

ADBPubs1 Help Index

ADBPubs1 is a demo application using the the PUBS database in SQL Server to demonstrate how to use dbLibrary through the VBSQL package. It is designed by Sten Sundblad and built by Per Sundblad of ADB-Arkitektur AB in Uppsala Sweden for the first release of SQL Server on Windows NT.

Overview

Detailed instructions:

The tool bar

The menu bar

QBE and queries

Refresh

Entering a new author

Modifying an existing author

Deleting an author

Showing SQL statements

NULL DBPROCESS

Overview

Simple

This is a very simple application. It can not do very much. However, you can use it to maintain information on authors in the pubs database, and it allows you to select sets of authors, answering to search conditions specified by the user giving examples in the form itself. From this set of authors you can select one and have all data about him or her shown in the form. Data on this author can be modified or deleted.

Not properly generalised

The application has not been generalised the way we usually do. Instead we have tried to make it as easy as possible to follow the logical flow of the program. For this reason, some code that could be placed in .BAS files for reuse are put in the form.

Remarks

There are too many remarks in the application. This is to help make everything as clear as possible for anybody studying this demo app. Hope it helps.

Concurrency Control

The application uses BROWSE MODE for Optimistic Concurrency Control. This means that your Authors table must have a unique index and a timestamp column. When you install the Pubs database there is a unique index but no timestamp column.

The sample application checks for the timestamp column at FORM_LOAD. If it is missing you will be asked if the application should add it.

If there is no unique index the application will refuse to do Update and Delete operations.

Alternative approach to Concurrency Control

An alternative approach for concurrency control is to use database cursors. We have decided not to use them in this sample. They are, however, very useful and powerful but maybe a little more complicated than Browse Mode.

The tool bar

The MDI form as well as every other main form (only Authors.FRM in this app) contains a vertical tool bar, placed just below the menu. The toolbar contains 9 push buttons. Starting from the left they are:

The NEW button.

The SAVE button

The DELETE button

The CLEAR button

The REFRESH button

The QUERY button

The CANCEL button

The LOGIN button

The LOGOUT button

The menu bar

The menu bar contains the following menus:

The File menu. Allows you to close the active form or exit the application.

The Edit menu. Contains the following options:

New

Save

Delete

Clear Form

Refresh

Query

Cancel

Copy To New

Release

The Forms menu. Allows you to open one or several instances of the Authors form.

The Window menu. Allow you to Arrange, Cascade or Tile the windows and icons as well as reset the active window to its normal size. Also contains a list of all open MDI child windows.

The Options menu. Toggles between the state where SQL statements are shown before they are sent to SQL Server and the state where they are not.

The Help menu. Gives access to an About box and this Help system.

QBE and queries

You may control queries that are sent to SQL Server by giving examples of what you want, then either push the Query button in the toolbar or select Query. in the edit menu Before you enter your example you need to clear the form, using the Clear button or selecting Clear Form. in the Edit menu (As an alternative you may select Release in theEdit menu.

Example:

If you wish to select all California authors you should do as follows:

1. Clear the form.
2. Enter CA in the States field.
3. Push the Query button or select Query in the Edit menu.

As a result you should get all California authors in the primary list box. The first one should be automatically selected and shown in the fields of the form. Select any name in the list box, and his or her details will be shown in the form.

You may combine examples from different fields. If you enter *Ring%* in the Last Name field and UT in the state field, you should get all authors whose last name starts with Ring and who live in California.

Release

Another way to give an example is to bring up some record on the screen, where much of the data looks like those you are interested in. This record will be connected to a row in the authors table in the database, but that connection could be released. Select Release in the Edit menu and then modify the record to form your example, then send the query to SQL Server.

Show SQL statements:

Check Show SQL in the Options menu to see the SQL statements that are generated and sent to the server.

Refresh

When you work in a multi user environment (like SQL Server) you never know for sure what is the actual state of the database. You know that the set you have in the list box was the set answering to your query when you sent it to the database. In the time between then and now somebody may have modified some of those records, deleted some and inserted some new ones.

The Refresh function remembers the SELECT statement that was last sent to the server and sends it again, refreshing your data.

Press on the Refresh button in the tool bar (the shower) or select Refresh from the Edit menu.

Entering a new author

Before you can enter information about a new author you need to clear the form as well as put it in the New Mode.

The new button in the tool bar

You can do this by pressing the new button (the first one from the left) in the tool bar.

The new option in the Edit menu

An alternative way is to select the New option in the Edit menu. This creates exactly the same effect as pressing the New button in the tool bar. In fact, when you select New in the menu the push button is automatically pressed by the application, guaranteeing the same result from both events.

When you forget...Copy to New

Sometimes you will forget to press the New button or select the New option before you start to enter information, especially when the form is just loaded and empty. In this case the application will refuse to save the data, since it has never been put in the New Mode. Stay cool! All you have to do is to select Copy to New in the Edit menu and then save.

Saving

Save the information by pressing the Save button in the tool bar or select Save in the Edit menu. Since the form is in New Mode it will know that the required save operation is an insert.

Modifying an existing author

Any author shown in the fields of the form may have its data modified, since the form will be in its SHOW MODE any time an author is shown in the form.

All you have to do is change the appropriate fields, then save the changes.

Saving

Save the information by pressing the Save button in the tool bar or select Save in the Edit menu. Since the form is in SHOW MODE it will know that the required save operation is an update.

Note: If you or anybody else has dropped the unique index on the Authors table the save will be refused.

Deleting an author

Any author shown in the fields of the form may be deleted, just as long as the author is shown in the form.

All you have to do is to select the Delete button in the tool bar (third from the left: a waste basket) or select Delete in the Edit menu. The effect is exactly the same. You will be prompted to affirm your wish to delete. If you do, a DELETE statement will be sent to SQL Server. If *Show SQL* is checked in the Options menu, the DELETE statement will appear in a message box before it is sent to the server.

Note: If you or anybody else has dropped the unique index on the Authors table the delete operation will be refused.

Saving information

Each form is sensitive to:

1. If there is anything to save.
2. If what there is to save should be inserted or updated.

The latter depends on the actual MODE of the form.

Save by pressing the Save button in the tool bar or select Save in the Edit menu. Since the form knows its Mode it will know if the required save operation is an insert or an update

If there is nothing to save your computer will just beep when you try..

Note: If you or anybody else has dropped the unique index on the Authors table, and if the save operation is an update (not an insert), the save will be refused.

Cancelling modifications

You may cancel any ongoing modifications on a record by pressing the Cancel button in the tool bar (the third from the right - the one with a large X on it). Any changes will be undone and the form will refresh the record you started to modify.

Modes

The form may be in different modes, depending on the operations that are going on. Possibles modes are:

| | |
|----------|---|
| SQLNEW: | You have at least prepared to enter a new record. |
| SQLSHOW: | A record from the database is shown in the form. |
| SQLREST: | When the form is not in SQLNEW or SQLSHOW modes, it is resting. This means that saving will not be enabled. Instead you are allowed to enter an example for <u>QBE</u> . |

Logging in and Logging out

When you start the application you will be prompted for Login information. You may log out at any time by pressing the Logout button (the rightmost button in the tool bar). When you do any open form will close. At any time, you may log in again by pressing the Login button, placed close to the Logout button.

NULL DBPROCESS

When you run the application from design mode rather than in a compiled form you may encounter the message *10001 NULL DBPROCESS encountered* when you log in. This is no big deal. You can go on running the program anyway. The reason for the message is as follows:

In order to set up you environment for VBSQL you need to execute the command SQLINIT. The countercommand SQLWINEXIT releases any traces of VBSQL. The call to SQLINIT is placed in the FORM_LOAD procedure of the MDI form, the SQLWINEXIT call in FORM_UNLOAD of the same form.

When running from design mode of VB, sometimes you break the program without passing FORM_UNLOAD of the MDI Form. In those cases your environment is not released from VBSQL. For that reason your next call to SQLINIT will fail, and you will get the message described above. The reason SQLINIT fails is that there is no need for it, your last call is still valid. Therefore: most of the time you may ignore this message. However, if you get this message the first time you try to log in there is probably something wrong with your installation.

When running a compiled version of the program you should normally not get this message.

