



Full credit is given to all the above companies including the Operating System that this PDF file was generated!

Windows PowerShell Get-Help on Cmdlet 'Export-Alias'

PS:\>Get-HELP Export-Alias -Full

NAME

Export-Alias

SYNOPSIS

Exports information about currently defined aliases to a file.

SYNTAX

```
Export-Alias [[-Name] <System.String[]>] [-Append] [-As {Csv | Script}] [-Description <System.String>] [-Force]
-LiteralPath <System.String> [-NoClobber] [-PassThru]
[-Scope <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Export-Alias [-Path] <System.String> [[-Name] <System.String[]>] [-Append] [-As {Csv | Script}] [-Description
<System.String>] [-Force] [-NoClobber] [-PassThru]
[-Scope <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

DESCRIPTION

The `Export-Alias` cmdlet exports the aliases in the current session to a file. If the output file does not exist, the cmdlet will create it.

`Export-Alias` can export the aliases in a particular scope or all scopes, it can generate the data in CSV format or as a series of Set-Alias commands that you can add to a session or to a PowerShell profile.

PARAMETERS

-Append <System.Management.Automation.SwitchParameter>

Indicates that this cmdlet appends the output to the specified file, rather than overwriting the existing contents of that file.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-As <Microsoft.PowerShell.Commands.ExportAliasFormat>

Specifies the output format. CSV is the default. The acceptable values for this parameter are:

- CSV. Comma-separated value (CSV) format. - Script. Creates a `Set-Alias` command for each exported alias. If you name the output file with a .ps1 file name

extension, you can run it as a script to add the aliases to any session.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Description <System.String>

Specifies the description of the exported file. The description appears as a comment at the top of the file, following the header information.

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

-Force <System.Management.Automation.SwitchParameter>

Forces the command to run without asking for user confirmation.

Overwrites the output file, even if the read-only attribute is set on the file.

By default, `Export-Alias` overwrites files without warning, unless the read-only or hidden attribute is set or the NoClobber parameter is used in the command.

The NoClobber parameter takes precedence over the Force parameter when both are used in a command.

The Force parameter cannot force `Export-Alias` to overwrite files with the hidden attribute.

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

-LiteralPath <System.String>

Specifies the path to the output file. Unlike Path , the value of the LiteralPath parameter 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.

Required? true
Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Name <System.String[]>

Specifies the names as an array of the aliases to export. Wildcards are permitted.

By default, `Export-Alias` exports all aliases in the session or scope.

Required? false

Position? 1

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? true

-NoClobber <System.Management.Automation.SwitchParameter>

Indicates that this cmdlet prevents `Export-Alias` from overwriting any files, even if the Force parameter is used in the command.

If the NoClobber parameter is omitted, `Export-Alias` will overwrite an existing file without warning, unless the read-only attribute is set on the file.

NoClobber takes precedence over the Force parameter, which permits `Export-Alias` to overwrite a file with the read-only attribute. NoClobber does not prevent the

Append parameter from adding content to an existing file.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-PassThru <System.Management.Automation.SwitchParameter>

Returns an object representing the item with which you are working. By default, this cmdlet does not generate any

output.

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

-Path <System.String>

Specifies the path to the output file. Wildcards are permitted, but the resulting path value must resolve to a single file name.

Required? true
Position? 0
Default value None
Accept pipeline input? False
Accept wildcard characters? true

-Scope <System.String>

Specifies the scope from which the aliases should be exported. The acceptable values for this parameter are:

- Global

- Local

- Script

- A number relative to the current scope (0 through the number of scopes where 0 is the current scope and 1 is its parent)

Required? false
Position? named
Default value None
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>).

None

You can't pipe objects to this cmdlet.

OUTPUTS

None

By default, this cmdlet returns no output.

`System.Management.Automation.AliasInfo`

When you use the `PassThru` parameter, this cmdlet returns an `AliasInfo` object representing the alias.

NOTES

Windows PowerShell includes the following aliases for `Export-Alias`:

- `epal`

* You can only Export-Aliases to a file.

----- Example 1: Export an alias -----

`Export-Alias -Path "alias.csv"`

This command exports current alias information to a file named `Alias.csv` in the current directory.

Example 2: Export an alias unless the export file already exists

`Export-Alias -Path "alias.csv" -NoClobber`

This command exports the aliases in the current session to an `Alias.csv` file.

Because the `NoClobber` parameter is specified, the command will fail if an `Alias.csv` file already exists in the current directory.

directory.

----- Example 3: Append aliases to a file -----

```
Export-Alias -Path "alias.csv" -Append -Description "Appended Aliases" -Force
```

This command appends the aliases in the current session to the Alias.csv file.

The command uses the Description parameter to add a description to the comments at the top of the file.

The command also uses the Force parameter to overwrite any existing Alias.csv files, even if they have the read-only attribute.

----- Example 4: Export aliases as a script -----

```
Export-Alias -Path "alias.ps1" -As Script  
Add-Content -Path $Profile -Value (Get-Content alias.ps1)  
$S = New-PSSession -ComputerName Server01  
Invoke-Command -Session $S -FilePath .\alias.ps1
```

This example shows how to use the script file format that `Export-Alias` generates.

The first command exports the aliases in the session to the Alias.ps1 file. It uses the As parameter with a value of Script to generate a file that contains a

Set-Alias command for each alias.

The second command adds the aliases in the Alias.ps1 file to the CurrentUser-CurrentHost profile. The path to the profile is saved in the `\$Profile` variable. The

command uses the `Get-Content` cmdlet to get the aliases from the Alias.ps1 file and the `Add-Content` cmdlet to add them to the profile. For more information, see

[about_Profiles \(./Microsoft.PowerShell.Core/About/about_Profiles.md\)](#).

The third and fourth commands add the aliases in the Alias.ps1 file to a remote session on the Server01 computer. The third command uses the `New-PSSession` cmdlet to

create the session. The fourth command uses the FilePath parameter of the `Invoke-Command` cmdlet Page 8/9 to

Alias.ps1 file in the new session.

RELATED LINKS

Online

Version:

https://learn.microsoft.com/powershell/module/microsoft.powershell.utility/export-alias?view=powershell-5.1&WT.mc_id=ps-gethelp

[Get-Alias](#)

[Import-Alias](#)

[New-Alias](#)

[Set-Alias](#)