



## Help for PicBtn

[Properties](#)

[Events](#)

### Registration Information

### Order Form

### Getting Custom Controls Written

### Description

This custom control for Visual Basic is a command button that has both text and a picture on it. The picture can be scaled and placed above, below, to the right, or to the left of the text. The text may be multiple lines.

### File Name

PICBTN1.VBX

### Object Type

MabryPictureButton

### Compatibility

VB 1.0 and up

**Usage Note** The Caption property can contain multiple lines of text. You can delimit each line with either a carriage return (Chr(13)) or a vertical bar (Chr(124)). Using the vertical bar will allow you to specify multiple lines of text at design-time.

**Distribution Note** When you develop and distribute an application that uses PicBtn, you should install the file PICBTN1.VBX into the users Windows SYSTEM directory. PicBtn has version information built into it. So, during installation, you should ensure that you are not overwriting a newer version of PicBtn.

---

## Properties

All of the properties that apply to this control are in this table. Properties that have special meaning for this control or that only apply to this control are marked with an asterisk (\*).

<u>*AutoSize</u>	<u>*Gap</u>	<u>*PictureDown</u>
<u>BackColor</u>	<u>Height</u>	<u>*PictureUp</u>
<u>Caption</u>	<u>HelpContextID</u>	<u>*Placement</u>
<u>DragIcon</u>	<u>hWnd</u>	<u>TabIndex</u>
<u>DragMode</u>	<u>Index</u>	<u>TabStop</u>
<u>Enabled</u>	<u>Left</u>	<u>Tag</u>
<u>FontBold</u>	<u>*Mode</u>	<u>Top</u>
<u>FontItalic</u>	<u>*MouseFocus</u>	<u>*Value</u>
<u>FontName</u>	<u>MousePointer</u>	<u>Visible</u>
<u>FontSize</u>	<u>Name</u>	<u>Width</u>
<u>FontStrikethru</u>	<u>Parent</u>	
<u>FontUnderline</u>	<u>*PictureDisabled</u>	

Value is the default value for the control.

## Events

All of the events that apply to this control are in this table. Events that have special meaning for this control or that only apply to this control are marked with an asterisk (\*).

[Click](#)

[DragDrop](#)

[DragOver](#)

[GotFocus](#)

[KeyDown](#)

[KeyPress](#)

[KeyUp](#)

[LostFocus](#)

[MouseDown](#)

[MouseMove](#)

[MouseUp](#)

## AutoSize Property

[See Also](#)

[Example](#)

### Description

Determines whether the bitmap is automatically resized to fit the button.

### Usage

*[form.]***AuoSize** = {True|False}

### Remarks

If this property is set to True, the bitmap(s) used for the button is automatically resized to fit the button. All three supported picture types may be re-sized.

### Data Type

Integer (boolean)

**See Also**

Properties:

[Gap](#)

[PictureDisabled](#)

[PictureDown](#)

[PictureUp](#)

[Placement](#)

## ButtonColor Property

### Description

Determines the color of the surface of the button.

### Usage

*[form.]***ButtonColor** [ = *color* ]

### Remarks

This property determines the color of the surface of the button. It defaults to light gray.

### Data Type

Color

## Gap Property

[See Also](#)

[Example](#)

### Description

Determines the distance between the edges, the bitmap, and the text.

### Usage

*[form.][control.]Gap[ = integer ]*

### Remarks

The value of this property determines the minimum distance between the edge and the bitmap. It also determines the distance between the bitmap and the text. This property is expressed in pixels. It defaults to 6.

### Data Type

Integer

**See Also**

Properties:

[AutoSize](#)

[PictureDisabled](#)

[PictureDown](#)

[PictureUp](#)

[Placement](#)



## Mode Property

[See Also](#)

### Description

Determines the way the button rises and lowers.

### Usage

*[form.][control.]Mode[ = integer ]*

### Remarks

This property specifies how the button acts. PicBtn can act like a normal button or a toggle button (for tool bars, etc.). This property may be one of these values:

Value	Description
0	Normal (default) - PicBtn depresses when the button is pressed and rises when the button is released.
1	Toggle - the button depresses on the first click, and rises on the second click. PicBtn will also change state via the <a href="#">Value</a> property.
2	Manual - PicBtn only changes state when set via the <a href="#">Value</a> property

### Data Type

Integer (enumerated)

**See Also**

Properties:

Value

## MouseFocus Property

### Description

Determines if the control will receive focus.

### Usage

*[form.][control.]MouseFocus* [ = *boolean* ]

### Remarks

This property determines whether or not the button will receive focus when clicked on with the mouse. If this property is set to False, the focus will remain with the previous control when the user clicks on the button with the mouse. If it is set to True, then the focus shifts to this control when the user clicks on it.

This is especially useful with toolbars where you don't want it to receive focus when clicked on. The button can still receive focus if the user tabs to it.

### Data Type

Integer (boolean)

## PictureDisabled Property

[See Also](#)

### Description

Determines the picture used for the control.

### Usage

*[form.]***[control.]PictureDisabled**[ = *picture* ]

### Remarks

The value of these properties determine the pictures that are displayed on the button. PictureUp is the bitmap that's displayed when the button is up; PictureDown is displayed when the button is down.

If PictureDown is not set, PictureUp is used. If neither are set, the button doesn't display bitmaps at all (i.e., it is text-only).

These properties can be bitmaps, icons, or Windows meta files. PicBtn will stretch or shrink any of these, if needed.

### Data Type

Picture

**See Also**

Properties:

[AutoSize](#)

[Gap](#)

[PictureDown](#)

[PictureUp](#)

[Placement](#)

## PictureDown and PictureUp Properties

[See Also](#)

### Description

Determines the picture used for the control.

### Usage

`[form.][control.]PictureDown[ = picture ]`

`[form.][control.]PictureUp[ = picture ]`

### Remarks

The value of these properties determine the pictures that are displayed on the button. PictureUp is the bitmap that's displayed when the button is up; PictureDown is displayed when the button is down.

If PictureDown is not set, PictureUp is used. If neither are set, the button doesn't display bitmaps at all (i.e., it is text-only).

These properties can be bitmaps, icons, or Windows meta files. PicBtn will stretch or shrink any of these, if needed.

### Data Type

Picture

**See Also**

Properties:

[AutoSize](#)

[Gap](#)

[PictureDisabled](#)

[Placement](#)

## Placement Property

[See Also](#)

[Example](#)

### Description

Determines the position of the picture relative to the text.

### Usage

[*form.*][*control.*]**Placement**[ = *integer* ]

### Remarks

The value of this property determines the where the bitmap is in relation to the text. This property is ignored if [PictureDown](#) is not set.

This property may be one of four values:

Value	Description
0	Left of text (default)
1	Above text
2	Right of text
3	Below text
4	Center of button (no text)

### Data Type

Integer (enumerated)



**See Also**

Properties:

[AutoSize](#)

[Gap](#)

[PictureDisabled](#)

[PictureDown](#)

[PictureUp](#)

## Value Property

[See Also](#)

### Description

Determines the state of the control (up or down).

### Usage

`[form.][control.]Value[ = {True|False}]`

### Remarks

This property is True when the button is depressed and False when the button is raised.

When [Mode](#) is zero (Normal), the Click event is fired when this property is set to non-zero. In all other modes, the Click event is fired whenever this property is changed.

### Data Type

Integer (boolean)

**See Also**

Properties:

Mode



## AutoSize, Gap, and Placement Properties Example

In this example, the program will show what happens when you adjust the various visual properties. To try this example, paste the code into the Declarations section of a form that contains a horizontal scroll bar (for Gap), a check box (for AutoSize), a command button (for Placement), and a PicBtn (with the PictureUp property set to a bitmap). Press F5. Play with the various controls.

```
Sub Check1_Click ()
    PicBtn1.AutoSize = Check1.Value
End Sub

Sub Command1_Click ()
    Select Case Val(Left(Command1.Caption, 1))
        Case 0:
            Command1.Caption = "1 - Above text"
            PicBtn1.Placement = 1
        Case 1:
            Command1.Caption = "2 - Right of text"
            PicBtn1.Placement = 2
        Case 2:
            Command1.Caption = "3 - Below text"
            PicBtn1.Placement = 3
        Case 3:
            Command1.Caption = "0 - Left of text"
            PicBtn1.Placement = 0
    End Select
End Sub

Sub Form_Load ()
    Form1.Caption = "PicBtn Sample #1"

    Form1.Height = 3120
    Form1.Width = 3000

    Check1.Top = 240
    Check1.Left = 240
    Check1.Width = 1800
    Check1.Caption = "AutoSize"
    Check1.Value = 1

    HScroll1.Top = 600
    HScroll1.Left = 240
    HScroll1.Width = 1800
    HScroll1.Value = 6
    HScroll1.Min = 0
    HScroll1.Max = 20

    Command1.Top = 960
    Command1.Left = 240
    Command1.Height = 480
    Command1.Width = 1800
    Command1.Caption = "0 - Left of text"

    PicBtn1.Top = 1560
    PicBtn1.Left = 240
```

```
        PicBtn1.Width = 2400
        PicBtn1.Height = 900
End Sub

Sub HScroll11_Change ()
    PicBtn1.Gap = HScroll11.Value
End Sub
```

## Registration Information

### Credits

PicBtn was written by James Shields. Inquiries can be sent to 71231,2066 on CompuServe, or mabry@halcyon.com on Internet. Our mailing address is:

Mabry Software, Inc.  
Post Office Box 31926  
Seattle, WA 98103-1926

### Registration

You can register this program by sending \$15 (\$20 for international orders) and your address. You can register PicBtn **and** its C source code by sending \$50 (\$55 for international orders). With your order, you will receive a copy of our manual documenting all of our controls.

For your convenience, an order form has been provided that you can print out directly from this help file.

### E-mail Discount

You may take a \$5 discount for e-mail delivery of this package (CompuServe or Internet). If you choose this option, please note: a printed manual is not included. Be sure to include your full mailing address with your order. Sometimes (on the Internet) the package cannot be e-mailed. So, we are forced to send it through the normal mails.

CompuServe members may also take the \$5 e-mail discount by registering this package in the software registration forum (GO SWREG). PicBtns SWREG ID number is 3434. The source code version's ID number is 3435.

### Credit Card Orders

We accept VISA and Mastercard. If you e-mail your order to us, please be sure to include your card number, expiration date, complete mailing address, and your phone number (in case we have any questions about your order).

© Copyright 1994-1995 by Mabry Software, Inc.





## PicBtn Order Form

Use the Print Topic.. command from the File menu to print this order form.

Mail this form to: Mabry Software, Inc.  
Post Office Box 31926  
Seattle, WA 98103-1926  
Phone: 206-634-1443  
Fax: 206-632-0272  
BBS: WinDev BBS 206-634-0783  
CompuServe: 71231,2066  
Internet: mabry@halcyon.com

Where did you get this copy of PicBtn?

\_\_\_\_\_

Ship to:

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Phone:

\_\_\_\_\_

Fax:

\_\_\_\_\_

E-Mail:

\_\_\_\_\_

MC/VISA:

\_\_\_\_\_ exp. \_\_\_\_\_

Disk Size:

(circle one)      3½      5¼

qty ordered \_\_\_\_\_

### REGISTRATION

\$15 each, postpaid (check or money order in hard currency). Outside of North America add \$5.00 shipping.

qty ordered \_\_\_\_\_

### SOURCE CODE AND REGISTRATION

\$50 each, postpaid (check or money order in hard currency). Outside of North America add \$5.00 shipping.

## Getting Custom Controls Written

If you or your organization would like to have custom controls written, you can contact me at the following:

James Shields  
Mabry Software, Inc.  
Post Office Box 31926  
Seattle, WA 98103-1926  
Phone: 206-634-1443  
Fax: 206-632-0272  
BBS: WinDev BBS 206-634-0783  
CompuServe: 71231,2066  
Internet: mabry@halcyon.com

You can also contact Zane Thomas. He can be reached at:

Zane Thomas  
Post Office Box 300  
Indianola, WA 98342  
CompuServe: 72060,3327



