



## ***Windows PowerShell Get-Help on Cmdlet 'Get-ItemProperty'***

***PS:\>Get-HELP Get-ItemProperty -Full***

### **NAME**

Get-ItemProperty

### **SYNOPSIS**

Gets the properties of a specified item.

### **SYNTAX**

Get-ItemProperty [[-Name] <System.String[]>] [-Credential <System.Management.Automation.PSCredential>] [-Exclude <System.String[]>] [-Filter <System.String>]

[-Include <System.String[]>] -LiteralPath <System.String[]> [-UseTransaction] [<CommonParameters>]

Get-ItemProperty [-Path] <System.String[]> [[-Name] <System.String[]>] [-Credential <System.Management.Automation.PSCredential>] [-Exclude <System.String[]>] [-Filter <System.String>] [-Include <System.String[]>] [-UseTransaction] [<CommonParameters>]

### **DESCRIPTION**

The `Get-ItemProperty` cmdlet gets the properties of the specified items. For example, you can use this cmdlet to get the value of the LastAccessTime property of a

file object. You can also use this cmdlet to view registry entries and their values.

## PARAMETERS

**-Credential** <System.Management.Automation.PSCredential>

> [!NOTE] > This parameter is not supported by any providers installed with Windows 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 the path to the current location of the property. 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

**-Name <System.String[]>**

Specifies the name of the property or properties to retrieve.

Required?                false  
Position?                1  
Default value            None  
Accept pipeline input?    False  
Accept wildcard characters? false

**-Path <System.String[]>**

Specifies the path to the item or items.

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

## <CommonParameters>

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

## INPUTS

System.String

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

## OUTPUTS

System.Boolean

System.String

System.DateTime

System.IO.FileInfo

System.IO.DirectoryInfo

This cmdlet returns an object for each item property that it gets. The object type depends on the object that is retrieved. For example, in a file system drive, it might return a file or folder.

## NOTES

Windows PowerShell includes the following aliases for ``Get-ItemProperty``:

- ``gp``

The ``Get-ItemProperty`` 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: Get information about a specific directory ----

```
Get-ItemProperty C:\Windows
```

----- Example 2: Get the properties of a specific file -----

```
Get-ItemProperty C:\Test\Weather.xls | Format-List
```

Example 3: Get the value name and data of a registry entry in a registry subkey

```
Get-ItemProperty -Path HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion -Name "ProgramFilesDir"
```

> [!NOTE] > This command requires that there is a PowerShell drive named ``HKLM:`` that is mapped to the `>`HKEY_LOCAL_MACHINE`` hive of the registry. > > A drive with

that name and mapping is available in PowerShell by default. Alternatively, the path > to this registry subkey can be specified by using the following alternative

path that begins with > the provider name followed by two colons: > >  
``Registry::HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion``.

Example 4: Get the value names and data of registry entries in a registry key

```
Get-ItemProperty -Path HKLM:\SOFTWARE\Microsoft\PowerShell1\PowerShellEngine
```

ApplicationBase : C:\Windows\system32\WindowsPowerShell\v1.0\

ConsoleHostAssemblyName : Microsoft.PowerShell.ConsoleHost, Version=1.0.0.0, Culture=neutral,

PublicKeyToken=31bf3856ad364e35, ProcessorArchitecture=msil

PowerShellVersion : 2.0

RuntimeVersion : v2.0.50727

CTPVersion : 5

PSCCompatibleVersion : 1.0,2.0

## RELATED LINKS

Online

Version:

<https://learn.microsoft.com/powershell/module/microsoft.powershell.management/get-itemproperty?view=powershell-5.1&W>

T.mc\_id=ps-gethelp

Clear-ItemProperty

Copy-ItemProperty

Move-ItemProperty

New-ItemProperty

Remove-ItemProperty

Rename-ItemProperty

Set-ItemProperty

about\_Providers