

40 Indispensable Powershell and Command Prompt Commands for Windows Administrators

In the realm of Windows administration, the mastery of PowerShell and Command Prompt commands is paramount. These powerful tools provide an unparalleled level of control and automation over a Windows system. This comprehensive guide presents a meticulously curated collection of 40 essential Powershell and Command Prompt commands that will empower you to manage your Windows environment with unparalleled efficiency.

PowerShell Commands

1. Get-Service

Retrieve information about Windows services, including their status, startup type, and dependencies.



40 Most Useful PowerShell and Command Prompt

Commands for Windows Administrators by Richard Potter

★★★★☆ 4.2 out of 5

Language : English

File size : 12595 KB

Text-to-Speech : Enabled

Screen Reader : Supported

Enhanced typesetting : Enabled

Print length : 68 pages

FREE

DOWNLOAD E-BOOK



```
Get-Service | Format-Table -AutoSize
```

2. Start-Service

Start a specified Windows service.

```
Start-Service "MyService"
```

3. Stop-Service

Stop a specified Windows service.

```
Stop-Service "MyService"
```

4. Restart-Service

Restart a specified Windows service.

```
Restart-Service "MyService"
```

5. Get-Process

Retrieve information about running processes, including their process ID, name, and memory usage.

```
Get-Process | Format-Table -AutoSize
```

6. Start-Process

Start a new process.

```
Start-Process "notepad.exe"
```

7. Kill-Process

Terminate a specified process.

```
Kill-Process -Id 1234
```

8. New-Item

Create a new file or directory.

```
New-Item -Path "C:\MyNewFile.txt" -Type File
```

9. Get-ChildItem

Retrieve a list of files and directories within a specified directory.

```
Get-ChildItem "C:\MyDirectory"
```

10. Remove-Item

Delete a specified file or directory.

```
Remove-Item -Path "C:\MyOldFile.txt"
```

Command Prompt Commands

11. dir

List files and directories in the current directory.

```
dir
```

12. cd

Change the current directory.

```
cd "C:\MyDirectory"
```

13. mkdir

Create a new directory.

```
mkdir "MyNewDirectory"
```

14. copy

Copy a file or directory.

```
copy "MyFile.txt" "C:\MyNewDirectory"
```

15. move

Move a file or directory.

```
move "MyFile.txt" "C:\MyNewDirectory"
```

16. del

Delete a file or directory.

```
del "MyFile.txt"
```

17. tasklist

List running processes.

```
tasklist
```

18. taskkill

Terminate a specified process.

```
taskkill /F /IM "notepad.exe"
```

19. ping

Test network connectivity to a specified host.

```
ping www.google.com
```

20. ipconfig

Display network configuration information.

```
ipconfig
```

Batch File Commands

21. @echo off

Suppress the display of commands in a batch file.

```
@echo off
```

22. set

Define or modify environment variables.

```
set MyVariable=MyValue
```

23. call

Execute another batch file.

```
call "MyOtherBatchFile.bat"
```

24. if

Execute commands conditionally.

```
if "%MyVariable%" == "MyValue" ( echo "MyVariable is set to MyValue" )
```

25. goto

Jump to a specified label in a batch file.

```
:MyLabel echo "This is my label" goto MyLabel
```

VBScript Commands

26. CreateObject

Create an instance of a COM object.

```
Set objShell = CreateObject("WScript.Shell")
```

27. Run

Execute a specified program or command.

```
objShell.Run "notepad.exe"
```

28. WScript.Sleep

Pause the script for a specified number of milliseconds.

```
WScript.Sleep 1000 'Pause for 1 second
```

29. MsgBox

Display a message box with a specified message and buttons.

```
MsgBox "Hello, world!"
```

30. FileSystemObject

Manipulate files and directories using the file system object.

```
Set objFSO = CreateObject("Scripting.FileSystemObject") objFSO.CreateFolder
```

WMI Commands

31. Get-WmiObject

Retrieve WMI objects and their properties.

```
Get-WmiObject -Class Win32_OperatingSystem
```

32. Set-WmiInstance

Modify properties of a WMI object.

```
Set-WmiInstance -Class Win32_OperatingSystem -PropertyValue @{Caption="M
```

33. Invoke-WmiMethod

Invoke a method on a WMI object.

```
Invoke-WmiMethod -Class Win32_OperatingSystem -Name Reboot
```

34. Get-CimInstance

Retrieve CIM instances and their properties.

```
Get-CimInstance -ClassName Win32_OperatingSystem
```

35. Set-CimInstance

Modify properties of a CIM instance.

```
Set-CimInstance -ClassName Win32_OperatingSystem -PropertyValue @{Captio
```

Advanced Commands

36. Get-EventLog

Retrieve events from the Windows event log.


```
Get-EventLog -LogName System
```

37. Set-EventLog

Write an event to the Windows event log.

```
Set-EventLog -LogName System -EventID 1000 -Message "My custom event"
```

38. Get-RegistryKey

Retrieve a registry key and its values.

```
Get-RegistryKey -Path "HKLM:\Software\Microsoft\Windows"
```

39.



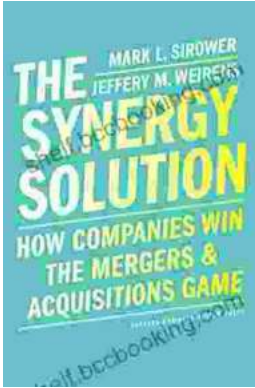
40 Most Useful PowerShell and Command Prompt

Commands for Windows Administrators by Richard Potter

★★★★☆ 4.2 out of 5

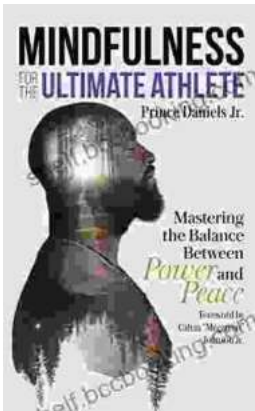
Language : English
File size : 12595 KB
Text-to-Speech : Enabled
Screen Reader : Supported
Enhanced typesetting: Enabled
Print length : 68 pages





How Companies Win the Mergers and Acquisitions Game: Unlocking the Secrets to Extraordinary Outcomes

In today's dynamic and ever-evolving business landscape, mergers and acquisitions (M&A) have become increasingly strategic for companies...



Mastering The Delicate Balance Between Power And Peace

In today's ever-evolving world, the interplay between power and peace has become increasingly complex and crucial. From personal relationships to global politics, striking the...