



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 'Find-PackageProvider'

PS C:\Users\wahid> Get-Help Find-PackageProvider

NAME

Find-PackageProvider

SYNOPSIS

Returns a list of Package Management package providers available for installation.

SYNTAX

```
Find-PackageProvider [-Name] <System.String[]> [-AllVersions] [-Credential  
<System.Management.Automation.PSCredential>] [-Force] [-ForceBootstrap]  
[-IncludeDependencies] [-MaximumVersion <System.String>] [-MinimumVersion  
<System.String>] [-Proxy <System.Uri>] [-ProxyCredential  
<System.Management.Automation.PSCredential>] [-RequiredVersion  
<System.String>] [-Source <System.String[]>] [<CommonParameters>]
```

DESCRIPTION

The `Find-PackageProvider` cmdlet finds matching PackageManagement providers that are available in package sources registered with PowerShellGet. These are package providers available for installation with the Install-PackageProvider

cmdlet. By default, this includes modules available in the PowerShell Gallery with the PackageManagement and Provider tags.

`Find-PackageProvider` also finds matching Package Management providers that are available in the Package Management Azure Blob store. Use the bootstrapper provider to find and install them.

PARAMETERS

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

Indicates that this cmdlet returns all available versions of the package provider. By default, `Find-PackageProvider` only returns the newest available version.

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

Specifies a user account that has permission to search for package providers.

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

Forces the command to run without asking for user confirmation. Currently, this is equivalent to the ForceBootstrap parameter.

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

Indicates that this cmdlet forces Package Management to automatically install the package provider.

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

Indicates that this cmdlet includes dependencies.

`-MaximumVersion <System.String>`

Specifies the maximum allowed version of the package provider that you want to find. If you do not add this parameter, `Find-PackageProvider` finds the highest available version of the provider.

-MinimumVersion <System.String>

Specifies the minimum allowed version of the package provider that you want to find. If you do not add this parameter, `Find-PackageProvider`` finds the highest available version of the package that also satisfies any maximum specified version specified by the `MaximumVersion` parameter.

-Name <System.String[]>

Specifies one or more package provider module names, or provider names with wildcard characters. Separate multiple package names with commas.

-Proxy <System.Uri>

Specifies a proxy server for the request, rather than connecting directly to the Internet resource.

-ProxyCredential <System.Management.Automation.PSCredential>

Specifies a user account that has permission to use the proxy server that is specified by the `Proxy` parameter.

-RequiredVersion <System.String>

Specifies the exact allowed version of the package provider that you want to find. If you do not add this parameter, `Find-PackageProvider`` finds the highest available version of the provider that also satisfies any maximum version specified by the `MaximumVersion` parameter.

-Source <System.String[]>

Specifies one or more package sources. You can get a list of available package sources by using the `Get-PackageSource`` cmdlet.

<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: Find all available package providers -----

Find-PackageProvider

This command gets a list of all package providers that are available on the repositories supported by Package Management. By default, those package providers are available on the PowerShell Gallery and by using the Package Management bootstrapping application.

----- Example 2: Find all versions of a provider -----

Find-PackageProvider -Name "Nuget" -AllVersions

This command finds all versions of the package provider named Nuget.

----- Example 3: Find a provider from a specified source -----

Find-PackageProvider -Name "Gistprovider" -Source "PSGallery"

This command finds a package provider available by using a specified package source.

REMARKS

To see the examples, type: "get-help Find-PackageProvider -examples".

For more information, type: "get-help Find-PackageProvider -detailed".

For technical information, type: "get-help Find-PackageProvider -full".

For online help, type: "get-help Find-PackageProvider -online"