The **DoWindow** operation for changing aspects of the graph window.

DisplayHelpTopic

DisplayHelpTopic [**/K=*k* /Z**] ***TopicString***

The DisplayHelpTopic operation displays a help topic as if a help link had been clicked in an Igor help file.

Parameters

TopicString is string expression containing the topic. It may be in one of three forms: <topic name>, <subtopic name>, <topic name>[<subtopic name>]. These forms are illustrated by the examples.

Make sure that your topic string is specific to minimize the likelihood that Igor will find the topic in a help file other than the one you intended. To avoid this problem, it is best to use the <topic name>[<subtopic name>] form if possible.

Flags

/K=*k* Determines when the help file is closed.

- *k*=0: Leaves the help file open indefinitely (default). Use this if the help topic may be of interest in any experiment.
- *k*=1: If the found topic is in a closed help file, the help file closes with the current experiment. Use this if the help topic is tightly associated with the current experiment.

/Z Ignore errors. If /Z is used, DisplayHelpTopic sets V_flag to 0 if the help topic was found or to a nonzero error code if it was not found. V_flag is set only when /Z is used.

Details

DisplayHelpTopic first searches for the specified topic in the open help files. If the topic is not found, it then searches all help files in the Igor Pro folder and subfolders.

If the topic is still not found, it then searches all help files in the current experiment's home folder, but not in subfolders. This puts a help file that is specific to a particular experiment in the experiment's home folder.

If the topic is still not found and if DisplayHelpTopic was called from a procedure and if the procedure resides in a stand-alone file on disk (i.e., it is not in the built-in procedure window or in a packed procedure file), Igor then searches all help files in the procedure file's folder, but not in subfolders. This puts a help file that is specific to a particular set of procedures in the same folder as the procedure file.

If Igor finds the topic, it displays it. If Igor can not find the topic, it displays an error message, unless /Z is used.

Examples

```
// This example uses the topic only.
DisplayHelpTopic "Modifying Traces"

// This example uses the subtopic only.
DisplayHelpTopic "Markers"

// This example uses the topic[subtopic] form.
DisplayHelpTopic "Modifying Traces[Markers]"
```

See Also

Chapter II-1, **Getting Help** for information about Igor help files and formats.

DisplayProcedure

DisplayProcedure [**flags**] [***functionOrMacroNameStr***]

The DisplayProcedure operation displays the named function, macro or line by bringing the procedure window it is defined in to the front with the function, macro or line highlighted.

Parameters

functionOrMacroNameStr is a string expression containing the name of the function or macro to display. If you omit *functionOrMacroNameStr* then you must use /W or /L.

functionOrMacroNameStr may be a simple name or may include independent module and/or module name prefixes to display static functions.

If you use /L to display a particular line then you must omit *functionOrMacroNameStr*.

To display a procedure window without changing its scrolling or selection, use /W and omit *functionOrMacroNameStr*.

Flags

/B=*winTitleOrName*	Brings up the procedure window just behind the window with this name or title.
/L=*lineNum*	If /W is specified, *lineNum* is a zero-based line number in the specified window. If /W is not specified, *lineNum* is a "global" line number. Each procedure window line has a unique global line number as if all of the procedure files were concatenated into one big file. The order of concatenation of files can change when procedures are recompiled. If you use /L then you must omit *functionOrMacroNameStr*.
/W=*procWinTitle*	Searches in the procedure window with this title. *procWinTitle* is a name, not a string, so you construct /W like this: `/W=$"New Polar Graph.ipf"` If you omit /W, DisplayProcedure searches all open (nonindependent module) procedure windows.

Details

If a procedure window has syntax errors that prevent Igor from determining where functions and macros start and end, then DisplayProcedure may not be able to locate the procedure.

winTitleOrName is not a string; it is a name. To position the found procedure window behind a window whose title has a space in the name, use the $ operator as in the second example, below.

If *winTitleOrName* does not match any window, then the found procedure window is placed behind the top target window.

lineNum is a zero-based line number: 0 is the first line of the window. Because each line of a procedure window is a paragraph, line numbers and paragraph numbers are the same. You can use the Procedure→Info menu item to show a selection's starting and ending paragraph/line number.

procWinTitle is also a name. Use `/W=$"New Polar Graph.ipf"` to search for the function or macro in only that procedure file.

Don't specify both *functionOrMacroNameStr* and /L=*lineNum* as this is ambiguous and not allowed.

Advanced Details

If `SetIgorOption IndependentModuleDev=1`, *procWinTitle* can also be a title followed by a space and, in brackets, an independent module name. In such cases searches for the function or macro are in the specified procedure window and independent module. (See **Independent Modules** on page IV-238 for independent module details.)

For example, if any procedure file contains these statements:

```
#pragma IndependentModule=myIM
#include <Axis Utilities>
```

The command

```
DisplayProcedure/W=$"Axis Utilities.ipf [myIM]" "HVAxisList"
```

opens the procedure window that contains the `HVAxisList` function, which is in the Axis Utilities.ipf file and the independent module `myIM`. The command uses the `$""` syntax because space and bracket characters interfere with command parsing.

Similarly, if `SetIgorOption IndependentModuleDev=1` then *functionOrMacroNameStr* may also contain an independent module prefix followed by the # character. The preceding command can be rewritten as:

```
DisplayProcedure/W=$"Axis Utilities.ipf" "myIM#HVAxisList"
```

or more simply

```
DisplayProcedure "myIM#HVAxisList"
```