



Contents

Introduction

[Read This First](#)
[What is VB Code Gen](#)
[Code Gen Features](#)
[Last Minute Changes](#)
[Copyright and Liabilities](#)
[The Future](#)
[Bugs](#)
[Technical Support](#)
[Register Code Gen](#)

The Main Screen



Button Bar



[Exit Code Gen](#)



[Copy to Database](#)



[Paste to VB](#)



[Color Dialogue](#)



[File Dialogue](#)



[Font Dialogue](#)



[Help Dialogue](#)



[Printer Dialogue](#)



[Message Box Dialogue](#)

[Menu Commands](#)

What is VB Code Gen V1.1

Code Gen© is a Visual Basic program that helps you use one of its most cumbersome controls - the Common dialogue. This control allows you to open and close files, select fonts, set printer settings, change colors and call the Windows help engine. Its settings are based on flags you set, and since most of us don't use it everyday, it's a hassle having to look the flags up and then test them to see if it worked. Code Gen will allow you to set the Flags with a click of the mouse and before generating the code in the clipboard, test it to see if it's what you wanted. Another added feature is that you can save your settings as a template and call them up at a later time. After you're satisfied with your settings, you can click the OK button and the code you **WOULD** have had to type in is put in the clipboard. All you have to do is switch to your application and paste where you need it. It's that simple! I've also created a Messagebox generator screen that works the same way as the common dialogue. There are two more features to Code Gen: the Access database reader/creator, and the 'snippets' code saver. The database tool works with Access databases and reads its structure and writes it out to a file. **Be aware, this will not work if you have referential integrity (RI) built into your database. RI creates multiple indexes for a single table that are flagged as Primary and the Create aborts.** After it's done, you add the "BUILD_DB.BAS" to your project and call the function Build_DB supplying an input file (the definition file) and an optional path (incase you want to put it in a different directory). I have another function "Read Database to Clipboard" that will read a database and create the code to recreate it in the clipboard. After it's done, you have a function that will create a database (tables, fields and indexes). The code snippet function is for the neat pieces of code that usually get scattered throughout your various projects. Instead of spending the rest of the day searching for it, you can paste it in the database and call it up when you need it. It's put in the clipboard and all you have to do is paste it where you want it to go. Code Gen was started because I needed to read a database that was too large. I created a little menu system for it and I kind of added on from there. Since this is version 1.0, I'm sure I'll have a couple of bugs to work out. If any are reported, I'll fix them as soon as I can. I don't have anyone to beta test it beside myself and it's been constantly evolving, so don't be too hard on me.

If you have any suggestions or comments, please contact me and I'll see if I can add them in the next version. I hope you enjoy this product and that it saves you time coding.

I would like to have this tied into VB, so if anyone has any ideas, I'm open. Of course I'll mention you for your efforts:)

If you need to contact me, I can be reached by:

Email: **Steven M. Perry at Compuserve:75251,3026**

US Mail: **Steve Perry**
13740 SE Grant Ct.
Portland, OR 97233

Exit Code Gen V1.1

Closes down the Code Gen database and exits the program.



Copy Code to Database

This option takes code that has already been [copied](#) to the Windows clipboard and allows you to put it in the Code Gen database. This could be variable declarations, a whole code module or a small piece of code. You have the opportunity of editing it before adding to the database. This is useful when you have special routines that aren't used a lot, but you need to keep track of them. Your options are to put a name you want to reference it by and also a description of what it does.



Paste Code to VB Application

This option allows you to [paste](#) a piece of code from the Code Gen database into your VB application. After selecting the snippet you want to paste, it's copied to the Windows clipboard. You then can switch back VB and the module you're editing and paste it.



Color Dialogue Code Generator

This screen will allow you to generate code for the Visual Basic [Common Dialogue Control](#) for selecting colors like you do when setting your Windows colors in the Control Panel. You have several defined areas They are: Flags, Common Dialogue Control Name, Receive Control Name or Variable, Templates, 'Quik' Help (not misspelled) and the command buttons.

[Flags](#)

[Common Dialogue Control Name](#)

[Receive Control Name or Variable](#)

[Templates](#)

[Quik Help](#)

[Command Buttons](#)



Command Dialogue Control Name

CMDialog1

Common Dialogue Control Name

The "Common Dialogue Control Name" is the name of the common dialogue control you've selected and put in your project or on your form. that you want code generated for. This can be the control name (i.e. CommonDlg1) or a more specific name like "Form1.ColorDlgCtl". Don't put an ending "." on it because I already do that.



Files Dialogue Code Generator

This screen will allow you to generate code for the [Visual Basic Common Dialogue Control](#) for opening and saving files. The dialogue that's generated is just like the one you get when selecting ALT-F, Run from the Windows Program Manager. You have several defined areas. They are: Action Properties, Flags, Dialogue Title, Default Extension, Common Dialogue Control Name, Receive Control Name or Variable, File Filters, Templates, 'Quik' Help (not misspelled) and the command buttons.

[Action Properties](#)

[Flags](#)

[Dialogue Title](#)

[Default Extension](#)

[Common Dialogue Control Name](#)

[Receive Control Name or Variable](#)

[File Filter - File Descriptions](#)

[File Filter - File Filter](#)

[File Filter - Command Buttons](#)

[Templates](#)

[Quik Help](#)

[Command Buttons](#)



Font Dialogue Code Generator

This screen will allow you to generate code for the Visual Basic Common Dialogue Control for changing Fonts. The dialogue that's generated is just like the one you get when using Word for Windows or some other word processor. You can select the font name, size, bold, italics, underlined, colors and some other settings. You have several defined areas on the screen. They are: Flags, Font Size, Common Dialogue Control Name, Receive Control Name or Variable, Templates, 'Quik' Help (not misspelled) and the command buttons.

[Flags](#)

[Font Size](#)

[Common Dialogue Control Name](#)

[Receive Control Name or Variable](#)

[Templates](#)

[Quik Help](#)

[Command Buttons](#)



Help Dialogue Code Generator

This screen will allow you to generate code for the Visual Basic Common Dialogue Control for calling the Windows Help Engine. The dialogue that's generated is just like the one you get when you press F1 from any Windows application. You can invoke it for a Context ID, Keyword, Help File Contents, Search and Help on Using Help. You have several defined areas on the screen. They are: Templates, Helpfile to use, Help Commands, Common Dialogue Control Name, 'Quik' Help (not misspelled) and the command buttons.

[Templates](#)

[Help Commands](#)

[Common Dialogue Control Name](#)

[Receive Control Name or Variable](#)

[Quik Help](#)

[Command Buttons](#)



Printer Dialogue Code Generator

This screen will allow you to generate code for the Visual Basic Common Dialogue Control for performing Printer functions. The dialogue that's generated is just like the one you get when you select the Printer Setup from most Windows word processors. You have several defined areas on the screen. They are: Flags, Common Dialogue Control Name, Templates, 'Quik' Help (not misspelled) and the command buttons.

[Templates](#)

[Common Dialogue Control Name](#)

[Quik Help](#)

[Command Buttons](#)



Message Box Code Generator

This screen will allow you to generate code for the Visual Basic "MessageBox" function. This is a nice screen because you get to test your selections before actually creating the code. You have several defined areas They are: Templates, Visible Buttons, Icons Displayed, Initial Button Focus, Build Logic, Modal, Code Generation Options, 'Quik' Help (not misspelled) and the command buttons.

[Templates](#)

[Visible Buttons](#)

[Icons Displayed](#)

[Initial Button Focus](#)

[Build Logic](#)

[Modal](#)

[Code Generation Options](#)

[Quik Help](#)

[Command Buttons](#)

Copying Text to the Windows Clipboard

The Windows Clipboard is a temporary buffer you can use to save and recall text and graphics. To copy to it, you need to first highlight the text by either clicking and holding the left mouse button down and dragging it to the end of the text you want to copy. Then, press the "Control" key and letter "C" at the same time or press the "Alt" key, the letter "E", and then the letter "C". Either way, whatever was highlighted will now be copied to the clipboard.

Paste Text to the Windows Clipboard

The Windows Clipboard is a temporary buffer you can use to save and recall text and graphics. To paste from it, you need to move to the position where you want the text inserted and, at the same time, press the "Shift" key and "Insert" key or the "Alt" key, the letter "E" and then "P". Either way, whatever was in the clipboard will now be pasted in your module.

Bugs

This is for any bugs that I'm aware of or that are forwarded to me. I'll start it by entering the first one (I'm not sure if it's a bug or VB limitation).

1. Don't use this unless you've run the compatibility layer so you can use Access 2.0 databases.
2. The database create will not work properly if you've defined referential integrity into the database. Access creates special indexes that I can't currently process (or drop) so your request to create will either abort with a 3283 - duplicate primary index, or 3284 - index already exists. I'm looking into this one.
3. See the readme file.

© Copyright and Liabilities

Code Gen V1.1 is copyrighted Software. Parts of the software are the copyright of Microsoft Corporation.

I will not be held responsible for damages, direct or consequential, from the use of this product. If you use this software, you are hereby bound by this disclaimer. Use it at your own risk.

If you find bugs, or want to make comments or suggestions, please feel free to contact me at:

Email: **Steven M. Perry at Compuserve:75251,3026**

US Mail: **Steve Perry**
13740 SE Grant Ct.
Portland, OR 97233

Copy Button

This button allows you to copy the text in the clipboard to the Code Gen database. You also have the option to edit it first.

Paste Button

This button allows you to paste data stored in the Code Gen database to the Windows clipboard. From there, you can go to your VB application and paste it where you want.

Color Button

This button allows you to create VB code for the common dialogue control for allowing the user to set Colors.

File Button

This button allows you to create VB code for the common dialogue control for allowing the user to open or save files.

Font Button

This button allows you to create VB code for the common dialogue control for allowing the user to set Fonts.

Help Button

This button allows you to create VB code for the common dialogue control for allowing the Windows Help Engine to be used. I left out a couple of flags that are available because I wasn't sure how they were used. If you need them available, let me know and I'll see what I can do.

Printer Button

This button allows you to create VB code for the common dialogue control for allowing the user to set Printer options.

Message Box Button

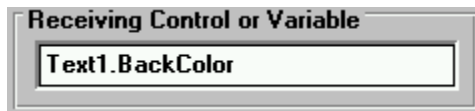
This button allows you to create VB code to generate Message Box dialogues. The nice thing is that you can test before you code.

Exit Button

Need I say more....

Flags

These are Hex values (i.e. &H8&) that are "or"ed together and passed to the common dialogue control to tell it what options you want available to the user, such as do you want them to have access to define their own colors or the basic ones or if you want a certain color set when opening it up. There is also a checkbox labeled "Generate Error If User Cancels", this is so you can trap this situation, instead of processing invalid selections. I provide code so that you can perform some certain action if this occurs. It's not an actual flag, but another property of the common dialogue, but I decided to put in the same frame anyway. I won't give you a detailed description of each flag setting available because that is provided by the "Quik" help on the screen.




Receive Control Name or Variable

The "Receiving Control Name or Variable" is if you want to set a control or variable to whatever the user selected.



Templates

The Template Frame is for times when you use the same basic settings all the time. Instead of having to select everything again, you type a name in the Template field name and click the "Save" button. To retrieve this template, you select it from the Template combo box and press the "Retrieve" button. If you don't need a certain template, select it in the combo box and click the "Delete" button. If you want to reuse a certain template, you can either delete it and then save it again or retrieve the old one, change what needs to be changed and save it again. You'll be prompted to confirm your changes. Another way you could save your settings, without using the template, would be to generate your code, by clicking the "OK" button, and then save it as a "snippet" by clicking the "Copy" button  [Copy Button](#).

Quik Help

The "Quik" Help frame is Screen specific help that's provided as a quick way of getting help without having to bring up the help file. In some instances, it provides more help than is included in the help file. Some people, mentioned that it was nice at first, but got annoying after awhile. I included a checkbox to turn it on or off. This is an application wide setting. If you turn it off for one screen, it's turned off for all screens. It's also remembered when you exit, so you don't have to keep resetting it.



Command Buttons

The Command buttons are pretty much self explanatory. The "Test" button allows you see if what you've selected is what you really want, the "OK" button is to actually generate the code. After clicking this button, you are brought back to the Main Screen. The "Cancel" button is if you decide you don't want to generate code for this option.

Technical Support

If you find bugs, or want to make comments or suggestions, please feel free to contact me at:

Email: **Steven M. Perry at Compuserve:75251,3026**

US Mail:**Steve Perry**
13740 SE Grant Ct.
Portland, OR 97233

Action Properties

The action option buttons appear in the Code Generator Screen for "Files". This sets the action property of the common dialogue control. Setting either one of these option buttons will cause some flags to be changed if that flag has no meaning to that action. For example, if I'm saving a file, I don't want the multiple selection flag to be set.

Dilalogue Title

This field allows you to set what is in the Common Dialogue title when it is opened up rather than accepting the default "Save File" or "Open File". This property was suppose to be available for all of the dialogue options, but I was only able to get it to work with the "Files" Option, so I didn't include it as an option with the other screens.

Default Extension

Use this property to specify a default file-name extension such as TXT or DOC. When a file with no extension is saved, the extension specified by this property is automatically appended to the file name. You can use this in conjunction with the "Different Extension" flag so that you can check if the file returned from the common dialogue is different. If both of these are set, you can check the flag on after the user has selected the file to see if it's different.

File Filters - File Description

This describes the files currently available for selection. For example, if the filter was set for *.MDB, the file description would be "All Access Databases".

File Filters - File Filter

This describes what files you want the user to choose. For example, if you wanted to show all "txt" files, you would put "*.txt" as the filter or all "Myfile" files, you would put "Myfile.*".

File Filter - Command Buttons

The file filter command buttons are for adding and removing file filters from the listbox. You need to have at least one filter set in the listbox for this to work properly. To use the "Add" button, fill in the File description (i.e. All Files) and the file filter (i.e. *.*) and press the "Add" button. If you decide you don't want a selection, click the mouse on the item in the listbox and click the "Remove" button

Font Size

This is in effect if you have selected the "Limit Font Size to Min/Max" flag. Otherwise these two fields (Max Font Size and Min Font Size) are ignored.

Help Commands

These commands are a property of the Common Dialogue Control (Help Command). and are set when either associating a helpfile with a common dialogue or when using the common dialogue to invoke the Windows Help Engine (Winhelp.exe). I've left 2 commands out because I wasn't sure how they are to be used. I'll put them back in when I get more information on them. They are: HELP_QUIT which notifies the Help application that the specified Help file is no longer in use and HELP_SETINDEX which sets the context specified by the HelpContext property as the current index for the Help file specified by the HelpFile property. This index remains current until the user accesses a different Help file (this value is only for Help files with more than one index).

The Future

I do plan to enhance this in the future. I know I can go a lot further with it. Somethings I plan to fix, besides any bugs, are to hook it into VB so You don't have to switch back and forth. I also want to get all flags for the common dialogue control working, figure out how to carry the referential integrity along with the database and rewrite this help file. Any other suggestions are welcome.

Register Code Gen

Mail this form to: Steve Perry
 13740 SE Grant Ct.
 Portland, OR 97233

MAKE CHECKS OR MONEY ORDERS OUT TO: Steve Perry

Where did you get this copy of Code Gen?

Ship to: _____

Phone: _____

Fax: _____

CompuServe: _____

Internet: _____

qty ordered _____

REGISTRATION
\$15 each, postpaid (Check in US funds)
Foreign addresses add \$3.00 shipping.
No additional shipping charge to Canada.

Code Gen Features

Some features that are part of Code Gen V1.1 are:

Database Features:

1. Read Access databases and create a definition file that can be read by a VB function I provide.
2. Read Access database and create the code necessary to recreate the database and copy it to the clipboard where you can paste it in your VB application as a sub where you can call it.
3. Read and recreate a database (Database Structure Only) on the fly. **ALL data will be lost!**
4. Generate report showing database structure: table names, field names and sizes, and index names and fields.

Code Library Features

1. Allow you save and recall pieces of code through the clipboard that can be pasted into your application.

Code Generation Features

1. Create code to recreate a database (above).
2. Create code for Message boxes.
3. Create code for the common dialogue control which allows you to control the Printer (set default), Fonts (change), Color (change), Files (open and save), and Help (call the help file).
4. Allow you to create [Templates](#) so you don't always have to go through and set your selections. It remembers for you!

BEST OF ALL - YOU CAN TEST BEFORE YOU CODE.

YOU DON'T NEED TO REMEMBER THE HEX VALUES FOR THE FLAGS.

Form Features

Double Click on the panel (area directly around buttons) or on the far right corner and it will either keep the window on top of all windows or allow other windows to be on top of it. This is kind of nice when going back and forth between Code Gen and VB.

Menu Commands

[Files Menu](#)

[Snippets Menu](#)

[Code Generator Menu](#)

[Help Menu](#)

About Screen

Information about the program and its author.

Help - Search

Allows you to search for a particular word, subject or category in the help file.

Files Menu

<u>F</u> ile
<u>R</u> ead Database to Clipboard
Read <u>D</u> atabase To Def File
Create Database <u>F</u> rom Def File De <u>l</u> ete and Redefine Database
<u>P</u> rint Def File
<u>E</u> xit

Snippets Menu

<u>S</u> nippets
<u>C</u> opy (From VB)
<u>P</u> aste (To VB)
<u>B</u> rowse
<u>D</u> elete (From Database)

Code Generators Menu

<u>C</u> ode Generators	
<u>C</u> ommon Dialogue...	<u>C</u> olor Dialogue
<u>M</u> essage Box	<u>F</u> ile Dialogue
	<u>F</u> ont Dialogue
	<u>H</u> elp
	<u>P</u> rinter

Help Menu

<u>H</u> elp
<u>C</u> ontents
<u>S</u> earch
<u>T</u> echnical Help
<u>A</u> bout

Browse

This option allows you to browse the CodeGen database for code snippets, copy a snippet from the database to the clipboard or delete a code snippet from the database.

Delete

This allows you to search the CodeGen database for a code snippet and then delete it from the database.

Visible Buttons

This allows you to select which buttons you want displayed in the message box.

Icons Displayed

Allows you to select which icons will be displayed in the message box.

Initial Button Focus

This allows you to set which button will have the initial focus when the message box is displayed. This option is not available if the "OK" button is selected because there's only button.

Build Logic

This allows you to have logic built for the user response. This will create a "Select Case" statement with the possible choices the user might have chose. This option is not available if the "OK" button is selected because there's only 1 choice.

Modal

This allows you to decide whether the application is held up until the user responds or if Windows is held up until the user responds.

Code Generation Options

You can choose single line sub - all on one line (ie. `Msgbox()`), single line function (ie. `UserResp% = Msgbox()`), or multiline and commented (declares variables and then uses them in the message box).



Common Dialogue Control

The Common Dialogue control is provided in Visual Basic to help you with five common Windows functions:

[Files \(Open and Save\)](#)

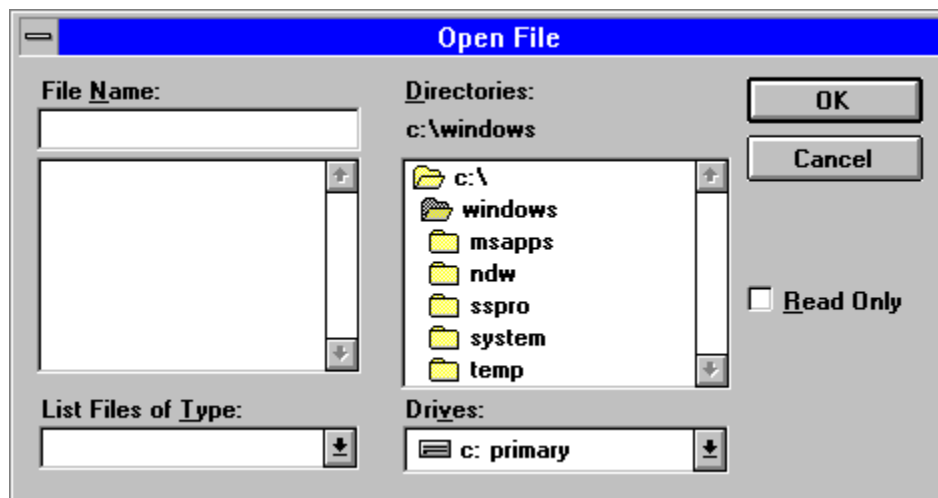
[Font Selection](#)

[Color Selection](#)

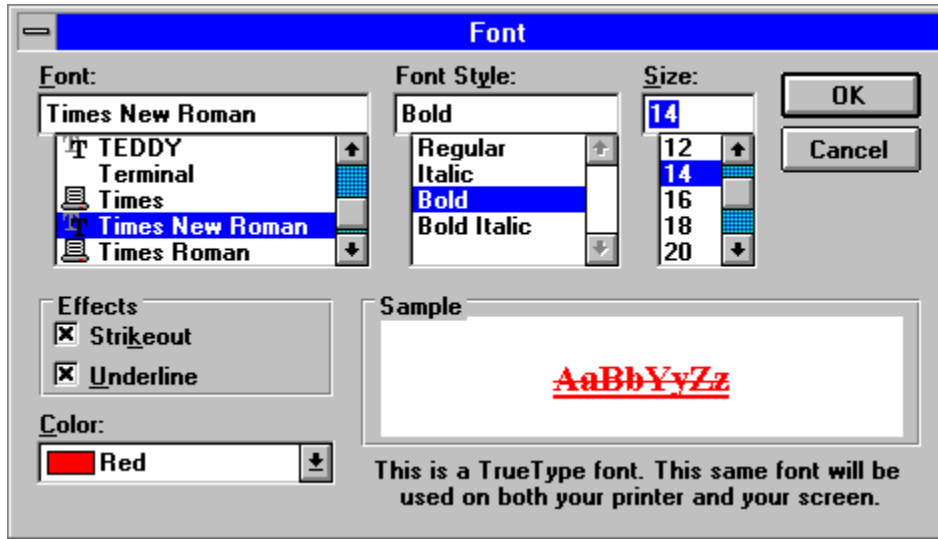
[Printer \(Printing and Setup\)](#)

[Starting the WinHelp Engine](#)

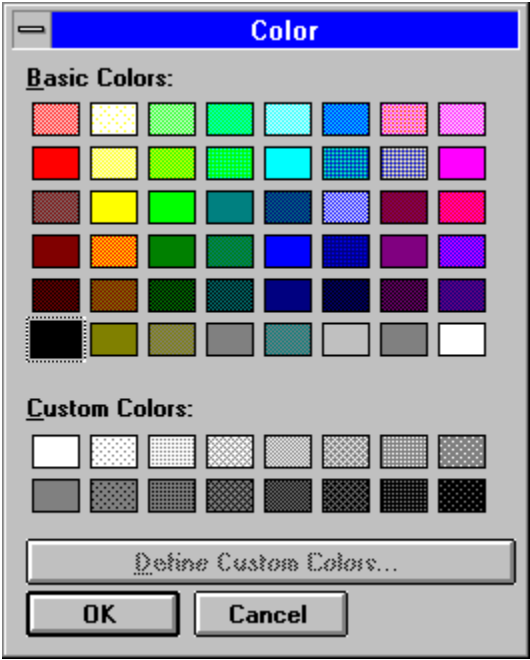
Files - Common Dialogue



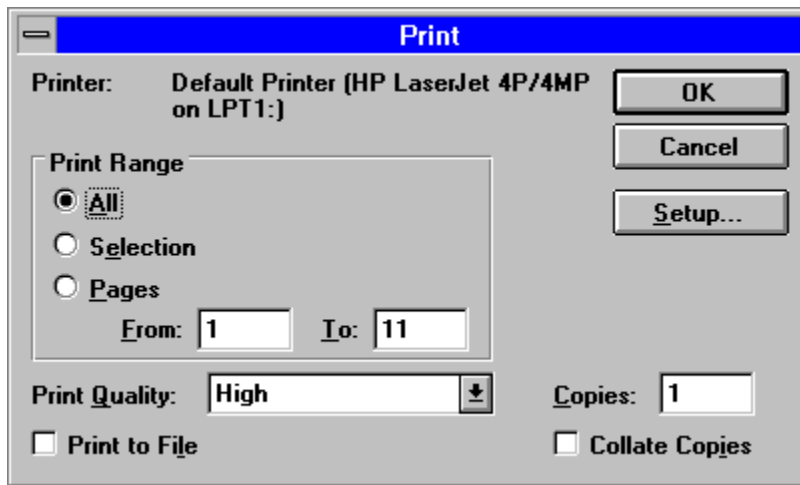
Font - Common Dialogue



Color - Common Dialogue



Printer - Common Dialogue



The image shows a standard Windows-style print dialog box. It has a title bar with a minus button and the word "Print". The main area is divided into sections for printer selection, print range, print quality, and other options. The printer is set to "Default Printer (HP LaserJet 4P/4MP on LPT1:)", and the print range is set to "All" pages. The print quality is set to "High", and the number of copies is set to 1. There are checkboxes for "Print to File" and "Collate Copies", both of which are currently unchecked. The dialog box has three buttons on the right: "OK", "Cancel", and "Setup...".

Print

Printer: Default Printer (HP LaserJet 4P/4MP on LPT1:)

Print Range

☒ All

☐ Selection

☐ Pages

From: 1 To: 11

Print Quality: High

Copies: 1

☐ Print to File

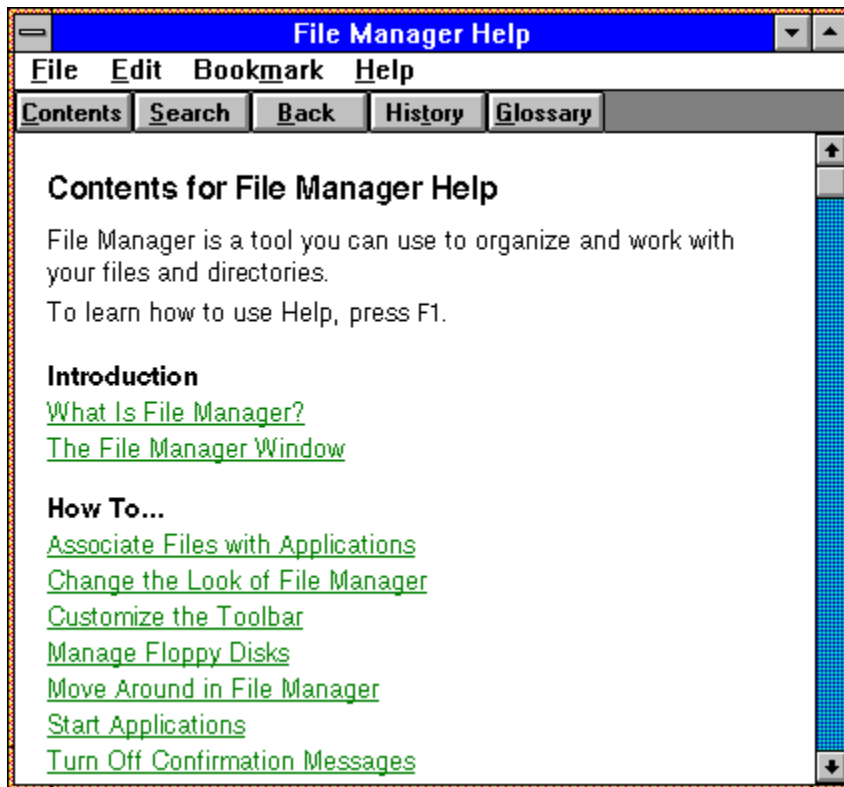
☐ Collate Copies

OK

Cancel

Setup...

Start Windows Help



Read Database To Clipboard

This will read an Access database (Structure only) and create code in the clipboard that will rebuild it. This will not retain your data, relationships or queries. Reasons for using this might be that you want to create a new test database and make sure it's clean - this will do it. I don't like carrying the database with me for an installation, I can call this routine and it will create it at install time. You might even want to keep old database structures archived for use at a later date. You can run this and add it to the database as a snippet. You have your own database library.

Read Database To Def File

This will read a database and create a "definiton" file that can be read by a routine I supply called **"BUILD_DB.BAS"** to rebuild it. This will not retain your data, relationships or queries. Reasons for using this might be that you want to create a new test database and make sure it's clean - this will do it. I don't like carrying the database with me for an installation, I can call **"BUILD_DB.BAS"** with this definition file and it will create the database at install time. You might even want to keep old database structures archived for use at a later date. This will do that and save space also.

Create Database From Def File

This will create an Access database from a definition file created using the Read Database To Def File.

Delete And Define Database

This will read and recreate an Access database. It **won't** keep the data. **ALL DATA WILL BE LOST!!!** Use this with care because you may spend a lot of time trying to recover from this if you expected your data to be put in the new database. This is good for retesting where you need to start with a "fresh" database.

Print Def File

This will print a database definition file created using the Read Database To Def File menu option. It's not fancy, but it will provide you with the information about the database within that file. It will generate a small report with the Database name and path, table names, field names and sizes and index names and fields. It uses "Times New Roman" font so if you don't have it installed in Windows, you may have trouble. I actually like it better than the Database Documenter that comes with Access 2.0. It doesn't have all the features, but it does provide you with a concise report about your database.

Last Minute Changes

This section was added because of some Last Minute changes that were added to Code Gen before releasing it.

1. There is now a Utilities menu that allows:

a. You can break up long strings into several lines and assign them to a variable. If you have used some of the Access SQL statements, you know how nice this will be. I also use it for comments. You have 3 fields: variable, source and preview. Variable is for the variable you want it assigned to (i.e. sSql). Source is the string to break up. This can be taken from the clipboard or typed in. Preview is what it will look like once it has been converted. The three buttons are self explanatory: Preview - previews the conversion, OK converts it and puts it in the clipboard, and Cancel backs out of converting it.

Note: The checkbox is for allowing assignment (sql = sql & "test") or for Sub procedure call (PrintRtn "print this"). I had a print routine that printed a bunch of lines and I added this option and it worked out nice, so I left it in.

b. You can now print an Access database structure without having to first create the definition file. Although the report is trivial, it has been much more useful than the one I get from Access and **it saves paper**.

2. I changed the database read procedure so that you are notified that Referential Integrity has been detected in your database and that VB CodeGen won't be able to create the database using the file (if you're using the definition file) or clipboard routine (if you're using the access to clipboard). The only check I currently make is for 2 or more indexes for one table that are both flagged as the Primary Key.

Read This First

Codegen has changed quite a bit between the time I was first going to release it and now. Unfortunately, I didn't update the help file with the new features. I have provided this file anyway because a lot of it is relevant. Of course the new stuff is not included. You should read the readme file to get a quick description of all features of Codegen.

You may see information about registering Codegen. This version is FREEware and isn't crippled. You can use it and pass it along to your friends without violating copyrights. If you decide to register it, you will get free bugfixes and upgrades, a regular install rather than just a zip file and free support. Since I did this on my own time, that's when tech support will be - ON MY OWN TIME. I hope you find this program useful. I have used it quite often for little things.

To read it [Press Here](#) and Alt-Tab through the running tasks to view it (or close this application).

