

Programmatically select multiple files in windows explorer

Asked 16 years, 4 months ago Modified 9 months ago

Viewed 23k times



I can display and select a single file in windows explorer like this:

26



```
explorer.exe /select, "c:\path\to\file.txt"
```



However, I can't work out how to select more than one file. None of the permutations of select I've tried work.



Note: I looked at these pages for docs, neither helped.

<https://support.microsoft.com/kb/314853>

<http://web.archive.org/web/20100716112458/http://www.in focellar.com:80/Win98/explorer-switches.htm>

windows

explorer

windows-shell

Share

Improve this question

Follow

edited Jun 25, 2020 at 8:33



Nfff3

429 ● 12 ● 28

asked Aug 13, 2008 at 0:35



Orion Edwards

123k ● 66 ● 245 ● 339

Can you remember which pages you searched? Both links are outdated now. – [Wolf](#) May 17, 2018 at 9:20

9 Answers

Sorted by:

Highest score (default)



This should be possible with the shell function

[SHOpenFolderAndSelectItems](#)

20

EDIT



Here is some sample code showing how to use the function in C/C++, without error checking:



```
//Directory to open
ITEMIDLIST *dir = ILCreateFromPath(_T("C:\\"));

//Items in directory to select
ITEMIDLIST *item1 =
ILCreateFromPath(_T("C:\\Program Files\\"));
ITEMIDLIST *item2 =
ILCreateFromPath(_T("C:\\Windows\\"));
const ITEMIDLIST* selection[] = {item1,item2};
UINT count = sizeof(selection) /
sizeof(ITEMIDLIST);

//Perform selection
SHOpenFolderAndSelectItems(dir, count, selection,
0);

//Free resources
ILFree(dir);
ILFree(item1);
ILFree(item2);
```

answered Jun 10, 2010 at 2:36



flashk

2,493 ● 3 ● 20 ● 31

Any more info on using this method would be helpful... there doesn't seem to be anything at pinvoke.net on it, and I'm not great with interop. – devios1 Jun 14, 2010 at 15:02

Something of note: the selection / count variables don't look like it, but they contain a 0-delimiter at the end. Without it the SHOpenFolderAndSelectItems function didn't select all requested files (that means in in case of calloc(), pass number of items + 1). – Karsten Apr 16, 2018 at 16:05

If you don't want to create a program yourself you can use a tool which I wrote using your guys answers in stack overflow: github.com/aurire/windows-explorer-files-selector . Check the source, feel free to modify, or just use a compiled exe file and install instructions. Right click on a list of files to be selected and hit select files, that's it. – Aurimas Rekštys Jun 24, 2018 at 17:33



The true way of selecting multiple files in Explorer is the next

9



Unmanaged code looks like this (compiled from China code posts with fixing its bugs)



```
static class NativeMethods
{
    [DllImport("shell32.dll", ExactSpelling =
```



```
true)]
    public static extern int
SHOpenFolderAndSelectItems(
    IntPtr pidlFolder,
    uint cidl,
    [In, MarshalAs(UnmanagedType.LPArray)]
IntPtr[] apidl,
    uint dwFlags);

    [DllImport("shell32.dll", CharSet =
CharSet.Auto)]
    public static extern IntPtr
ILCreateFromPath([MarshalAs(UnmanagedType.LPTStr)]
string pszPath);

    [ComImport]

    [InterfaceType(ComInterfaceType.InterfaceIsIUnknown)]
    [Guid("000214F9-0000-0000-C000-000000000046")]
    public interface IShellLinkW
    {
        [PreserveSig]
        int GetPath(StringBuilder pszFile, int
cch, [In, Out] ref WIN32_FIND_DATAW pfd, uint
fFlags);

        [PreserveSig]
        int GetIDList([Out] out IntPtr ppidl);

        [PreserveSig]
        int SetIDList([In] ref IntPtr pidl);

        [PreserveSig]
        int GetDescription(StringBuilder pszName,
```

Share Improve this answer

answered Aug 26, 2010 at 15:02

Follow



[Siarhei Kuchuk](#)

5,476 ● 1 ● 30 ● 31



3

There are COM Automation LateBinding IDispatch interfaces, these are easy to use from PowerShell, Visual Basic.NET and C#, some sample code:



```

$shell = New-Object -ComObject Shell.Application

function SelectFiles($filesToSelect)
{
    foreach ($fileToSelect in $filesToSelect)
    {
        foreach ($window in $shell.Windows())
        {
            foreach ($folderItem in
$window.Document.Folder.Items())
            {
                if ($folderItem.Path -eq
$fileToSelect)
                {
                    $window.Document.SelectItem($folderItem, 1 + 8)
                }
            }
        }
    }
}

```

-

Option Strict Off

Imports Microsoft.VisualBasic

Public Class ExplorerHelp

```

    Shared ShellApp As Object =
CreateObject("Shell.Application")

```

```

    Shared Sub SelectFile(filepath As String)
        For Each i In ShellApp.Windows
            For Each i2 In

```

```
i.Document.Folder.Items()  
    If i2.Path = filepath Then  
        i.Document.SelectItem(i2, 1 +  
8)  
        Exit Sub  
    End If  
Next  
Next  
End Sub  
End Class
```

<https://learn.microsoft.com/en-us/windows/win32/shell/shellfolderview-selectitem>

Share Improve this answer
Follow

answered Dec 29, 2019 at 5:51



stax76

500 ● 5 ● 14



it cannot be done through explorer.exe

2

Share Improve this answer
Follow

answered Aug 13, 2008 at 0:47



Nick Berardi

54.8k ● 15 ● 117 ● 136



how do you know? – [Paul Sumpner](#) Apr 14, 2020 at 5:26



Depending on what you actually want to accomplish you may be able to do it with [AutoHotKey](#). It is an amazing free tool for automating things you normally can't do. It

2



should come with Windows. This script will select your file and highlight the next two files below it when you hit F12.

```
F12::  
    run explorer.exe /select`, "c:\path\to\file.txt"  
    SendInput {Shift Down}{Down}{Down}{Shift Up}  
    return
```

It is also possible to just put those two middle lines in a text file and then pass it as a param to autohotkey.exe. They have an option to compile the script also, which would make it a standalone exe that you could call. Works great with a great help file.

@Orion, It is possible to use autohotkey from C#. You can make an autohotkey script into a standalone executable (about 400k) that can be launched by your C# app (just the way you are launching explorer). You can also pass it command line parameters. It does not have any runtime requirements.

[Share](#) [Improve this answer](#)

[edited Aug 21, 2008 at 16:18](#)

[Follow](#)

answered Aug 13, 2008 at 5:56



[bruceatk](#)


5,138 ● 2 ● 27 ● 36

Won't work well if the files are not after each other though?

– [Svish](#) Dec 6, 2009 at 11:56

That is why I said "depending" and "may be able to". There are many options within autohotkey including automating a

search, I just gave one example. – [bruceatk](#) Dec 8, 2009 at 19:42

- 1 I'm not sure why this is marked down. AutoHotkey is an excellent solution when you need to do something that you just can't do using an existing API. It has many safeguards that can be used to ensure that the correct program is targeted. There are many possibilities. The answer that is marked as an answer, isn't an answer. It just an obvious statement which is the reason the question was asked in the first place. I would suggest anyone marking down this answer should first look into AutoHotkey and learn what it can do. It should be built into windows. – [bruceatk](#) Jan 20, 2010 at 17:37 
-

@bruceatk agreed. Voted up. RATM. Possibly someone was being overly pedantic about how SO doesn't like it when you solve problems with library suggestions, but this type of answer (specific about context, example code and explanation of limitations and power) is totally within bounds and applicable to the problem at hand. – [Max von Hippel](#) Jul 16, 2015 at 18:01

There is now a better AHK solution that also supports selecting multiple files
github.com/FuPeiJiang/just_SHOpenFolderAndSelectItems
– [skygate](#) Feb 20 at 14:15



1



This is one of those questions where it may be good to consider what you're trying to achieve, and whether there's a better method.

To add some more context - Our company develops a C# client application, which allows users to load files and do



stuff with them, kind of like how iTunes manages your MP3 files without showing you the actual file on disk.

It's useful to select a file in the application, and do a 'Show me this file in Windows Explorer' command - this is what I'm trying to achieve, and have done so for single files.

We have a ListView which allows users to select multiple files within the application, and move/delete/etc them. It would be nice to have this 'show me this file in windows' command work for multiple selected files - at least if all the source files are in the same directory, but if it's not possible then it's not a major feature.

Share Improve this answer

edited Dec 23, 2009 at 23:57

Follow

answered Aug 17, 2008 at 20:52



Orion Edwards

123k ● 66 ● 245 ● 339

Replace 'show me this file in windows' by 'show in windows folder' ('windows' is optional here) -- problem solved. – [jfs](#)
Dec 3, 2008 at 23:55

@J.F.Sebastian sorry... could you expound on this a bit. I don't understand how this solves the OrionEdwards's problem... tia. – [Flak DiNenno](#) Nov 5, 2014 at 14:27

@FlakDiNenno: I meant that (as an inferior option) you could just open the parent folder that contains the files (without selecting the files). [flashk's answer](#) shows how to open the folder *and select the files*. – [jfs](#) Dec 1, 2014 at 17:25

ahh got it. you didn't downvote me for simply asking you to explain a bit more did you? – [Flak DiNenno](#) Dec 2, 2014 at 15:33



0



I suppose you can use `FindWindowEx` to get the `SysListView32` of Windows Explorer, then use `SendMessage` with `LVM_SETITEMSTATE` to select the items. The difficulty being to know the position of the items... Perhaps `LVM_FINDITEM` can be used for this.



Share Improve this answer

answered Oct 24, 2008 at 11:17



Follow



[PhiLho](#)

41.1k ● 6 ● 99 ● 136



0



Grr i would like to do this as well. Media Player does it when you select 2+ files and right click and do "open file location" but not exactly sure how (nor do i really feel like spending the time w/ procmon to figure it out).

Share Improve this answer

answered Dec 3, 2008 at 23:48



Follow



Brad



I found this command line tool, works great.

0

OpenFolderAndSelectItem



Tiny EXE to open folder and select a file/folder in Windows Explorer.



Supported OS

- Windows XP to Windows 11

Tiny overhead

- 3.0KB EXE for 32-bit Windows (i386)
- 4.5KB EXE for 64-bit Windows (AMD64)
- No additional DLL/redistribute/runtime needed

Easy to use

Open "C:\Windows" and select "System32" folder.

```
OpenFolderAndSelect.exe "C:\Windows\System32"
```

Open "C:\Windows\System32" and select "cmd.exe".

```
OpenFolderAndSelect.exe  
"C:\Windows\System32\cmd.exe"
```

Open "C:\Windows\System32" and select all items where name ended with ".exe".

```
OpenFolderAndSelect.exe  
"C:\Windows\System32\*.exe"
```

Open "C:" and select two folders ("Program Files" and "Windows").

```
OpenFolderAndSelect.exe C:\ "Program Files"  
Windows
```

[Share](#) [Improve this answer](#)

answered Mar 3 at 4:38

[Follow](#)



[skygate](#)

373 ● 2 ● 12
