SAVEWARN

Applies to

Window.

Description

Set to true by the system when changes are made to data in bound controls. Reset by the system to false after the READ, WRITE, and CLEAR events.

Usage

status = **Get_Property** (*windowname*, **"SAVEWARN"**)

oldstatus = **Set_Property** (*windowname*, **"SAVEWARN"**, *truefalse*)

Remarks

SAVEWARN specifies whether or not the user should be warned that changes have been made, for example, when the user tries to close the form.

To suppress these messages, reset SAVEWARN to false (0).

Since the system, on CLOSE, SAVE, and CLEAR, checks if the DEFPROP property of the current control has been changed from the GOTFOCUS_VALUE, you may want to send the GOTFOCUS event to the control before resetting SAVEWARN and sending the CLOSE event, so that the GOTFOCUS_VALUE matches DEFPROP. This allows the window to close unconditionally. See the Example section for code implementing this method.

See Also

MODIFIED property

Example

```
* The following code snippet handles the turning off of the SAVEWARN message during the
* CLOSE event of the window.  This code should be placed into the CLOSE Event Handler.
declare function Set_Property, Get_Property, Send_Event

* determine which control has focus
GotFocus_Control = Get_Property(@window,"GOTFOCUS_CONTROL")

* force the matching of the GOTFOCUS and DEFPROP properties on the control which has focus
x = Send_Event(GotFocus_Control,"GOTFOCUS")

* Turn the SAVEWARN message off
x = Set_Property(@window,"SAVEWARN",0)
```

Note from Anonymous User

I found that this example code will not work when to current control with focus is an OLE control. A workaround that I came up with was to add the following line after getting the GOTFOCUS_CONTROL:

Set_Property(GotFocus_Control, "FOCUS", 1)