]

**Remarks**

The Preserve property defines the set of standard colors which will not be converted to the Windows system colors when displayed.   By default, the following color conversions are performed:

| Source | Converted Color |
|---|---|
| Black | COLOR_BTNTEXT |
| Dark Gray | COLOR_BTNSHADOW |
| Light Gray | COLOR_BTNFACE |
| White | COLOR_BTNHILIGHT |

The Preserve property is a bit field, for which the following bit values are accepted:

| Constant | Value | Meaning |
|---|---|---|
| PRESERVE_NONE | 0 | Convert all colors as shown above. (Default) |
| PRESERVE_BLACK | &H100 | Do not convert black. |
| PRESERVE_DKGRAY | &H200 | Do not convert dark gray. |
| PRESERVE_LTGRAY | &H400 | Do not convert light gray. |
| PRESERVE_WHITE | &H800 | Do not convert white. |
| PRESERVE_ALL | &HF00 | Do not convert any colors. |

**Note** No conversion is performed for White in Windows 3.0, because the button hilight color cannot be set from the Control Panel.

**Data Type**

Integer

**Example**

The example displays the button with the black and light gray colors on a ToolButton preserved while the button is pressed.   To try this example, create a form with a ToolButton (set StandardButton = "1 - Edit Cut").   Use the Control Panel to set alternate button colors.

## 'Preserve Property Example

```
Sub ToolButton1_MouseDown (Button As Integer, Shift As Integer, X As Single, Y
As Single)

        Const PRESERVE_BLACK = &H100
        Const PRESERVE_LTGRAY = &H400

        ' Preserve black and light gray
        ToolButton1.Preserve = PRESERVE_BLACK Or PRESERVE_LTGRAY

End Sub

Sub ToolButton1_MouseUp (Button As Integer, Shift As Integer, x As Single, Y
As Single)

        Const PRESERVE_NONE = 0

        ' Allow all color conversions
        ToolButton1.Preserve = PRESERVE_NONE

End Sub
```

# StandardButton Property

**Applies To**

ToolButton

**Description**

Sets or returns the standard button face to be displayed.

**Usage**

[*form*.]*toolbutton*.**StandardButton**[ = *button%*]

**Remarks**

The StandardButton property settings are:

| Setting | Description |
| --- | --- |
| 0 | (Default) Blank |
| 1 | Edit Cut |
| 2 | Edit Copy |
| 3 | Edit Paste |
| 4 | File New |
| 5 | File Open |
| 6 | File Save |
| 7 | File Print |
| 8 | Help |
| 9 | Context-Sensitive Help |

**Data Type**

Integer (Enumerated)

⊞ **Tool Button**

**Description**
  A tool button displays a graphical button face from a standard set or a custom set.   It can behave as either a command button, or an attribute button.

**Remarks**
  The ButtonSource property allows you to choose between the standard set of button faces and a custom set.

  The ButtonType property allows you to choose between command and attribute button behavior.   All standard buttons are of the command type; only custom buttons may be of the attribute type.

  The Picture property represents a custom set of button faces.   It may be set to (none) or (Bitmap); metafile and icon formats are not supported.   If it is set to (Bitmap), the picture must be a horizontal sequence of button faces.   Use the CustomCount property to indicate the number of button faces in the custom set, and the CustomButton property to select which button face to display.   Setting the Picture property automatically resets the CustomCount and CustomButton properties to zero.

  When using custom button faces, create a bitmap containing all of the button faces for a particular form, and assign this bitmap to the Picture property of only one button at design time.   In the Form_Load event, copy the Picture and CustomCount properties, and assign the appropriate CustomButton value.   This method uses fewer Windows resources than assigning the Picture property of every custom button at design time.

  To make custom buttons the same size as standard buttons, create button faces which are 16 pixels wide and 15 pixels high.

  **Note:**   Non-standard properties are preceded by an asterisk (*) in the list below.

**Properties**

| | | |
|---|---|---|
| About | DragIcon | Picture |
| AutoSize | DragMode | *StandardButton |
| BackColor | Enabled | Tag |
| *ButtonSource | Height | Top |
| *ButtonType | Index | *Value |
| Name | Left | Visible |
| *CustomButton | MousePointer | Width |
| *CustomCount | Parent | |

**Events**

| | | |
|---|---|---|
| Click | DragOver | MouseMove |
| DblClick | MouseDown | MouseUp |
| DragDrop | | |

**Methods**

| | | |
|---|---|---|
| Drag | Move | Refresh |

# Value Property

**Applies To**
  ToolButton

**Description**
  Sets or returns the button state.

**Usage**
  [*form*.]*toolbutton*.**Value**[ = *state%*]

**Remarks**
  The Value property settings are:

| Setting | Description |
| --- | --- |
| 0 | (Default) Up |
| 1 | Down |
| 2 | Indeterminate |

The Value settings are treated differently depending on the value of the ButtonType property.   For a command button, the button state is always 0 (up).   Setting Value to 1 (down) in code invokes the button's Click event, but does not actually change Value. Setting value to 0 (up) has no effect, and attempting to set it to 2 (indeterminate) results in an error.

For an attribute button, Value automatically changes when the user clicks on the button.   If the Value is 1 (down) when the button is clicked, Value will be changed to 0 (up); otherwise, it will be changed to 1 (down).   Value may also be set from code, but it never invokes a Click event.

The indeterminate state is useful to indicate special attribute conditions.   Consider a button to represent the "bold" attribute in a word processor; if the user selects a block of text which contains some bold characters and some non-bold characters, it is not appropriate to display the button as either down or up.   Instead, the indeterminate state is used.

**Data Type**
  Integer (Enumerated)