



## **ASSOCIATION OF SHAREWARE PROFESSIONALS (ASP) OMBUDSMAN STATEMENT**

**Global Majic Software, Inc.** is a member of the **Association of Shareware Professionals (ASP)**. ASP wants to make sure that the shareware principle works for you. If you are unable to resolve a shareware-related problem with **Global Majic Software, Inc.** by contacting them directly, ASP may be able to help. The ASP Ombudsman can help you resolve a dispute or problem with an ASP member, but does not provide technical support for members' products.

Please write to the ASP Ombudsman at:

545 Grover Road  
Muskegon, MI 49442-9427 USA  
FAX 616-788-2765

or send a CompuServe message via CompuServe Mail to:

ASP Ombudsman 70007,3536

# AUTOREDRAW PROPERTY

## Description

Determines whether the control is redraw manually or automatically.

## Usage

[*form.*]control.**AutoRedraw**[ = {TRUE|FALSE} ]

## Setting

The property settings are:

Setting	Description
<b>TRUE</b>	Automatic (default) - The operating system will redraw the control when it has time.
<b>FALSE</b>	Manual - The user is responsible for all redraw commands.

## Remarks

If AutoRedraw=**TRUE**, then the control will be redrawn after any property is changed. If several properties are being changed rapidly, then the control may seem slow and/or may not update when desired. In this case, it may be wise to set AutoRedraw=**FALSE** and issue a Redraw command after all the desired property changes are made.

## Related Property

Redraw

## Data Type

Integer (Boolean)

# BACKPICTURE PROPERTY

## Description

Determines the graphic to be displayed in the background of the control.

## Usage

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

## Setting

The BackgroundPicture property settings are:

<b>Setting</b>	<b>Description</b>
(none)	No picture is displayed.
(bitmap)	At design time specify the bitmap file name to be displayed. At run-time specify the bitmap using Visual Basic's <b>LoadPicture</b> (or comparable) function.

## Remarks

When setting the picture at design-time, the picture will be saved with the form and will be compiled into the executable.

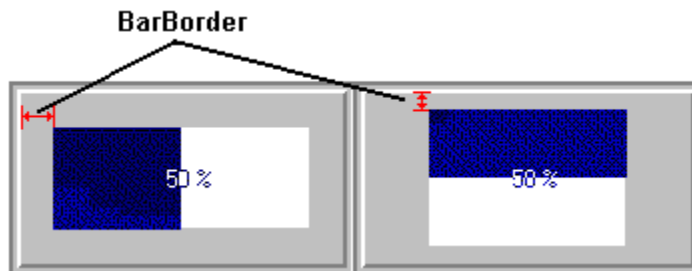
## Data Type

Picture

# BARBORDER PROPERTY

## Description

If the Orientation property is set to horizontal, than this property determines the border size along the left and right sides of the percent indicator as shown in the figure below. If the Orientation property is set to vertical, however, than it determines the border size along the top and bottom of the percent indicator (see figure below). This property is based on a unitless scale and typically has values between 0.0 and 1.0.



## Usage

`[form.]control.BarBorder[ = single ]`

## Related Properties

BarInner, BarOuter and Orientation

## Data Type

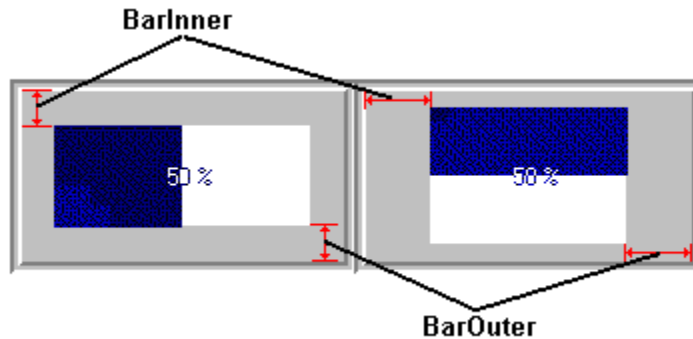
Single

# BARINNER PROPERTY

# BAROUTER PROPERTY

## Description

If the Orientation property is set to horizontal, than these properties determine the top and bottom extents of the percentage indicator as shown in the figure below. If the Orientation property is set to vertical, however, than they determine the left and right extents of the percentage indicator (see figure). These properties are based on a unitless scale and typically have values between 0.0 and 1.0.



## Usage

[form.]control.BarInner[ = single ]

[form.]control.BarOuter[ = single ]

## Remarks

The inner value should be less than the outer value.

## Related Properties

BarBorder and Orientation

## Data Type

Single

# BEVELINNER PROPERTY BEVELOUTER PROPERTY

## Description

Sets or returns the inner or outer shadow style of the control.

## Usage

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

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

## Setting

These property settings are:

Setting	Description
0	None
1	Raised
2	Inset

## Remarks

This property has no affect when BevelWidth=0.

## Related Properties

BevelWidth and BorderWidth

## Data Type

Integer (Enumerated)

## BEVELWIDTH PROPERTY

### Description

Sets or returns the shadow sizes of the inner and outer bevels of the control.

### Usage

`[form.]control.BevelWidth[ = integer ]`

### Related Properties

[BevelInner](#), [BevelOuter](#) and [BorderWidth](#)

### Data Type

Integer

## **BORDERWIDTH PROPERTY**

### **Description**

Sets or returns the border size between the inner and outer bevels of the control.

### **Usage**

`[form.]control.BorderWidth[ = integer ]`

### **Related Properties**

BevelInner, BevelOuter and BevelWidth

### **Data Type**

Integer



# CAPTION PROPERTY

## Description

Determines the text displayed on the control for the caption currently selected by CaptionID. The number of captions displayed is set by the Captions property.

## Usage

`[form.]control.Caption[ = string ]`

## Remarks

See the example for more information on setting caption properties.

## Related Properties

CaptionColor, CaptionFontID, CaptionID, Captions, CaptionX and CaptionY

## Data Type

String

## CAPTIONCOLOR PROPERTY

### Description

Determines the text color for the caption currently selected by [CaptionID](#).

### Usage

`[form.]control.CaptionColor[ = color ]`

### Remarks

This property can be set using Visual Basic's **RGB** or **QBColor** (or comparable) functions. See the [example](#) for more information on setting caption properties.

### Related Properties

[Caption](#), [CaptionFontID](#), [CaptionID](#), [Captions](#), [CaptionX](#) and [CaptionY](#)

### Data Type

Long

## CAPTIONFONTID PROPERTY

### Description

Determines which font (designated by FontID) is used for the caption currently selected by CaptionID.

### Usage

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

### Remarks

See the example for more information on setting caption properties.

### Related Properties

Caption, CaptionColor, CaptionID, Captions, CaptionX and CaptionY

### Data Type

Integer

# CAPTIONID PROPERTY

## Description

Assigns a unique ID to each caption. This property must be set before any other caption property (except Captions). The total number of captions is determined by the Captions property and CaptionID has valid values from 0 to Captions-1.

## Usage

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

## Remarks

The number of Captions must be set before this property can be set. See the example for more information on setting caption properties.

## Related Properties

Caption, CaptionColor, CaptionFontID, Captions, CaptionX and CaptionY

## Data Type

Integer

# CAPTIONX PROPERTY

# CAPTIONY PROPERTY

## Description

Determines the vertical and horizontal position of the caption currently selected by the CaptionID property. These properties are based on a unitless scale and typically have values between -1.0 and 1.0 where a value of 0.0 is located at the center of the control.

## Usage

*[form.]control.CaptionX[ = single ]*

*[form.]control.CaptionY[ = single ]*

## Remarks

See the example for more information on setting caption properties.

## Related Properties

Caption, CaptionColor, CaptionFontID, CaptionID and Captions

## Data Type

Single

## CAPTIONS PROPERTY

### Description

Determines the number of captions displayed on the control. This property must be set before all other caption properties are entered (see [example](#)). The [CaptionID](#) property is used to select the caption to which caption properties apply.

### Usage

`[form.]control.Captions[ = integer ]`

### Remarks

See the [example](#) for more information on setting caption properties.

### Related Properties

[Caption](#), [CaptionColor](#), [CaptionFontID](#), [CaptionID](#), [CaptionX](#) and [CaptionY](#)

### Data Type

Integer

A **CHANGE** event is fired every time the left button is released when using the mouse to change the value of the control.



## Global Majic Software, Inc.



### Percentage Indicator Control

[Properties](#)

[Events](#)

[Product Support](#)

[Copyright](#)

#### Description:

The Percentage Indicator Control can quickly be configured to display percentage bars (horizontal or vertical), ellipses or pre-defined "tanks". Additionally, a user-defined polygon may be used. The display can be filled in a linear or radial (pie) fashion - forward or backward. Bitmaps or colors may be utilized to define the *ON* and *OFF* regions of the display. Percentages are calculated manually or automatically.



## **COPYRIGHT INFORMATION**

All Global Majic Software, Inc. software programs, shareware, and freeware are protected under the copyright laws of the United States and foreign countries. All rights are reserved to Global Majic Software, Inc. Violations of copyright laws are investigated by the FBI. Distribution of Global Majic Software, Inc. products implies that you have read and agreed to the distribution terms described below:

### **INTENT**

Global Majic Software, Inc. seeks to distribute its shareware as widely as possible. However, we want the end-users of our software to be properly informed that it is shareware.

### **DISTRIBUTOR INFORMATION AND LICENSE INFORMATION**

The license information and distribution requirements in this document supersede all previous license statements. To continue to distribute Global Majic Software, Inc. products, you must adhere to the licensing and distribution requirements below.

If you are a mail order or BBS-type distributor of shareware software, you may distribute these programs as they are, without any changes other than expanding files contained in the ZIP archives. However, you have the responsibility to check from time to time, at a minimum interval of 6 months, for new versions of these programs, and to update your copies in a timely manner. Global Majic Software, Inc. will gladly send you a diskette containing the current versions on request.

You must fully identify all Global Majic Software, Inc. programs in your advertising, by the program's full name and version, and indicate the registration fee in the program description. The words Global Majic Software, Inc. must appear in all program descriptions.

### **SHAREWARE DISCLOSURE REQUIRED**

All advertising and packaging information including references to Global Majic Software, Inc. products must contain a statement explaining the shareware concept (see Registration). Specifically, that statement must explain that shareware software **MUST** be registered by the user, after a trial period, by paying a registration fee, and that all monies paid for the shareware version are duplication and distribution charges only. All such statements must be clearly displayed in a position where they are likely to be read by potential customers.

### **RETAIL RACK AND CD-ROM DISTRIBUTION**

If you distribute shareware in a retail setting in racks, store displays, vending machines, at computer fairs, or in any way other than normal BBS or catalog-based sales, you must contact Global Majic Software, Inc. for permission to distribute any Global Majic Software, Inc. program. Rack or retail-like sales require a special distribution license, normally requiring royalties paid to Global Majic Software, Inc. If you distribute shareware on CD-ROM disks, you must also contact Global Majic Software, Inc. before including any Global Majic Software, Inc. shareware programs on a CD-ROM disk. Normally, permission is granted, but current versions must be included and all old versions of any Global Majic Software, Inc. program removed from any CD-ROM disk containing Global Majic Software, Inc. products.

## DIRECTION PROPERTY

### Description

Selects either a forward or backward fill for the control.

### Usage

`[form.]control.Direction[ = integer ]`

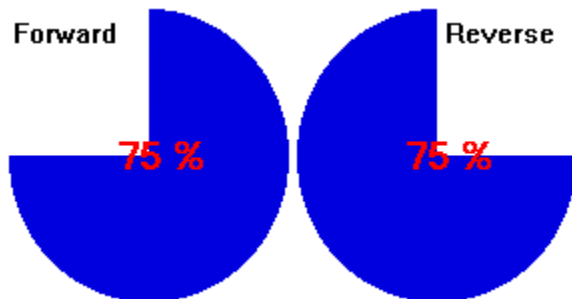
### Setting

The Direction property settings are:

Setting	Description
0	<b>Forward.</b> This is left to right for horizontal orientation, top to bottom for vertical orientation and clockwise for radial <u>DisplayMode</u> .
1	<b>Backward.</b> This is right to left for horizontal orientation, bottom to top for vertical orientation and counter-clockwise for radial <u>DisplayMode</u> .

### Remarks

This picture depicts a control with a vertical Orientation and DisplayMode set to radial. If the Orientation was horizontal, the percentage will fill from the X-axis instead of the Y-axis as depicted below.



### Data Type

Integer (Enumerated)

# DISPLAYMODE PROPERTY

## Description

Determines if the control will be filled in a linear or radial fashion.

## Usage

`[form.]control.DisplayMode[ = integer ]`

## Setting

The DisplayMode property settings are:

Setting	Description
0	<b>Linear</b> (right to left, left to right, top to bottom or bottom to top)
1	<b>Radial</b> (Clockwise or Counter-clockwise)

## Related Properties

[Direction](#) and [Orientation](#)

## Data Type

Integer (Enumerated)

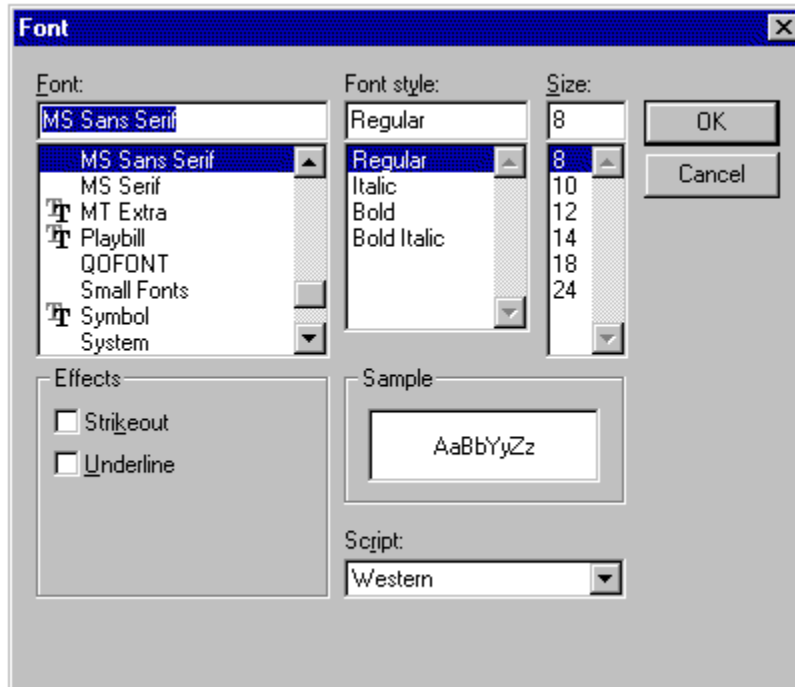
**Events:**

Change  
Click  
GotFocus  
LostFocus  
MouseDown  
MouseMove  
MouseUp

## FontDialog PROPERTY

### Description

Selecting this property (in design mode) launches the font dialog box shown below. This dialog sets the font properties for the font currently selected by FontID.



### Usage

This property can only be used at design time. Use standard font properties to set fonts in code.

### Related Properties

FontBold, FontID, FontItalic, FontName, Fonts, FontSize, FontStrike and FontUnder

### Data Type

N/A

# FONTID PROPERTY

## Description

This property allows the control to display several different fonts by assigning a unique ID to each font. The total number of fonts is determined by the Fonts property and FontID has valid values from 0 to Fonts-1.

## Usage

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

## Remarks

The desired font is obtained by selecting the corresponding FontID (through the use of CaptionFontID for example). See the **example** for more information on setting font properties.

## Related Properties

CaptionFontID, FontBold, FontDialog, FontItalic, FontName, Fonts, FontSize, FontStrike, FontUnder and PercentFontID

## Data Type

Integer

## FONTS PROPERTY

### Description

Determines the number of fonts displayed on the control. This property must be set before all other font properties are entered (see [example](#)). The [FontID](#) is used to select the font to which other font properties apply.

### Usage

`[form.]control.Fonts[ = integer ]`

### Remarks

See the [example](#) for more information on setting font properties.

### Related Properties

[CaptionFontID](#), [FontBold](#), [FontDialog](#), [FontID](#), [FontItalic](#), [FontName](#), [FontSize](#), [FontStrike](#), [FontUnder](#) and [PercentFontID](#)

### Data Type

Integer

# MAX PROPERTY

# MIN PROPERTY

## Description

Sets or returns the maximum or minimum value for the percentage indicator. The percentage caption displays the Percent property and is calculated by changing the Value property:

$$\text{Percent} = ((\text{Value} - \text{Min}) / (\text{Max} - \text{Min})) * 100$$

The Value property can be changed by changing the Percent property:

$$\text{Value} = ((\text{Percent} - \text{Min}) / 100) * (\text{Max} - \text{Min})$$

## Usage

[form.]control.Max[ = single ]

[form.]control.Min[ = single ]

## Remarks

The Min value should be less than the Max value. If this is not the case the control will change either the Min value or the Max value to make the Min less than the Max.

## Data Type

Single



# MOUSECONTROL PROPERTY

## Description

Enables or disables mouse input to the control.

## Usage

`[form.]control.MouseControl[ = {TRUE|FALSE} ]`

## Setting

The MouseControl property settings are:

Setting	Description
<b>True</b>	Allows the control's value to be modified with mouse input.
<b>False</b>	Disables mouse input to the control.

## Data Type

Integer (Boolean)

## **OFFCOLOR PROPERTY**

## **ONCOLOR PROPERTY**

### **Description**

Determines the colors used in the "filled" and "unfilled" portions of the percentage indicator. As the percentage increases, the OffColor will change to the OnColor.

### **Usage**

```
[form.]control.OffColor[ = color ]  
[form.]control.OnColor[ = color ]
```

### **Remarks**

These properties can be set using Visual Basic's **RGB** or **QBColor** (or comparable) functions.

### **Data Type**

Long

# OFFPICTURE PROPERTY

## ONPICTURE PROPERTY

### Description

Determines the graphic to be displayed in the in the "filled" and "unfilled" portions of the percentage indicator. These graphics share the same space in the control so as the percentage increases, the OffPicture will gradually disappear leaving the OnPicture.

### Usage

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

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

### Setting

The property settings are:

<b>Setting</b>	<b>Description</b>
<b>(none)</b>	No picture is displayed.
<b>(bitmap)</b>	At design-time, specify the bitmap file name to be displayed. At run-time, specify the bitmap using Visual Basic's <b>LoadPicture</b> (or comparable) function.

### Remarks

When setting the picture at design-time, the picture will be saved with the form and will be compiled into the executable.

### Data Type

Picture

# ORIENTATION PROPERTY

## Description

Determines how the percentage indicator is arranged within the control's bounds.

## Usage

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

## Setting

The Orientation property settings are:

Setting	Description
0	Horizontal
1	Vertical

## Remarks

When used in conjunction with the Direction and DisplayMode properties, the user can control the way the control fills (left to right-linear fill, top to bottom-radial fill, etc.).

## Data Type

Integer (Enumerated)

# PERCENT PROPERTY

## Description

Sets or returns the control's percentage using the equation below:

$$\text{Percent} = \frac{\text{Value}}{(\text{Max}-\text{Min})} * 100$$

This percentage is displayed as a caption at the center of the control if PercentStyle is **NOT** 2.

## Usage

[*form.*]control.**Percent**[ = *single* ]

## Remarks

The Percent value is rounded to the nearest integer when displayed in the percentage caption.

## Related Properties

PercentColor, PercentFontID and PercentStyle

## Data Type

Single

# PERCENTCOLOR PROPERTY

## Description

Determines the color used for the percentage caption.

## Usage

*[form.]control.PercentColor* [ = *color* ]

## Remarks

This property is only applied if the PercentStyle = 1. It can be set using Visual Basic's **RGB** or **QBColor** (or comparable) functions.

## Related Properties

Percent, PercentFontID and PercentStyle

## Data Type

Long

## PERCENTFONTID PROPERTY

### Description

Determines which font (designated by FontID) is used for the percentage caption.

### Usage

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

### Related Properties

Percent, PercentColor and PercentStyle

### Data Type

Integer

# PERCENTSTYLE

## Description

Determines how the percentage caption is displayed on the control.

## Usage

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

## Setting

The PercentStyle property settings are:

Setting	Description
0	<b>On/Off Negative.</b> The percentage caption is made by swapping the two colors defined by <u>OffColor</u> and <u>OnColor</u> .
1	<b>PercentColor.</b> The percentage caption is a solid color defined by <u>PercentColor</u> .
2	<b>None.</b> No percentage caption is displayed.

## Related Properties

Percent, PercentColor and PercentFontID

## Data Type

Integer (Enumerated)



## PRODUCT SUPPORT

Product support for all products is available to registered users by contacting **Global Majic Software, Inc.** at any of the following locations:

**CompuServe:** 73261,3642

**AmericaOnline:** GMagic

**Internet:** gms@globalmajic.com

**Snail Mail:** Global Majic Software, Inc.  
P.O. Box 322  
Madison, Alabama 35758

**TEL/FAX:** (205) 864-0708

**Home Page:** <http://www.globalmajic.com>

Product Support is free for a period of three (3) months from the date of registration.

If you have a shareware-related problem or dispute that you are unable to resolve with **Global Majic Software, Inc.**, please feel free to contact the [Association of Shareware Professionals](#).

## Properties:

AutoRedraw  
BackColor  
BackPicture  
BarBorder  
BarInner  
BarOuter  
BevelInner  
BevelOuter  
BevelWidth  
BorderWidth  
Caption  
CaptionColor  
CaptionFontID  
CaptionID  
Captions  
CaptionX  
CaptionY  
Direction  
DisplayMode

Enabled  
FontBold  
FontDialog  
FontID  
FontItalic  
FontName  
Fonts  
FontSize  
FontStrike  
FontUnder  
Height  
Index  
Left  
Max  
Min  
MouseControl  
MousePointer  
Name  
OffColor

OffPicture  
OnColor  
OnPicture  
Orientation  
Percent  
PercentColor  
PercentFontID  
PercentStyle  
Redraw  
Shape  
ShapeStyle  
TabIndex  
TabStop  
Tag  
Top  
Value  
Visible  
Width

# REDRAW PROPERTY

## Description

Issues a redraw command to the control if AutoRedraw=**FALSE**.

## Usage

[*form.*]control.Redraw[ = {TRUE|FALSE} ]

## Setting

The property settings are:

Setting	Description
<b>TRUE</b>	Issue a redraw command.
<b>FALSE</b>	Does not issue redraw command.

## Remarks

If AutoRedraw=**TRUE**, then the control will be redrawn after any property is changed. If several properties are being changed rapidly, then the control may seem slow and/or may not update when desired. In this case, it may be wise to set AutoRedraw=**FALSE** and issue a Redraw command after all the desired property changes are made.

## Related Property

AutoRedraw

## Data Type

Integer (Boolean)

## SHAPE PROPERTY

### Description

Determines the shape of a user defined percentage control. This property only applies when ShapeStyle = 3 (User Defined) and defines a polygon made from a list of x,y coordinate pairs. The shape is defined using a coordinate system in which (0,0) is located at the center of the control.

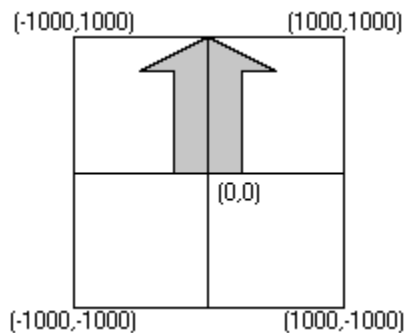
### Usage

`[form.]control.Shape[ = string ]`

### Example

`prcnt1.Shape = "250,0, 250,750, 500,750, 0,1000, -500,750, -250,750, -250,0"`

The following image displays this shape polygon defined in a coordinate system with origin at (0,0) and (1000,1000) extents.



### Related Properties

ShapeStyle

### Data Type

String

# SHAPESTYLE PROPERTY

## Description

Determines how the percentage indicator is displayed on the control.

## Usage

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

## Setting

The ShapeStyle property settings are:

Setting	Description
0	Rectangle
1	Ellipse
2	Tank
3	User Defined (used the <u>Shape</u> property)

## Remarks

The BarBorder, BarInner and BarOuter properties may be used to alter the shape of all ShapeStyles **except** user defined.

## Related Properties

Shape

## Data Type

Integer (Enumerated)

## VALUE PROPERTY

### Description

Sets or returns the control's current value (not the percentage value). This property is related to other properties through the following equation:

$$\underline{\text{Percent}} = (\text{Value}-\text{Min}) / (\underline{\text{Max}}-\underline{\text{Min}}) * 100$$

### Usage

*[form.]control.Percent[ = single ]*

### Remarks

The value should fall between the Max and Min values.

### Data Type

Single

## EXAMPLE - HOW TO BUILD A PERCENTAGE INDICATOR

### General Information

This example will explain the process of building a percentage indicator. The final product is shown in Figure 1. This example explains how several of the control's properties relate to each other. For example, it depicts how to use the properties to setup multiple captions to obtain the desired appearance. In general, the steps for setting up percentage indicators are as follows: **1)** set the control's general properties (BarBorder, BarInner, etc.); **2)** set the desired number of fonts (Prct1.Fonts=2); **3)** set the other font properties; and **4)** change FontID and repeat step 3 for all desired fonts. The caption properties work in a similar fashion. For simplicity, the steps for setting up this example are written out in Visual Basic script. However, the values may also be assigned using the property list at design time.

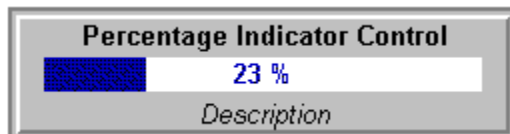


Figure 1

### Sample Code

#### 'general setup

```
Prct1.BarBorder = 0.05  
Prct1.BarInner = 0.35  
Prct1.BarOuter = 0.65  
Prct1.Value = 23.3
```

#### 'setup fonts

```
Prct1.Fonts = 2  
  
Prct1.FontID = 0  
Prct1.FontBold = True  
Prct1.FontName = "Arial"  
Prct1.FontSize = 12
```

```
Prct1.FontID = 1  
Prct1.FontBold = False  
Prct1.FontItalic = True  
Prct1.FontName = "Arial"  
Prct1.FontSize = 12
```

#### 'setup captions

```
Prct1.Captions = 2  
  
Prct1.CaptionID = 0  
Prct1.CaptionFontID = 0  
Prct1.Caption = "Percentage Indicator Control"  
Prct1.CaptionX = 0.5  
Prct1.CaptionY = 0.15
```

```
Prct1.CaptionID = 1  
Prct1.CaptionFontID = 1  
Prct1.Caption = "Description"  
Prct1.CaptionX = 0.5  
Prct1.CaptionY = 0.85
```

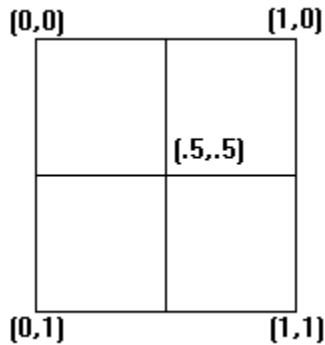
FontBold, FontItalic, FontName, FontSize, FontStrike and FontUnder



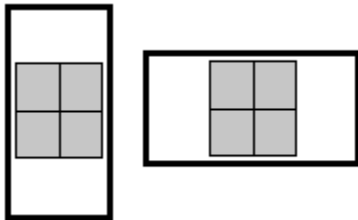
## UNITLESS SCALE

### Description:

Properties which are used to specify position or length use a scale which is based on the size of the control (instead of twips or pixels). For a control which is square, the coordinate system used is depicted with its origin at the center of the control and its width and height measured from -1 to 1.



For controls which are not square, the origin is still at the center of the control. The unit scale used, however, is based on the width or height whichever is smaller so that a unit square fits completely within the control as shown in the two controls below.



**NOTE:** For some variables (i.e., radii, width, etc.), the valid range of the unitless scale is from 0 to 1 (negative values have no meaning)



