

Message Box Wizzard

Current Version 1.4.0

1) History of Updates:

- a) **Version 1.4** - 9/26/98 - Updated to work in Visual Basic 6.0. Tested and works with Windows98/Windows95 and WindowsNT 4.0. Not for use with VB4 or VB5. Utilizes VB6's Connect event to control the connection as an addin. Changed Add-in to be an ActiveX DLL vice EXE

(1) Ignore setup instructions below. Simply copy DLL and this RTF file to a directory and run RegSvr32 with the MsgBoxWiz98.dll as its parameter.

- b) **Version 1.3** - 4/6/97 - With the following updates:

(1) Version 1.3 is for Visual Basic 5.0 and 4.0 32-bit only. If you need MsgBoxWiz for VB 4.0 16-bit, please use version 1.2

(2) Upgraded code to new constants used by Visual Basic 5.0

- (a) vbMsgBoxRight
- (b) vbMsgBoxRTLReading
- (c) vbMsgBoxSetForeGround
- (d) vbDefaultButton4

(3) Added support for Help Files to be called from Message Boxes

(4) Updated install of MsgWiz.EXE to install settings into VB.INI and VBADDIN.INI by simply running MsgWiz.EXE once outside the VB environment after software installation.

- c) **Version 1.2** – 12/25/96 – Fixed a bug in the code generator for Message Boxes that failed to create a string properly. The error caused only the last line of the whole text to be displayed in a multi-line message box.

(e.g. `Msg = "This is a test"`
`Msg = Msg & "This is a test also"`

‘ Error occurred here on the code

`Msg = "This is a test"`

‘ This resulted in only the third line being displayed in a

‘ message box once code was copied to your program

‘ Corrected error in generated code.

`Msg = Msg & "This is a test")`

- d) **Version 1.1** – 12/1/1996 – Added the ability to set users own variable names vice Message Box Wizzards created names.

- e) **Initial release** – 11/28/96

2) Registering the software: Send **\$25.00** U.S. funds in Check or Money Order to:

Leo Mrozek
317 W. Treehouse Lane
Round Lake, IL 60073

Registering will entitle you to free upgrades, a copy of the source code and the key to turn off the please register nag screen.

3) Installation of Program: (This now only applies to Version 1.3 and below) See Version 1.4 changes for Installation instructions

a) **Windows95** – Run the Setup program which will perform the following:

- (1) Installation of all required DLL's (version dependent) and the Message Box Wizzard Executable.
- (2) Update your VB.INI file for using the Message Box Wizzard with both 16 and 32 bit versions of Visual Basic 4.0
- (3) Use the Visual Basic Add-In Manager to activate the Message Box Wizzard.
- (4) If upon startup you receive error –31037 (File not found) switch to the windows system directory (Usually C:\WINDOWS\SYSTEM) and manually run MSGWIZ.EXE. Nothing will appear to happen, however this is registering the add-in which did not register correctly in the setup.

b) **Windows NT** – (Note: This has only been tested with NT 4.0 Workstation) Run the Setup program which will perform the following:

- (1) Installation of all required DLL's (version dependent) and the Message Box Wizzard Executable.
- (2) You must however manually edit the VB.INI file in NT 4.0. Add the following lines to the sections noted. (NOTE: One or both of these may have been entered by the setup program. This is an issue we are currently researching as to why it does not perform correctly every time)

```
[Add-Ins32]
MsgBoxWiz.clsMsgBoxWiz=0
```

```
[Add-Ins16]
MsgBoxWiz.clsMsgBoxWiz=0
```

Note: Setting both the above to 1 vice 0 will result in the Wizard being available to VB without having to perform step 3 below.

- (3) Use the Visual Basic Add-In Manager to activate the Message Box Wizzard.
 - (4) If upon startup you receive error –31037 (File not found) switch to the windows system directory (Usually C:\WINNT\SYSTEM32) and manually run MSGWIZ.EXE. Nothing will appear to happen, however this is registering the add-in which did not register correctly in the setup.
- c) Windows 3.1/3.11 – Currently the program does not support 16 bit Windows 3.1/3.11 but will work in Win95/WinNT 16 bit version properly. Due to Microsoft's rumored non-support of 16-bit programs in VB 5.0 the 16-bit version is not being considered for development.

4) Using the Program:

a) To create a message box:

- (1) Select Message Box from the Type Message Box Option group.
- (2) Enter a Title for the Message Box in the Title Block
- (3) Enter the message for the Message Box in the Message Block.

- (4) Select an Icon Style by clicking on an Icon. The currently selected Icon will be displayed on the right hand side of the Icon group.
- (5) Select a button style
- (6) Select Modal Style from the Modal Style option group.
- (7) Select which button you want for the default button from the Default Button options group. (Available buttons for default are based on selection in v) above.
- (8) Select whether you want explicit variable declaration with the code. This is handy to be turned off if you publicly declare the variables in the current form/module and then reuse the variables for all message boxes.
- (9) Select Preview Message to view the Message Box. As a helping hand, since the text you enter the message into is using a TrueType font, any spacing you use in multi-line messages will be carried into the message box and give the same lineup of the message in the message box as displayed in the text entry box.
- (10) Select View Code to view the generated code for the message box. You may cut and paste required parts to the Clipboard from the displayed code, or select Copy Code to copy the entire code window to the Clipboard and close the code window at the same time. Click on the Close button to close the code preview window.
- (11) Click on Close to close the Message Box Wizzard Add-in.

b) To create an Input Box:

- (1) Select Input Box from the Type Message Box Option group.
- (2) Enter a Title for the Input Box in the Title Block
- (3) Enter the message for the Input Box in the Message Block.
- (4) Enter any default value you want to appear in the Inputbox text line in the Default Value text box.
- (5) If you require an initial x-position and y-position, enter them in the Positioning text boxes. Remember this measurement is based on twips (1440 twips per inch). The VB4 Help file states that entering one of the x or y positions will result in a default being entered for the other one. This however does not work as described. Instead, if only one is entered the other is ignored by VB4. The code has been setup to take advantage of what the help file describes in the event this is fixed in future versions of VB.
- (6) Select whether you want explicit variable declaration with the code. This is handy to be turned off if you publicly declare the variables in the current form and then reuse the variables for all message boxes.
- (7) Select Preview Message to view the Input Box. As a helping hand, since the text you enter the message into is using a TrueType font, any spacing you use in multi-line messages will be carried into the message box and give the same lineup of the message in the message box as displayed in the text entry box.
- (8) Select View Code to view the generated code for the message box. You may cut and paste required parts to the Clipboard from the displayed code, or select Copy Code to copy the entire code window to the Clipboard and close the code window at the same time. Click on the Close button to close the code preview window.
- (9) Click on Close to close the Message Box Wizzard Add-in.

5) User Options

a) Users can set their own variable names to be generated in the code by Message Box Wizzard.

- (1) Select File/Variable Names from the Main Menu. You will be given a combo box listing of all user definable variables and the current variable's name. Simply enter the new variable's name and click on Save. The next time you view code, this variable's name will be inserted into the code. By using this feature, you will not have to go through the code generated by Message Box Wizzard and replace variable names with the variable names you are currently using in code.