# PowerShell Basics

All the Powershell command is of the form [**Action-Object] or [Verb-Noun]**

|  |  |  |
| --- | --- | --- |
| Command | Alias | Description |
| Start-Transcript |  | Creates a record of all or part of a Windows PowerShell session to a text file. |
| Get-Command |  | Gets all commands in powershell |
| Get-Command -Noun S\* |  | Get command which contains ‘s’ |
| Get-Command -Noun service |  | Get command which contains ‘Service’ |
| Get-Service |  | One of the command |
| Get-Help |  | Gets help |
| Update-Help |  | Update help directory of powershell from online |
| Get-help Get-Service |  | Gets help of the command get-service |
| Get-help Get-Service -examples |  | Gets examples |
| Get-help Get-Service -online |  | Opens online help of the command |
| Get-Alias |  | Alias |
| Clear\_Host | cls | Clears the display in the host program |
| Get-Alias cls |  | Alias Clear-Host |
| Get-Help Get-Alias -examples |  | Get-alias Example |
| Get-Help Get-Alias -online |  | Open online Help of Get-Alias |
| Get-Process |  | Get process |
| Get-Process -Name iexplore |  | Get process with name ‘iexplore’ |
| Get-Process -Name iexplore |Get-Member |  | Opens member of object ‘iexplore’ |
| $ab=Get-Process iexplore |  | Variable assign |
| $ab.Name |  | Get-Process -Name iexplore |
| get-history | h | Get history of all the command we typed in console |
| Get-PSDrive |  | Gets drives in the current session. |
| Select-Object | select | Selects objects or object properties  Eg:  Select-Object \*  Select-Object property1,property2  Get-Process -Name iexplore | Select-Object \*  Get-Process -Name iexplore | select Id  Get-Process -ID 8880 | Select-Object StartTime |
| Where-Object | ? | Is similar to where in linq .  Selects objects from a collection based on their property values.  Eg:  ?{$\_.Free -gt 1}  Where(x=>x.Free>1)  Where (checks the current item (ie. $\_.) is greater than 1 |
| ForEach-Object | % | Performs an operation against each item in a collection of input objects. |
| "{0:N0}" -f 1000000 |  | Formats the number 1000000 to 1,000,000 with zero decimal points  <https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-powershell-1.0/ee692795(v=technet.10)?redirectedfrom=MSDN> |

* PS C:\Users\Administrator> Get-PSDrive|where-object{$\_.Free -gt 1}

Name Used (GB) Free (GB) Provider Root CurrentLocation

---- --------- --------- -------- ---- ---------------

C 52.50 197.15 FileSystem C:\ Users\Administrator

* PS C:\Users\Administrator> Get-PSDrive|where-object{$\_.Free -gt 1}|ForEach-Object{Write-Host "The free space in " $\_.Root " is :" $\_.Free -ForegroundColor "red"}

The free space in C:\ is : 211688271872

* PS C:\Users\Administrator> Get-PSDrive|where-object{$\_.Free -gt 1}|ForEach-Object{Write-Host "The free space in "$\_.Root " is :" ($\_.Free/1024/1024/1024) -ForegroundColor "red"}

Converted $\_.Free (bits) to ($\_.Free/1024/1024/1024) (in GB)

Converted $\_.Free (bits) to ($\_.Free/1gb) (in GB)

The free space in C:\ is : 197.15002822876

* PS C:\Users\Administrator> Get-PSDrive|where-object{$\_.Free -gt 1}|ForEach-Object{Write-Host "The free space in "$\_.Root "is :" ("{0:N2}" -f ($\_.Free/1gb)) -ForegroundColor "red"}

The free space in C:\ is : 197.15

**Foreach Loop Explained**

* PS C:\Users\Administrator> Get-PSDrive|where-object{$\_.Used -gt 1}|ForEach-Object{$c=0; Write-Host "This step runs Once when foreach loop is strated" -ForegroundColor "Green" }{$c=$C+1;Write-Host "This Runs for Every object in the Loop and The value of C is "-ForegroundColor "red" $c}{Write-Host "This Runs when the loop end . The value of C is " -ForegroundColor "yellow" $c}

This step runs Once when foreach loop is strated

This Runs for Every object in the Loop and The value of C is 1

This Runs for Every object in the Loop and The value of C is 2

This Runs when the loop end. The value of C is 2

This is a typical example of foreach loop which have 3 phases.

1. Initialize counter variable
2. Loop through all items
3. End

**Complex powershell Example**

* PS C:\Users\Administrator> Get-PSDrive | ?{$\_.Used -gt 1} | %{$Count = 0; Write-Host "";} { $\_.Name + ": Used: " + "{0:N2}" -f ($\_.Used/1gb) + " Free: " + "{0:N2}" -f ($\_.free/1gb) + " Total: " + "{0:N2}" -f (($\_.Used/1gb)+($\_.Free/1gb)); $Count = $Count + $\_.Free;}{Write-Host"";Write-Host "Total Free Space " ("{0:N2}" -f ($Count/1gb)) -backgroundcolor red}

C: Used: 52.55 Free: 197.10 Total: 249.66

D: Used: 4.24 Free: 0.00 Total: 4.24

Total Free Space 197.10

**Explanation of above powershell script** :

|  |  |
| --- | --- |
| 1. Get-PSDrive | Get Object |
| 1. ? - | Where-Object |
| 1. {$\_.Used -gt 1} | Item.Used greater than 1 |
| 1. % | Foreach-Object |
| 1. {$Count = 0; Write-Host "";} | Initialize counter |
| 1. { $\_.Name + ": Used: " + "{0:N2}" -f ($\_.Used/1gb) + " Free: " + "{0:N2}" -f ($\_.free/1gb) + " Total: " + "{0:N2}" -f (($\_.Used/1gb)+($\_.Free/1gb)); $Count = $Count + $\_.Free;} | Do the loop for each item |
| 1. {Write-Host"";Write-Host "Total Free Space " ("{0:N2}" -f ($Count/1gb)) -backgroundcolor red} | Final End Statement |