Details

index starts from zero, and returns the top-most window matching the parameters.

The window names are ordered in window-stacking order, as returned by WinList.

`DoWindow/B` moves the window to the back and changes the index needed to retrieve its name to the greatest index that returns any name.

Hiding or showing a window (with `SetWindow hide=1` or `Notebook visible=0` or by manual means) does not affect the index associated with the window.

windowTypes is a bitwise parameter:

1:	Graphs.
2:	Tables.
4:	Layouts.
16:	Notebooks.
64:	Panels.
128:	Procedure windows.
4096:	XOP target windows.
16384:	Camera windows in Igor Pro 7.00 or later
65536:	Gizmo windows in Igor Pro 7.00 or later

See **Setting Bit Parameters** on page IV-12 for details about bit settings.

Examples

```
Print WinName(0,1)          // Prints the name of the top graph.
Print WinName(0,3)          // Prints the name of the top graph or table.
String win=WinName(0,1)     // The name of the top visible graph.
SetWindow $win hide=1       // Hide the graph (it may already be hidden).
Print WinName(0,1)          // Prints the name of the now-hidden graph.
Print WinName(0,1,1)        // Prints the name of the top visible graph.
Print WinName(0,64,1,1)     // Name of the top visible NewPanel/FLT=1 window.
```

See Also

WinList, **DoWindow** (/F and /B flags), **SetWindow** (hide keyword), **Notebook (Miscellaneous)** (visible keyword), **NewPanel** (/FLT flag).

WinRecreation

WinRecreation(*winStr*, *options*)

The WinRecreation function returns a string containing the window recreation macro (or style macro) for the named window.

Parameters

winStr is the name of a graph, table, page layout, panel, notebook, Gizmo, camera, or XOP target window or the title of a procedure window or help file. If *winStr* is "" and options is 0 or 1, information for the top graph, table, page layout, panel, notebook, or XOP target window is returned.

As of Igor Pro 7.00, *winStr* may be a subwindow path. The returned recreation macro is generated as if the subwindow were extracted from its host as a standalone window. See **Subwindow Syntax** on page III-92 for details on forming the subwindow path.

The meaning of *options* depends on the type of window as described in the following sections.

Target Window Details

Target windows include graphs, tables, page layouts, panels, notebooks, and XOP target windows.

If *options* is 0, WinRecreation returns the window recreation macro.

If *options* is 1, WinRecreation returns the style macro or an empty string if the window does not support style macros.

Graphs Details

If *options* is 2, WinRecreation returns a recreation macro in which all occurrences of wave names are replaced with an ID number having the form ##<number>## (for instance, ##25##). These ID numbers can be found easily using the **strsearch** function. This is intended for applications that need to alter the recreation macro by replacing wave names with something else, usually other wave names. The ID numbers are the same as those returned by the **GetWindow** operation with the wavelist keyword.

Graphs and Panels Details

If *options* is 4, WinRecreation returns the window recreation macro without the default behavior of causing the graph to revert to "normal" mode (as if the GraphNormal operation had been called). This allows the use of WinRecreation when a graph or panel is in drawing tools mode without exiting that mode. For windows other than graphs or panels, this is equivalent to an *options* value of 0.

Notebooks Details

If *options* is -1, WinRecreation returns the same text that the Generate Commands menu item would generate with the Selected paragraphs radio button selected and all the checkboxes selected (includes text commands).

If *options* is 0, WinRecreation returns the same text that the Generate Commands menu item would generate with the Entire document radio button selected and all the checkboxes *except* "Generate text commands" selected).

If *options* is 1, WinRecreation returns the same text that the Generate Commands menu item would generate with the Entire document radio button selected and all the checkboxes selected (includes text commands).

Regardless of the value of *options* the text returned by WinRecreation for notebook always ends with 5 lines of file-related information formatted as comments:

```
// File Name: MyNotebook.txt
// Path: "Macintosh HD:Desktop Folder:"
// Symbolic Path: home
// Selection Start: paragraph 100, position 31
// Selection End: paragraph 100, position 31
```

Help Windows Details

WinRecreation returns the same 5 lines of file-related information as described above for notebooks.

Set *options* to -3 to ensure that *winStr* is interpreted as a help window title (help windows have only titles, not window names).

Procedures Details

WinRecreation returns the same 5 lines of file-related information as described above for notebooks.

Set *options* to -2 to ensure that *winStr* is interpreted as a procedure window title (procedure windows have only titles, not window names).

If `SetIgorOption IndependentModuleDev=1` is in effect, *winStr* can also be a procedure window title followed by a space and, in brackets, an independent module name. In such cases WinRecreation returns text from or information about the specified procedure file which is part of that independent module. (See **Independent Modules** on page IV-238 for independent module details.)

For example, in an experiment containing:

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

code like this:

```
String text=WinRecreation("Axis Utilities.ipf [myIM]",-2)
```

will return the file-related information for the Axis Utilities.ipf procedure window, which is normally a hidden part of the myIM independent module.

To get the text content of a procedure window, use the **ProcedureText** function.

Examples

```
WinRecreation("Graph0",0)     // Returns recreation macro for Graph0.

WinRecreation("",1)                      // Style macro for top window.

String win= WinName(0,16,1)                 // top visible notebook
String str= WinRecreation(str,-1)           // Selected Text commands
Variable line= itemsInList(str,"\r")-5      // First file info line
Print StringFromList(line, str,"\r")        // Print File Name:
```