SWBMenuButton Control

Displays a button with an optional dropdown menu that is displayed when the button is clicked

Remarks

The SWBMenuButton control creates a 3-D button that displays a small down-facing arrow to the right of the button caption. The button will optionally display a drop-down menu when clicked.

The control allows design-time or run-time configuration of its properties and customization of the menu including options to show the "drop-down" arrow, set button caption text color, and background color for the control. The drop-down menu can be aligned with the left or right edge of the button. The button can also be disabled like the standard button control.

Creating a menu button with SWBMenuButton

Properties

Caption
Enabled
BackColor
ForeColor
Font
ShowDropArrow
MenuAlignment
MenuItems

Methods

AddMenuItem
ClearMenu
Refresh
DeleteMenuItem

Events

Click
DropMenu

Creating a Menu Button

To create a button bar with the SWBMenuButton control:

1 Place the SWBMenuButton on a form.

2 Right-click on the button bar and select Properties from the popup menu.

3 On the Button property page, type the button caption -- the text of the button itself. You can also select how you want the drop-down menu aligned, and whether you want the drop-down arrow to appear on the right side of the button.

4 On the Menu Items tab, youu can type the name of a selection you want to appear on the menu at the Caption prompt. Then click the Add buttoin to add it to the list of menu items below. Repeat this process to add all the menu options you want to have on the drop-down menu. Use the "&" before the character you want to be the "shortcut" key for that selection (optional). You can also add a seperator bar by entering a single hyphen character ("-") as the Caption property. Each item can be enabled or disabled by highlighting the option in the list, and checking or clearing the Enabled checkbox.

5 When you have added all menu selections you need, click OK to update the control.

Caption Property

Type: String

Access: Read/Write

Description:

Sets or returns the text caption that appears on the button.

Enabled Property

Type: Boolean

Access: Read/Write

Description:

Sets or returns the state of the button, either active or inactive. When the button is not enabled (False), it will appear with visually "disabled" text and a "disabled" arrow image (assuming ShowDropArrow is True). When the button is not enabled, no Click events will occur.

BackColor Property

Type: OLE_COLOR

Access: Read/Write

Description:

Sets or returns the color of the background of the button. This color does not affect the drop-down menu.

ForeColor Property

Type: OLE_COLOR

Access: Read/Write

Description:

Sets or returns the color of the foreground, or text color, of the button. The caption of the button will be displayed using this color. This color does not affect the drop-down menu.

Font Property

Type: Font

Access: Read/Write

Description:

Sets or returns the font to be used to display the Caption text on the face of the button. This font does not affect the text on the drop-down menu.

ShowDropArrow Property

Type: Boolean

Access: Read/Write

Description:

Sets or returns the visibility of the drop-down arrow that appears on the right side of the button, indicating that a drop-down menu is available. This option does not, however, control the availability of the menu. That is, the menu will appear even if this property is False.

MenuAlignment Property

Type: Integer

Access: Read/Write

Description:

Sets or returns the method used to visually position the drop-down menu.

State	Setting	Description
maLeftAlign	1	Left edge of menu aligns with left edge of button.
maRightAlign	2	Right edge of menu aligns with right edge of button.

MenuItems Property

Type: Collection of CMenuItem Type

Access: Read/Write

Description:

Allows setting or retrieving of properties of the individual MenuItems on the drop-down list. Note that the items within the collection are numbered starting at 1.

MenuItem Properties

MenuItem Properties

Caption The text caption that appears on the drop-down menu for this selection. This property can incude the "&" character before the letter to be used as the "shortcut" key for this menu selection. You can include a seperator bar by setting the caption to a single hyphen character ("-").

Enabled Determines if this menu selection is enabled when the drop-down menu appears. True = Enabled; False = Disabled.

AddMenuItem Method

```
SWBMenuBtn.AddMenuItem   Caption as String, Optional Enabled as Boolean,
Optional Position as Integer
```

Adds an item to the drop-down menu. Once an item is added to the button bar, it becomes a member of the MenuItems collection, through which these properties can be retrieved or updated (except for Position).

Arguments:

Caption The caption that will appear on the drop-down menu. Use the "&" character before a letter in the caption to make that letter the "shortcut" key for this selection. Use a single hyphen ("-") to create a seperator bar.

Enabled (Optional) Determines if this selection is enabled on the drop-down menu. The default is True

Position (Optional) The ordinal position of this item in the drop-down menu. The default is last (bottom-most). The first position (top-most) is one (1). If an item already exists at the specified position, all existing items are moved down, and the new item is inserted at the position specified.

ClearMenu Method

```
SWBMenuBtn.ClearMenu
```

This method removes *all* menu selections from the drop-down menu. If the menu has no selections it will not appear when the button is clicked.

Refresh Method

```
SWBMenuBtn.Refresh
```

This method causes the button to be redrawn.   It does not affect the drop-down menu contents or appearance.

DeleteMenuItem Method

```
SWBMenuBtn.DeleteMenuItem  Index as Variant
```

Removes the specified item from the drop-down menu.

Arguments:

Index The (integer) position of the menu item to be deleted from the drop-down menu. The position is one-based; that is, the top-most entry on the menu is position one (1).

Click Event

```
Click(MenuItemIndex as Integer, MenuItemCaption as String)
```

Occurs whenever the user selects an item on the drop-down menu.   This event is not fired if the user clicks the button, but does not select an item on the menu.

Arguments:

MenuItemIndex	The position of the menu item that was selected (one-based)
MenuItemCaption	The caption of the menu item that was selected..

DropMenu Event

```
DropMenu()
```

Occurs whenever the user clicks the button and the drop-down menu appears.

Arguments: None