WINDOX Version 1.0

This program allows you to create and modify Paradox(TM) tables under Windows 3.0(TM) and then generates C or C++ code for you to use in your programs in order to manipulate those tables.

It is assumed that you are familiar with Paradox tables, data types, etc., and with C programming and that you have the Paradox Engine and its related tools which are necessary for using the code generated by this program.

Although this program is a complete working version and has not been crippled or limited for demo purposes, this does not mean that it is free. If you find it useful and continue using it, you are required to send a fee of $25.00 to the author. This will authorize you to continue using the program and you will receive a registered version that does not have the registration reminder. You will also be informed of any future updates and enhancements. You are, of course, encouraged to make as many copies of this program as you want and pass them to your friends and colleagues.

I am contemplating a Pascal version of this utility, or rather, the addition of Pascal to the existing one. If that is of interest to you, please let me know.

Please send your registration fee and all your comments to:

Computer Solutions
1501 Broadway, Suite 2002
New York, NY 10036
Att: Kamran Bayegan

I can also be reached on Compuserve: 73627,1573

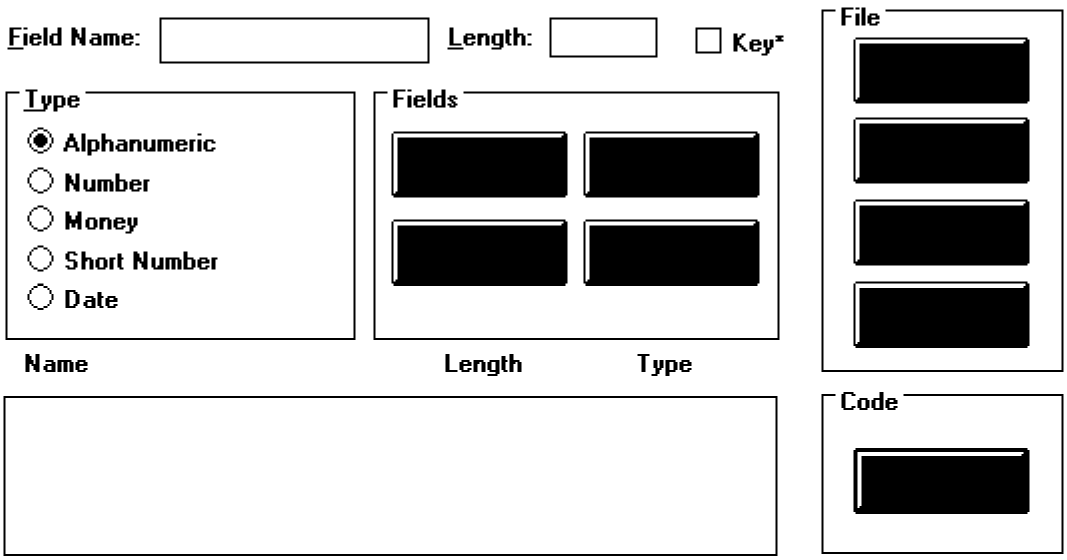
All your comments and ideas for the improvement of this program are highly appreciated and welcomed.

Contents

WINDOX.EXE	The executable module.
PXENGWIN.DLL	Paradox DLL necessary to run WINDOX.EXE.
WINDOX.WRI	Documentation (Windows Write format)
TEST.DB	A sample database for testing
READ.ME	

Description

The program is so simple and straightforward that it really does not require any description. Remember that for the program to run properly, PXENGWIN.DLL must be in your Windows directory. The main screen looks like this:

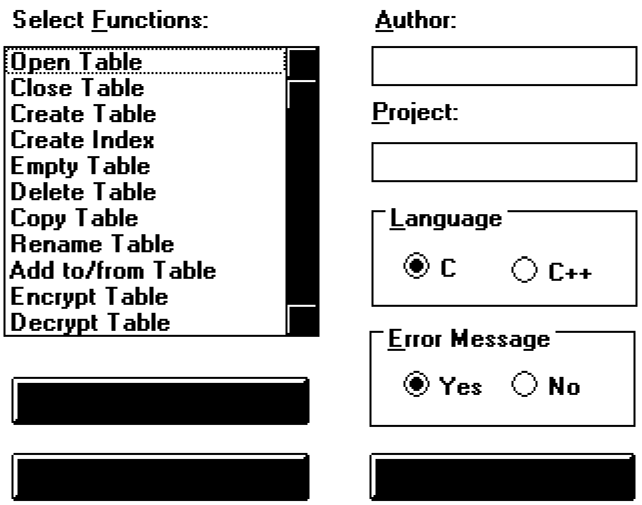


On this screen, you can start creating a **NEW** (items in bold are buttons) Paradox table by entering the properties of the fields and using the **ADD** button to add them to the file structure. Each new field that you add will be placed in the lower left box. Highlighting each field in that box causes its properties to be displayed in the upper section. You can change these properties and **MODIFY** the field. You can alaso **INSERT** fields among existing ones or **DELETE** existing fields. I am sure that in a couple of minutes you will get the feel of this screen. The **KEY** field only marks the fields that you want to include in the primary index. This only generates the necessary code for the creation of the primary index and does not actually generate the index. You should also be careful that although Paradox allows you to have field names of more than one word, i.e., "First Name", C does not allow such a variable. So please resist that temptation and use one-word field names.

The files that you create from scratch, or modify can be **SAVE**d and you can use **SAVE AS...** to give them a new name.

You can also **OPEN** existing files. The structure of the file will appear in the lower left box and properties of each highlighted field is displayed in the upper section. You can modify this structure and save the file again.

Once you are satisfied with the structure of the file, you press the **SETUP...** button to prepare for code generation. The code generation screen looks like this:



Select Functions:

- Open Table
- Close Table
- Create Table
- Create Index
- Empty Table
- Delete Table
- Copy Table
- Rename Table
- Add to/from Table
- Encrypt Table
- Decrypt Table

Author:

Project:

Language ● C ○ C++

Error Message ● Yes ○ No

On this screen, you can enter your name, the name of your project and choose whether you want C code or C++ code. Marking "Yes" in "Error Messages" generates a macro that, upon an error, will display a box on your screen with the appropriate error message. This is very good for the development phase of your project, but you may decide to handle your errors in a different way, in which case, you will mark "No". Then the functions will just return the error code to your application.

The upper left box contains the functions that you can generate. You can choose one, a few or all of them, via **SELECT ALL**, or clear the selections with **UNSELECT ALL**. Pressing **GO** genrates your code, which consists of a header file and a code file. The names of these files are made up of the first four letters of your table name. This is also the prefix that is attached to all function names. The program also genrates two default functions, ????Get() and ????Put(), which are used for receiving data from the buffer and putting data into the buffer. The functions are all short and self explanatory and are named similar to the Paradox engine functions. Please take a few minutes to review them.

That's about it!!!

Please feel free to contact me with your suggestions, ideas and wish lists. Good luck and enjoy!!