

## Using Your Own Help File for F1 Help

You can customize Microsoft Developer Studio to get F1 Help on keywords documented in a private or third-party Help file. Once this feature is set up, a command appears on the **Help** menu allowing you to toggle the feature on or off. When the command is active, all source-file F1 Help requests are passed to a specified external file or files rather than to the online documentation or Microsoft Development Library. WinHelp 4.0 displays the topic.

**Note** Compiling with Microsoft Help Workshop (hcx.exe) is recommended. You must use Help files compiled with hc31.exe or greater to take advantage of this feature.

### Enabling Extension Help

To enable the **Use Extension Help** feature, you need to:

1. Combine your Help index with the index for Extension Help (ExtHelp.hlp), a file supplied with Developer Studio to support Extension Help. This file is setup in the product \SharedIDE\Help subdirectory of the installation root directory.
2. Optionally, you can include your Contents file (.cnt) with the Extension Help Contents file (ExtHelp.cnt).
3. Edit your registry.

**Note** ExtHelp.cnt has been created to combine multiple Help files. If you have only one Help file, you can skip steps 1 and 2 above. However, you will need to edit your registry.

### See Also

[Combining Another Help Index with the Extension Help Index](#)

[Including Another Contents File with the Extension Help Contents File](#)

[Editing the Registry to Enable Extension Help](#)

[Notes on Using Extension Help](#)

## Combining Another Help Index with the Extension Help Index

You can combine Help file indexes by adding an **:Index** command to a base Contents (.cnt) file. The **:Index** command references the Help index you want to combine. When the base Help file opens, WinHelp automatically merges the files into a single index. Extension Help searches that single index for Help keywords.

### ► To combine your index

1. Open ExtHelp.cnt in a text editor. This file is located in the \SharedIDE\Help subdirectory of the installation root directory.
2. Add the **:Index** command to specify the name of your Help file. The **:Index** command follows the **:Title** command that identifies Extension Help. The **:Index** command takes two parameters:

*help\_title* Specifies the title of the Help file (to display in the **Topics Found** dialog box).

*HLP-filename* Specifies the name of a Help (.hlp) file.

Example:

```
:Base ExtHelp.hlp  
  
:Title Microsoft Developer Studio Extension Help  
  
:Index Your Own Help File=MyHelp.Hlp ;my own help file
```

3. Save your changes and open Extension Help to create a combined index.

Use a separate **:Index** command for each Help file whose keywords you want included in the index. You can include as many Help files as you want.

Use a space before an equal sign (=) that indicates a Help file, or a semicolon (;) that indicates a comment. Use a backslash (\) before an equal sign or semicolon that is to be treated literally.

## Including Another Contents File with the Extension Help Contents File

The **:Include** command in a Contents file specifies the name of another Contents file to be included in the current Contents file. When a Help file opens with an **:Include** command as part of its Contents file, the contents of the other file are listed in the **Contents** tab of the **Help Topics** dialog box. You may want this for navigational purposes; it is unnecessary for F1 Help support.

### ► To include your contents file

1. Open ExtHelp.cnt in a text editor. This file is located in the \SharedIDE\Help subdirectory of the installation root directory.
2. Add the **:Include** command to specify the name of your Help file. This command can be placed anywhere in a Contents file.

The **:Include** command takes a single parameter:

*cnt-filename* Specifies the name of the Contents file that you want to insert in the current Contents file.

Example:

```
:Include MyFile.cnt ;this is my own contents file
```

3. Save your changes and open Extension Help to include your Contents file.

You can include as many Contents files as you want.

If WinHelp cannot find the file specified with the **:Include** command when the user opens the Help file, no error message is displayed to the user. If WinHelp does find the specified file, the topics in the included Contents file are added to the **Contents** tab at the position where the **:Include** command exists in the Contents file.

The **:Include** command is used typically to merge Contents files that are being managed by different authors.

If the parent Contents file has a **:Base** command that specifies a default window, and you want the included Contents file to have the same default window, the included Contents file must also contain a **:Base** command specifying the same window.

Use a space before a semicolon (;) that indicates a comment. Use a backslash (\) before a semicolon or equal sign (=) that is to be treated literally.

**Note** For more information on creating a Contents file, see the Microsoft Help Workshop documentation (hcw.hlp).

## Editing the Registry to Enable Extension Help

You must edit the registry to add a command to the **Help** menu to toggle the extension help feature on and off. When the command is active, all source-file F1 Help requests are passed to a specified external file or files rather than to the online documentation provided with Microsoft Developer Studio products. WinHelp 4.0 displays the topic.

### ► To edit your registry

1. Open the Registry editor to the Extension key under:

```
HKEY_CURRENT_USER/Software/Microsoft/DevStudio/5.0/Help
```

There are three values under Extension key:

**Enable** Adds the **Use Extension Help** command to the **Help** menu. The default value is 0.

**Filename** Lists the path to the ExtHelp Help file. The default value is the following filename and path, which is a subdirectory of the installation root directory:

```
...\SharedIDE\Help\ExtHelp.hlp
```

**State** Holds a value to indicate whether **Use Extension Help** is on or off. The default value is 0, indicating off.

2. Double-click the Enable value. In the **Value Data** box, type in the value 1.
3. If you are specifying only one file, double-click the Filename value. List the path to the file.
4. Restart Developer Studio to read the new registry value or values. The **Use Extension Help** command is now added to the **Help** menu.
5. On the **Help** menu in Developer Studio, click **Use Extension Help**. A checkmark appears next to the command and the feature is enabled.

### ► To toggle Use Extension Help on and off

On the **Help** menu, click **Use Extension Help**. A checkmark appears next to the menu item when the command is on.

**Tip** You can assign a keyboard shortcut to this command. The command is listed as **HelpToggleExtensionHelp**. To display the appropriate command list, select **Help** in the **Category** box on the **Keyboard** tab in the **Customize** dialog box (**Tools** menu, **Customize** command).

## Notes on Using Extension Help

1. If you have combined the index of a Help file compiled with hc31.exe, the WinHelp 4.0 **Help Topics** dialog box opens without a listing in the **Contents** tab. A **Contents** tab is not needed for F1 Help. However, you can create a Contents file for any Windows 3.1 or Windows NT 3.5 (or earlier) Help file, then combine it with ExtHelp.cnt. For more information on creating a Contents file, see the Microsoft Help Workshop documentation (hcx.hlp).
2. When the development environment parses a keyword in a source file, it looks for an exact match to the keyword in the WinHelp file index. If WinHelp cannot exactly match the keyword, it opens the WinHelp 4.0 **Index** tab to the closest match. The WinHelp index is created from K footnotes in .rft topics. For the best results, include the bare keyword in the topic K footnote, i.e., "MyMember", rather than "the MyMember function." When a parsed keyword is exactly matched in the Help index, the Help Topic window opens rather than the **Index** tab.
3. When the **Use Extension Help** command is toggled on, the F1 Help searches external Help exclusively. This means, for example, that F1 Help on errors searches external Help, but not the online documentation provided with Developer Studio products. Toggle off the **Use Extension Help** command to search the online documentation supplied with the product. Assigning a keyboard shortcut to the **HelpToggleExtensionHelp** command is a convenient way to toggle this command on and off.
4. Third-party setup programs can automatically combine indexes and content files with the default ExtHelp files.
5. The development environment will automatically restore registry information to the default values if the DevStudio key or Help key is deleted. Repeat the steps to edit the registry to enable the Extension Help again.
6. If ExtHelp.hlp or ExtHelp.cnt is deleted, you can create your own as long as the files have the same name as ExtHelp.hlp or ExtHelp.cnt. You can also edit the Filename value under the Extension key to provide alternative Help and Contents files. However, this is not recommended since third-party tools may try to combine indexes with the default ExtHelp files.

