



Windows PowerShell Get-Help on Cmdlet 'Invoke-Item'

PS:\>Get-HELP Invoke-Item -Full

NAME

Invoke-Item

SYNOPSIS

Performs the default action on the specified item.

SYNTAX

```
Invoke-Item [-Credential <System.Management.Automation.PSCredential>] [-Exclude <System.String[]>] [-Filter  
<System.String>] [-Include <System.String[]>] -LiteralPath  
<System.String[]> [-UseTransaction] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Invoke-Item [-Path] <System.String[]> [-Credential <System.Management.Automation.PSCredential>] [-Exclude  
<System.String[]>] [-Filter <System.String>] [-Include  
<System.String[]>] [-UseTransaction] [-Confirm] [-WhatIf] [<CommonParameters>]
```

DESCRIPTION

The 'Invoke-Item' cmdlet performs the default action on the specified item. For example, it runs an executable file or opens a document file in the application

associated with the document file type. The default action depends on the type of item and is determined by the PowerShell provider that provides access to the data.

PARAMETERS

-Credential <System.Management.Automation.PSCredential>

> [!NOTE] > This parameter is not supported by any providers installed with PowerShell. > To impersonate another user, or elevate your credentials when running

this cmdlet, > use Invoke-Command (../Microsoft.PowerShell.Core/Invoke-Command.md).

Required? false

Position? named

Default value Current user

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Exclude <System.String[]>

Specifies, as a string array, an item or items that this cmdlet excludes in the operation. The value of this parameter qualifies the Path parameter. Enter a path

element or pattern, such as *.txt. Wildcard characters are permitted. The Exclude * parameter is effective only when the command includes the contents of an

item, such as C:\Windows*, where the wildcard character specifies the contents of the C:\Windows directory.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? true

-Filter <System.String>

Specifies a filter to qualify the Path parameter. The FileSystem (../Microsoft.PowerShell.Core/About/about_FileSystem_Provider.md)provider is the only installed

PowerShell provider that supports the use of filters. You can find the syntax for the FileSystem filter language in

about_Wildcards

(../Microsoft.PowerShell.Core/About/about_Wildcards.md). Filters are more efficient than other parameters, because the provider applies them when the cmdlet gets the objects rather than having PowerShell filter the objects after they are retrieved.

| | |
|-----------------------------|-------|
| Required? | false |
| Position? | named |
| Default value | None |
| Accept pipeline input? | False |
| Accept wildcard characters? | true |

-Include <System.String[]>

Specifies, as a string array, an item or items that this cmdlet includes in the operation. The value of this parameter qualifies the Path parameter. Enter a path

element or pattern, such as `"*.txt"`. Wildcard characters are permitted. The Include `*` parameter is effective only when the command includes the contents of an

item, such as ``C:\Windows*``, where the wildcard character specifies the contents of the ``C:\Windows`` directory.

| | |
|-----------------------------|-------|
| Required? | false |
| Position? | named |
| Default value | None |
| Accept pipeline input? | False |
| Accept wildcard characters? | true |

-LiteralPath <System.String[]>

Specifies a path to one or more locations. The value of LiteralPath is used exactly as it is typed. No characters are interpreted as wildcards. If the path

includes escape characters, enclose it in single quotation marks. Single quotation marks tell PowerShell not to interpret any characters as escape sequences.

For more information, see [about_Quoting_Rules](#) (../Microsoft.Powershell.Core/About/about_Quoting_Rules.md).

| | |
|-----------|------|
| Required? | true |
|-----------|------|

Position? named
Default value None
Accept pipeline input? True (ByPropertyName)
Accept wildcard characters? false

-Path <System.String[]>

Specifies the path to the selected item. Wildcard characters are permitted.

Required? true
Position? 0
Default value None
Accept pipeline input? True (ByPropertyName, ByValue)
Accept wildcard characters? true

-UseTransaction <System.Management.Automation.SwitchParameter>

Includes the command in the active transaction. This parameter is valid only when a transaction is in progress. For more information, see [about_transactions](#)

(../Microsoft.PowerShell.Core/About/about_Transactions.md).

Required? false
Position? named
Default value False
Accept pipeline input? False
Accept wildcard characters? false

-Confirm <System.Management.Automation.SwitchParameter>

Prompts you for confirmation before running the cmdlet.

Required? false
Position? named
Default value False
Accept pipeline input? False
Accept wildcard characters? false

-WhatIf <System.Management.Automation.SwitchParameter>

Shows what would happen if the cmdlet runs. The cmdlet is not run.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see about_CommonParameters (<https://go.microsoft.com/fwlink/?LinkID=113216>).

INPUTS

System.String

You can pipe a string that contains a path to this cmdlet.

OUTPUTS

None

This cmdlet returns no output of its own, but the items it invokes may return their own output.

NOTES

Windows PowerShell includes the following aliases for `Invoke-Item`:

- `ii`

This cmdlet is designed to work with the data exposed by any provider. To list the providers available in your session, type ``Get-PSPProvider``. For more

information, see `about_Providers` (`../Microsoft.PowerShell.Core/About/about_Providers.md`).

----- Example 1: Open a file -----

```
Invoke-Item "C:\Test\aliasApr04.doc"
```

----- Example 2: Open all files of a specific type -----

```
Invoke-Item "C:\Users\User1\Documents\*.xls"
```

RELATED LINKS

Online

Version:

https://learn.microsoft.com/powershell/module/microsoft.powershell.management/invoke-item?view=powershell-5.1&WT.mc_id=ps-gethelp

Clear-Item

Copy-Item

Get-Item

Move-Item

New-Item

Remove-Item

Rename-Item

Set-Item

`about_Providers`