



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 'Get-Location'

PS C:\Users\wahid> Get-Help Get-Location

NAME

Get-Location

SYNOPSIS

Gets information about the current working location or a location stack.

SYNTAX

```
Get-Location [-PSDrive <System.String[]> [-PSProvider <System.String[]>]  
[-UseTransaction] [<CommonParameters>]
```

```
Get-Location [-Stack] [-StackName <System.String[]> [-UseTransaction]  
[<CommonParameters>]
```

DESCRIPTION

The `Get-Location` cmdlet gets an object that represents the current directory, much like the print working directory (pwd) command.

When you move between PowerShell drives, PowerShell retains your location in each drive. You can use this cmdlet to find your location in each drive.

You can use this cmdlet to get the current directory at run time and use it in functions and scripts, such as in a function that displays the current directory in the PowerShell prompt.

You can also use this cmdlet to display the locations in a location stack. For more information, see the Notes and the descriptions of the Stack and StackName parameters.

PARAMETERS

`-PSDrive <System.String[]>`

Gets the current location in the specified PowerShell drive.

For instance, if you are in the ``Cert:`` drive, you can use this parameter to find your current location in the ``C:`` drive.

`-PSProvider <System.String[]>`

Gets the current location in the drive supported by the specified PowerShell provider. If the specified provider supports more than one drive, this cmdlet returns the location on the most recently accessed drive.

For example, if you are in the ``C:`` drive, you can use this parameter to find your current location in the drives of the PowerShell Registry provider.

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

Indicates that this cmdlet displays the locations added to the current location stack. You can add locations to stacks by using the ``Push-Location`` cmdlet.

To display the locations in a different location stack, use the StackName

parameter. For information about location stacks, see the Notes (#notes).

`-StackName <System.String[]>`

Specifies, as a string array, the named location stacks. Enter one or more location stack names.

To display the locations in the current location stack, use the `Stack` parameter. To make a location stack the current location stack, use the ``Set-Location`` cmdlet.

This cmdlet cannot display the locations in the unnamed default stack unless it is the current stack.

`-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`).

`<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: Display your current drive location -----

```
PS C:\Windows> Get-Location
```

```
Path
```

```
----
```

```
C:\Windows
```

For instance, if you are in the `Windows` directory of the `C:` drive, it displays the path to that directory.

Example 2: Display your current location for different drives

```
PS C:\> Set-Location C:\Windows
```

```
PS C:\Windows> Set-Location HKLM:\Software\Microsoft
```

```
PS HKLM:\Software\Microsoft> Set-Location "HKCU:\Control Panel\Input Method"
```

```
PS HKCU:\Control Panel\Input Method> Get-Location -PSDrive C
```

Path

C:\Windows

```
PS HKCU:\Control Panel\Input Method> Get-Location -PSDrive HKLM
```

Path

HKLM:\Software\Microsoft

```
PS HKCU:\Control Panel\Input Method> Set-Location C:
```

```
PS C:\Windows> Get-Location -PSProvider Registry
```

Path

HKCU:\Control Panel\Input Method

----- Example 3: Get locations using stacks -----

```
PS C:\> Push-Location C:\Windows
```

```
PS C:\Windows>Push-Location System32
```

```
PS C:\Windows\System32>Push-Location WindowsPowerShell -StackName Stack2
```

```
C:\Windows\System32\WindowsPowerShell>Get-Location -Stack
```

Path

C:\Windows

C:\

```
C:\Windows\System32\WindowsPowerShell>Get-Location -StackName Stack2
```

Path

C:\Windows\System32

----- Example 4: Customize the PowerShell prompt -----

```
PS C:\>
```

```
function prompt { 'PowerShell: ' + (Get-Location) + '> ' }
```

```
PowerShell: C:\>
```

The function that defines the prompt includes a `Get-Location`` command, which is run whenever the prompt appears in the console.

The format of the default PowerShell prompt is defined by a special function named `prompt``. You can change the prompt in your console by creating a new function named `prompt``.

To see the current prompt function, type the following command: `Get-Content`

```
Function:\prompt`
```

REMARKS

To see the examples, type: `"get-help Get-Location -examples"`.

For more information, type: `"get-help Get-Location -detailed"`.

For technical information, type: `"get-help Get-Location -full"`.

For online help, type: `"get-help Get-Location -online"`

