



python



PowerShell

FPDF Library
PDF generator

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

PowerShell Get-Help on command 'Set-Item'

PS C:\Users\wahid> Get-Help Set-Item

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`).

`-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.`

`-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.

-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.

-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.

-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](#)).

-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

`-PassThru <System.Management.Automation.SwitchParameter>`

Passes an object that represents the item to the pipeline. By default, this cmdlet does not generate any output.

`-Path <System.String[]>`

Specifies a path of the location of the items. Wildcard characters are permitted.

`-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.

-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](#)).

-Value <System.Object>

Specifies a new value for the item.

-Confirm <System.Management.Automation.SwitchParameter>

Prompts you for confirmation before running the cmdlet.

-WhatIf <System.Management.Automation.SwitchParameter>

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

<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>).

----- 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"
```

REMARKS

To see the examples, type: "get-help Set-Item -examples".

For more information, type: "get-help Set-Item -detailed".

For technical information, type: "get-help Set-Item -full".

For online help, type: "get-help Set-Item -online"