```
// Function example
Function Test()
   Make/O/N=(2,2) data= 0
   FilterIIR/COEF=data/LO=999/Z data    // Purposely wrong /LO value
   Print GetErrMessage(V_Flag,3)        // Substitution assuming user-defined function
End
// Executing Test() prints: "expected /LO frequency between 0 and 0.5"

// Multiple error example
// Because of the first error, an assignment to a null wave reference,
// the substitution information for the FilterIIR operation is not available.
Function MultipleErrors()
   Make/O/N=(2,2) data= 0
   WAVE ww = $""

   // Generates error because ww is not valid
   ww = 0

   // Generates another error because of purposely wrong /LO value
   FilterIIR/COEF=data/LO=999/Z data

   Print GetErrMessage(V_Flag,3)        // Substitution assuming user-defined function
End
// Executing MultipleErrors() prints:
// "expected  between  and "         // Wave error masks /LO error reporting
```

See Also

Flow Control for Aborts on page IV-48, **GetRTErrMessage**, **GetRTError**

GetFileFolderInfo

GetFileFolderInfo [***flags***][***fileOrFolderNameStr***]

The GetFileFolderInfo operation returns information about a file or folder.

Parameters

fileOrFolderNameStr specifies the file (or folder) for which information is returned. It is optional if /P=*pathName* and /D are specified, in which case information about the directory associated with *pathName* is returned.

If you use a full or partial path for *fileOrFolderNameStr*, see **Path Separators** on page III-451 for details on forming the path.

Folder paths should not end with single Path Separators. See the **MoveFolder Details** section.

If Igor can not determine the location of the file from *fileOrFolderNameStr* and *pathName*, it displays a dialog allowing you to specify the file to be examined. Use /D to select a folder.

Flags

/D	Uses the Select Folder dialog rather than Open File dialog when *pathName* and *fileOrFolderNameStr* do not specify an existing file or folder.
/Omit	
/P=*pathName*	Specifies the folder to look in for the file. *pathName* is the name of an existing symbolic path.
/Q	No information printed to the history area.
/UTC[=*u*]	If you include /UTC or /UTC=1, GetFileFolderInfo returns creation and modification dates in UTC (coordinated universal time). If you omit /UTC or specify /UTC=0, GetFileFolderInfo returns creation and modification dates in local time. The default, used if you omit /UTC, is local time. The /UTC flag was added in Igor Pro 9.00.

/Z[=*z*] Prevents procedure execution from aborting if GetFileFolderInfo tries to get information about a file or folder that does not exist. Use /Z if you want to handle this case in your procedures rather than having execution abort.

- /Z=0: Same as no /Z.
- /Z=1: Used for getting information for a file or folder only if it exists. /Z alone has the same effect as /Z=1.
- /Z=2: Used for getting information for a file or folder if it exists and displaying a dialog if it does not exist.

Output Variables

GetFileFolderInfo returns information in the following output variables:

`V_flag` 0: File or folder was found.
-1: User cancelled the Open File dialog.
Other: An error occurred, such as the specified file or folder does not exist.

`S_path` Full file system path to the specified file or folder using Macintosh path syntax.

`V_isFile` 1: *fileOrFolderNameStr* is a file.

`V_isFolder` 1: *fileOrFolderNameStr* is a folder.

`V_isInvisible` 1: File is invisible (*Macintosh*) or Hidden (*Windows*).

`V_isReadOnly` Set if the file is locked (*Macintosh*) or is read-only (*Windows*).

On Macintosh, V_isReadOnly is either 0 (unlocked) or 1 (locked). To set this manually, display the Finder Info window for the file and then check or uncheck the "Locked" checkbox.

On Windows, V_isReadOnly is either 0 (unlocked) or 1 (locked). To set this manually, display the Properties window for the file and then check or uncheck the "Read-only" checkbox.

On both Macintosh and Windows, V_isReadOnly tells you only about the property set in the Finder or Windows desktop. It does not tell you if you have write permission for the file or for the folder containing the file. If your goal is to determine if you can write to the file, the only way to do that is to try to write to it and catch any resulting error.

`V_creationDate` Number of seconds since midnight on January 1, 1904 when the file or folder was first created in local time or UTC depending on the /UTC flag. Use **Secs2Date** to format the date as text.

`V_modificationDate` Number of seconds since midnight on January 1, 1904 when the file or folder was last modified in local time or UTC depending on the /UTC flag. Use **Secs2Date** to format the date as text.

`V_isAliasShortcut`

1: File is an alias (*Macintosh*) or a shortcut (*Windows*) and S_aliasPath is also set.

If *fileOrFolderNameStr* refers to a file (not a folder), GetFileFolderInfo returns additional information in the following variables:

`S_aliasPath` If the specified file is an alias or shortcut, S_aliasPath is the full path to the target of the specified file. Otherwise it is "".

S_aliasPath uses Macintosh path syntax. When the source is a folder, it ends with a ":" character.

`V_isStationery` 1: The stationery bit is set (*Macintosh*) or (*Windows*) the file type is one of the stationery file types (.pxt, .uxt, .ift).

S_fileType	Four-character file type code, such as 'TEXT' or 'IGsU' (packed experiment). On Windows, these codes are fabricated by translating from the equivalent file name extensions, such as .txt and .pxp.
S_creator	Four-character creator code, such as 'IGR0' (Igor Pro creator code). On Windows, S_creator is set to 'IGR0' if the file name extensions is one of those registered to Igor Pro, such as .pxp or .bwav (but not .txt). For other registered extensions, S_creator is set to the full file path of the registered application. Otherwise it is set to "".
V_logEOF	Number of bytes in the file data fork. For other forks, use **Open**/F and **FStatus**.
V_version	Version number of the file. On Macintosh, this is the value in the vers(1) resource. On Windows, a file version such as 3.10.2.1 is returned as 4.021: use S_fileVersion to avoid the problem of the second digit overflowing into the first digit. "0": File version can't be determined, or the file can't be examined because it is already open.
S_fileVersion	The file version as a string. On Macintosh, this is just a string representation of V_Version. On Windows, a file version such as 3.10.2.1 is returned as "3.10.2.1". "0": (*Macintosh*) file version can't be determined. "0.0.0.0": (*Windows*) file version can't be determined.

Details

You can change some of the file information by using **SetFileFolderInfo**.

On Windows shortcuts have ".lnk" file name extensions that are hidden on the desktop. Prior to Igor Pro 9, *fileOrFolderNameStr* was required to include the ".lnk" extension. For consistency with operations such as NewPath and OpenNotebook, in Igor Pro 9.00 and later, it is optional. When *fileOrFolderNameStr* refers to a shortcut, the S_path output variable includes the ".lnk" extension.

Examples

Print the modification date of a file:

```
GetFileFolderInfo/Z "Macintosh HD:folder:afile.txt"
if( V_Flag == 0 && V_isFile )                    // file exists
   Print Secs2Date(V_modificationDate,0), Secs2Time(V_modificationDate,0)
endif
```

Determine if a folder exists (easier than creating a path with NewPath and then using PathInfo):

```
GetFileFolderInfo/Z "Macintosh HD:folder:subfolder"
if( V_Flag && V_isFolder )
   Print "Folder Exists!"
endif
```

Find the source for a shortcut or alias:

```
GetFileFolderInfo/Z "Macintosh HD:fileThatIsAlias"
if( V_Flag && V_isAliasShortcut )
   Print S_aliasPath
endif
```

See Also

The **SetFileFolderInfo**, **PathInfo**, and **FStatus** operations. The **IndexedFile**, **Secs2Date**, and **ParseFilePath** functions.