Code.Print for Visual Basic

Professional Edition

Table of Contents

System Requirements

Computer

IBM-PC/AT class or higher.

Operating System/Environment

MS-DOS or *PC-DOS* 3.3 or later. Microsoft *Windows* version 3.1 or later.

Host

Microsoft *Visual Basic* version 1.0, 2.0, 3.0,4.0 or 5.0. For information on compatibility with the earlier versions of Visual Basic, see Compatibility with Visual Basic 1.0 and 2.0.

In order for Code.Print to read and print your code, all forms and modules must be saved in text format. For information on how to use Code.Print to convert Visual Basic files from binary to text format, see Converting Visual Basic Files to Text Format.

System Resources

When Code.Print is printing a job, it can use as much as 5% GDI and 5% User resource space. When you are laying out a job, Code.Print's hit may be a total of 10% in both categories. Global memory requirements are negligible.

Disk Space

An installation of Code.Print can require as much as 1.5 MB of free space on a hard disk drive. As much as 1 MB may be used on the drive where your WINDOWS\SYSTEM directory is located. The exact amount will vary, since some of the files used by Code.Print may already be on your system and thus will not need to be installed. For more information, see Files Installed by Setup.

In addition, you will need sufficient disk space for spooling print files. On average, a print job of 100 pages will create a temporary print file of about 1 MB. Exact sizes vary with the printer driver and the fonts used. You can minimize the size of print files by turning on the Print While Processing option.

Code.Print for Visual Basic

Professional Edition

Table of Contents

Instructions

Code.Print for Visual Basic

Professional Edition

Table of Contents

Code.Print for Visual Basic
Professional Edition

Table of Contents

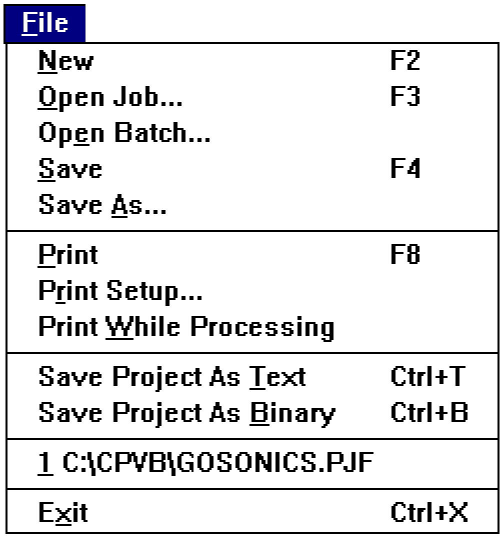
Code.Print for Visual Basic
Professional Edition

Table of Contents

Code.Print for Visual Basic
Professional Edition

Table of Contents

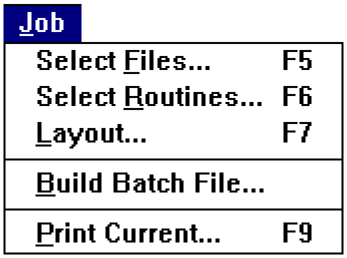
Click on the menu command for which you want help.



Code.Print for Visual Basic
Professional Edition

Table of Contents

Click on the menu command for which you want help.



Code.Print for Visual Basic

Professional Edition

Table of Contents

Click on the menu command for which you want help.

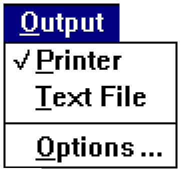


Table of Contents

Table of Contents

Instructions

Operating the Table of Contents

To expand a topic heading, click on a folder with a plus sign +
To collapse an expanded topic, click on the open folder +
To jump to a help topic, click on a page symbol +.
In some cases the link to a help topic is not indicated by a page symbol but instead by a picture of a menu or toolbar button. You can click on those, too.

Rule of Thumb: In this help system, if it's a picture the odds are very good that clicking on it will give you some information or take you somewhere where you can get information about the picture.

Close

Topic

New (File menu)

Related

Open Job (File menu)

Running Code.Print

Save As (File menu)

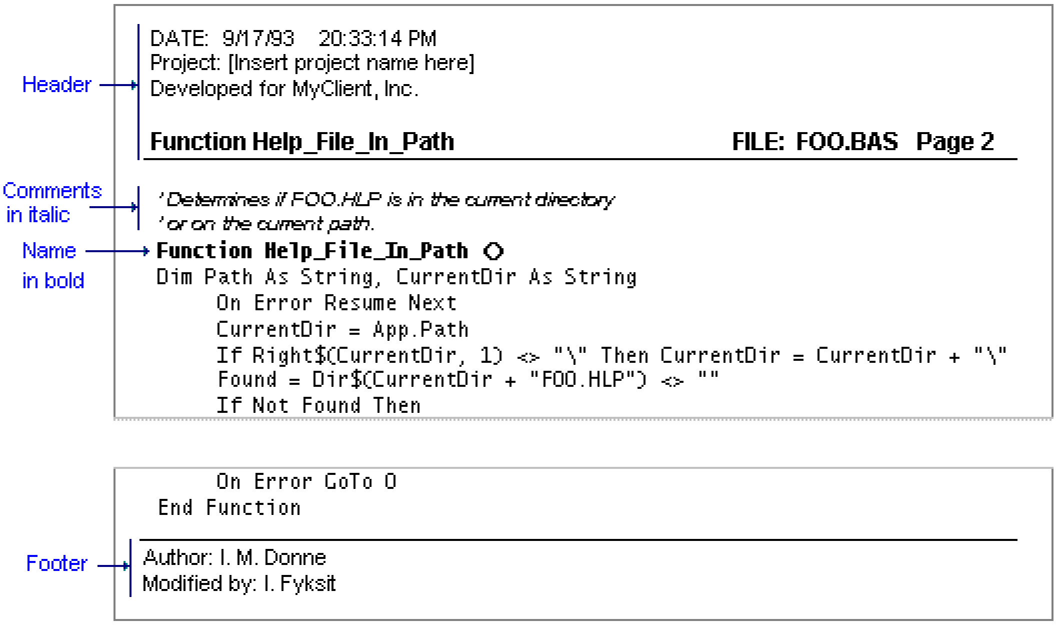
Setting Layout Defaults

Overview

Code.Print Pro is a tool designed for anyone who needs to create and maintain printed source code written in Microsoft Visual Basic. We suspect these are mainly people who make their living programming in Visual Basic. That's why we call it Code.Print *Pro*.

On the most superficial level Code.Print Pro is a pretty printer. You can create attractive, desktop-publishing quality printouts with headers and footers and any fonts your printer will support.

More important, though, is Code.Print's ability to organize your printouts so that they are easy to find your way around. You can put useful information in the header of every page--the date, time, file name, procedure name, and whatever other text you like. You can highlight the beginning of a routine by printing procedure names in bold type. You can print code in one font, comments in another.


Header
Comments
in italic
Name
in bold
Footer

DATE: 9/17/93 20:33:14 PM
Project: [Insert project name here]
Developed for MyClient, Inc.

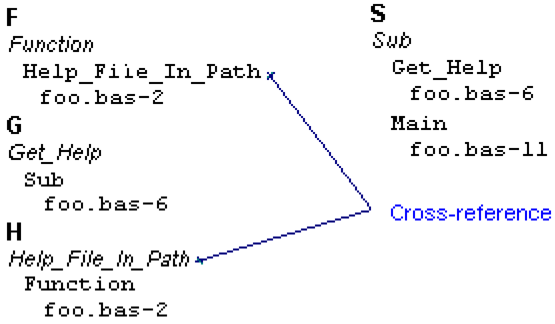
Function Help_File_In_Path FILE: FOO.BAS Page 2

```
'Determines if FOO.HLP is in the current directory
'or on the current path.
Function Help_File_In_Path ()
Dim Path As String, CurrentDir As String
     On Error Resume Next
     CurrentDir = App.Path
     If Right$(CurrentDir, 1) <> "\" Then CurrentDir = CurrentDir + "\"
     Found = Dir$(CurrentDir + "FOO.HLP") <> ""
     If Not Found Then
```

```
     On Error GoTo 0
End Function
```

Author: I. M. Donne
Modified by: I. Fyksit

And you can create a back-of-the-book index cross-referencing every control, procedure, and file in the project.


Cross-reference

```
F
Function
  Help_Fiie_In_Path
    foo.bas-2
G
Get_Help
  Sub
    foo.bas-6
H
Help_File_In_Path
  Function
    foo.bas-2
```

```
S
Sub
  Get_Help
    foo.bas-6
  Main
    foo.bas-11
```

Just as useful are features designed to make printing easy. You can run Code.Print and Visual Basic together, calling Code.Print's print engine rather than Visual Basic's whenever you press Ctrl+P. You can create reusable print jobs that Code.Print can crank out when Visual Basic is not running--or that you can simply drag from File Manager, drop on Print Manager, and then go on to other work while Code.Print handles the printing.

If you do much printing at all, you'll like Code.Print's incremental processing feature. Why print every single file in a project--a job that can easily total 100 pages or more--when you can print only those files that have changed?

Better yet, try putting incremental processing together with batch processing. Why hang around the printer waiting for your code when you can schedule the job for a time when the printer is free? And why hang around at all when Code.Print can print the job unattended?

Features

Listed below are some good reasons for making Code.Print Pro your tool of choice whenever you need to print your code. For detailed information on how to put Code.Print to work, see the topics listed in the Table of Contents under How to...

Visual Basic	**Code.Print Pro**	
●	●	Print current project
●	●	Print current file
●	●	Print current routine
	●	Print current selection
	●	Print a project that is not loaded in Visual Basic
	●	Print only the files that have changed
	●	Print today's work tonight, after you've gone home
	●	Sort procedures before printing
	●	Highlight procedure names with bold type
	●	Print code in one font and comments in another
	●	Print the date, procedure name, and file name in the header of every page
	●	Create an index
	●	Create a default style for all source code printouts

Topic

System Requirements

Related

Compatibility with Visual Basic 1.0 and 2.0

Converting Visual Basic Files to Text Format

Files Installed by Setup

Print While Processing (File menu)

What Setup Did

Setup created a directory for storing Code.Print files. It gave the directory the name you said you want it to have, and put it on the drive you specified. Setup then installed a number of files in that directory and some others in your WINDOWS\SYSTEM directory. For a list of these files, see Files Installed by Setup.

Setup did not modify your AUTOEXEC.BAT or CONFIG.SYS in any way, nor did it add anything to your WIN.INI. However, the first time you run Code.Print a section labeled [Code.Print VB] will be added to your WIN.INI. This is where Code.Print stores the last position of its window and the names of the most recently used files.

Setup registered Code.Print by adding entries to the file REG.DAT in your WINDOWS directory. These entries associate file extensions .PJF and .PJB with CPVB.EXE, the Code.Print executable file. For information on how to use these file associations, see Running Code.Print. To modify or delete Code.Print's registration data, you'll need to run the program REGEDIT.EXE located in your WINDOWS directory. For more information, see your Microsoft Windows *Users Guide* or the Windows Help file REGEDIT.HLP.

Finally, Setup asked if you wanted it to create a program group for Code.Print. If you said No, then it's up to you to add a program item for Code.Print to whatever program group you prefer.

To add Code.Print to a program group

1. In Program Manager, click the group to which you want to add Code.Print.
2. From the File menu, choose New.

   The New Program Object dialog will appear.
3. In the New Program Object dialog, choose Program Item. Then push the Ok button.

   The Program Item Properties dialog will appear.
4. In the Program Item Properties dialog, push the Browse button.

   The Browse dialog will appear.
5. In the Browse dialog, set Drives to the drive where you installed Code.Print.
6. Set Directories to the directory where you installed Code.Print.
7. In the File Name list box, choose CPVB.EXE. Then push the Ok button.

   Focus will return to the Program Item Properties dialog.
8. Set whatever other options you prefer. Then push the Ok button.

   The Code.Print icon will appear in the program group.

   Note Several startup options are available for Code.Print. For an explanation of these options, see Running Code.Print.

Topic

What Setup Did

Related

Files Installed by Setup

Files Created by Code.Print

Running Code.Print

Files Installed by Setup

Setup installs these files in your Code.Print directory:

CPVB.EXE
CPVB.HLP
REGISTER.EXE
UNINSTALL.EXE
MODEMS.INI
INSTALL.LOG

Setup installs these files in your WINDOWS\SYSTEM directory:

CBK.VBX
CALL32.DLL
CMDIALOG.VBX
DWSPYDLL.DLL
REGLOAD.EXE
SBC.VBX
SBCKBD.VBX
SS3D.VBX
VBIS23MX.DLL
VBRUN300.DLL

Before overwriting an existing file, Setup checks the file's date/time stamp and version resource. If your system has a more recent version of a file than the one on Code.Print's installation disk, the file on your system is not overwritten.

Note If you decide to remove Code.Print from your system, you may not want to delete all the files that Setup placed in your WINDOWS\SYSTEM directory, as some of them may be used by other programs, including programs you have written.

Topic

Files Installed by Setup

Related

Files Created by Code.Print

Files Created by Code.Print

In addition to the files installed by Setup, several others are created as you use Code.Print.

- If you save layout options as the user-defined default, Code.Print will create the file **CPVB.INI**. This file will be placed in the Code.Print directory.
- Each time you create and save a print job, the configuration of the job is stored in a file with the extension **.PJF**. These files can reside in any directory you choose.
- Each time you create a batch job, the information needed to run the batch is stored in a file with the extension **.PJB**. These files can reside in any directory you choose.
- Each time you run a batch job, Code.Print records the date and time it prints each file in the batch. This information is kept in a log file with the same root name as the batch job file and the extension **.LOG**. These files will be stored in the same directory as the corresponding batch job file.
- Any temporary files that Code.Print needs while it is working are created in your Windows TEMP directory, if you have one, or in the root directory of the current drive. These files have the prefix ~CP and are deleted when Code.Print no longer needs them, so long as the program terminates normally.

Topic

Files Created by Code.Print

Related

Files Installed by Setup

Converting Visual Basic 3.0 or 2.0 Files to Text Format

Visual Basic forms and modules created with version 3.0 or 2.0 are stored in either text or binary format. For Code.Print to be able to read and print your code, all forms and modules included in the print job must be saved in text format.

To convert files from binary to text using Code.Print

1. Start Visual Basic and open your project.
2. From Code.Print's File menu, choose Save Project As Text.

⌨ For a shortcut to the Save Project As Text command, press Ctrl+T.

Code.Print will issue a Visual Basic Save File As command for each form or module listed in the project's .MAK file. You will probably see the Save File As dialog flash on the screen several times. You do not need to make any entry (in fact, you *cannot* make any entry) in this dialog.

To convert files from binary to text in Visual Basic

1. Select a form or module.
2. From the file menu, choose Save File As.
3. Click on the Save as Text checkbox.
4. Click on the OK button.

Repeat these steps for each form or module in your project.

To make text the Visual Basic default format

You can save some time by setting Visual Basic's default file format to text. However, the default setting applies only to new forms or modules. The default for existing files is the current format, whatever that is.

1. From Visual Basic's Options menu, choose Environment.
2. Set the Default Save As Format option to Text.

Topic

Overview

Related

Compatibility with Visual Basic 1.0 and 2.0

Converting Visual Basic Files to Text Format

Creating a Print Job

Printing from the Visual Basic Environment

Running Code.Print

When you installed Code.Print on your system, the Setup program asked if you wanted it to create a program group for Code.Print. If you said No, then it's up to you to add a program item for Code.Print to whatever program group you prefer.

To start Code.Print from the Windows Program Manager

1. In the Program Manager window, open the program group that contains the Code.Print icon.
2. Double-click the Code.Print icon.

To start Code.Print from the Windows File menu

1. From the Program Manager File menu, choose Run.
2. Type the path to Code.Print and CPVB.EXE. For example:

   c:\cpvb\cpvb.exe

Running Code.Print with Visual Basic

Whenever Code.Print and Visual Basic are running at the same time, Visual Basic's Window menu will include the item *Code.Print*. In addition, Code.Print will intercept the Visual Basic shortcut key combination Ctrl+P. For more information, see Printing from the Visual Basic Environment.

Note You cannot run two instances of Code.Print at once. Code.Print will prevent a second instance of itself from being loaded in order to avoid messaging conflicts when it subclasses the Visual Basic menu.

Command Line Options

- To open a print job or batch job file automatically when you start Code.Print, type

  cpvb.exe *file name***.pjf**

  or

  cpvb.exe *file name***.pjb**

  If the file is not in Code.Print's working directory, you must prefix a path to the file.

- To open and immediately print a print job or batch job, type

  cpvb.exe /p *file name***.pjf**

  or

  cpvb.exe /p *file name***.pjb**

Here's a tip.... If you print the same print job or run the same batch file every day, you may want to create an icon for the job in your Visual Basic program group. Just set the Command Line field in the Program Item Properties dialog using one of the forms shown above. Then you can load and run the job simply by double-clicking on the icon.

Drag and Drop

Setup registered Code.Print by adding entries to the file REG.DAT in your WINDOWS directory. These entries associate file extensions .PJF and .PJB with CPVB.EXE, the Code.Print executable file. Double-clicking in File Manager on a file name with one of those extensions will automatically run Code.Print and load the file. If you drag a file name with one of those extensions from File Manager and drop it on Print Manager, Code.Print will run and, if the file is a print job file, print the file. If the file is a batch file, Code.Print will execute the batch job.

Topic

Running Code.Print

Related

Creating a Print Job

Printing Several Jobs At Once

Printing from the Visual Basic Environment

Compatibility with Other Versions of Visual Basic

Code.Print can print code written with Visual Basic version 5.0 or earlier. However, it cannot process Visual Basic source files directly unless they are saved in ASCII text format. Version 4.0 and 5.0 source files are always saved as text. Versions 2.0 and 3.0 of Visual Basic offer the option to save forms and modules in text format. Version 1.0 does not. However, Visual Basic 1.0 through 3.0 offer the option to save the contents of .FRM and .BAS files as ASCII text files with the extension .TXT. Code.Print can print .TXT files.

If you are using version 2.0 or 3.0 of Visual Basic, convert any files saved in binary format to text format before including them in a print job. For information on how to convert an entire project at once, see Converting Visual Basic Files to Text Format.

If you are using Visual Basic 1.0, Code.Print's Save Project As Text command will be unavailable (dimmed) on the File menu. Sorry. You must create .TXT files for each form and module that you wish to include in a print job by choosing Save Text from Visual Basic's Code menu. When you define the print job with Code.Print's Select Files command, be sure you select only the .TXT versions of your forms and modules. If a file you select is not in ASCII format, Code.Print will display a warning message.

VB 5.0 Support

Code.Print supports all versions of Visual Basic from release 1.0 to 5.0. When operating on Win95 and WinNT, Code.Print supports long file names for both file selection and as referenced in Visual Basic .vbp project files. All new VB5 syntax and constructs are supported including property procedures. Class files are also supported.

IDE integration

Code.Print Pro can hook into the Visual Basic 16-bit IDE only. Because of the nature of protected memory space on 32-bit operating systems such as Win95 and WinNT, the 32-bit version of VB4 and VB5 are inaccessible to 16-bit programs such as Code.Print Pro. Code.Print Pro can still process all 32-bit VB4 and VB5 files but they have to be selected and saved manually in print job files.

IDE integration is still supported for all versions of VB 1.0 through 3.0. In addition, IDE integration is fully functional under all flavors of Windows.

Note: In order for Code.Print to successfully hook into Visual Basic when running under Win95 or WinNT, Visual Basic and Code.Print must be running in the same memory space. Do not run Visual Basic or Code.Print in separate memory spaces otherwise Code.Print will be unable to connect to the VB IDE.

Topic

Compatibility with Other Versions of Visual Basic

Related

Converting Visual Basic Files to Text Format

Select Files (Job menu)

Topic

Contacting Technical Support

Related

Before You Call...

New command (File menu or Toolbar)

Closes the current print job, resets all layout options, and creates a new (untitled) job.

When you first run Code.Print you do not need to use the New command to create a new print job. A new (untitled) job is created automatically each time you start the program. You can immediately begin configuring the new job by selecting files or layout options.

However, once you have configured an untitled job or opened a job previously saved in a print job file, you must close the current job before you can begin defining a new one. Use the New command to do this.

If you have unsaved changes to the active print job, Code.Print will display a dialog asking if you want to save the changes. To save, press the Yes button. If the current job is untitled, the Save As dialog will appear. Otherwise, Code.Print will save the changes to the open print job file and then close the file.

> **Note** Before you save a new print job for the first time, Code.Print's window caption displays the word *Untitled*. After you save and name the job, the window caption displays the file name. When you select File, New the job is closed and the window caption again displays the word *Untitled*.

After Code.Print has closed the current print job, it will reset all layout options to the default settings. You can then begin configuring a new job.

Topic

Converting Visual Basic Files

Related

Save Project As Text (File menu)

Printing from the 16 bit Visual Basic Environment

Whenever Code.Print and 16 bit Visual Basic (version 3.0 or 4.0) are running at the same time, Code.Print appears as an item on Visual Basic's Window menu. In addition, Code.Print will intercept the key combination Ctrl+P, normally used to execute Visual Basic's Print command. *This feature is unavailable when running 32 bit Visual Basic.*

Choosing Code.Print from Visual Basic's Window menu or pressing Ctrl+P will execute Code.Print's Print command. You will be given the option to print the currently highlighted code, the current routine, the current file, or the entire project.

> **Note** Code.Print can print the current selection or the current routine even if the Visual Basic file is stored in binary format. However, if you want to print an entire file or an entire project, you must first save the files in text format. For information on how to use Code.Print to convert Visual Basic files from binary to text format, see Converting Visual Basic Files to Text Format.

To print from within Visual Basic

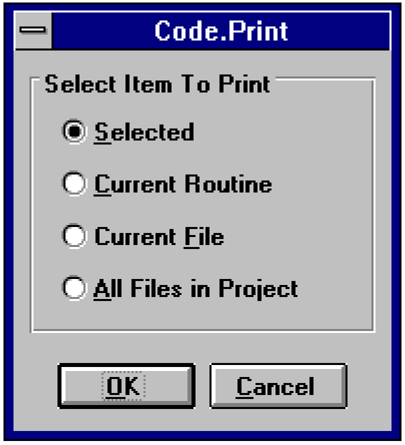
1. From the Visual Basic File menu, choose Open.

   The File Open dialog will appear. Select the project you wish to print.

2. From the Visual Basic Window menu, choose Code.Print.

   For a shortcut, press Ctrl+P.

   The Code.Print dialog will appear.



● Selected
○ Current Routine
○ Current File
○ All Files in Project

3. Select the option you prefer by clicking on the appropriate radio button.

   Code.Print will send its output to the currently selected printer. (To select a printer, choose Print Setup from Code.Print's File menu.) Text will be formatted using the current layout defaults.

Topic

Printing from the Visual Basic Environment

Related

Output menu

Print Current... (Job menu)

Print Setup (File menu)

Setting Layout Defaults

Creating a Print Job

When you use Code.Print to print from within the Visual Basic environment, you can print only the current routine, the current file, or the current project. That may be all you want to do, much of the time. For information on how to do that, see Printing from the Visual Basic Environment.

When you print from Code.Print itself, on the other hand, you can print any selection of Visual Basic files or routines, even when they are not currently loaded in Visual Basic, and you can save your selection so that you can use it again and again. You do this by creating a print job and saving it as a print job file.

Once you've done that, you gain access to still more powerful features of Code.Print, such as its ability to print only modified files (incremental processing) and to print several jobs at once (batch processing). And with batch processing you also gain the ability to schedule a print job so that it will execute automatically, even on the weekend when you're off doing something else.

The key, as we've said, is to create a print job specification that can be stored on disk and used again. To learn how to do that, read the topics listed below.

What to read next

- **Selecting Files and Routines to Print**
- **Using a .MAK File to Organize a Job**
- **Selecting Paper Size and Orientation**
- **Setting Margins, Line-Spacing, and Page Breaks**
- **Controlling Headers and Footers**
- **Working with Fonts**
- **Adding or Removing an Index**
- **Saving Your Work**
- **Setting Layout Defaults**

Topic

Creating a Print Job

Related

Printing from the Visual Basic Environment.

Printing Only Modified Files

Printing Several Jobs At Once

Setting Layout Defaults

Selecting Files and Routines to Print

The first step in creating a reusable print job is to tell Code.Print what it is that you want to print as part of a single job. You can include an entire project, selected files, or just certain routines within files.

To select files to print

1. From Code.Print's Job menu, choose Select Files. (Or push the Files button on the toolbar.)

   The Select Files dialog will appear.
2. Set the drive and directory you want to search just as you do with any Windows common dialog.
3. From the Type combo box, choose the file extensions you want displayed.

   Code.Print can print any ASCII text file. Thus acceptable file extensions include any standard Visual Basic file, and of course .TXT.

   Note For Code.Print to be able to read and print Visual Basic forms and modules, they must be saved in text format. See Converting Visual Basic Files to Text Format.
4. From the File list box, select one or more files and then press the Add button.

   Note Selecting a .MAK file is a special case. See Using .MAK Files to Organize a Print Job.

   The files you have selected will appear in the Selected Files list.

   If you want to remove a file from the list, highlight the file name in the Selected Files list and then push the Delete button. If you want to remove all files from the list and start over, push the Clear button.
5. (Optional) Code.Print prints your files in the order in which they appear in the Selected Files list. If you want to arrange the files in alphabetical order, push the Sort button. If you want to move a file up or down in the list, highlight the file name and then push the Up button or the Down button until the file is where you want it.
6. When you have selected all the files that you want included in the job, push the OK button.

By default, Code.Print will print all routines in every Visual Basic file you have selected. If the file is a form, Code.Print will print the declarations and form information as well. If you do not want to print everything in every file, you can easily select just those parts of a file that you do want printed.

To specify which parts of a file to print

1. Add one or more files to the job by following the steps listed above.
2. From the Job menu, choose Select Routines. (Or push the Routines button on the toolbar.)

   The Select Routines dialog will appear.
3. From the File list, choose the name of a Visual Basic file.
4. In the Sections group to the right of the File list, set the options you want.

   Form Information Select this option to print form and control properties and their values.

   Declarations Select this option to print the declarations section.

   Routines Select this option to print procedures and functions. By default, all routines in a file will be included in the print job. You can limit the job to individual routines from the current file by selecting specific routines from the Available list.

   Sort Select this option to sort routines in alphabetical order. When this option is turned off, routines will be printed in the order in which they appear in the file.

   Set as Default Press this button to make the current settings for options in the Sections group the default settings whenever you use Code.Print. Each time you create a new print job the options you save as default will automatically be set for that job.

   Set All Press this button to apply the current settings for options in the Sections group to all files in the current job. You can then set custom options for only those files that you want printed differently.

5. From the Available list in the Routines group, select the subroutines that you want to print and then push the Add button.

   The routines you have selected will appear in the Selected list on the right.

   Note If you want to print all subroutines in the file, do not select any routines listed in the Available list. Whenever the Selected list is empty, Code.Print assumes you want to print all routines. Just be sure you have checked the Routines box in the Sections group.

6. Code.Print prints routines in the order in which they appear in the Selected list. If you want to arrange them in alphabetical order, push the Sort button. If you want to move a routine up or down in the list, highlight the name and then push the Up button or the Down button until the routine is where you want it.
7. Repeat steps 1-6 for each file for which you want to specify custom options. When you have finished, push the OK button.

Topic

Selecting Files and Routines for a Print Job

Related

Converting Visual Basic Files to Text Format

Printing from the Visual Basic Environment

Select Files (Job menu)

Select Routines (Job menu)

Using .MAK Files to Organize a Print Job

Using a Project File to Organize a Print Job

Visual Basic project files can be just as useful in organizing a print job as they are in pulling together the parts of a project. You can use [roject files with Code.Print in two ways.

Finding a Project's Files

One method is to use a project file to help you find the files belonging to a project so you can include some of them in a print job. That way you can be sure the code you print is the code that belongs with your project even though you may have several files with similar names.

To select files from a .MAK/.VBP

1. From the Job menu, choose Select Files.

   The Select Files dialog will appear.

2. From the File list box, select the .MAK file for your project and then push the Add button.
3. When Code.Print asks if you want to view the files in the .MAK/.VBP, push the Yes button

   A list of files in the project will appear above the Selected Files list.

4. From the list box labeled Files in .MAK/.VBP, select the files you want to include in the print job, and then push the Add button.

   The files you have selected will appear in the Selected Files list. You can then add other files to the job by selecting them from the Files list.

5. When you have selected all the files you want included in the job, push the OK button.

Defining an Open-Ended Job

The other method is to tell Code.Print to print all of the files in a project without specifying individual file names. That way your print job will stay in sync with your project. When you add, delete, or rename files in the project, you will not need to make corresponding changes to the print job. Each time it prints the job, Code.Print will look at the .MAK or .VBP and print the files it finds listed there.

To print all the files in a Project

1. From the Job menu, choose Select Files.

   The Select Files dialog will appear.

2. In the File list box, select the .MAK/.VBP file for your project and then push the Add button.
3. When Code.Print asks if you want to view the files in the .MAK/.VBP, push the No button

   The project file will appear in the Selected Files list. You can then add other files to the print job, including other project files.

   Note Remember, however, that you do not need to add any of the files listed in the .MAK. If you do individually select files that are also listed in the .MAK/.VBP, the files will be printed twice, once as a part of the .MAK/.VBP and again as an individual file.

4. When you have selected all the files you want included in the job, push the OK button.

Topic

Using a .MAK File to Organize a Print Job

Related

Compatibility with Visual Basic 1.0 and 2.0

Converting Visual Basic Files to Text Format

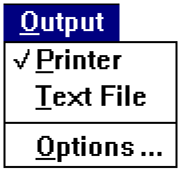
Selecting Paper Size and Orientation

How you set the width and height of your paper depends on whether you intend to send the job to a printer or to a text file. When you send output to a printer, paper size is set by choosing Print Setup from the File menu. When you send output to a text file, paper size is set by choosing Options from the Output menu.

To select either printer or text file output

1. From Code.Print's main menu, choose Output.

   The Output menu will drop down.



2. The menu items Printer and Text File are mutually exclusive options. The current setting is indicated by a check mark.

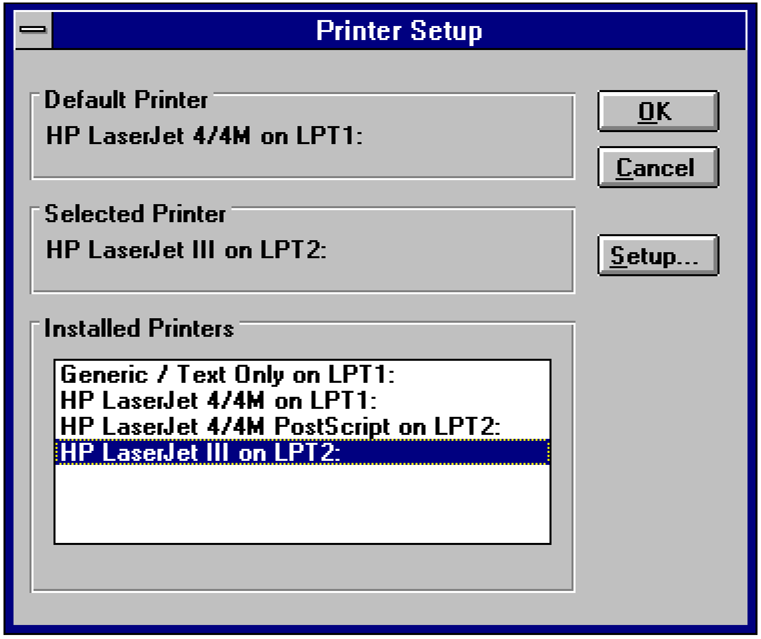
- If you want to send output to a printer and the item Printer is not checked, click on Printer now.
- If you want to send output to a text file and the item Text File is not checked, click on Text File now.

Once you have selected an output device, you may proceed to setting the paper size and orientation.

To set paper size and orientation for printed output

1. From Code.Print's File menu, choose Print Setup.

   The Printer Setup dialog will appear.



Default Printer
HP LaserJet 4/4M on LPT1:

Selected Printer
HP LaserJet III on LPT2:

Installed Printers
Generic / Text Only on LPT1:
HP LaserJet 4/4M on LPT1:
HP LaserJet 4/4M PostScript on LPT2:
HP LaserJet III on LPT2:

2. In the Installed Printers list box, highlight the printer you want to use for this job.

   The name of the device you have selected will appear in the Selected Printer box.

3. Push the Setup button.

   A Setup dialog will appear. The layout of the dialog and the options available will vary with the printer driver.

4. Set the paper size and orientation you prefer. Then push the Ok button.

   Focus will return to the Printer Setup dialog.

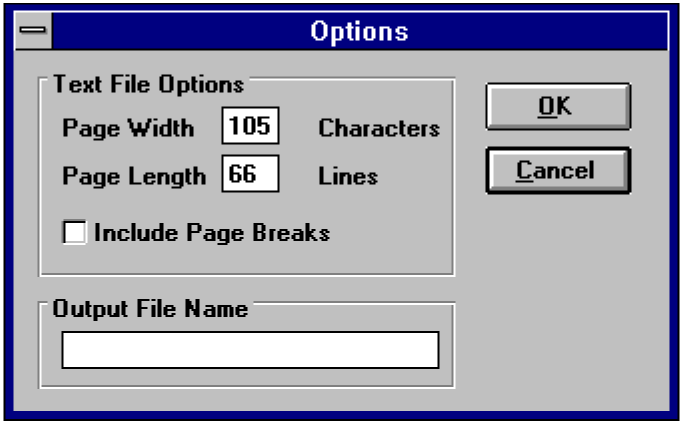
5. In the Printer Setup dialog, push the Ok button.

The printer and printer options you have selected will be saved as part of the print job specification when you save the job by choosing Save or Save As from Code.Print's File menu.

To set paper size for text file output

1. From the Output menu, choose Options.

   The Options dialog will appear.



Text File Options

Page Width 105 Characters

Page Length 66 Lines

□ Include Page Breaks

Output File Name

When Code.Print's output is directed to a text file, the size of the (virtual) paper is determined by the settings you choose for Page Width, specified as a number of characters per line, and Page Length, specified as a number of lines per page. An 8.5 X 11 inch page translates to 105 characters by 66 lines.

2. Type the settings you prefer.
3. When you are satisfied with your settings, push the Ok button.

   Your settings will be saved as part of the print job specification when you save the job by choosing Save or Save As from Code.Print's File menu.

If a check mark appears in front of the word Printer, then output will go to the currently selected printer.

To select a printer, choose Print Setup from the File menu.

If a check mark appears in front of the words Text File, then output will go to an ASCII file.

By default, text output will be written to a file with the same root name as the print job and the extension .TXT. To specify another file name, choose Options from the Output menu.

Choose Options if you want to set paper size for Text File output. You can also specify a file name for text output.

The current Windows default printer. This is the printer that Code.Print will use if you have not explicitly selected another printer.

Printer selected for this print job. When you save a print job, the name of the Selected Printer is stored in the print job file. If you later open a print job file or print one as part of a batch job, Code.Print will automatically use the printer associated with that job.

A list of printers installed on your system. To select a printer, highlight its name in the list. The name will appear in the Selected Printer box. Press Ok to confirm your choice, or Cancel to abort.

When you press the Setup button, Code.Print will display a dialog for selecting printer options. Available options vary with the printer driver. Typically, you can choose paper source, size, and orientation (Portrait or Landscape). Whatever printer options you choose will be saved with the print job.

Topic

Selecting Paper Size

Related

Options (Output menu)

Print Setup (File menu)

Printer (Output menu)

Setting Margins, Line-Spacing, and Page Breaks

Text File (Output menu)

Setting Margins, Line Spacing, and Page Breaks

You can control margins, line-spacing and page breaks by setting the Page options in the Layout dialog.

To set margins, line-spacing, and page breaks

1. From the Job menu, choose Layout.

   The Layout dialog will appear. Near the bottom right corner of the dialog is a box labeled **Settings**. In this box you will find a list of layout elements for which you can choose settings. Selecting an item from the list changes the contents of the dialog.

2. From the Settings listbox, choose Page.

   Three groups of controls will appear in the Layout dialog: **Margins**, **Page Breaks**, and **Line Spacing**.

3. In the Margins group, set the left, right, top, and bottom margins. The default unit of measurement is inches. Select centimeters, if you prefer.

   Note If a line of code is too long to fit within the margins you specify, Code.Print will break the line and indicate a continuation with the symbol **->**.

4. (Optional) In the Page Breaks group, choose either option or neither.

Page Breaks on New Routine Code.Print will insert a page break before beginning a new routine. Some routines will be printed on a single page; longer routines will span several pages. But all will begin at the top of a page.

Fit Routines onto Page Choose this option if you want Code.Print to insert a page break only when all of a subroutine will not fit on the current page. The difference between this option and the first one is that Fit Routines onto Page can cause several small routines to appear on one page, whereas Page Breaks on New Routine never will.

5. In the Line Spacing group, choose the setting you prefer.

6. When you are satisfied with your choices, push the Ok button.

Topic

Setting Margins, Line-Spacing, and Page Breaks

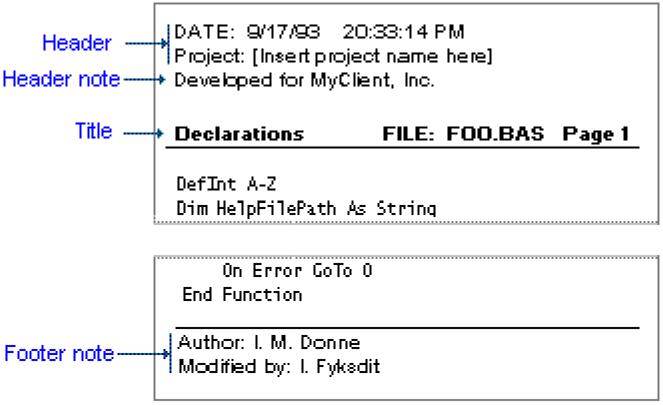
Related

Layout (Job menu)

Selecting Paper Size

Controlling Headers and Footers

Each page of a printout produced by Code.Print includes a header, a title line, and a footer. You can choose what information you want to appear in each of those elements.



```
DATE: 9/17/93   20:33:14 PM
Project: [Insert project name here]
Developed for MyClient, Inc.

Declarations          FILE: FOO.BAS   Page 1

DefInt A-Z
Dim HelpFilePath As String
```

```
      On Error GoTo 0
 End Function

Author: I. M. Donne
Modified by: I. Fyksdit
```

To specify what gets printed in the header and title

1. From the Job menu, choose Layout.

   The Layout dialog will appear. Near the bottom right corner of the dialog is a box labeled **Settings**. In this box you will find a list of layout elements for which you can choose settings. Selecting an item from the list changes the contents of the dialog.

2. From the Settings listbox, choose Header.

   Three groups of controls will appear in the Layout dialog: **Header**, **Title Line**, and **Borders**.

3. Check the boxes next to the items you want to be printed in each of these areas.
4. When you are satisfied with your choices, push the Ok button.

To add notes to the header and footer

1. From the Job menu, choose Layout.

   The Layout dialog will appear. Near the bottom right corner of the dialog is a box labeled **Settings**. In this box you will find a list of layout elements for which you can choose settings. Selecting an item from the list changes the contents of the dialog.

2. From the Settings listbox, choose Notes.

   Edit lines for entering header and footer notes will appear.

3. Enter whatever text you want added to the header and footer.
4. Check the box labeled **Print Notes**. If you want the notes to appear only on the first page of the printout for each file in the job, check the box labeled **First Page Only** too.
5. Push the Ok button to confirm your settings.

Topic

Controlling Headers and Footers

Related

Layout (Job menu)

Working with Fonts

Key parts of your printout can be highlighted by printing them in a distinctive font and font style. You can select individual fonts for the header, the title line, code, and comments. You can also choose to print the names of routines and form objects in bold type.

The fonts that are available will vary with the printer driver you plan to use. Before making any font selections, you should select a printer by choosing Print Setup from the File menu. Your selection will be saved with other information pertaining to the print job.

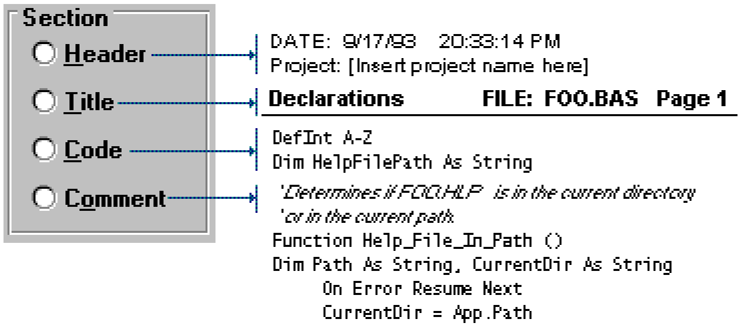
To select fonts

1. From the Job menu, choose Layout.

   The Layout dialog will appear. Near the bottom right corner of the dialog is a box labeled **Settings**. In this box you will find a list of layout elements for which you can choose settings. Selecting an item from the list changes the contents of the dialog.

2. From the Settings listbox, choose Fonts.

   The controls for selecting fonts will appear in the dialog. In the left center of the dialog is a group labeled **Section.**



```
DATE:  9/17/93   20:33:14 PM
Project: [Insert project name here]
Declarations          FILE:  FOO.BAS   Page 1
DefInt A-Z
Dim HelpFilePath As String
'Determines if FOO.HLP  is in the current directory
'or in the current path
Function Help_File_In_Path ()
Dim Path As String, CurrentDir As String
     On Error Resume Next
     CurrentDir = App.Path
```

3. In the **Section** group, click on the radio button beside the name of the element for which you want to choose a font.
4. In the **Font** combo box, choose the name of a font.
5. In the **Size** combo box, choose the font size you prefer.

   The range of sizes available varies with the font.

6. In the **Style** group, check whatever options you want.

   A sample showing your choices will appear in the **Sample** box.

   Note Although Code.Print can enumerate the fonts supported by the currently selected printer,it may not be able to display samples of all of them. If you select a resident font that Code.Print cannot display, the **Sample** box will show the message *Unable to Display*.

7. Repeat steps 3 through 6 for each of the remaining sections (Header, Title Line, Code, and Comments).
8. When you are satisfied with your choices, push the Ok button.

To print the names of routines and form objects in bold type

1. From the Job menu, choose Layout.

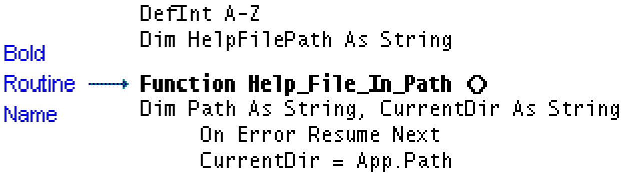
   The Layout dialog will appear. Near the bottom right corner of the dialog is a box labeled **Settings**. In this box you will find a list of layout elements for which you can choose settings. Selecting an item from the list changes the contents of the dialog.

2. From the Settings listbox, choose General.

   The controls for setting general layout options will appear in the dialog.

3. Check the box labeled **Bold Routine Name** to print the names of subroutines and functions in bold type.

   When routine names are printed in bold, it will be easier to find the start of a routine as you scan through a printout of your code.



```
DefInt A-Z
Dim HelpFilePath As String

Function Help_File_In_Path
Dim Path As String, CurrentDir As String
     On Error Resume Next
     CurrentDir = App.Path
```

4. Check the box labeled **Bold Form Objects** to print the names of controls in bold type.
5. When you are satisfied with your choices, push the Ok button.

Topic

Working with Fonts

Related

Layout (Job menu)

Print Setup

Adding or Removing an Index

By default, Code.Print prints an index at the end of the print job. Every file in the job will have an index entry showing the number of the page in the job where the text can be found. Each routine and, optionally, each form object will be indexed by name and cross-referenced by type.

To add or remove an index

1. From the Job menu, choose Layout.

   The Layout dialog will appear. Near the bottom right corner of the dialog is a box labeled **Settings**. In this box you will find a list of layout elements for which you can choose settings. Selecting an item from the list changes the contents of the dialog.

2. From the Settings listbox, choose General.

   Controls for setting general layout options will appear in the dialog.

3. The checkbox labeled **Print Index** controls whether an index is printed with every printout.

   If you want an index printed, check the box.

   If you do not want an index printed, clear the box.

4. (Optional) If you want form objects to be indexed as well as subroutines, check the box labeled **Index Form Objects**.

5. When you are satisfied with your choices, push the Ok button.

Topic

Adding or Removing an Index

Related

Layout (Job menu)

Saving Your Work

Before you save a new print job for the first time, Code.Print's window caption displays the word *Untitled*. To save and name the print job, choose Save As from the File menu or push the Save button on the toolbar. After you save the job, the window caption will display the file name.

All selections you have made will be saved in a print job file (.PJF) so that you can later reuse them. This includes options you have set in the Files, Routines, and Layout dialogs, any selections you have made from the Output menu, and the printer you have selected and printer options you have set with the Print Setup command.

To begin defining a new print job, choose New from the File menu or push the New button on the toolbar.

Topic

Saving Your Work

Related

New (File menu)

Save As (File menu)

Print Setup (File menu)

Setting Layout Defaults

If you find that you frequently set the same options with the Layout command, you may want to save those options as the default. Each time you create a new print job, it will automatically be configured with your default settings for fonts, margins, page numbering, header/footer text, and so on.

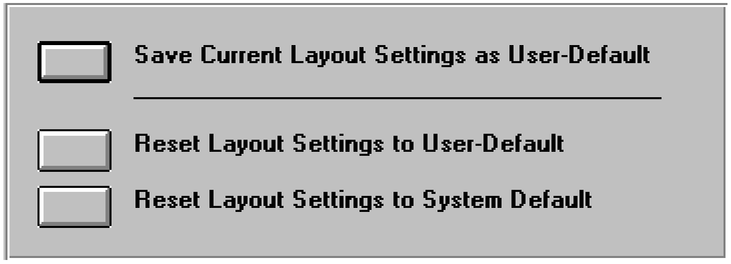
To create a default layout

1. From the Job menu, choose Layout.

   The Layout dialog will appear.

2. Change the state of each checkbox and radio button to the setting you want Code.Print to store as the default.
3. From the Layout dialog's Settings group, choose Default.

   Three pushbuttons will appear.


Save Current Layout Settings as User-Default
Reset Layout Settings to User-Default
Reset Layout Settings to System Default

4. Push the button labeled Save Current Settings as User-Default.

   Code.Print will store your default settings in an initialization (.INI) file. The next time you begin a new (untitled) print job, the job's layout will be automatically configured with your default layout. You can then change individual settings in the normal way if you want to customize a particular print job.

After you have changed layout options from their defaults, you may want to start over. No problem. You can easily reset all options to either your own defaults or to Code.Print's system defaults.

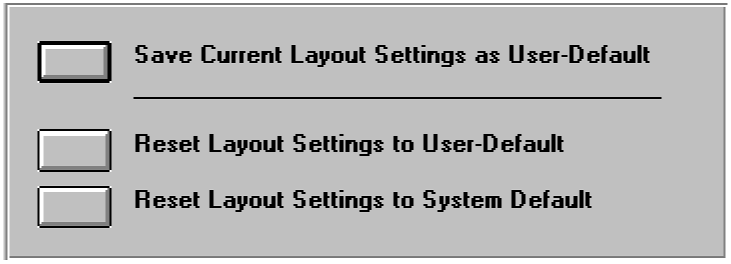
To reset layout options to your defaults

1. From the Job menu, choose Layout.

   The Layout dialog will appear.

2. From the Layout dialog's Settings group, choose Default.

   Three pushbuttons will appear.


Save Current Layout Settings as User-Default
Reset Layout Settings to User-Default
Reset Layout Settings to System Default

3. Push the button labeled Reset Layout Settings to User-Default.

   All options will be restored to the settings last stored with the Save Current Layout Settings as User-Default button.

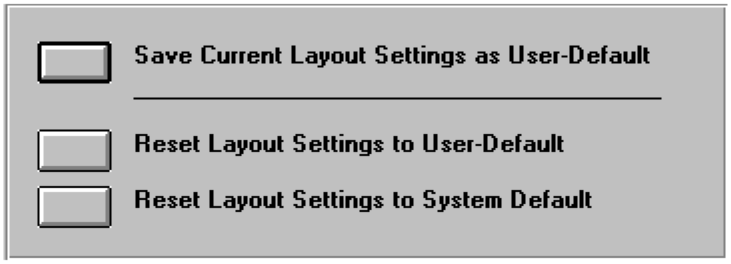
To reset layout options to Code.Print's system defaults

1. From the Job menu, choose Layout.

   The Layout dialog will appear.

2. From the Layout dialog's Settings group, choose Default.

   Three pushbuttons will appear.


Save Current Layout Settings as User-Default
Reset Layout Settings to User-Default
Reset Layout Settings to System Default

3. Push the button labeled Reset Layout Settings to System Default.

   All options will be restored to the settings they had when you first installed Code.Print.

Topic

Setting Layout Defaults

Related

Layout (Job menu)

Saves the current setting of each option selected in the Layout dialog as the default.

Resets the options in the Layout dialog to the settings last saved as the user-defined default.

Resets the options in the Layout dialog to the settings they had when you first installed Code.Print.

Sending a Job to the Printer

You can print a Code.Print print job from within Code.Print or from the Windows File Manager. In either case, the job will be printed by the device that you selected when you created the job. For information on selecting a printer, see Print Setup.

To print a job from Code.Print

1. From Code.Print's File menu, choose Open Job. Or simply push the Open button on the toolbar.

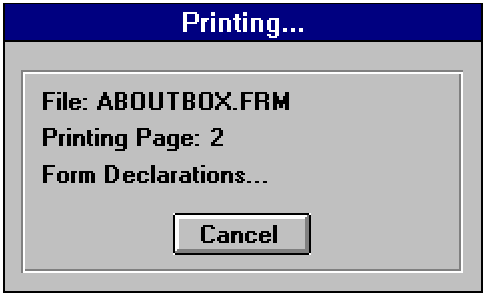
   The Open dialog will appear.

2. In the Open dialog, select the print job file you want to print. Then push the Ok button.

   The print job file name will appear in the caption of Code.Print's main window.

3. From Code.Print's File menu, choose Print. Or just push the Print button on the toolbar.

   When Code.Print begins printing, it will display a message box showing the progress of the job.



File: ABOUTBOX.FRM

Printing Page: 2

Form Declarations...

To print a job from File Manager

1. Run the Windows Print Manager.
2. In the Windows File Manager, click on the name of the print job file you want to print. Then, while holding down the left mouse button, drag the file to Print Manager.
3. When the file is over Print Manager, drop it by releasing the left mouse button.

   Code.Print will run and begin printing the job. When Code.Print begins printing, it will display a message box showing the progress of the job.

Topic

Sending a Job to the Printer

Related

Print (File menu)

Print Setup (File menu)

Printing only modified files

Printing from the Visual Basic environment

Printing several jobs at once

Printing Only Modified Files

Successive printouts of a large print job can consume a lot of paper and time. You can save time and spare a tree by printing only those files that have changed since the last time you printed the job.

To print only modified files

1. From the toolbar, select Layout.

   The Layout dialog will appear.

2. From the Settings group, choose General.

   The General settings group will appear in the Layout dialog.

3. Look for the checkbox marked Incremental Processing. If this box is not checked, click on it to check it. Then press the OK button to dismiss the Layout dialog.

   Note The Incremental Processing option will be unavailable (dimmed) if you have checked either the Header option for Continuous page-numbering or the Line Numbers option for Continuous line-numbering.

   If you want Incremental Processing to become part of the print job's normal configuration, save the job now by choosing Save from the File menu or by pressing the Save button on the toolbar.

4. From the toolbar, select Print.

   The job will be sent to the current output device. Code.Print will not print a file if its date/time stamp indicates that it has not been modified since the last time it was printed.

Topic

Printing Only Modified Files

Related

Layout (Job menu)

Print (File menu)

Printing Several Jobs At Once

You can execute several print jobs with a single command by creating a batch job. More important, batch jobs can be scheduled for automatic execution at a time of your choosing. At night after you've left for the day, for instance.

Batch jobs are designed to run unattended. They will run through to completion without user intervention. If for some reason an error should occur while a batch job is running--say a file that should be available is not where Code.Print expects it to be--Code.Print will log the error and go on with the rest of the job.

What to read next

- **Creating a Batch Job**
- **Running a Batch Job**
- **Using Several Printers At Once**
- **Closing a Batch File**
- **Modifying a Batch File**

An ASCII text file with the same root name as the batch file and the extension .LOG will be created or, if it already exists, appended whenever you run a batch job. In this file, Code.Print will record the date and time it prints each file in the batch and any errors that occur as it processes the files.

Topic

Printing Several Jobs At Once

Related

Build Batch File (Job menu)

Open Batch (File menu)

Creating a Batch Job

A batch job consists of one or more print job files and, optionally, a scheduled start time.

To build a batch file

1. From the Job menu, choose Build Batch File.

   The Build Batch File dialog will appear.

2. In the **Type** list box, select the extension .PJF to display print job files.

   Batch jobs may include only print job files (.PJF). You should select the extension for batch job files (.PJB) only if you want to edit an existing batch job.

3. In the **File** list box, highlight the names of the files you want to include. Then push the Add button.

+ If you highlight the name of a **print job file** and then push the Add button, the file name will appear in the Selected Files list.

+ If you highlight the name of a **batch job file** and then push the Add button, Code.Print will ask if you want to edit the batch job.

   - Press Yes to edit the batch file.

     The current configuration of the batch job will be loaded into the Build Batch File dialog. You can then add or delete files and change the start time. When you have finished editing, you can save the batch file under the same name or choose a new name.

   - Press No if you do not want to edit the batch file.

     You cannot add batch job files to a batch job. The name of the batch job file will not appear in the Selected Files list.

4. (Optional) In the space labeled **Batch Job Start Time,** enter a time of day in twenty-four hour format. For instance, to run a job at 8:30 am, enter 08:30. To run it at 8:30 pm, enter 20:30.

   Note If your system clock is inaccurate, the job may not take place when you expect it to.

   If you plan to run the batch job manually, leave **Batch Job Start Time** blank.

5. When you have configured the batch job to your satisfaction, click Ok.

   The Save As dialog will appear.

6. Enter a name for the batch job file.

7. Click Ok to save your work or Cancel to abandon it.

   If you click Ok the batch job you have just created will be loaded. The batch file name will appear in the window caption, as will the start time if you scheduled one. You can then run the job or close the file and go on to another task.

Topic

Creating a Batch Job

Related

Running a Batch Job

Closing a Batch File

Modifying a Batch File

Running a Batch Job

Code.Print must be running and the batch file must be open before you can run the job. For information on how to start Code.Print and immediately run a batch file, see the explanation of command line options under Running Code.Print.

To run a batch file

1. From the File menu, choose Open Batch.

   Select the name of the batch file you want to run. Then click Ok. The name of the batch job file will appear in the caption of Code.Print's main window. If the job has a start time, it too will appear on the window caption.

2. What you do next depends on whether you want to start the job manually or let Code.Print execute the job at its scheduled time.

- If you want to run the job immediately, choose Print from the File menu, or push the Print button on the toolbar.
- If you have scheduled the job for a particular time, just leave Code.Print running and the batch job open while you go do something else.

When Code.Print completes the job, it will display a message informing you that its work is done. An ASCII text file with the same root name as the batch file and the extension .LOG will be created or, if it already exists, appended to. In this file, Code.Print will record the date and time it prints each file in the batch and any errors that occur as it processes the files.

Topic

Running a Batch Job

Related

Creating a Batch Job

Closing a Batch File

Modifying a Batch File

Using Several Printers At Once

Unless your printer has an unlimited paper supply, it may not be able to handle several large print jobs chained together in a batch job.

One way to overcome this limitation is to create a batch job that sends each print job to a separate printer. To do this, you'll need to follow these steps for each print job in the batch:

1. Open one of the print job files by choosing Open Job from the File menu.
2. Choose Print Setup from the File menu.
3. Set the Selected Printer to the printer you want this job to go to.
4. Save the print job by choosing Save from the File menu.

When you have completed these steps for all of the print jobs you intend to print, create a batch job by following the procedure described under Creating a Batch Job.

Topic

Using Several Printers At Once

Related

Creating a Batch Job

Running a Batch Job

Closing a Batch File

Modifying a Batch File

Closing a Batch File

To close a batch file and open another file, choose Open Job or Open Batch from the File menu. To close a batch file and begin a new (untitled) print job, choose New from the File menu or push the New button on the toolbar.

Topic

Closing a Batch File

Related

Creating a Batch Job

Running a Batch Job

Modifying a Batch File

Modifying a Batch File

You can easily change the start time of a batch file or add files to the job. The process is nearly identical to the one described under Creating a Batch Job.

To edit a batch job file

1. From the Job menu, choose Build Batch File.

   The Build Batch File dialog will appear.

2. In the **Type** list box, select the extension .PJB.
3. In the **File** list box, highlight the name of the batch file you want to edit. Then push the Add button.

   Code.Print will ask if you want to edit the batch job.

4. Press Yes.

   The current configuration of the batch job will be loaded into the Build Batch File dialog. You can then add or delete files and change the start time.

5. When you have configured the batch job to your satisfaction, click Ok.

   The Save As dialog will appear.

6. Enter a name for the batch job file.

   You can save the batch file under the same name or choose a new name.

7. Click Ok to save your work or Cancel to abandon it.

   If you click Ok, the batch job you have just edited will be loaded. The batch file name will appear in the window caption, as will the start time if you scheduled one. You can then run the job or close the file and go on to another task.

Topic

Modifying a Batch File

Related

Creating a Batch Job

Running a Batch Job

Closing a Batch File

Select Files... command (Job menu or Toolbar)

Selects the files to be included in a print job.

Use the Select Files command to specify which files to include in a new (untitled) print job or to add files to an existing print job.

> **Note** For Code.Print to be able to read and print your code, all forms and modules included in the print job must be saved in text format. See Converting Visual Basic Files to Text Format.

When you choose Select Files from the Job menu or click the Files button on the toolbar, the Select Files dialog appears.

Select Files dialog

Directory Location of files to be included in the print job. Changing the directory also changes the files listed in the File list box.

Type Extensions of files to be displayed in the File list box. Changing the extension either narrows or broadens the list displayed in the File list box.

File List of files to choose from. Highlight the name of a file you want to include in the print job and then press the Add button. (Or simply double-click on the file name.) The file name will appear in the Selected Files list.

If you attempt to add a project file to the Selected Files list, Code.Print will ask if you want to see the files listed in the project.

To add all files in the project to the print job, press No.

Even though only the project file appears in the Selected Files list, all files for the project will be included in the print job.

The advantage of using a project to define a print job is that file selection is dynamic. If you later modify the project, your changes will propagate to the print job as well. Adding files to or deleting files from the project automatically adds them to or deletes them from the print job.

The disadvantage of using a project to define the job is that you cannot omit parts of the project. All files and all routines within the files are automatically part of the print job. (This does not mean that all files need to be printed every time you send a job to the printer, however. See the explanation of incremental processing in the discussion of the Layout command.)

To add only selected files from the project file, press Yes.

A list of files in the project will appear in the Select Files dialog. You can then select individual files just as you would from the File list box. However, in this case changes to the project file will not propagate to the print job. If you later add or delete files from the project, you will need to add or delete files from the print job as well.

Selected Files List of files selected for inclusion in the print job. To add a file to the list, highlight the name of the file in the Files list and then press the Add button. To remove a file from the list, highlight the file name in the Selected Files list and then press the Delete button.

Add Adds a file to the Selected Files list. Highlight the name of a file in the File list box and then press the Add button. The file name will appear in the Selected Files list.

Delete Deletes a file from the Selected Files list. Highlight the name of a file in the Selected Files list box and then press the Delete button. The file name will disappear from the Selected Files list.

Clear Clears all file names from the Selected Files list. Press this button if you want to start over.

Sort Sorts the file names in the Selected Files list in alphabetical order. The order in which files are listed determines the order in which they are printed.

Up Moves a file name up in the Selected Files list. Highlight the name you want to reposition and then press the Up button. The file name will move up one position in the Selected Files list.

Down Moves a file name down in the Selected Files list. Highlight the name you want to reposition and then press the Down button. The file name will move down one position in the Selected Files list.

You *can* start Code.Print and immediately load a print job file. For an explanation of startup options, see Running Code.Print.

You can set defaults for most of the options available in the Layout dialog. For information on setting user-defined defaults, see Setting Layout Defaults. For an explanation of the options available for layout, see the discussion of the Layout command.

Open Job... command (File menu or Toolbar)

Opens a print job file and loads the job configuration.

Use this command to open an existing print job file. The file name will appear in the window caption and Code.Print will load the print job configuration so that you can modify or print the job.

Code.Print can keep only one print job open at a time. If you have another print job open and have made changes that you wish to keep, you must save the current job before opening another. To save a new (untitled) print job, choose the Save As command from the File menu. To save an existing job, choose Save.

When you choose Open from the File menu or press the Open button on the toolbar, the Open dialog appears.

Open Job dialog

Select the print job file that you want to open.

Directories Print job files may be in any directory.

List Files of Type Print job files have the file extension .PJF.

File Name A print job file may have any base file name you like, so long as it is a legal DOS file name.

For information on starting Code.Print and immediately opening a job, see the explanation of startup options under Running Code.Print.

Topic

Open Job... (File menu)

Related

New (File menu)

Save (File menu)

Save As (File menu)

Open Batch... command (File menu)

Opens a batch file so that you can run it.

Use this command to open a batch job file created with the Build Batch command. You can then choose to run the batch job manually or, if the job has a scheduled start time, let Code.Print run the job unattended. For additional information on batch-processing, see Printing Several Jobs At Once.

When you choose Open Batch from the File menu, the Open Batch dialog appears.

Open Batch dialog

Select the file that you want to open.

Directories Batch job files may be in any directory.

List Files of Type Batch job files have the file extension .PJB.

File Name A batch job file may have any base file name you like, so long as it is a legal DOS file name.

When the file is opened the file name appears in the window caption, as does the job's start time if one was scheduled. Menu commands for defining print jobs will be unavailable.

Closing a Batch Job

To close a batch file and open another file, choose Open Job or Open Batch from the File menu. To close a batch file and begin a new (untitled) print job, choose New from the File menu or push the New button on the toolbar.

Before you can run a batch job, you must first start Code.Print and then open the batch job file.

- If you want to run the job immediately, choose Print from the File menu, or push the Print button on the toolbar.
- If you have scheduled the job for a particular time, just leave Code.Print running and the batch job open while you go do something else. When the appointed time arrives, Code.Print will execute the job. When the job is complete, Code.Print will display a message box informing you that its work is done.

Topic

Open Batch... command (File menu)

Related

Build Batch File (Job menu)

Printing Several Jobs At Once

Save command (File menu or Toolbar)

Saves changes to an open print job file.

Use this command to save changes to an open print job file. To save a new (untitled) print job, choose the Save As command from the File menu.

> **Note** The name of the currently open file is displayed in the window caption. If no print job file is open, Code.Print's window caption displays the word *Untitled* and the Save command will be unavailable (dimmed) on the File menu.

All changes you have made will be saved in the active print job file. This includes options you have set in the Files, Routines, and Layout dialogs, any selections you have made from the Output menu, and any change you have made by selecting a new printer or setting new printer options with the Print Setup command.

Topic

Save (File menu or Toolbar)

Related

Save As (File menu)

Save As... command (File menu)

Saves and names a new print job or copies and renames an open job.

Use this command to save and name a new print job or to save a copy of an open print job file under a new name or in a new location. For a new (untitled) job, you can also execute the Save As command by pushing the Save button on the toolbar.

To save an open print job file with its existing name and location, choose the Save command from the File menu.

> **Note** Before you save a new print job for the first time, Code.Print's window caption displays the word *Untitled*. After you save and name the job, the window caption displays the file name.

When you choose either Save or Save As, all selections you have made from Code.Print's menus and dialogs are saved in a print job file (.PJF) so that you can later reuse them. This includes the options you have set in the Files, Routines, and Layout dialogs, as well as any selections you have made from the Output menu. The name of the currently selected printer is also stored with the print job, and so are any printer options you have set.

When you select Save As from the File menu or, for a new job, when you press the Save button on the toolbar, the Save As dialog appears.

Save As dialog

Directories Print job files may be in any directory.

Save File as Type Print job files have the file extension .PJF.

File Name A print job file may have any base file name you like, so long as it is a legal DOS file name.

For information on loading a saved print job configuration, see the discussion of the Open Job command.

Topic

Save As... (File menu)

Related

New (File menu)

Open Job (File menu)

Save (File menu)

Print command (File menu or Toolbar)

Sends a print job to the current output device or runs a batch job.

Use this command to execute a print job or to run a batch job. In either case the print job file or batch job file must be open.

To open a print job, choose Open Job from the File menu. To open a batch job, choose Open Batch. For information on opening and immediately printing a job, see the explanation of startup options under Running Code.Print.

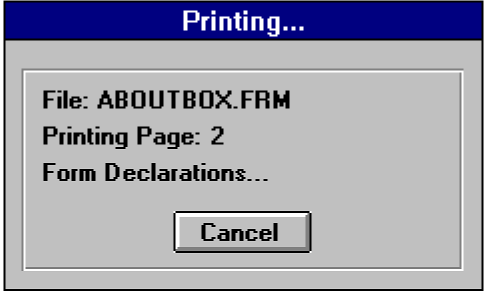
> **Note** You are not limited to printing prepared print jobs. To print code that you are editing in the Visual Basic environment, choose Print Current from the Job menu. For additional information, see Printing from the Visual Basic environment.

Output

Output may be sent either to a printer or to a text file. To select a destination, choose either Printer or Text File from the Output menu.

By default, all files included in the job will be printed. However, you can choose to print only those files that have changed since the last printout. See Printing only modified files.

When Code.Print begins printing, it displays a message box showing the progress of the job.



File: ABOUTBOX.FRM

Printing Page: 2

Form Declarations...

The name of the file that Code.Print is currently processing. If you have selected Print While Processing, a separate print file will be created for each Visual Basic file in the job.

The current page number. For information on how to specify page numbering, see the explanation of the Layout dialog.

The part of the current file that is being printed. For information on how to select parts of Visual Basic forms for printing, see the explanation of the Routines dialog.

You may cancel the job at any time by pushing the Cancel button.

Topic

Print (File menu)

Related

Output menu

Printing only modified files

Printing from the Visual Basic environment

Printing several jobs at once

Print While Processing (File menu)

Running Code.Print (Startup options)

Selecting files and routines to print

Print Setup... command (File menu)

Selects a printer.

Code.Print will assume that you want printer output to go to the default Windows printer unless you have designated another printer. Use this command to change printers.

When you select Print Setup from the File menu, the Print Setup dialog appears.

Print Setup dialog

Default Printer The current Windows default printer. This is the printer that Code.Print will use if you have not explicitly selected another printer.

Selected Printer Printer selected for this print job. When you save a print job, the name of the Selected Printer is stored in the print job file. If you later open a print job file or print one as part of a batch job, Code.Print will automatically use the printer associated with that job.

> **Note** When you start Code.Print, the Windows default printer is automatically selected for new (untitled) print jobs. If you select another printer, *that* printer will be automatically selected for new print jobs. Existing print jobs, on the other hand, use whatever was the selected printer when the job was last saved.

Installed Printers A list of printers installed on your system. To select a printer, highlight its name in the list. The name will appear in the Selected Printer box. Press Ok to confirm your choice, or Cancel to abort.

Setup When you press the Setup button, Code.Print will display a dialog for selecting printer options. Available options vary with the printer driver. Typically, you can choose paper source, size, and orientation (Portrait or Landscape). Whatever printer options you choose will be saved with the print job.

Topic

Print Setup... (File menu)

Related

Output menu

Print (File menu)

Print While Processing command (File menu)

Toggles the option to print each file in the job as soon as it is processed.

Use this command to control whether Code.Print begins sending to the current output device as soon as each Visual Basic file in a job is processed, or whether it waits until all files are processed before spooling the job.

Print While Processing has two settings: Off (unchecked) or On (checked). The setting toggles each time you choose Print While Processing from the File menu.

Settings

Off (unchecked) This is the default setting. Code.Print will process all files included in a job before sending the first output. A single print file will be spooled.

On (checked) Code.Print will send output for each Visual Basic file in the job as soon as it is processed. Several smaller print files will be spooled. This setting may be useful if your system does not have the free disk space needed for spooling very large print files. It can also reduce the time required to print a large project.

> **Note** The Print While Processing option does not apply when output is sent to a text file by choosing Text File from the Output menu. In this case, only one output file will be created for a job no matter what the setting for Print While Processing.

Whether output goes to the printer or to a text file depends on the setting selected from the Output menu.

Code.Print requires about 1MB disk space for every 100 pages of printout. The exact figure varies with the printer driver and the fonts used in creating the document.

Topic

Print While Processing (File menu)

Related

Output menu

Save Project As Text command (File menu)

Saves the current Visual Basic project (.MAK) and all its forms and modules in text format. (VB 3.0 and 2.0 only)

Visual Basic forms and modules are stored in either text or binary format. For Code.Print to be able to read and print your code, all forms and modules included in the print job must be saved in text format.

> **Note** Choosing Save Project As from Visual Basic's File menu will save only the project's .MAK file. The individual files in the project will remain in whatever format was selected when the files were last saved. To convert a project's files from within Visual Basic, you must choose File Save As from Visual Basic's File menu for each of the files in the project.

You can convert all of a project's files at once with Code.Print's Save Project As Text command.

To convert from binary to text format

1. Start Visual Basic and open your project.
2. From Code.Print's File menu, choose Save Project As Text.

The keyboard shortcut to the Save Project As Text command is Ctrl+T.

Code.Print will issue a Visual Basic Save File As command for each form or module listed in the project's .MAK file. You will probably see the Save File As dialog flash on the screen several times. You do not need to make any entry (in fact, you *cannot* make any entry) in this dialog.

Topic

Save Project As Text

Related

Compatibility with Visual Basic 1.0 and 2.0

Converting Visual Basic Files to Text Format

Save Project As Binary

Keyboard Shortcuts

Ctrl+B	Save Project as Binary (File menu)
Ctrl+P	Printing from the Visual Basic Environment
Ctrl+T	Save Project as Text (File menu)
Ctrl+X	Exit (File menu)
F2	New (File menu)
F3	Open Job (File menu)
F4	Save (File menu)
F5	Select Files (Job menu)
F6	Select Routines (Job menu)
F7	Layout (Job menu)
F8	Print (File menu)
F9	Print Current (Job menu)

Save Project As Binary command (File menu)

Saves the current Visual Basic project (.MAK) and all its forms and modules in binary format. (VB 3.0 and 2.0 only)

Visual Basic forms and modules are stored in either text or binary format. For Code.Print to be able to read and print your code, all forms and modules included in the print job must be saved in text format. However, after printing your code you may want to convert the files to binary format.

> **Note** Choosing Save Project As from Visual Basic's File menu will save only the project's .MAK file. The individual files in the project will remain in whatever format was selected when the files were last saved. To convert a project's files from within Visual Basic, you must choose File Save As from Visual Basic's File menu for each of the files in the project.

You can convert all of a project's files at once with Code.Print's Save Project As Binary command.

To convert from text to binary

1. Start Visual Basic and open your project.
2. From Code.Print's File menu, choose Save Project As Binary.

The keyboard shortcut to the Save Project As Binary command is Ctrl+B.

Code.Print will issue a Visual Basic Save File As command for each form or module listed in the project's .MAK file. You will probably see the Save File As dialog flash on the screen several times. You do not need to make any entry (in fact, you *cannot* make any entry) in this dialog.

Topic

Save Project As Binary

Related

Save Project As Text

Exit command (File menu)

Closes Code.Print.

Choose Exit from the File menu to quit Code.Print.

The keyboard shortcut to the Exit command is Ctrl+X.

You will be prompted to save any unsaved changes to the print job file. If Visual Basic is running, Code.Print will be removed from the Visual Basic Window menu.

Text File command (Output menu)

Designates a text file as the destination for output.

Use this command to send output to a text file rather than to a printer. The next time you choose Print from the File menu or press the Print button on the Toolbar, output will be sent to an ASCII text file.

Note When you save a print job, the current output setting is saved with the job as well.

You can verify the current destination for output by looking for the item on the Output menu with a check mark beside it. If the item Text File is checked, then output will go to a text file. If another item is checked and you want to send output to a text file, choose Text File. The item Text File will be checked the next time you pull down the Output menu. (The destinations on the Output menu are mutually exclusive. You cannot simultaneously send output to a printer and to a text file.)

By default, Code.Print will create a file with the same base file name as the print job and the extension .TXT. To specify another name, choose the Options command from the Output menu. In either case, the text file will be created in the directory where the print job file is stored. If the directory has another file of the same name as the output text file, it will be overwritten.

Note The Print While Processing option (File menu) does not apply when output is sent to a text file. Only one output file will be created for a job.

Whatever paper size you may have specified with the Print Setup command does not apply when you send output to a text file. To control the size of the (virtual) paper in a text file, choose the Options command from the Output menu.

Most Recently Used Files

Code.Print keeps track of the print job files (up to four) that you have used most recently. To open one of the print job files in the list, just click on the file name.

Printer command (Output menu)

Designates the printer as the destination for output. (Default)

Each time you start Code.Print, the destination for output is automatically set to the default device. You can change the destination by selecting another choice from the Output menu. To reset the destination to the printer, choose Printer.

Note When you save a print job, the current output setting is saved with the job as well.

You can verify the current destination for output by looking for the item on the Output menu with a check mark beside it. If the item Printer is checked, then output will go to the printer. If another item is checked and you want to send output to the printer, choose Printer. The item Printer will be checked the next time you pull down the Output menu. (The destinations on the Output menu are mutually exclusive. You cannot simultaneously send output to a printer and to a text file.)

Printed output is sent to the default Windows printer. To select a specific printer, choose Print Setup from the File menu.

Topic

Select Files... (Job menu)

Related

Converting Visual Basic Files to Text Format

Using .MAK Files to Organize a Print Job

Select Routines... (Job menu)

Select Routines... command (Job menu or Toolbar)

Selects which parts of Visual Basic forms or modules to include in a print job. (The default is all.)

By default, Code.Print prints all routines in every file selected with the Select Files command. If you want to print only certain routines from one or more files, use the Select Routines command to tell Code.Print which routines to print.

In addition, you can specify what parts of a Visual Basic form you want printed. You can print or omit:

- Form information
- Declarations
- Routines

You can also use the Select Routines command to determine the order in which routines are printed. If you choose to print all routines from a file, you can tell Code.Print to sort routines alphabetically before printing them. If you select individual routines from a file, you have the option to sort the selected routines alphabetically or to arrange them in any order you wish.

When you choose Select Routines from the Job menu or click the Routines button on the toolbar, the Select Routines dialog appears.

Select Routines dialog

File

This is a combo box with a drop-down list of the files selected with the Select Files command. The options you choose from the Sections and Routines groups apply only to the currently selected file.

Sections

If neither Form Information, Declarations, nor Routines is selected, Code.Print will print a blank page. You should select at least one of these three options.

Form Information (aka Control Information) Select this option to print form and control properties and their values.

Declarations Select this option to print the declarations section.

Routines Select this option to print procedures and functions. By default, all routines in a file will be included in the print job. You can limit the job to individual routines from the current file by selecting specific routines from the Available list.

Sort Select this option to sort routines in alphabetical order. When this option is turned off, routines will be printed in the order in which they appear in the file.

Set as Default Press this button to make the current settings for options in the Sections group the default settings whenever you use Code.Print. Each time you create a new print job the options you save as default will automatically be set for that job.

Set All Press this button to apply the current settings for options in the Sections group to all files in the current job. You can then set custom options for only those files that you want printed differently.

Routines

Available List of routines in the file currently selected in the File combo box.

Tab Order This check box appears only if the currently selected File is a Visual Basic form. Check the box if you want the routines to be sorted in tab order.

Selected List of individual routines to be printed. If the Selected list is empty, all routines will be printed. To print only certain routines, highlight one or more routines in the Available list and then press the Add button. (Or double-click on a name in the Available list.) The names of the routines will appear in the Selected list.

Add Adds a routine to the Selected list. Highlight the name of a routine in the Routines list box and then press the Add button. The name of the routine will appear in the Selected list.

Note Double-clicking on a file name will also add it to the selected list. You can also add multiple files with one click by highlighting the names of several files and then pushing the Add button. However, multi-selection is not enabled within the Selected Files list box itself.

Delete Deletes a routine from the Selected list. Highlight the name of a routine in the Selected list box and then press the Delete button. The name of the routine will disappear from the Selected list.

Clear Clears all routines from the Selected list. Press this button if you want to start over.

Sort Sorts the names in the Selected list in alphabetical order. The order in which routines are listed determines the order in which they are printed. (If you choose to print all routines, you can still choose to sort them in alphabetical order. See the Sort option in the Sections group.)

Up Moves the name of a routine up in the Selected list. Highlight the name you want to reposition and then press the Up button. The name will move up one position in the Selected list.

Down Moves the name of a routine down in the Selected list. Highlight the name you want to reposition and then press the Down button. The name will move down one position in the Selected list.

Topic

Select Routines... (Job menu)

Related

Select Files... (Job menu)

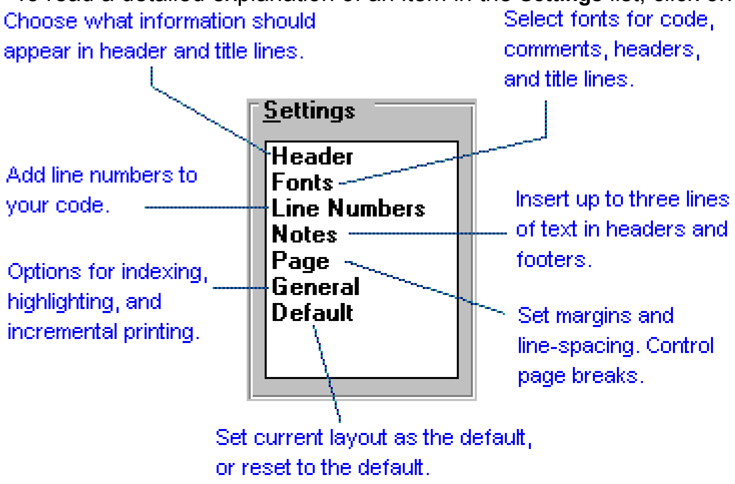
Layout... command (Job menu or Toolbar)

Formats a print job.

Use the Layout command to tell Code.Print how you want a job formatted. You can also use the layout command to save layout options as the default, making it possible to create a "house style" for printouts.

When you choose Layout from the Job menu, the Layout dialog appears. Near the bottom right corner of the dialog is a box labeled **Settings**. In this box you will find a list of layout elements for which you can choose settings. Selecting an item from the list changes the contents of the dialog.

To read a detailed explanation of an item in the **Settings** list, click on the name of the item in the picture below:



Choose what information should appear in header and title lines.

Select fonts for code, comments, headers, and title lines.

Add line numbers to your code.

Insert up to three lines of text in headers and footers.

Options for indexing, highlighting, and incremental printing.

Set margins and line-spacing. Control page breaks.

Set current layout as the default, or reset to the default.

Topic

Layout... (Job menu)

Related

Header

Fonts

Line Numbers

Notes

Page

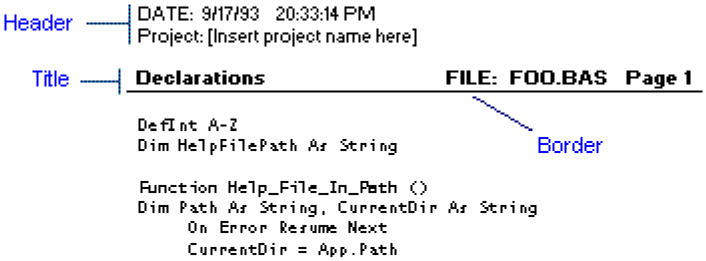
General

Default

Layout dialog: Header

Selects what information should appear in header and title lines.

Use the Header settings group of the Layout dialog to tell Code.Print what information to include in the header and title lines. In addition, you can specify whether you want Code.Print to draw a border separating headers and footers from the main body of the page.



```
DATE: 9/17/93   20:33:14 PM
Project: [Insert project name here]

Declarations                         FILE:  FOO.BAS   Page 1

DefInt A-Z
Dim HelpFilePath As String

Function Help_File_In_Path ()
Dim Path As String, CurrentDir As String
    On Error Resume Next
    CurrentDir = App.Path
```

Settings in the dialog are organized by groups labeled Header, Title Line, and Border.

Header

To print items in this group on the first page only, check the item and then check the box next to it labeled First Page Only.

Date and Time Prints the system date and time in the header.

Print Job Name Prints a job name or title in the header. The name printed is whatever text you enter in Name Text.

Name Text Text to print when Print Job Name is checked.

Title Line

To print items in this group on the first page only, check the item and then check the box next to it labeled First Page Only.

Print Routine Name Prints the name of the current routine at the top of the page.

File Name Prints the name of the current file.

Include Path Adds the path to the file name. Applies only if File Name is also selected.

Page #'s Prints page numbers.

Continuous Numbers pages continuously from the first page of the job to the last. When this option is turned off, page numbering will restart with each file. The option to number pages continuously will be unavailabel (dimmed) if you have checked the General option for Incremental Processing.

Start Number for the first page.

Borders

Print Borders Draws border lines separating headers and footers from the body of the page.

Topic

Header

Related

Fonts

Line Numbers

Notes

Page

General

Default

Layout dialog: Fonts

Selects fonts for code, comments, headers, and title lines.

In the left center of the dialog, in a group labeled **Section**, are radio buttons labeled **Header**, **Title**, **Code**, and **Comment**. You can make information in each of these elements stand out by choosing a distinctive font and font style.

To select a font

1. In the **Section** group, click on the radio button beside the name of the element for which you want to choose a font.
2. In the **Font** combo box, choose the name of a font.

   The fonts in the list will vary with the currently selected printer. To select a printer for this job, choose Print Setup from the File menu.
3. In the **Size** combo box, choose the font size you prefer.

   The range of sizes available varies with the font.
4. In the **Style** group, check whatever options you want.

   A sample showing your choices will appear in the **Sample** box.

   Note Although Code.Print can enumerate the fonts supported by the currently selected printer,it may not be able to display samples of all of them. If you select a resident font that Code.Print cannot display, the **Sample** box will show the message *Unable to Display*.

Topic

Fonts

Related

Header

Line Numbers

Notes

Page

General

Default

Layout dialog: Line Numbers

Adds line numbers to your code.

Use the Line Numbers settings group of the Layout dialog to tell Code.Print to print line numbers with your code. You have the option to number all lines in the print job from the first to the last, or to restart numbering with each file or with each routine.

Keep in mind that you cannot choose to number lines continuously and also choose Incremental Processing from the General options group. The Incremental Processing option is intended for those who want to maintain a printout of a project's code while it is undergoing development or maintenance. When you select this option, Code.Print prints only those files that have changed since the last printout. In this case, line numbers for the modified sections would not be in sequence with those for the rest of the project.

Line Numbers

Print Line Numbers Check this box if you want Code.Print to number lines of code.

Increment Sets the increment by which numbers increase from one line to the next.

Reset On...

New File Restarts line-numbering with each new file.

New Routine Restarts line-numbering with each new routine.

Continuous Numbers lines of code continuously from the first line in the job to the last. This option will be unavailable (dimmed) if you have checked the General option for Incremental Processing.

Topic

Line Numbers

Related

Header

Fonts

Notes

Page

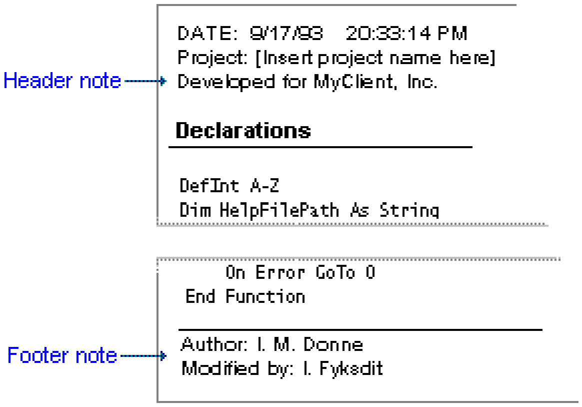
General

Default

Layout dialog: Notes

Adds lines of text to headers and footers.

Use the Notes settings group of the Layout dialog to specify text that you want printed in page headers and footers. This is in addition to whatever information you choose to print in the header and title lines by setting Header options.



```
DATE: 9/17/93   20:33:14 PM
Project: [Insert project name here]
Developed for MyClient, Inc.

Declarations

DefInt A-Z
Dim HelpFilePath As String
```

```
      On Error GoTo 0
 End Function

Author: I. M. Donne
Modified by: I. Fyksdit
```

Header Note Space is provided for three lines of text.

Footer Note Space is provided for three lines of text.

Print Notes Check this box if you want the text to be printed.

First Page Only Check this box, as well as the **Print Notes** box, if you want the text to be printed only on the first page of the printout for each file in the job.

Topic

Notes

Related

Header

Fonts

Line Numbers

Page

General

Default

Layout dialog: Page

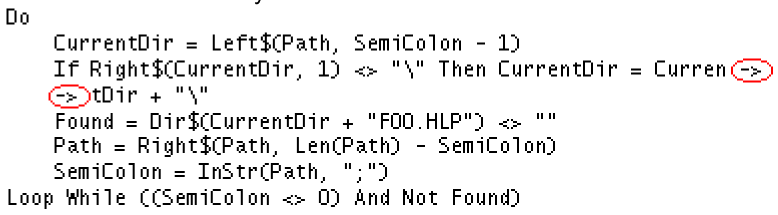
Sets margins, line-spacing, and page breaks,

Use the Page settings group of the Layout dialog to control margins, line-spacing, and page breaks for printed output.

Margins

Set the left, right, top, and bottom margins. The default unit of measurement is inches. Select centimeters, if you prefer.

If a line of code is too long to fit within the margins you specify, Code.Print will break the line and indicate a continuation with the symbol **->**.



```
Do
    CurrentDir = Left$(Path, SemiColon - 1)
    If Right$(CurrentDir, 1) <> "\" Then CurrentDir = Curren->
    ->tDir + "\"
    Found = Dir$(CurrentDir + "FOO.HLP") <> ""
    Path = Right$(Path, Len(Path) - SemiColon)
    SemiColon = InStr(Path, ";")
Loop While ((SemiColon <> 0) And Not Found)
```

Page Breaks

Page Breaks on New Routine Code.Print will insert a page break before beginning a new routine. Some routines will be printed on a single page; longer routines will span several pages. But all will begin at the top of a page. You may want to choose this option if you also choose Print Routine Name as a Header option.

Fit Routines onto Page Choose this option if you want Code.Print to insert a page break only when all of a subroutine will not fit on the current page. The difference between this option and the first one is that Fit Routines onto Page can cause several small routines to appear on one page, whereas Page Breaks on New Routine never will.

Line Spacing

You may choose between single, double, triple, or quadruple spacing.

Topic

Page

Related

Header

Fonts

Line Numbers

Notes

General

Default

Layout dialog: General

Sets options for indexing, highlighting, and incremental printing.

Use the General settings group of the Layout dialog to select general options for the printout.

Incremental Processing Prints only modified files. This option can save time and paper if you are maintaining a printout of the current state of a project during development.

If you check the Incremental Processing option, the Header option for Continuous page numbering will be unavailable (dimmed). The Line Numbers option Continuous will also be unavailable.

Print Index Prints an index at the end of the print job. Every file in the job will have an index entry showing the number of the page in the job where the text can be found. If you choose the Print Routines option in the Select Routines dialog, each routine will be indexed by name and cross-referenced by type.

Index Form Objects If you choose Print Form Information in the Select Routines dialog, every control will be indexed by name.

Bold Routine Name Highlights the names of routines by printing them in bold type.

Bold Form Objects Highlights the names of controls by printing them in bold type.

Topic

General

Related

Header

Fonts

Line Numbers

Notes

Page

Default

Select Routines (Job menu)

Layout dialog: Default

Sets current layout as the default, or resets it to the default.

If you find that you frequently set the same options with the Layout command, you may want to save those options as the default. Each time you create a new print job, it will automatically be configured with your default settings for fonts, margins, page numbering, header/footer text, and so on. Creating a user-defined default can also be useful for printing from the Visual Basic environment, since Code.Print uses the layout defaults when printing something you are currently editing in Visual Basic.

Save Current Layout Settings as User-Defined Default Stores the current layout options in a initialization (.INI) file. The next time you begin a new (untitled) print job, the job's layout will be automatically configured with your default layout. You can then change individual settings in the normal way if you want to customize a particular print job.

Reset Layout Settings to User-Defined Default Restores all layout options to the settings last stored with the Save Layout as User-Defined Default command.

Reset Layout Settings to System Default Restores all layout options to the settings they had when you first installed Code.Print.

Topic

Default

Related

Header

Fonts

Line Numbers

Notes

Page

General

Build Batch File... command (Job menu)

Creates or modifies a batch-processing file.

Use this command to create or edit a batch job consisting of one or more print jobs. Batch jobs are designed to run unattended and may be initiated either manually or automatically at a scheduled start time. For additional information on batch-processing, see Printing Several Jobs At Once.

When you choose Build Batch File from the Job menu, the Build Batch File dialog appears.

Build Batch File dialog

Batch Job Start Time If you plan to run the batch job manually, leave this blank. Otherwise, specify a time of day in twenty-four hour format. For instance, to run a job at 8:30 am, enter 08:30. To run it at 8:30 pm, enter 20:30.

> **Note** Code.Print will execute a batch job at the appointed time even when it is minimized.

In order to account for the possibility that the processor may be busy when the scheduled time arrives, Code.Print will execute a job that is as much as one hour overdue. That is, if you schedule the job for 16:00 and Code.Print doesn't get a chance to look at the clock until 16:30, the batch job will run at 16:30. However, if you schedule the job for 16:00 and it is now 17:30, the job will run tomorrow at four in the afternoon, assuming you leave Code.Print running until then.

Type Extensions of files to be displayed in the File list box. Print job files have the extension .PJF. Batch job files have the extension .PJB. You cannot include another batch job in a batch job. However, you can open a batch job, edit it, and save modified file under the same name or under a new name.

Directory Location of files to be included in the batch job. Changing the directory also changes the files listed in the File list box.

File List of files to choose from. Highlight the name of a file you want to include in the batch job and then press the Add button. (Or simply double-click on the file name.)

- If you highlight the name of a print job file and then push the Add button, the file name will appear in the Selected Files list.

- If you highlight the name of a batch job file and then push the Add button, Code.Print will ask if you want to edit the batch job.

  To edit the batch job, press Yes.

  The current configuration of the batch job will be loaded into the Build Batch File dialog. You can then add or delete files and change the start time. When you have finished editing, you can save the batch file under the same name or choose a new name.

  To cancel the operation, press No.

  You cannot add batch job files to a batch job. The name of the batch job file will not appear in the Selected Files list.

Selected Files List of files selected for inclusion in the batch job. To add a file to the list, highlight the name of the file in the Files list and then press the Add button. To remove a file from the list, highlight the file name in the Selected Files list and then press the Delete button.

Add Adds a file to the Selected Files list. Highlight the name of a file in the File list box and then press the Add button. The file name will appear in the Selected Files list.

Delete Deletes a file from the Selected Files list. Highlight the name of a file in the Selected Files list box and then press the Delete button. The file name will disappear from the Selected Files list.

Clear Clears all file names from the Selected Files list. Press this button if you want to start over.

Sort Sorts the file names in the Selected Files list in alphabetical order. The order in which files are listed determines the order in which they are printed.

Up Moves a file name up in the Selected Files list. Highlight the name you want to reposition and then press the

Up button. The file name will move up one position in the Selected Files list.

Down Moves a file name down in the Selected Files list. Highlight the name you want to reposition and then press the Down button. The file name will move down one position in the Selected Files list.

Save Batch File As dialog

When you press the Ok button in the Build Batch File dialog, the Save Batch File As dialog appears. Use this dialog to name and save the batch file.

Directories Batch job files may be in any directory.

Save File as Type Batch job files have the file extension .PJB.

File Name A batch job file may have any base file name you like, so long as it is a legal DOS file name.

After You Have Built a Batch File...

Once you have created a batch job, it will remain loaded until you exit Code.Print or start defining another job.

To	Do this
Run the job immediately	Choose Print from the File menu or push the Print button on the toolbar.
Close the job and open another file	Choose Open Job or Open Batch from the File menu.
Open a closed batch file and run it	Choose Open Batch from the File menu. If you want to run the job immediately, choose Print from the File menu or push the Print button on the toolbar. If the job has a scheduled start time, simply leave Code.Print running until that time arrives.
Open a closed batch file and edit it	Choose Build Batch File from the Job menu. Select the name of the batch file from the Files list and push the Add button. Code.Print will ask if you want to edit the batch file. Push the Yes button. Then make whatever changes you want.

Topic

Build Batch File... (Job menu)

Related

Open Batch... (File menu)

Printing Several Jobs At Once

Print Current... command (Job menu)

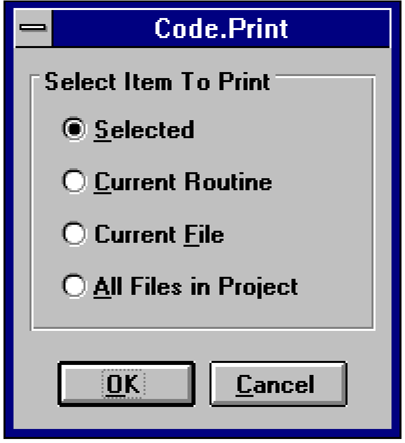
Prints a selection highlighted in the 16 bit Visual Basic editing window, or the current routine, file, or project.

When Code.Print and 16 bit Visual Basic are running at the same time, Code.Print can "see" the files you have opened in the Visual Basic environment. Thus you can use Code.Print to print code you are editing. The output will be formatted using Code.Print's current layout defaults.

> **Note** Code.Print can print the current selection or the current routine even if the Visual Basic file is stored in binary format. However, if you want to print an entire file or an entire project, you must first save the files in text format. For information on how to use Code.Print to convert Visual Basic files from binary to text format, see Converting Visual Basic Files to Text Format.

The easiest way to print something you are working on in Visual Basic is to choose Code.Print from Visual Basic's Window menu. Or you can just press Ctrl+P. In either case, a Code.Print dialog will appear, and you will have the option to print the currently highlighted selection, the current routine, the current file, or the entire project.

You can, if you prefer, switch to Code.Print itself and choose Print Current from the Job menu. The effect will be the same as choosing Code.Print from the Window menu in Visual Basic. The same dialog appears in both cases:



Select Item To Print

● Selected

○ Current Routine

○ Current File

○ All Files in Project

Code.Print will print the currently highlighted selection.

Code.Print will print the procedure or function that you are currently editing.

Code.Print will print the form or module that is currently selected in the Project window.

Code.Print will print the entire project.

Topic

Print Current... (Job menu)

Related

Converting Visual Basic Files to Text Format

Print (File menu)

Print Setup... (File menu)

Printing from the Visual Basic Environment

Setting Layout Defaults

Topic

Printer (Output menu)

Related

Print

Print Setup (File menu)

Text File (Output menu)

Before You Call...

Before obtaining technical support, please review the questions our staff will ask. This information will help expedite a solution.

- Have you read README.TXT? Your problem may have been encountered before and a workaround may be given in this file.

- Are you using a replacement for the Windows Print Manager? If so, does the problem go away when you unload the replacement and switch to Print Manager? (We know of no such conflict, but if you are experiencing one we'd like to hear about it.)

- Are you using a font or type manager? If so, does the problem go away when you disable that program?(We'd like to hear about this, too.)

- Does the problem go away if you switch to another printer? If so, what is the filename and date of the printer driver you are using?

- Does the problem occur only with a particular print job? If so, can you determine how this job is different from others where the problem does not occur?

- Can you duplicate the problem consistently? If so, retrace your steps until you come to the point where the problem occurs. Then press CTRL+SHIFT+L. This key combination will cause Code.Print to log the current state of your machine and all Code.Print options to a file named CPVB.LOG in your Code.Print directory. Open the file using the Windows Notepad or other text editor, print it out, and have it handy when you call us.

Topic

Text File (Output menu)

Related

Options... (Output menu)

Print (File menu)

Options... command (Output menu)

Sets options for text file output.

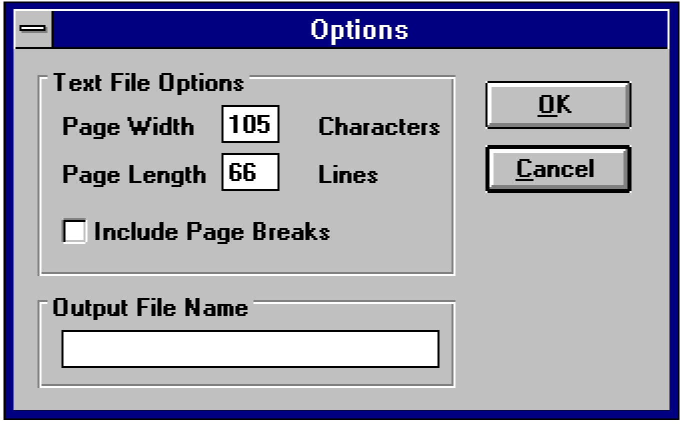
By default, Code.Print will create a file for text output with the same base file name as the print job and the extension .TXT. To specify another name, choose the Options command from the Output menu. When the Text File Options dialog appears, type the file name of your choice in the space provided.

You can also use the Options command to control the size of the (virtual) paper for text output. When the Options dialog appears, set the paper width by specifying a number of characters per line, and the length by specifying a number of lines per page. If you want Code.Print to insert control characters for page breaks, set the Include Page Breaks option on.

> **Note** The Options command sets *paper size* not *page margins*. Set the margins for text file output, just as you do for printer output, by choosing Page from the Layout dialog.

Options dialog

Click on the part of the dialog box for which you want help.



Text File Options

Page Width 105 Characters

Page Length 66 Lines

□ Include Page Breaks

Output File Name

The default is 105 characters per line. Type the line length you prefer.

The default is 66 lines per page. Type the page length you prefer.

The default is not to include control characters for page breaks. To include page breaks, click on this check box.

Supply a complete file name (including extension) for the text file. If you do not supply a name, Code.Print will create one with the base file name of the print job file and the extension .TXT. If this is a new (untitled) print job, Code.Print will name the file OUTPUT.TXT.

Topic

Options... (Output menu)

Related

Layout (Job menu)

Print Setup (File menu)

Text File (Output menu)

Batch Job File

A file where Code.Print saves the names of print job files composing a batch job and the (optional) time of day to run the job. Batch job files can be identified by the file extension .PJB.

Copyright Notice

Contacting Technical Support

Before you contact technical support, be sure you have completed your user registration. If you purchased direct from Caladonia then you are already registered. Here is how to reach us:

- **Telephone** our technical support staff at **206/780-3351** on any weekday from 8:30 am to 5:30 pm, Pacific Time. It will help if you first make note of your product version and serial number.
- **Visit** Caladonia's Web Site. http://www.caladonia.com. Leave your comments or questions for our support staff. You may also download demos, sample programs, and maintenance releases posted on the system.
- Post your questions by **Email.** Send email to **support@caladonia.com**
- **Fax** your questions to us at **206/727-8086**.
- **Mail** a letter addressed to:

  Technical Support Department
  Caladonia Systems, Inc.
  9650 NE Green Spot Pl.
  Bainbridge, WA 98110

Print Job File

A file where Code.Print saves information about page layout and the names of the Visual Basic files and routines to be printed. Print job files can be identified by the file extension .PJF.

Print Job

A single print job may consist of one routine, several, or all code in a Visual Basic project. For more information, see Creating a Print Job.

Batch Job

A batch job may include one or more print jobs. Batch jobs are designed to run unattended and may be initiated either manually or automatically at a scheduled start time. For more information, see Printing Several Jobs At Once.

About command (Help menu)

Use this command to display the version number of your copy of Code.Print Professional for Visual Basic and the copyright notice.