LDrive MsgBar

LDrive MsgBar is an ActiveX control for displaying multiline status messages in a Microsoft Outlook-style bar.

Properties

Methods

Events

MsgBar Properties

- AutoSize
- CustomIcon
- IconType
- LineCount
- LineItems
- PostType
- WordWrap

MsgBar Methods

About
Clear
ClearLine
Post

MsgBar Events

Click
DblClick

AutoSize Property

Returns or sets a value that determines whether the height of the control is automatically resized to display its contents.

Syntax

object.**AutoSize** [= *integer*]

The **AutoSize** property syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
integer	Determines the sizing method to be used, as described in Settings, below.

Settings

The settings for *integer* are:

Constant	Value	Description
msbasNone	0	Keeps the size of the control constant. Contents are clipped when they exceed the area of the control.
mbasTight	1	The height of the control is automatically resized to display the contents.
mbasRelaxed	2	(Default) Same as **mbasTight**, but a small area is added to the bottom of the control.
mbasOutlook	3	Same as **mbasRelaxed** when the control's **LineCount** property is 1, **mbasTight** otherwise.

Remarks

If the **WordWrap** property is **True**, then the control will resize to display its entire contrents. If **WordWrap** is **False**, then some contents may be clipped if the control is not wide enough.

CustomIcon Property

Returns or sets the graphic to be displayed when the **IconType** property is set to **mbiCustom**.

Syntax

Set *object*.**CustomIcon** [= *picture*]

The **CustomIcon** property syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
picture	A string expression specifying a file containing a graphic, as described in Settings.

Settings

The settings for *picture* are:

Setting	Description
(none)	(Default) No picture.
(Bitmap, icon, metafile, GIF, JPEG)	Specifies a graphic. You can load the graphic from the Properties window at design time. At run time, you can also set this property using the **LoadPicture** function on a bitmap, icon, or metafile.

Remarks

The standard icons are 16x16 graphics.

IconType Property

Returns or sets a value indicating which icon to display on the control.

Syntax

object.**IconType** [= *integer*]

The **IconType** property syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
integer	Determines the icon to display, as described in Settings, below.

Settings

The settings for *integer* are:

Constant	Value	Description
mbiNone	0	No icon.
mbiCritical	16	Display **Critical Message** icon.
mbiQuestion	32	Display **Warning Query** icon.
mbiExclamation	48	Display **Warning Message** icon.
mbiInformation	64	(Default) Display **Information Message** icon.
mbiCustom	99	Display icon from **CustomIcon** property.

Remarks

The **mbiCritical**, **mbiQuestion**, **mbiExclamation**, and **mbiInformation** constant values match the **vbCritical**, **vbQuestion**, **vbExclamation**, and **vbInformation** constants of the **MsgBox** function.

LineCount Property

Returns or sets a value indicating the number of message lines the control can display.

Syntax

object.**LineCount** [= *integer*]

The **LineCount** property syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
integer	An Integer expression specifying the number of message lines.

Remarks

The maximum number of lines is 256.

LineItems Property

Returns or sets the message line items contained in the control. The list is a string array in which each element is a line item.

Syntax

object.**LineItems***(index)* [= *string*]

The **CancelCaption** property syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
index	The number of a specific item in the list.
string	A string expression specifying the list item.

Remarks

Valid *index* values range from 1 to **LineCount**.   Note that **LineItems** is an array and not a collection, so the **For Each** syntax is not supported.

PostType Property

Determines how the control will add new lines to the control when the **Post** method is used.

Syntax

object.**PostType** [= *integer*]

The **PostType** property syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
integer	Determines the style to be used, as described in Settings.

Settings

The settings for *integer* are:

Constant	Value	Description
mbpPostScrollUp	0	(Default) Lines are added from the top down. If all of the lines have been posted to, they are scrolled up.
mbpPostScrollDown	1	Lines are added from the bottom up. If all of the lines have been posted to, they are scrolled down.

Remarks

Setting the text of a line via the **LineItems** property is the same as posting to a line.

To clear lines for (re)posting, use the **Clear** and **ClearLine** methods.

WordWrap Property

Returns or sets a value indicating whether the control will wrap the text specified in its **LineItems** property.

Syntax

object.**WordWrap** [= *boolean*]

The **WordWrap** property syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
boolean	A Boolean expression specifying whether the text will wrap, as described in Settings.

Settings

The settings for *boolean* are:

Setting	Description
True	(Default) The text wraps to fit the width of the control.
False	The text doesn’t wrap.

Remarks

none.

About Method

Displays the About box for the control.

Syntax

object.**About**

The *object* placeholder an object expression that evaluates to a **MsgBar** object.

Remarks

This is the same as clicking About in the Properties window.

Clear Method

Removes all text from the **MsgBar** control.

Syntax

object.**Clear**

The *object* placeholder an object expression that evaluates to a **MsgBar** object.

Remarks

none.

ClearLine Method

Clears the text from a single line of the control.

Syntax

object.**ClearLine***(index)*

The **ClearLine** method syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
index	The number of a specific item in the list.

Remarks

Valid *index* values range from 1 to **LineCount**, and corresponded to the valid *index* values for the **LineItems** property.

Post Method

Adds a line of text to the control in the next available position, scrolling existing lines if necessary.

Syntax

object.**Post** [*string*]

The **Post** method syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
string	Optional. A string expression to display.

Remarks

Normally, message lines are scrolled off the control when the **Post** method is used. If the **Clear** or **ClearLine** methods are used, however, the control will consider the cleared lines “available” and will put *string* in the first available position.

Click Event

Occurs when the user presses and then releases a mouse button over the object.

Syntax

Private Sub *object*_**Click(**[*index* **As Integer**,] *objectnumber* **As Integer)**

The **Click** event syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
index	An integer that identifies a control if it is in a control array.
objectnumber	An integer that contains a number corresponding to the part of the control that was clicked, as described in Settings, below.

Settings

The settings for *objectnumber* are:

Value	Description
-1	The underlying control was clicked.
0	The icon was clicked.
1..**LineCount**	The message line corresponding to **LineItems(***objectnumber***)** was clicked.

Remarks

none.

DblClick Event

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Syntax

Private Sub *object*_**DblClick(**[*index* **As Integer**,] *objectnumber* **As Integer)**

The **DblClick** event syntax has these parts:

Part	Description
object	An object expression that evaluates to a **MsgBar** object.
index	An integer that identifies a control if it is in a control array.
objectnumber	An integer that contains a number corresponding to the part of the control that was double-clicked, as described in Settings, below.

Settings

The settings for *objectnumber* are:

Value	Description
-1	The underlying control was double-clicked.
0	The icon was double-clicked.
1..**LineCount**	The message line corresponding to **LineItems(***objectnumber***)** was double-clicked.

Remarks

none.

Constants

AutoSize property

Constant	Value	Description
msbasNone	0	Keeps the size of the control constant. Contents are clipped when they exceed the area of the control.
mbasTight	1	The height of the control is automatically resized to display the contents.
mbasRelaxed	2	(Default) Same as **mbasTight**, but a small area is added to the bottom of the control.
mbasOutlook	3	Same as **mbasRelaxed** when the control's **LineCount** property is 1, **mbasTight** otherwise.

IconType property

Constant	Value	Description
mbiNone	0	No icon.
mbiCritical	16	Display **Critical Message** icon.
mbiQuestion	32	Display **Warning Query** icon.
mbiExclamation	48	Display **Warning Message** icon.
mbiInformation	64	(Default) Display **Information Message** icon.
mbiCustom	99	Display icon from **CustomIcon** property.

PostType property

Constant	Value	Description
mbpPostScrollUp	0	(Default) Lines are added from the top down. If all of the lines have been posted to, they are scrolled up.
mbpPostScrollDown	1	Lines are added from the bottom up. If all of the lines have been posted to, they are scrolled down.

Runtime Distribution Requirements

This release of LDrive MsgBar was compiled using Microsoft Visual Basic 5 Service Pack 3. No additional files other than the VB 5 runtime files are needed. A .DEP file was installed by Setup.

Differences Between the VB5 and VB6 Versions

LDrive MsgBar is available in both VB5 and VB6 runtime versions. The control license allows either, or both, versions to be shipped with your products. This page explains the differences between the two.

Note: Either version of the controls can be used with *any* OLE-based hosting environment. The term “Version” refers only to the runtime requirements of the control. VB5-based versions require MSVBVM50.DLL and OLE 2.2+, while VB6-based versions require MSVBVM60.DLL and OLE 2.3+. Information on runtime file versions can be found in the .DEP file shipped with the control.

Distinct GUIDs and ProgIDs

Each version uses its own GUID. VB6 versions of the controls have a “6” appended to the VB5 ProgID. For example, the VB5 version ProgID for **MsgBox** is “MsgBoxLib”, whereas the VB6 version ProgID is “MsgBoxLib6”. In either version, the control name (in the example, “MsgBox”) remains the same.

Using distinct ProgIDs and GUIDs allows the developer to use either or both controls in a project.

Unique File Names

VB6 versions append a “6” to the control’s file name. For example, the VB6 version of the **MsgBox** control is “MsgBox6.ocx”.

Common File Versions

File major, minor, and build version numbers (e.g. 1.10.0056) are shared between all versioned builds of the control. This reduces confusion when updating libraries.

Distinct Library Descriptions

VB6 versions of controls are identified in the **Components** and **References** dialogs with a “(VB6)” designator. Version resource information also identifies a VB6 version control.

Separate Distribution Packages

Each package contains version-appropriate help and samples (if available) created with the control included in the package. VB6 shortcuts created by Setup are identified with a “(VB6)” designator.

Known Issues

none.

Registration and Support

For complete information on LDrive software products, including updates, pricing, payment options, and availability visit the LDrive web site at:

http://www.ldrive.com

You can also send check or money order along with your name, company, address and e-mail address to:

LogicDriven Incorporated
61 Alafaya Woods Blvd.
Suite 273
Oviedo, FL   32765

Don’t forget to indicate which LDrive product you are purchasing.

Contact us via e-mail at: **info@ldrive.com**

Bug reports, feedback, and suggestions are welcome from anyone, however, support is provided only to registered users.   Send your comments, questions, observations, gripes and complaints to:

feedback@ldrive.com

LogicDriven Software License Agreement

This License Agreement is a legal agreement between you (either an individual or a single entity) and LogicDriven, Incorporated (“LogicDriven”). for the software product identified above, which includes computer software and "online" or electronic documentation and may include associated media and printed materials ("the Software"). By installing, copying, or otherwise using the Software, you agree to be bound by the terms of this License Agreement.

LogicDriven Software License

LogicDriven software and the separately copyrighted components included therewith, and any enhancements or modifications thereto, whether delivered electronically or otherwise ("the Software") and the User Documentation accompanying the Software ("the User Documentation") are licensed to you for your personal use only.

Copyright laws and international copyright treaties, as well as other intellectual property laws and treaties protect the Software. The Software is licensed, not sold.

This Agreement grants you the following limited, non-exclusive rights:

You may install and use the Software on any number of computers, and may make any number of copies of the Software, provided that only one copy of the software is in use by you at any one time.

You may modify the sample source code included with the Software to design, develop, and test your Application. You may also reproduce and distribute your derivative works of the Sample Code in source and object code forms, provided that you comply with the Distribution Requirements described below. For purposes of this section, "modifications" shall mean enhancements to the functionality of the Sample Code.

Portions of the Software are designated as "Redistributable". Your rights to distribute the Redistributable Code are subject to the following requirements:

You may copy and redistribute the Sample Code and Redistributable Code (collectively "Redistributable Components") as described above provided that: (a) you distribute the Redistributable Components only in conjunction with and as a part of your Application; (b) you agree to indemnify, hold harmless, and defend LogicDriven from and against any claims or lawsuits, including attorney's fees, that arise or result from the use or distribution of your Application; (c) you grant third parties the rights to redistribute the Redistributable Components, which they acquire from your Application, only pursuant to terms no less restrictive as those contained in this Agreement.

LogicDriven retains all rights not expressly granted.

Limited Warranty

LogicDriven warrants that the Software will conform substantially to published specifications, documentation, and authorized advertising. If the Software and User Documentation was distributed in a non-electronic form, that media shall be free from defects in materials and workmanship for a period of ninety (90) days from the date of delivery to you.

Except for the limited warranty set forth above, the software and the user documentation are provided "as is".

LogicDriven makes no other warranty, either express or implied, with respect to the Software and/or the User Documentation and specifically disclaims the implied warranties of merchantability and fitness for a particular purpose. LogicDriven does not warrant that the Software and/or the User Documentation will meet your requirements or expectations or that the operation of the software will be uninterrupted and/or error free.   You are solely responsible for the selection of the software to achieve your intended results and for the results actually obtained.

Some states do not allow the exclusion or limitation of implied warranties, so the above exclusions and limitations may not apply to you.   This warranty gives you specific rights, and you may also have other rights that vary from state to state.

Limitation of Remedies and Liability

In the event the Software and/or the User Documentation fail to meet the Limited Warranty, LogicDriven's entire liability and your exclusive remedies shall be:

A) The replacement of the Software and/or the User Documentation not meeting LogicDriven's Limited Warranty, which is returned to LogicDriven with your proof of payment.

B) If LogicDriven cannot or will not replace the Software and/or User Documentation, you may terminate this Agreement by returning the Software and the User Documentation to LogicDriven with your proof of payment, and your license fee will be refunded.

Under no circumstances, and notwithstanding any failure of the essential purpose of any limited remedy provided for herein, shall LogicDriven be liable to you for any damages, claims or losses whatsoever, including but not limited to any claims for lost profits, lost savings or other special incidental or consequential damages arising out of the use or inability to use the Software and/or the User Documentation regardless of the circumstances.

General

In the event it is determined that any provision contained in this license is unlawful, void, or unenforceable, such determination shall solely affect such unlawful, void, or unenforceable provision and shall not affect the validity or enforceability of the remaining provisions of this license.   The laws of the State of Florida will govern this agreement.

standard property
Information on this property can be found in the standard Visual Basic documentation. No other information is available (or needed).