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

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

NAME

   Set-Item

SYNOPSIS

   Changes the value of an item to the value specified in the command.

SYNTAX

      Set-Item   [[-Value]   <System.Object>]   [-Credential   <System.Management.Automation.PSCredential>]   [-Exclude
<System.String[]>] [-Filter <System.String>] [-Force]

                        [-Include   <System.String[]>]   -LiteralPath   <System.String[]>   [-Options
<System.Management.Automation.ScopedItemOptions>] [-PassThru] [-Type

   <Microsoft.Win32.RegistryValueKind>] [-UseTransaction] [-Confirm] [-WhatIf] [<CommonParameters>]

                  Set-Item   [-Path]   <System.String[]>   [[-Value]   <System.Object>]   [-Credential
<System.Management.Automation.PSCredential>] [-Exclude <System.String[]>] [-Filter

                  <System.String>]   [-Force]   [-Include   <System.String[]>]   [-Options
<System.Management.Automation.ScopedItemOptions>] [-PassThru] [-Type

   <Microsoft.Win32.RegistryValueKind>] [-UseTransaction] [-Confirm] [-WhatIf] [<CommonParameters>]

DESCRIPTION

The `Set-Item` cmdlet changes the value of an item, such as a variable or registry key, to the value specified in the command.

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

-Force <System.Management.Automation.SwitchParameter>

Forces the cmdlet to set items that cannot otherwise be changed, such as read-only alias or variables. The cmdlet cannot change constant aliases or variables.

Implementation varies from provider to provider. For more information, see about_Providers (../Microsoft.PowerShell.Core/About/about_Providers.md). Even using the

Force parameter, the cmdlet cannot override security restrictions.

Required?   false

Position?   named

Default value   False

Accept pipeline input?   False

Accept wildcard characters? false

-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

-Options <System.Management.Automation.ScopedItemOptions>

This is a dynamic parameter made available by the Alias and Function providers. For more information, see about_Alias_Provider

(../Microsoft.PowerShell.Core/About/about_Alias_Provider.md)and about_Function_Provider (../Microsoft.PowerShell.Core/About/about_Function_Provider.md).

Specifies the value of the Options property of an alias.

Valid values are:

- `None`: The alias has no constraints (default value)

- `ReadOnly`: The alias can be deleted but can't be changed without using the Force parameter - `Constant`: The alias

can't be deleted or changed

- `Private`: The alias is available only in the current scope

- `AllScope`: The alias is copied to any new scopes that are created

- `Unspecified`: The option isn't specified

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

-PassThru <System.Management.Automation.SwitchParameter>

Passes an object that represents the item to the pipeline. 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 a path of the location of the items. Wildcard characters are permitted.

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

-Type <Microsoft.Win32.RegistryValueKind>

This is a dynamic parameter made available by the Registry provider. The Registry provider and this parameter are only available on Windows.

Specifies the type of property that this cmdlet adds. The acceptable values for this parameter are:

- `String`: Specifies a null-terminated string. Used for REG_SZ values. - `ExpandString`: Specifies a null-terminated string that contains unexpanded references

to environment variables that are expanded when the value is retrieved. Used for REG_EXPAND_SZ values. - `Binary`: Specifies binary data in any form. Used for

REG_BINARY values. - `DWord`: Specifies a 32-bit binary number. Used for REG_DWORD values. - `MultiString`: Specifies an array of null-terminated strings

terminated by two null characters. Used for REG_MULTI_SZ values. - `Qword`: Specifies a 64-bit binary number. Used for REG_QWORD values. - `Unknown`: Indicates

an unsupported registry data type, such as REG_RESOURCE_LIST values.

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

-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
```

-Value <System.Object>

Specifies a new value for the item.

    Required?   false
    Position?   1
    Default value   None
    Accept pipeline input?   True (ByPropertyName, ByValue)
    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.Object

You can pipe an object that represents the new value of the item to this cmdlet.

OUTPUTS

None

By default, this cmdlet returns no output.

System.Object

When you use the PassThru parameter, this cmdlet returns an object representing the item.

NOTES

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

- `si`

- `Set-Item` is not supported by the PowerShell FileSystem provider. To change the values of items in   the file system, use the `Set-Content` cmdlet. - In the

Registry drives, `HKLM:` and `HKCU:`, `Set-Item` changes the data in the (Default) value of a   registry key.   - To create and change the names of registry keys,

use the `New-Item` and `Rename-Item` cmdlet.   - To change the names and data in registry values, use the `New-ItemProperty`, `Set-ItemProperty`, and

`Rename-ItemProperty` cmdlets. - `Set-Item` is designed to work with the data exposed by any provider.   To list the providers available in your session, type

`Get-PsProvider`.   For   more   information,   see   about_Providers (../Microsoft.PowerShell.Core/About/about_Providers.md).

------------------ Example 1: Create an alias ------------------

Set-Item -Path alias:np -Value "c:\windows\notepad.exe"

---- Example 2: Change the value of an environment variable ----

Set-Item -Path env:UserRole -Value "Administrator"

------------ Example 3: Modify your prompt function ------------

Set-Item -Path function:prompt -Value {'PS '+ (Get-Date -Format t) + " " + (Get-Location) + '> '}

------- Example 4: Set options for your prompt function -------

Set-Item -Path function:prompt -Options "AllScope,ReadOnly"

RELATED LINKS

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

Clear-Item

Copy-Item

Get-Item

Invoke-Item

Move-Item

New-Item

Remove-Item

Rename-Item

about_Providers