


Getting information about commands

08/27/20182 minutes to read

In this article

Displaying available command by type

The PowerShell `Get-Command` displays commands that are available in your current session. When you run the `Get-Command cmdlet`, you see something similar to the following output:

output

Copy

CommandType sion	Source	Name	Ver-
-----		----	-----
--	-----		
Cmdlet		Add-Computer	
3.1.0.0	Microsoft.PowerShell.Management		
Cmdlet		Add-Content	
3.1.0.0	Microsoft.PowerShell.Management		
Cmdlet		Add-History	
3.0.0.0	Microsoft.PowerShell.Core		
Cmdlet		Add-JobTrigger	
1.1.0.0	PSScheduledJob		
Cmdlet		Add-LocalGroupMember	
1.0.0.0	Microsoft.PowerShell.LocalAccounts		
Cmdlet		Add-Member	
3.1.0.0	Microsoft.PowerShell.Utility		
Cmdlet		Add-PSSnapin	
3.0.0.0	Microsoft.PowerShell.Core		
Cmdlet		Add-Type	

3.1.0.0 Microsoft.PowerShell.Utility

...

This output looks a lot like the Help output of **cmd.exe**: a tabular summary of internal commands. In the excerpt of the `Get-Command` command output shown above, every command shown has a `CommandType` of `Cmdlet`. A cmdlet is PowerShell's intrinsic command type. This type corresponds roughly to commands like `dir` and `cd` in **cmd.exe** or the built-in commands of Unix shells like `bash`.

The `Get-Command` cmdlet has a **Syntax** parameter that returns syntax of each cmdlet. The following example shows how to get the syntax of the `Get-Help` cmdlet:

PowerShell

Copy

`Get-Command Get-Help -Syntax`

output

Copy

```
Get-Help [[-Name] <String>] [-Path <String>]
[-Category <String[]>] [-Component
<String[]>] [-Functionality <String[]>]
[-Role <String[]>] [-Full] [-Online] [-Ver-
bose] [-Debug] [-ErrorAction <ActionPrefer-
ence>] [-WarningAction <ActionPreference>] [-
ErrorVariable <String>] [-WarningVariable
<String>] [-OutVariable <String>] [-OutBuffer
<Int32>]
```

```
Get-Help [[-Name] <String>] [-Path <String>]
[-Category <String[]>] [-Component
<String[]>] [-Functionality <String[]>]
[-Role <String[]>] [-Detailed] [-Online] [-
```

```
Verbose] [-Debug] [-ErrorAction <ActionPreference>] [-WarningAction <ActionPreference>] [-ErrorVariable <String>] [-WarningVariable <String>] [-OutVariable <String>] [-OutBuffer <Int32>]
```

```
Get-Help [[-Name] <String>] [-Path <String>] [-Category <String[]>] [-Component <String[]>] [-Functionality <String[]>] [-Role <String[]>] [-Examples] [-Online] [-Verbose] [-Debug] [-ErrorAction <ActionPreference>] [-WarningAction <ActionPreference>] [-ErrorVariable <String>] [-WarningVariable <String>] [-OutVariable <String>] [-OutBuffer <Int32>]
```

```
Get-Help [[-Name] <String>] [-Path <String>] [-Category <String[]>] [-Component <String[]>] [-Functionality <String[]>] [-Role <String[]>] [-Parameter <String>] [-Online] [-Verbose] [-Debug] [-ErrorAction <ActionPreference>] [-WarningAction <ActionPreference>] [-ErrorVariable <String>] [-WarningVariable <String>] [-OutVariable <String>] [-OutBuffer <Int32>]
```

Displaying available command by type

The `Get-Command` command lists only the cmdlets in the current session. PowerShell actually supports several other types of commands:

- Aliases
- Functions

- Scripts

External executable files, or files that have a registered file type handler, are also classified as commands.

To get all commands in the session, type:

```
PowerShell
```

Copy

```
Get-Command *
```

This list includes external commands in your search path so it can contain thousands of items. It is more useful to look at a reduced set of commands.

Note

The asterisk (*) is used for wildcard matching in PowerShell command arguments. The * means "match one or more of any characters". You can type `Get-Command a*` to find all commands that begin with the letter "a". Unlike wildcard matching in **cmd.exe**, PowerShell's wildcard will also match a period.

Use the **CommandType** parameter of `Get-Command` to get native commands of other types. `cmdlet`.

To get command aliases, which are the assigned nicknames of commands, type:

```
PowerShell
```

Copy

```
Get-Command -CommandType Alias
```

To get the functions in the current session, type:

PowerShell

Copy

`Get-Command -CommandType Function`

To display scripts in PowerShell's search path, type:

PowerShell

Copy

`Get-Command -CommandType Script`