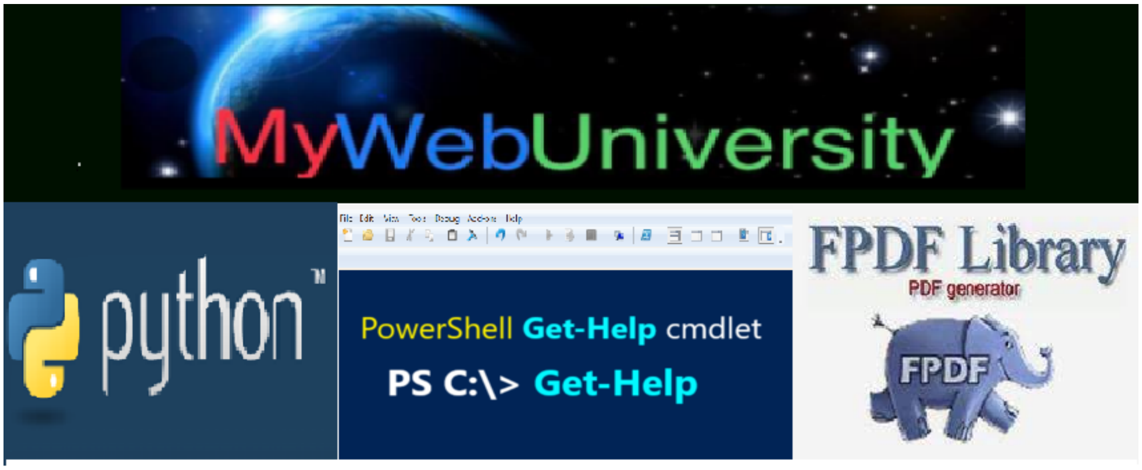

Full credit is given to all the above companies including the Operating System that this PDF file was generated!

Windows PowerShell Get-Help on Cmdlet 'New-Service'

PS:\>Get-HELP New-Service -Full

NAME

New-Service

SYNOPSIS

Creates a new Windows service.

SYNTAX

New-Service [-Name] <System.String> [-BinaryPathName] <System.String> [-Credential <System.Management.Automation.PSCredential>] [-DependsOn <System.String[]>]

[-Description <System.String>] [-DisplayName <System.String>] [-StartupType {Boot | System | Automatic | Manual | Disabled}] [-Confirm] [-WhatIf] [<CommonParameters>]

DESCRIPTION

The `New-Service` cmdlet creates a new entry for a Windows service in the registry and in the service database. A new service requires an executable file that runs

during the service.

The parameters of this cmdlet let you set the display name, description, startup type, and dependencies of the service.

PARAMETERS

-BinaryPathName <System.String>

Specifies the path of the executable file for the service. This parameter is required.

The fully qualified path to the service binary file. If the path contains a space, it must be quoted so that it is correctly interpreted. For example, `d:\my
share\myservice.exe` should be specified as `'"d:\my share\myservice.exe"'`.

The path can also include arguments for an auto-start service. For example, `'"d:\my share\myservice.exe" arg1 arg2'`. These arguments are passed to the service
entry point.

For more information, see the lpBinaryPathName parameter of CreateServiceW (/windows/win32/api/winsvc/nf-winsvc-createservicew)API.

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

-Credential <System.Management.Automation.PSCredential>

Specifies the account used by the service as the Service Logon Account (/windows/desktop/ad/about-service-logon-accounts).

Type a user name, such as User01 or Domain01\User01 , or enter a PSCredential object, such as one generated by the `Get-Credential` cmdlet. If you type a user
name, this cmdlet prompts you for a password.

Credentials are stored in a PSCredential (/dotnet/api/system.management.automation.pscredential)object and the password is stored as a SecureString

(/dotnet/api/system.security.securestring).

> [!NOTE] > For more information about SecureString data protection, see > How secure is SecureString? (/dotnet/api/system.security.securestring#how-secure-is-securestring).

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

-DependsOn <System.String[]>

Specifies the names of other services upon which the new service depends. To enter multiple service names, use a comma to separate the names.

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

-Description <System.String>

Specifies a description of the service.

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

-DisplayName <System.String>

Specifies a display name for the service.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Name <System.String>

Specifies the name of the service. This parameter is required.

Required? true

Position? 0

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-StartupType <System.ServiceProcess.ServiceStartMode>

Sets the startup type of the service. The acceptable values for this parameter are:

- Automatic - The service is started or was started by the operating system, at system start-up. If an automatically started service depends on a manually

started service, the manually started service is also started automatically at system startup. - Disabled - The service is disabled and cannot be started by a

user or application. - Manual - The service is started only manually, by a user, using the Service Control Manager, or by an application. - Boot - Indicates

that the service is a device driver started by the system loader. This value is valid only for device drivers. - System - Indicates that the service is a device

driver started by the 'IOInitSystem()' function. This value is valid only for device drivers.

The default value is Automatic .

Required? false

Position? named

Default value Automatic

```
        Accept pipeline input?       False
        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
    None
        You can't pipe objects to this cmdlet.


OUTPUTS
```

System.ServiceProcess.ServiceController

This cmdlet returns an object representing the new service.

NOTES

To run this cmdlet, start PowerShell by using the Run as administrator option.

To delete a service, use Sc.exe, or use the `Get-CimInstance` cmdlet to get the Win32_Service object that represents the service and then use the Delete method to

delete the service. The object that `Get-Service` returns does not have a delete method.

----------------- Example 1: Create a service -----------------

```
New-Service -Name "TestService" -BinaryPathName 'C:\WINDOWS\System32\svchost.exe -k netsvcs'
```

This command creates a service named TestService.

Example 2: Create a service that includes description, startup type, and display name

```
$params = @{
  Name = "TestService"
  BinaryPathName = 'C:\WINDOWS\System32\svchost.exe -k netsvcs'
  DependsOn = "NetLogon"
  DisplayName = "Test Service"
  StartupType = "Manual"
  Description = "This is a test service."
}
New-Service @params
```

This command creates a service named TestService. It uses the parameters of `New-Service` to specify a description, startup type, and display name for the new service.

--------------- Example 3: View the new service ---------------

```
Get-CimInstance -ClassName Win32_Service -Filter "Name='testservice'"
```

```
ExitCode  : 0
Name      : testservice
ProcessId : 0
StartMode : Auto
State     : Stopped
Status    : OK
```

This command uses `Get-CimInstance` to get the Win32_Service object for the new service. This object includes the start mode and the service description.

----------------- Example 4: Delete a service -----------------

```
sc.exe delete TestService
# - or -
(Get-CimInstance -Class Win32_Service -Filter "name='TestService'").delete()
```

This example shows two ways to delete the TestService service. The first command uses the delete option of `Sc.exe`. The second command uses the Delete method of the

Win32_Service objects that `Get-CimInstance` returns.

RELATED LINKS

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

Get-Service

Restart-Service

Resume-Service

Set-Service

Start-Service

Stop-Service

Suspend-Service