

Message_Box Function

Description

Displays a Windows message in event context

Syntax

button = **message_box**(*ownerwindow*, *text*, *title*, *options*)

Parameters

The Message_Box function has the following parameters:

Parameter	Description
<i>ownerwindow</i>	The window ID of the window to use as a parent. This parameter can be null to use the screen as a parent. Typically, this parameter is @window . For MDI children, pass the frame ID.
<i>text</i>	Pass the message text. Multiple lines are delimited by CR/LF. CR/LF can be coded as \0D0A\ or char(13):char(10) . The constant MSG_NL\$ is defined in the insert record Message_Box_Equates and can be used to separate multiple lines of text.
<i>title</i>	Pass the title for the message.
<i>options</i>	The options are defined in the insert record Message_Box_Equates under the sections "button arrangement", "default button", and "message icon". To specify multiple options, add the options together. For example:

MSG_BTN_YES_NO_CAN\$ + MSG_DEFAULT2\$ + MSG_ICON_QUESTION\$

Returns

The return values from Message_Box are defined in the insert record Message_Box_Equates. Each button (OK, CANCEL, YES, NO, ABORT, RETRY, IGNORE) returns a different value.

See also

[Msg\(\)](#)

Example: Yes/No/Cancel, Default to 'No' Button

```
declare function Message_Box
$insert Message_Box_Equates
retval = Message_Box( @window , 'Yes/No/Cancel, default to No', |
    'Please Confirm', MSG_BTN_YESNOCAN$ + MSG_DEFAULT2$)
if retval = MSG_RET_YES$ then
    * user clicked 'YES', so do processing, then display message
    retval = Message_Box ( @window, 'Processing completed!', 'Done!', MSG_BTN_OK$)
end
```

Example: Message with Information Icon

```
$insert Message_Box_Equates
Message_Box(@window, "Connection opened!", "Data Download", MSG_ICON_INFO$)
```