

All Commands :

Show-Command

Get-Command

Get-Service

Get-ComputerInfo

Get-Process

Get-Module

Get-Process -Name winlogo

Get-ExecutionPolicy -List

Get-Process | Where-Object {\$_.CPU -gt 1}

Get-Process | Where-Object {\$_.CPU -gt 1 -and \$_.WorkingSet -gt 10mb}

Get-EventLog -LogName Application -Newest 10

Test-Path -Path "C:\Documents and Settings\DavidC"

New-PSDrive -Name "Public" -PSProvider "FileSystem" -Root "\\Server01\Public"

Get-Process | Out-File E:\wiprofiledwc\processfile

Get-NetIPConfiguration | Out-File E:\wiprofiledwc\configurationfile

Get-PSDrive -PSProvider FileSystem | Out-File E:\wiprofiledwc\datafile.txt

Get-Command | Out-File E:\wiprofiledwc\datafilea.txt

Get-ChildItem | Out-File E:\wiprofiledwc\file.txt

Get-Service | Out-File E:\wiprofiledwc\servicefile.txt

Get-Service | Where status -ne Running | Out-File E:\wiprofiledwc\runningservicefile.txt

Write-Host "Hello Everyone"

Read-Host "Please Enter your name "

\$name = Read-Host "Please Enter Name "

Write-Output "Hello Lets welcome \$name in DWS Session "

Format Table :

Get-Process | Format-Table -Property Name, CPU, StartTime

Format List :

Get-Service | Format-List -Property Name, Status , DisplayName

Format Wide :

Get-ChildItem | Format-Wide -Column 3

Pipeline

Get-Process | Where-Object {\$_.CPU -gt 1 } | Sort-Object CPU -Descending | Select-Object Name , CPU

Conditional statement

\$age = 17

if(\$age -gt 18)

{

Write-Host "Eligible to vote "

}

elseif(\$age -lt 18)

{

Write-Host "Not eligible to vote "

}

Variables

\$name = "Omkar"

\$age = 20

\$isValid = \$true

Array

\$myArray = 10,56,78,"Banana" ,"Mango "

\$myArray[0]

\$myArray[-1]

\$myArray[-2]

\$myArray+="Orange"

Switch

\$day = "Wednesday"

switch (\$day){

"Monday"{Write-Host "Start of the week "}

"Saturday"{Write-Host "End of the week "}

default{Write-Host "Mid of Week "}}

String Formatting

```
$name = "Atharav "
```

```
$age = 20
```

```
" My name is {0} and I am {1} years old " -f $name , $age
```

```
'hello' -eq ' Hello '
```

```
"Hello " -eq "Hello "
```

Looping

```
for($i=1 ; $i -le 5 ; Si++)  
{  
    Write-Host "Iteration :$i"  
}
```

Execute MSI

```
Execute-MSI -Action 'Install' -Path 'Adobe_FlashPlayer_11.2.202.233_x64_EN.msi'
```

```
Execute-MSI -Action 'Install' -Path 'Adobe_FlashPlayer_11.2.202.233_x64_EN.msi' -Transform  
'Adobe_FlashPlayer_11.2.202.233_x64_EN_01.mst' -Parameters '/QN'
```

```
Execute-MSI -Action 'Uninstall' -Path '{26923b43-4d38-484f-9b9e-de460746276c}'
```

```
Execute-MSI -Action 'Patch' -Path 'Adobe_Reader_11.0.3_EN.msp'
```

```
Execute-MSI -Action 'Patch' -Path 'Adobe_Reader_11.0.3_EN.msp'
```

```
Execute-MSI -Action Install -Path $AppMSIName -SkipMSIAlreadyInstalledCheck -ContinueOnError  
$False -LogName "${AppMSIName}_MSI"
```

Execute Process:

```
Execute-Process -Path 'uninstall_flash_player_64bit.exe' -Parameters '/uninstall' -WindowStyle  
'Hidden'
```

```
Execute-Process -Path "$dirFiles\Bin\setup.exe" -Parameters '/S' -WindowStyle 'Hidden'
```

```
Execute-Process -Path 'setup.exe' -Parameters '/S' -IgnoreExitCodes '1,2'
```

```
Execute-Process -Path 'setup.exe' -Parameters "-s -f2""$configToolkitLogDir\$installName.log`""
```

```
Execute-Process -Path 'setup.exe' -Parameters "/s /v""ALLUSERS=1 /qn /L*  
""$configToolkitLogDir\$installName.log`""
```

Execute MSP

Execute-MSP -Path 'Adobe_Reader_11.0.3_EN.msp'

Execute-MSP -Path 'AcroRdr2017Upd1701130143_MUI.msp' -AddParameters 'ALLUSERS=1'

Copy File

Copy-File -Path "\$dirSupportFiles*.*)" -Destination "\$envTemp\tempfiles"

Copy-File -Path "\$dirSupportFiles\MyApp.ini" -Destination "\$envWinDir\MyApp.ini"

Execute MSUpdates

Install-MSUpdates -Directory "\$dirFiles\MSUpdates"

Set-ActiveSetup

Set-ActiveSetup -StubExePath 'C:\Users\Public\Company\ProgramUserConfig.vbs' -Arguments '/Silent' -Description 'Program User Config' -Key 'ProgramUserConfig' -Locale 'en'

Set-ActiveSetup -StubExePath "\$envWinDir\regedit.exe" -Arguments "/S \"%SystemDrive%\Program Files (x86)\PS App Deploy\PSAppDeployHKCUSettings.reg\"" -Description 'PS App Deploy Config' -Key 'PS_App_Deploy_Config' -ContinueOnError \$true

Get-RegistryKey

Get-RegistryKey -Key 'HKLM:SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{1AD147D0-BE0E-3D6C-AC11-64F6DC4163F1}'

Set INI Value

Set-IniValue -FilePath "C:\Windows\Path\to\my.ini" -Section 'MySection' -Key 'MyKey' -Value 'MyValue'

#Create/Delete/Modify Registries

Create new key

New-Item -Path "HKLM:\Software\MyCompany" -Type Directory -ErrorAction SilentlyContinue

Delete a Key

Remove-Item -Path "HKLM:\Software\MyCompany" -Recurse -Force -ErrorAction SilentlyContinue

Set a registry value

Set-ItemProperty -Path "HKLM:\Software\MyCompany\MyKey" -Name "MyValue" -Value "MyData"

Get a registry value

Get-ItemProperty -Path "HKLM:\Software\MyCompany\MyKey" -Name "MyValue"

#Start/Stop Services

Start Service

Start-Service -Name "YourServiceName"

Start-Service -Name "wuauserv"

Stop Service

Stop-Service -Name "YourServiceName"

Stop-Service -Name "wuauserv"

Stop-Service -Name "YourServiceName" -Force

Start Service with dependencies

Start-ServiceAndDependencies -Name "YourServiceName"

Start-ServiceAndDependencies -Name "wuauserv"

Stop Service with dependencies

Stop-ADTServiceAndDependencies -Name "YourServiceName"

Stop-ADTServiceAndDependencies -Name "wuauserv"