List of basic commands

donderdag 25 januari 2018   21:28

Command alias	Cmdlet name	Description of command
%	ForEach-Object	Performs an operation against each item in a collection of input objects.
?	Where-Object	Selects objects from a collection based on their property values.
ac	Add-Content	Appends content, such as words or data, to a file.
asnp	Add-PSSnapIn	Adds one or more Windows help Shell snap-ins to the current session.
cat	Get-Content	Gets the contents of a file.
cd	Set-Location	Sets the current working location to a specified location.
chdir	Set-Location	Sets the current working location to a specified location.
clc	Clear-Content	Deletes the contents of an item, but does not delete the item.
clear	Clear-Host	Clears the display in the host program.
clhy	Clear-History	Deletes entries from the command history.
cli	Clear-Item	Deletes the contents of an item, but does not delete the item.
clp	Clear-ItemProperty	Deletes the value of a property but does not delete the property.
cls	Clear-Host	Clears the display in the host program.
clv	Clear-Variable	Deletes the value of a variable.
cnsn	Connect-PSSession	Reconnects to disconnected sessions
compare	Compare-Object	Compares two sets of objects.
copy	Copy-Item	Copies an item from one location to another.
cp	Copy-Item	Copies an item from one location to another.
cpi	Copy-Item	Copies an item from one location to another.
cpp	Copy-ItemProperty	Copies a property and value from a specified location to another location.
curl	Invoke-WebRequest	Gets content from a webpage on the Internet.
cvpa	Convert-Path	Converts a path from a Windows PowerShell path to a Windows PowerShell provider path.
dbp	Disable-PSBreakpoint	Disables the breakpoints in the current console.
del	Remove-Item	Deletes files and folders.
diff	Compare-Object	Compares two sets of objects.
dir	Get-ChildItem	Gets the files and folders in a file system drive.
dnsn	Disconnect-PSSession	Disconnects from a session.
ebp	Enable-PSBreakpoint	Enables the breakpoints in the current console.

echo	Write-Output	Sends the specified objects to the next command in the pipeline. If the command is the last command in the pipeline, the objects are displayed in the console.
epal	Export-Alias	Exports information about currently defined aliases to a file.
epcsv	Export-Csv	Converts objects into a series of comma-separated (CSV) strings and saves the strings in a CSV file.
epsn	Export-PSSession	Imports commands from another session and saves them in a Windows PowerShell module.
erase	Remove-Item	Deletes files and folders.
etsn	Enter-PSSession	Starts an interactive session with a remote computer.
exsn	Exit-PSSession	Ends an interactive session with a remote computer.
fc	Format-Custom	Uses a customized view to format the output.
fl	Format-List	Formats the output as a list of properties in which each property appears on a new line.
foreach	ForEach-Object	Performs an operation against each item in a collection of input objects.
ft	Format-Table	Formats the output as a table.
fw	Format-Wide	Formats objects as a wide table that displays only one property of each object.
gal	Get-Alias	Gets the aliases for the current session.
gbp	Get-PSBreakpoint	Gets the breakpoints that are set in the current session.
gc	Get-Content	Gets the contents of a file.
gci	Get-ChildItem	Gets the files and folders in a file system drive.
gcm	Get-Command	Gets all commands.
gcs	Get-PSCallStack	Displays the current call stack.
gdr	Get-PSDrive	Gets drives in the current session.
ghy	Get-History	Gets a list of the commands entered during the current session.
gi	Get-Item	Gets files and folders.
gjb	Get-Job	Gets Windows PowerShell background jobs that are running in the current session.
gl	Get-Location	Gets information about the current working location or a location stack.
gm	Get-Member	Gets the properties and methods of objects.
gmo	Get-Module	Gets the modules that have been imported or that can be imported into the current session.
gp	Get-ItemProperty	Gets the properties of a specified item.
gps	Get-Process	Gets the processes that are running on the local computer or a remote computer.
group	Group-Object	Groups objects that contain the same value for specified properties.
gsn	Get-PSSession	Gets the Windows PowerShell sessions on local and remote computers.
gsnp	Get-PSSnapIn	Gets the Windows PowerShell snap-ins on the computer.
gsv	Get-Service	Gets the services on a local or remote computer.

gu	Get-Unique	Returns unique items from a sorted list.
gv	Get-Variable	Gets the variables in the current console.
gwmi	Get-WmiObject	Gets instances of Windows Management Instrumentation (WMI) classes or information about the available classes.
h	Get-History	Gets a list of the commands entered during the current session.
history	Get-History	Gets a list of the commands entered during the current session.
icm	Invoke-Command	Runs commands on local and remote computers.
iex	Invoke-Expression	Runs commands or expressions on the local computer.
ihy	Invoke-History	Runs commands from the session history.
ii	Invoke-Item	Performs the default action on the specified item.
ipal	Import-Alias	Imports an alias list from a file.
ipcsv	Import-Csv	Creates table-like custom objects from the items in a CSV file.
ipmo	Import-Module	Adds modules to the current session.
ipsn	Import-PSSession	Imports commands from another session into the current session.
irm	Invoke-RestMethod	Sends an HTTP or HTTPS request to a RESTful web service.
ise	powershell_ise.exe	Explains how to use the PowerShell_ISE.exe command-line tool.
iwmi	Invoke-WMIMethod	Calls Windows Management Instrumentation (WMI) methods.
iwr	Invoke-WebRequest	Gets content from a web page on the Internet.
kill	Stop-Process	Stops one or more running processes.
lp	Out-Printer	Sends output to a printer.
ls	Get-ChildItem	Gets the files and folders in a file system drive.
man	help	Displays information about Windows PowerShell commands and concepts.
md	mkdir	Creates a new item.
measure	Measure-Object	Calculates the numeric properties of objects, and the characters, words, and lines in string objects, such as files of text.
mi	Move-Item	Moves an item from one location to another.
mount	New-PSDrive	Creates temporary and persistent mapped network drives.
move	Move-Item	Moves an item from one location to another.
mp	Move-ItemProperty	Moves a property from one location to another.
mv	Move-Item	Moves an item from one location to another.
nal	New-Alias	Creates a new alias.
ndr	New-PSDrive	Creates temporary and persistent mapped network drives.
ni	New-Item	Creates a new item.
nmo	New-Module	Creates a new dynamic module that exists only in memory.
npssc	New-PSSessionConfig	Creates a file that defines a session configuration.

	urationFile	
nsn	New-PSSession	Creates a persistent connection to a local or remote computer.
nv	New-Variable	Creates a new variable.
ogv	Out-GridView	Sends output to an interactive table in a separate window.
oh	Out-Host	Sends output to the command line.
popd	Pop-Location	Changes the current location to the location most recently pushed to the stack. You can pop the location from the default stack or from a stack that you create by using the Push-Location cmdlet.
ps	Get-Process	Gets the processes that are running on the local computer or a remote computer.
pushd	Push-Location	Adds the current location to the top of a location stack.
pwd	Get-Location	Gets information about the current working location or a location stack.
r	Invoke-History	Runs commands from the session history.
rbp	Remove-PSBreakpoint	Deletes breakpoints from the current console.
rcjb	Receive-Job	Gets the results of the Windows PowerShell background jobs in the current session.
rcsn	Receive-PSSession	Gets results of commands in disconnected sessions.
rd	Remove-Item	Deletes files and folders.
rdr	Remove-PSDrive	Deletes temporary Windows PowerShell drives and disconnects mapped network drives.
ren	Rename-Item	Renames an item in a Windows PowerShell provider namespace.
ri	Remove-Item	Deletes files and folders.
rjb	Remove-Job	Deletes a Windows PowerShell background job.
rm	Remove-Item	Deletes files and folders.
rmdir	Remove-Item	Deletes files and folders.
rmo	Remove-Module	Removes modules from the current session.
rni	Rename-Item	Renames an item in a Windows PowerShell provider namespace.
rnp	Rename-ItemProperty	Renames a property of an item.
rp	Remove-ItemProperty	Deletes the property and its value from an item.
rsn	Remove-PSSession	Closes one or more Windows PowerShell sessions (PSSessions).
rsnp	Remove-PSSnapin	Removes Windows PowerShell snap-ins from the current session.
rujb	Resume-Job	Restarts a suspended job
rv	Remove-Variable	Deletes a variable and its value.
rvpa	Resolve-Path	Resolves the wildcard characters in a path, and displays the path contents.
rwmi	Remove-WMIObject	Deletes an instance of an existing Windows Management Instrumentation (WMI) class.

sajb	Start-Job	Starts a Windows PowerShell background job.
sal	Set-Alias	Creates or changes an alias (alternate name) for a cmdlet or other command element in the current Windows PowerShell session.
saps	Start-Process	Starts one or more processes on the local computer.
sasv	Start-Service	Starts one or more stopped services.
sbp	Set-PSBreakpoint	Sets a breakpoint on a line, command, or variable.
sc	Set-Content	Replaces the contents of a file with contents that you specify.
select	Select-Object	Selects objects or object properties.
set	Set-Variable	Sets the value of a variable. Creates the variable if one with the requested name does not exist.
shcm	Show-Command	Creates Windows PowerShell commands in a graphical command window.
si	Set-Item	Changes the value of an item to the valu

Tags Fundamentals getting started Scripting Guy! Windows PowerShell

Van <https://blogs.technet.microsoft.com/heyscriptingguy/2015/06/11/table-of-basic-powershell-commands/>

Cheat sheet

PowerShell Cheat Sheet
comparitech

Basic Commands

Cmdlet	Commands built into shell written in .NET
Functions	Commands written in PowerShell language
Parameter	Argument to a Cmdlet/Function/Script
Alias	Shortcut for a Cmdlet or Function
Scripts	Text files with .ps1 extension
Applications	Existing windows programs
Pipelines	Pass objects Get-process word | Stop-Process
Ctrl+c	Interrupt current command
Left/right	Navigate editing cursor
Ctrl+left/right	Navigate a word at a time
Home / End	End Move to start / end of line
Up/down	Move up and down through history
Insert	Toggles between insert/overwrite mode
F7	Command history in a window
Tab / Shift-Tab	Command line completion

Variables

$var = "string"	Assign variable
$a,$b = 0 or $a,$b = 'a','b'	Assign multiple variables
$a,$b = $b,$a	Flip variables
$var=[int]5	Strongly typed variable

Help

Get-Command	Get all commands
Get-Command -Module RGHS	Get all commands in RGHS module
Get-Command Get-p*	Get all commands starting with get-p
Get-help get-process	Get help for command
Get-Process | Get-Member	Get members of the object
Get-Process| format-list -properties *	Get-Process as list with all properties

Scripts

Set-ExecutionPolicy -ExecutionPolicy Bypass	Set execution policy to allow all scripts
."\\c-is-ts-91\c$\scripts\script.ps1"	Run Script.PS1 script in current scope
&"\\c-is-ts-91\c$\scripts\script.ps1"	Run Script.PS1 script in script scope
.\Script.ps1	Run Script.ps1 script in script scope
$profile	Your personal profile that runs at launch

Import, Export, Convert

Export-CliXML	Import-CliXML
ConvertTo-XML	ConvertTo-HTML
Export-CSV	Import-CSV
ConvertTo-CSV	ConvertFrom-CSV

Flow Control

```
If(){} Elseif(){} Else{}
while(){}
For($i=0; $i -lt 10; $i++){}
Foreach($file in dir C:\){$file.name}
1..10 | foreach($_)
```

Comments, Escape Characters

#Comment	Comment
<#comment#>	Multiline Comment
"`test`'"	Escape char `
`t	Tab
`n	New line
`	Line continue

Parameters

-Confirm	Prompt whether to take action
-WhatIf	Displays what command would do

Assignment, Logical, Comparison

=, +=, -=, ++,--	Assign values to variable
-and, -or, -not,!	Connect expressions / statements
-eq, -ne	Equal, not equal
-gt, -ge	Greater than, greater than or equal
-lt, -le	Less than, less than or equal
-replace	"Hi"-replace "H", "P"
-match, -notmatch	Regular expression match
-like, -notlike	Wildcard matching
-contains, -notcontains	Check if value in array
-in, -notin	Reverse of contains, notcontains.

Common cmdlets

cd, chdir, sl	Set-Location
cat, gc, type	Get-Content
ac	Add-Content
sc	Set-Content
copy, cp, cpi	Copy-Item
del, erase, rd, ri, rm, rmdir	Remove-Item
mi, move, mv	Move-Item
si	Set-Item
ni	New-Item
sleep	Start-Sleep
sajb	Start-Job
compare, diff	Compare-Object
group	Group-Object
curl, iwr, wget	Invoke-WebRequest
measure	Measure-Object
nal	New-Alias
rvpa	Resolve-Path
rujb	Resume-Job
set, sv	Set-Variable
shcm	Show-Command
sort	Sort-Object
sasv	Start-Service
saps, start	Start-Process
sujb	Suspend-Job
wjb	Wait-Job
?, where	Where-Object
echo, write	Write-Output

Common Aliases

gcm	Get-Command
foreach,%	Foreach-Object
sort	Sort-Object
where, ?	Where-Object
diff, compare	Compare-Object
dir, ls, gci	Get-ChildItem
gi	Get-Item
copy, cp, cpi	Copy-Item
move, mv, mi	Move-Item
del, rm	Remove-Item
rni, ren	Rename-Item
fFt	Format-Table
fl	Format-List
gcim	Get-CimInstance
cat, gc, type	Get-Content
sc	Set-Content
h, history, ghy	Get-History
ihy, r	Invoke-History
gp	Get-ItemProperty
sp	Set-ItemProperty
pwd, gl	Get-Location
gm	Get-Member
sls	Select-String
cd, chdir, sl	Set-Location
cls, clear	Clear-Host

Arrays Objects

$arr = "a", "b"	Array of strings
$arr = @()	Empty array
$arr[5]	Sixth array element
$arr[-3..-1]	Last three array elements
$arr[1,4+6..9]	Elements at index 1,4, 6-9
$arr[1] += 200	Add to array item value
$z = $arA + $arB	Two arrays into single array
[pscustomobject]@{x=1;z=2}	Create custom object
(Get-Date).Date	Date property of object

Writing output and reading

"This displays a string"	String is written directly to output
Write-Host "color" -ForegroundColor Red -NoNewLine	String with colors, no new line at end
$age = Read-host "Please enter your age"	Set $age variable to input from user
$pwd = Read-host "Please enter your password" -asSecureString	Read in $pwd as secure string
Clear-Host	Clear console