

Assignments:-17

Module:- COSA(Power Shell)

Name:- Prithviraj Nikam

Lab Assignment :-

1. Find a command that lists the network adapters installed in your computer.

```
Administrator: Command Prompt - PowerShell
Microsoft Windows [Version 10.0.19044.1645]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ME>PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\ME> gwmi win32_networkadapter

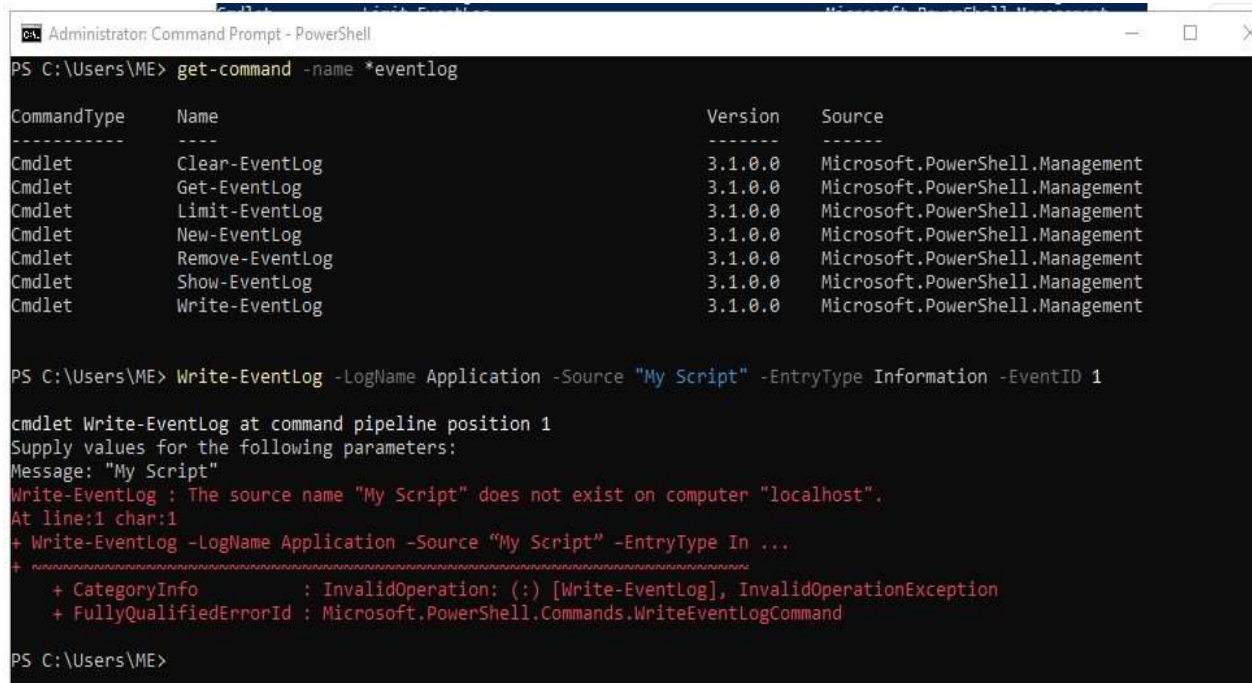
ServiceName      : kdnic
MACAddress       :
AdapterType      :
DeviceID         : 0
Name             : Microsoft Kernel Debug Network Adapter
NetworkAddresses :
Speed            :

ServiceName      : rt640x64
MACAddress       : 28:D2:44:AB:3C:2C
AdapterType      : Ethernet 802.3
DeviceID         : 1
Name             : Realtek PCIe GBE Family Controller
NetworkAddresses :
Speed            : 9223372036854775807

ServiceName      : RTWlanE01
MACAddress       : 9C:AD:97:84:CA:31
AdapterType      : Ethernet 802.3
DeviceID         : 2
Name             : Realtek RTL8723BE Wireless LAN 802.11n PCI-E NIC
NetworkAddresses :
Speed            : 63100000

ServiceName      : vwifimp
MACAddress       : 9E:AD:97:84:CA:31
AdapterType      : Ethernet 802.3
DeviceID         : 3
Name             : Microsoft Wi-Fi Direct Virtual Adapter
NetworkAddresses :
Speed            : 9223372036854775807
```

2. Find a command that lets you write to an event log.



The screenshot shows a PowerShell console window titled "Administrator: Command Prompt - PowerShell". The user enters the command `get-command -name *eventlog`, which returns a table of event log-related cmdlets. Then, the user enters `Write-EventLog -LogName Application -Source "My Script" -EntryType Information -EventID 1`. This command fails with an error: "Write-EventLog : The source name 'My Script' does not exist on computer 'localhost'". The error details show an `InvalidOperationException` from the `Write-EventLog` command.

```
PS C:\Users\ME> get-command -name *eventlog

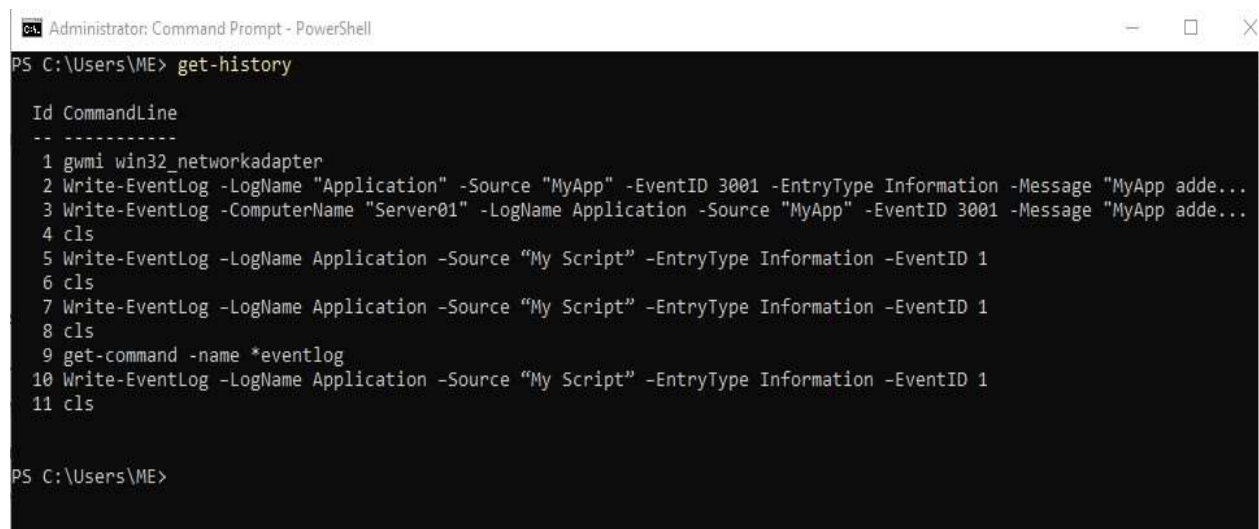
CommandType      Name                                           Version      Source
-----
Cmdlet            Clear-EventLog                               3.1.0.0      Microsoft.PowerShell.Management
Cmdlet            Get-EventLog                                3.1.0.0      Microsoft.PowerShell.Management
Cmdlet            Limit-EventLog                               3.1.0.0      Microsoft.PowerShell.Management
Cmdlet            New-EventLog                                3.1.0.0      Microsoft.PowerShell.Management
Cmdlet            Remove-EventLog                              3.1.0.0      Microsoft.PowerShell.Management
Cmdlet            Show-EventLog                                3.1.0.0      Microsoft.PowerShell.Management
Cmdlet            Write-EventLog                               3.1.0.0      Microsoft.PowerShell.Management

PS C:\Users\ME> Write-EventLog -LogName Application -Source "My Script" -EntryType Information -EventID 1

cmdlet Write-EventLog at command pipeline position 1
Supply values for the following parameters:
Message: "My Script"
Write-EventLog : The source name "My Script" does not exist on computer "localhost".
At line:1 char:1
+ Write-EventLog -LogName Application -Source "My Script" -EntryType In ...
+ ~~~~~
+ CategoryInfo          : InvalidOperationException: (:) [Write-EventLog], InvalidOperationException
+ FullyQualifiedErrorId : Microsoft.PowerShell.Commands.WriteEventLogCommand

PS C:\Users\ME>
```

3. Find a command that lists the history of PowerShell commands that you have run in the current session



The screenshot shows a PowerShell console window titled "Administrator: Command Prompt - PowerShell". The user enters the command `get-history`, which returns a list of commands run in the current session, including `gwmi win32_networkadapter`, several `Write-EventLog` commands, `cls`, and `get-command -name *eventlog`.

```
PS C:\Users\ME> get-history

Id CommandLine
--
1 gwmi win32_networkadapter
2 Write-EventLog -LogName "Application" -Source "MyApp" -EventID 3001 -EntryType Information -Message "MyApp adde...
3 Write-EventLog -ComputerName "Server01" -LogName Application -Source "MyApp" -EventID 3001 -Message "MyApp adde...
4 cls
5 Write-EventLog -LogName Application -Source "My Script" -EntryType Information -EventID 1
6 cls
7 Write-EventLog -LogName Application -Source "My Script" -EntryType Information -EventID 1
8 cls
9 get-command -name *eventlog
10 Write-EventLog -LogName Application -Source "My Script" -EntryType Information -EventID 1
11 cls

PS C:\Users\ME>
```

4. Find commands that will let you work with aliases in the shell.

Step- 1: Get all aliases in the current session

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> get-Alias
```

CommandType	Name	Version	Source
Alias	% -> ForEach-Object		
Alias	? -> Where-Object		
Alias	ac -> Add-Content		
Alias	asnp -> Add-PSSnapin		
Alias	cat -> Get-Content		
Alias	cd -> Set-Location		
Alias	CFS -> ConvertFrom-String	3.1.0.0	Microsoft.PowerShell.Utility
Alias	chdir -> Set-Location		
Alias	clc -> Clear-Content		
Alias	clear -> Clear-Host		
Alias	clhy -> Clear-History		
Alias	cli -> Clear-Item		
Alias	clp -> Clear-ItemProperty		
Alias	cls -> Clear-Host		
Alias	clv -> Clear-Variable		
Alias	cnsn -> Connect-PSSession		
Alias	compare -> Compare-Object		
Alias	copy -> Copy-Item		
Alias	cp -> Copy-Item		
Alias	cpi -> Copy-Item		
Alias	cpx -> Copy-ItemProperty		
Alias	curl -> Invoke-WebRequest		
Alias	cvpa -> Convert-Path		

Step- 2: Get aliases by name

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-Alias -Name gp*, sp* -Exclude *ps
```

CommandType	Name	Version	Source
Alias	gp -> Get-ItemProperty		
Alias	gpv -> Get-ItemPropertyValue		
Alias	sp -> Set-ItemProperty		
Alias	spjb -> Stop-Job		
Alias	spsv -> Stop-Service		

```
PS C:\Users\ME>
```

Step- 3: Get aliases for a cmdlet

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-Alias -Definition Get-ChildItem

CommandType      Name                                     Version          Source
-----
Alias            dir -> Get-ChildItem                   -----
Alias            gci -> Get-ChildItem
Alias            ls -> Get-ChildItem

PS C:\Users\ME>
```

Step-4:-Get aliases by name and filter by beginning letter

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-Alias -Definition "*-PSSession" -Exclude e* -Scope Global

CommandType      Name                                     Version          Source
-----
Alias            cnsn -> Connect-PSSession
Alias            dnsn -> Disconnect-PSSession
Alias            gsn -> Get-PSSession
Alias            ipsn -> Import-PSSession
Alias            nsn -> New-PSSession
Alias            rcsn -> Receive-PSSession
Alias            rsn -> Remove-PSSession

PS C:\Users\ME>
```

5. Find a command that will list Windows Firewall rules

```

Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetFirewallRule -PolicyStore ActiveStore|more

Name                : SNMPTRAP-In-UDP
DisplayName          : SNMP Trap Service (UDP In)
Description          : Inbound rule for the SNMP Trap Service to allow SNMP traps. [UDP 162]
DisplayGroup        : SNMP Trap
Group               : @firewallapi.dll,-50323
Enabled             : False
Profile             : Private, Public
Platform            : {}
Direction           : Inbound
Action              : Allow
EdgeTraversalPolicy  : Block
LooseSourceMapping  : False
LocalOnlyMapping     : False
Owner               :
PrimaryStatus       : Inactive
Status              : The rule was parsed successfully from the store. (65536)
EnforcementStatus   : Disabled
PolicyStoreSource    : PersistentStore
PolicyStoreSourceType : Local

Name                : SNMPTRAP-In-UDP-NoScope
DisplayName          : SNMP Trap Service (UDP In)
Description          : Inbound rule for the SNMP Trap Service to allow SNMP traps. [UDP 162]
DisplayGroup        : SNMP Trap
Group               : @firewallapi.dll,-50323
Enabled             : False
Profile             : Domain
Platform            : {}
Direction           : Inbound
Action              : Allow
EdgeTraversalPolicy  : Block
LooseSourceMapping  : False
LocalOnlyMapping     : False
Owner               :
PrimaryStatus       : Inactive
Status              : The rule was parsed successfully from the store. (65536)
EnforcementStatus   : Disabled
PolicyStoreSource    : PersistentStore
PolicyStoreSourceType : Local

```

6. Find a command that will list in use IP addresses.

Administrator: Command Prompt - PowerShell

```
PS C:\Users\ME> Get-NetIPAddress -AddressFamily IPV4
```

```
IPAddress      : 192.168.213.1
InterfaceIndex : 13
InterfaceAlias : VMware Network Adapter VMnet8
AddressFamily  : IPv4
Type           : Unicast
PrefixLength   : 24
PrefixOrigin   : Dhcp
SuffixOrigin   : Dhcp
AddressState   : Preferred
ValidLifetime  : 00:19:35
PreferredLifetime : 00:19:35
SkipAsSource   : False
PolicyStore    : ActiveStore

IPAddress      : 192.168.58.1
InterfaceIndex : 8
InterfaceAlias : VMware Network Adapter VMnet1
AddressFamily  : IPv4
Type           : Unicast
PrefixLength   : 24
PrefixOrigin   : Dhcp
SuffixOrigin   : Dhcp
AddressState   : Preferred
ValidLifetime  : 00:19:35
PreferredLifetime : 00:19:35
SkipAsSource   : False
PolicyStore    : ActiveStore

IPAddress      : 169.254.14.178
InterfaceIndex : 16
InterfaceAlias : Local Area Connection* 2
AddressFamily  : IPv4
Type           : Unicast
PrefixLength   : 16
PrefixOrigin   : WellKnown
SuffixOrigin   : Link
AddressState   : Tentative
ValidLifetime  : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime : Infinite ([TimeSpan]::MaxValue)
SkipAsSource   : False
PolicyStore    : ActiveStore
```

7. How would you write a command that retrieved all services whose names started with S and with W?

Step-1: Get all services on the computer

Administrator: Command Prompt - PowerShell

PS C:\Users\ME> Get-Service

Status	Name	DisplayName
Running	AarSvc_4f05d	Agent Activation Runtime_4f05d
Running	AdobeARMservice	Adobe Acrobat Update Service
Stopped	AJRouter	AllJoyn Router Service
Stopped	ALG	Application Layer Gateway Service
Running	AnyDesk	AnyDesk Service
Stopped	AppIDSvc	Application Identity
Running	AppInfo	Application Information
Stopped	AppMgmt	Application Management
Stopped	AppReadiness	App Readiness
Stopped	AppVClient	Microsoft App-V Client
Running	AppXSvc	AppX Deployment Service (AppXSVC)
Running	arwsrv	Realtime Behavior Detection
Stopped	AssignedAccessM...	AssignedAccessManager Service
Running	AudioEndpointBu...	Windows Audio Endpoint Builder
Running	Audiosrv	Windows Audio
Stopped	autotimesvc	Cellular Time
Stopped	AxInstSV	ActiveX Installer (AxInstSV)
Stopped	BcastDVRUserSer...	GameDVR and Broadcast User Service_...
Stopped	BDESVC	BitLocker Drive Encryption Service
Running	Behavior Detect...	Behavior Detection System
Running	BFE	Base Filtering Engine
Stopped	BITS	Background Intelligent Transfer Ser...
Stopped	BluetoothUserSe...	Bluetooth User Support Service_4f05d
Running	BrokerInfrastru...	Background Tasks Infrastructure Ser...
Running	BTAGService	Bluetooth Audio Gateway Service
Running	BthAvctpSvc	AVCTP service
Running	bthserv	Bluetooth Support Service
Running	camsvc	Capability Access Manager Service
Running	CaptureService_...	CaptureService_4f05d
Running	cbdhsvc_4f05d	Clipboard User Service_4f05d
Running	CDPSvc	Connected Devices Platform Service
Running	CDPUserSvc_4f05d	Connected Devices Platform User Ser...
Stopped	CertPropSvc	Certificate Propagation
Running	ClipSVC	Client License Service (ClipSVC)
Stopped	cloudidsvc	Microsoft Cloud Identity Service
Stopped	COMSysApp	COM+ System Application
Stopped	ConsentUxUserSv...	ConsentUX_4f05d
Running	Core Mail Prote...	Core Mail Protection
Stopped	Core Scanning S...	Core Scanning Server
Running	Core Scanning S...	Core Scanning ServerEx

Step-2: Get services that begin with a search string “s”

Administrator: Command Prompt - PowerShell

PS C:\Users\ME> Get-Service "s*"

Status	Name	DisplayName
Running	SamSs	Security Accounts Manager
Stopped	SAService	Conexant SmartAudio service
Running	ScanWscS	Quick Heal Helper Service WSC
Stopped	SCardSvr	Smart Card
Stopped	ScDeviceEnum	Smart Card Device Enumeration Service
Running	Schedule	Task Scheduler
Stopped	SCPolicySvc	Smart Card Removal Policy
Running	ScSecSvc	Core Browsing Protection
Stopped	SDRSVC	Windows Backup
Stopped	seclogon	Secondary Logon
Running	SecurityHealthS...	Windows Security Service
Running	SEMgrSvc	Payments and NFC/SE Manager
Running	SENS	System Event Notification Service
Stopped	Sense	Windows Defender Advanced Threat Pr...
Stopped	SensorDataService	Sensor Data Service
Stopped	SensorService	Sensor Service
Stopped	SensrSvc	Sensor Monitoring Service
Stopped	SessionEnv	Remote Desktop Configuration
Running	SgrmBroker	System Guard Runtime Monitor Broker
Stopped	SharedAccess	Internet Connection Sharing (ICS)
Stopped	SharedRealitySvc	Spatial Data Service
Running	ShellHWDetection	Shell Hardware Detection
Stopped	shpamsvc	Shared PC Account Manager
Stopped	smphost	Microsoft Storage Spaces SMP
Stopped	SmsRouter	Microsoft Windows SMS Router Service.
Stopped	SNMPTRAP	SNMP Trap
Stopped	spectrum	Windows Perception Service
Running	Spooler	Print Spooler
Stopped	sppsvc	Software Protection
Running	SSDPSPRV	SSDP Discovery
Stopped	ssh-agent	OpenSSH Authentication Agent
Running	SstpSvc	Secure Socket Tunneling Protocol Se...
Running	StateRepository	State Repository Service
Stopped	stisvc	Windows Image Acquisition (WIA)
Running	StorSvc	Storage Service
Stopped	svsvc	Spot Verifier
Stopped	swprv	Microsoft Software Shadow Copy Prov...
Running	SynTPEnhService	SynTPEnh Caller Service
Running	SysMain	SysMain
Running	SystemEventsBroker	System Events Broker

String"W"

Administrator: Command Prompt - PowerShell

PS C:\Users\ME> Get-Service "W*"

status	Name	DisplayName
stopped	W32Time	Windows Time
stopped	WaaSMedicSvc	Windows Update Medic Service
stopped	WalletService	WalletService
stopped	WarpJITSvc	WarpJITSvc
stopped	wbengine	Block Level Backup Engine Service
Running	WbioSrv	Windows Biometric Service
Running	WcmSvc	Windows Connection Manager
stopped	wcncsvc	Windows Connect Now - Config Registrar
Running	WdiServiceHost	Diagnostic Service Host
stopped	WdiSystemHost	Diagnostic System Host
stopped	WdNisSvc	Microsoft Defender Antivirus Networ...
stopped	WebClient	WebClient
stopped	WeSvc	Windows Event Collector
stopped	WEHOSTSV	Windows Encryption Provider Host Se...
stopped	werclpsupport	Problem Reports Control Panel Support
stopped	WerSvc	Windows Error Reporting Service
stopped	WFDSCongMgrSvc	Wi-Fi Direct Services Connection Ma...
stopped	WiaRpc	Still Image Acquisition Events
stopped	WinDefend	Microsoft Defender Antivirus Service
Running	WinHttpAutoProx...	WinHTTP Web Proxy Auto-Discovery Se...
Running	Winmgmt	Windows Management Instrumentation
stopped	WinRM	Windows Remote Management (WS-Manag...
stopped	wisvc	Windows Insider Service
Running	WlanSvc	WLAN AutoConfig
stopped	wlidsvc	Microsoft Account Sign-in Assistant
stopped	wlpasvc	Local Profile Assistant Service
stopped	WManSvc	Windows Management Service
stopped	wmiApSrv	WMI Performance Adapter
stopped	WMPNetworkSvc	Windows Media Player Network Sharin...
stopped	workfolderssvc	Work Folders
stopped	WpcMonSvc	Parental Controls
stopped	WPDBusEnum	Portable Device Enumerator Service
Running	WpnService	Windows Push Notifications System S...
Running	WpnUserService_...	Windows Push Notifications User Ser...
Running	wscsvc	Security Center
Running	WSearch	Windows Search
stopped	wuauerv	Windows Update
stopped	WwanSvc	WWAN AutoConfig

8. Find a command that produces the current date and time

Step-1: Get the current date and time

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-Date

1 December, 2022 12:27:36 AM

PS C:\Users\ME>
PS C:\Users\ME> Tuesday, June 25, 2019 14:53:32
```

Step-2:- Get the date and time with a .NET format specifier

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-Date -Format "dddd MM/dd/yyyy HH:mm K"
Thursday 12-01-2022 00:29 +05:30
PS C:\Users\ME>
PS C:\Users\ME> Tuesday 06/25/2019 16:17 -07:00
```

9. display a list of only physical network adapters.

Step-1:-Get all physical network adapters

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetAdapter -Name * -Physical
```

Name	InterfaceDescription	ifIndex	Status	MacAddress	LinkSpeed
Wi-Fi	Realtek RTL8723BE Wireless LAN 802.1...	15	Up	9C-AD-97-84-CA-31	72.2 Mbps
Ethernet	Realtek PCIe GBE Family Controller	6	Disconnected	28-D2-44-AB-3C-2C	0 bps

```
PS C:\Users\ME>
```

Step-2:-Get a network adapter by the specified name

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetAdapter -Name "Ethernet"
```

Name	InterfaceDescription	ifIndex	Status	MacAddress	LinkSpeed
Ethernet	Realtek PCIe GBE Family Controller	6	Disconnected	28-D2-44-AB-3C-2C	0 bps

```
PS C:\Users\ME>
```

Step-3:-Get all visible and hidden network adapters

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetAdapter -Name * -IncludeHidden
```

Name	InterfaceDescription	ifIndex	Status	MacAddress	LinkSpeed
Bluetooth Network Conn...	Bluetooth Device (Personal Area Netw...	20	Disconnected	9C-AD-97-84-CA-32	3 Mbps
Local Area Connection* 8	WAN Miniport (IP)	19	Up		0 bps
Ethernet (Kernel Debug...	Microsoft Kernel Debug Network Adapter	18	Not Present		0 bps
Local Area Connection* 4	WAN Miniport (IKEv2)	17	Disconnected		0 bps
Local Area Connection* 2	Microsoft Wi-Fi Direct Virtual Ada...#2	16	Disconnected	9C-AD-97-84-CA-31	0 bps
Wi-Fi	Realtek RTL8723BE Wireless LAN 802.1...	15	Up	9C-AD-97-84-CA-31	72.2 Mbps
Local Area Connection* 6	WAN Miniport (PPTP)	14	Disconnected		0 bps
VMware Network Adapte...	VMware Virtual Ethernet Adapter for ...	13	Up	00-50-56-C0-00-08	100 Mbps
Local Area Connection* 5	WAN Miniport (L2TP)	12	Disconnected		0 bps
Local Area Connection* 3	WAN Miniport (SSTP)	11	Disconnected		0 bps
Teredo Tunneling Pseud...		10	Not Present		0 bps
Local Area Connection* 7	WAN Miniport (PPPOE)	9	Disconnected		0 bps
VMware Network Adapte...	VMware Virtual Ethernet Adapter for ...	8	Up	00-50-56-C0-00-01	100 Mbps
Local Area Connection* 10	WAN Miniport (Network Monitor)	7	Up		0 bps
Ethernet	Realtek PCIe GBE Family Controller	6	Disconnected	28-D2-44-AB-3C-2C	0 bps
Local Area Connection* 9	WAN Miniport (IPv6)	5	Up		0 bps
Microsoft IP-HTTPS Pla...		4	Not Present		0 bps
Local Area Connection* 1	Microsoft Wi-Fi Direct Virtual Adapter	3	Disconnected	9E-AD-97-84-CA-31	0 bps
6to4 Adapter		2	Not Present		0 bps

```
PS C:\Users\ME>
```

Step- 4:- Get all visible network adapters

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetAdapter -Name *
```

Name	InterfaceDescription	ifIndex	Status	MacAddress	LinkSpeed
Bluetooth Network Conn...	Bluetooth Device (Personal Area Netw...	20	Disconnected	9C-AD-97-84-CA-32	3 Mbps
Wi-Fi	Realtek RTL8723BE Wireless LAN 802.1...	15	Up	9C-AD-97-84-CA-31	72.2 Mbps
VMware Network Adapte...	VMware Virtual Ethernet Adapter for ...	13	Up	00-50-56-C0-00-08	100 Mbps
VMware Network Adapte...	VMware Virtual Ethernet Adapter for ...	8	Up	00-50-56-C0-00-01	100 Mbps
Ethernet	Realtek PCIe GBE Family Controller	6	Disconnected	28-D2-44-AB-3C-2C	0 bps

```
PS C:\Users\ME>
```

Step- 5:- Display the common properties for the specified network adapter

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetAdapter -Name "Ethernet" | Format-List -Property *

MacAddress           : 28-D2-44-AB-3C-2C
Status               : Disconnected
LinkSpeed            : 0 bps
MediaType            : 802.3
PhysicalMediaType    : 802.3
AdminStatus          : Up
MediaConnectionState : Disconnected
DriverInformation     : Driver Date 2015-05-05 Version 10.1.505.2015 NDIS 6.40
DriverFileName       : rt640x64.sys
NdisVersion          : 6.40
IfOperStatus         : Down
IfAlias              : Ethernet
InterfaceAlias       : Ethernet
IfIndex              : 6
IfDesc               : Realtek PCIe GBE Family Controller
IfName               : ethernet_32769
DriverVersion        : 10.1.505.2015
LinkLayerAddress     : 28-D2-44-AB-3C-2C
Caption              :
Description          :
ElementName          :
InstanceID           : {4AD7EDCA-1E28-489F-8E66-BE5E6BF36FA4}
CommunicationStatus  :
DetailedStatus       :
HealthState          :
InstallDate          :
Name                 : Ethernet
OperatingStatus      :
OperationalStatus    :
```

10.display a list of drives that use the FileSystem PSProvider.

Step-1:-Get drives in the current session

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-PSDrive

Name      Used (GB)  Free (GB) Provider      Root      CurrentLocation
-----
Alias     123.97    168.44  FileSystem    C:\       Users\ME
Cert      47.80     136.17  Certificate    \
D         0.10      58.49   FileSystem    E:\
Env       86.20     50.52   Environment    F:\
Function  118.19    26.53   FileSystem    G:\
H         108.92    24.81   FileSystem    H:\
HKCU      0         0       Registry      HKEY_CURRENT_USER
HKLM      0         0       Registry      HKEY_LOCAL_MACHINE
Variable  0         0       Variable
WSMan     0         0       WSMAN
```

Step-2:- Get a drive on the computer

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> get-psdrive D

Name      Used (GB)  Free (GB) Provider      Root      CurrentLocation
-----
D         47.80     136.17  FileSystem    D:\
```

Step-3:-Get all the drives that are supported by the Windows PowerShell file system provider


```
PS C:\Users\ME> Get-PSDrive -PSProvider FileSystem
```

Name	Used (GB)	Free (GB)	Provider	Root	CurrentLocation
C:	123.97	168.43	FileSystem	C:\	Users\ME
D:	47.80	136.17	FileSystem	D:\	
E:	0.10	58.49	FileSystem	E:\	
F:	86.20	50.52	FileSystem	F:\	
G:	110.10	26.53	FileSystem	G:\	
H:	108.92	24.81	FileSystem	H:\	

11.Display a list of Windows Firewall rules that are enabled.

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetFirewallRule -PolicyStore ActiveStore|more
```

```
Name : SNMPTRAP-In-UDP
DisplayName : SNMP Trap Service (UDP In)
Description : Inbound rule for the SNMP Trap Service to allow SNMP traps. [UDP 162]
DisplayGroup : SNMP Trap
Group : @firewallapi.dll,-50323
Enabled : False
Profile : Private, Public
Platform : {}
Direction : Inbound
Action : Allow
EdgeTraversalPolicy : Block
LooseSourceMapping : False
LocalOnlyMapping : False
Owner :
PrimaryStatus : Inactive
Status : The rule was parsed successfully from the store. (65536)
EnforcementStatus : Disabled
PolicyStoreSource : PersistentStore
PolicyStoreSourceType : Local
```

```
Name : SNMPTRAP-In-UDP-NoScope
DisplayName : SNMP Trap Service (UDP In)
Description : Inbound rule for the SNMP Trap Service to allow SNMP traps. [UDP 162]
DisplayGroup : SNMP Trap
Group : @firewallapi.dll,-50323
Enabled : False
Profile : Domain
Platform : {}
Direction : Inbound
Action : Allow
EdgeTraversalPolicy : Block
LooseSourceMapping : False
LocalOnlyMapping : False
Owner :
PrimaryStatus : Inactive
Status : The rule was parsed successfully from the store. (65536)
EnforcementStatus : Disabled
PolicyStoreSource : PersistentStore
PolicyStoreSourceType : Local
```

12.Display a list of all currently defined variables.

Step-1:-Get variables by letter

Administrator: Command Prompt - PowerShell

```
PS C:\Users\ME> Get-Variable m*
```

Name	Value
Matches	{0}
MaximumAliasCount	4096
MaximumDriveCount	4096
MaximumErrorCount	256
MaximumFunctionCount	4096
MaximumHistoryCount	4096
MaximumVariableCount	4096
MyInvocation	System.Management.Automation.InvocationInfo

Step-2:- Get variable values by letter

Administrator: Command Prompt - PowerShell

```
PS C:\Users\ME> Get-Variable m* -ValueOnly
```

Name	Value
0	ReadOnly
4096	
4096	
256	
4096	
4096	
4096	

MyCommand	: Get-Variable m* -ValueOnly
BoundParameters	: {}
UnboundArguments	: {}
ScriptLineNumber	: 0
OffsetInLine	: 0
HistoryId	: 92
ScriptName	:
Line	:
PositionMessage	:
PSScriptRoot	:
PSCommandPath	:
InvocationName	:
PipelineLength	: 2
PipelinePosition	: 1
ExpectingInput	: False
CommandOrigin	: Runspace
DisplayScriptPosition	:

```
PS C:\Users\ME>
```

Step-3:-Get variables by two letters

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-Variable -Include M*,P*

Name                           Value
----                           -
Matches                        {0}
MaximumAliasCount              4096
MaximumDriveCount              4096
MaximumErrorCount              256
MaximumFunctionCount           4096
MaximumHistoryCount            4096
MaximumVariableCount           4096
MyInvocation                   System.Management.Automation.InvocationInfo
PID                             1992
PROFILE                        C:\Users\ME\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1
ProgressPreference             Continue
PSBoundParameters              {}
PSCommandPath                  {}
PSCulture                      en-US
PSDefaultParameterValues       {}
PSEdition                      Desktop
PSEmailServer                  {}
PSHOME                         C:\Windows\System32\WindowsPowerShell\v1.0
PSScriptRoot                   {}
PSSessionApplicationName       wsman
PSSessionConfigurationName     http://schemas.microsoft.com/powershell/Microsoft.PowerShell
PSSessionOption                System.Management.Automation.Remoting.PSSessionOption
PSUICulture                    en-US
PSVersionTable                 {PSVersion, PSEdition, PSCompatibleVersions, BuildVersion...}
PWD                             C:\Users\ME

PS C:\Users\ME>
```

13. Create a variable named x and populate it with the number 100.

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Set-Variable -Name "x" -Value "100"
PS C:\Users\ME> Get-Variable -Name "x"

Name                           Value
----                           -
x                               100

PS C:\Users\ME>
```

14. Completely remove the variable named x.

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Set-Variable -Name "x" -Value "100"
PS C:\Users\ME> Get-Variable -Name "x"

Name          Value
-----
x             100

PS C:\Users\ME> Remove-Variable x
PS C:\Users\ME> Get-Variable -Name "x"
Get-Variable : Cannot find a variable with the name 'x'.
At line:1 char:1
+ Get-Variable -Name "x"
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (x:String) [Get-Variable], ItemNotFoundException
+ FullyQualifiedErrorId : VariableNotFound,Microsoft.PowerShell.Commands.GetVariableCommand

PS C:\Users\ME>
```

15.Place all running processes into a new variable named procs.

16.Using the Win32_Service class, display a list of services that are set to start automatically, but that are not currently running

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-WmiObject -List win32*service*

NameSpace: ROOT\cimv2

Name          Methods          Properties
-----
Win32_ServiceControl      {}               {Arguments, Caption, Description, Event...}
Win32_ServiceSpecification {Invoke}         {Caption, CheckID, CheckMode, Dependencies...}
Win32_ServiceSpecificationService {}               {Check, Element}
Win32_BaseService         {StartService, St... {AcceptPause, AcceptStop, Caption, CreationClassName...}
Win32_Service             {StartService, St... {AcceptPause, AcceptStop, Caption, CheckPoint...}
Win32_TerminalService     {StartService, St... {AcceptPause, AcceptStop, Caption, CheckPoint...}
Win32_ApplicationService {StartService, St... {Caption, CreationClassName, Description, InstallDate...}
Win32_SystemServices      {}               {GroupComponent, PartComponent}
Win32_LoadOrderGroupServiceMembers {}               {GroupComponent, PartComponent}
Win32_LoadOrderGroupServiceDepen... {}               {Antecedent, Dependent}
Win32_DependentService    {}               {Antecedent, Dependent, TypeOfDependency}

PS C:\Users\ME>
```

17.Display a list of configurations where DHCP is enabled. Display only the interface index and description of each

Step-1:-Get all IP configuration details

Administrator: Command Prompt - PowerShell

PS C:\Users\ME> Get-NetIPConfiguration -All

```
InterfaceAlias      : VMware Network Adapter VMnet8
InterfaceIndex      : 13
InterfaceDescription : VMware Virtual Ethernet Adapter for VMnet8
IPv4Address         : 192.168.213.1
IPv6DefaultGateway  :
IPv4DefaultGateway  :
DNSServer           : fec0:0:0:ffff::1
                   : fec0:0:0:ffff::2
                   : fec0:0:0:ffff::3

InterfaceAlias      : VMware Network Adapter VMnet1
InterfaceIndex      : 8
InterfaceDescription : VMware Virtual Ethernet Adapter for VMnet1
IPv4Address         : 192.168.58.1
IPv6DefaultGateway  :
IPv4DefaultGateway  :
DNSServer           : fec0:0:0:ffff::1
                   : fec0:0:0:ffff::2
                   : fec0:0:0:ffff::3

InterfaceAlias      : Wi-Fi
InterfaceIndex      : 15
InterfaceDescription : Realtek RTL8723BE Wireless LAN 802.11n PCI-E NIC
NetProfile.Name     : BC_2.4G
IPv6Address         : 2401:4900:1cb8:ae58:65d1:4182:e449:5dbd
IPv4Address         : 192.168.1.8
IPv6DefaultGateway  : fe80::1
IPv4DefaultGateway  : 192.168.1.1
DNSServer           : fe80::1
                   : 192.168.1.1

InterfaceAlias      : Local Area Connection* 2
InterfaceIndex      : 16
InterfaceDescription : Microsoft Wi-Fi Direct Virtual Adapter #2
NetAdapter.Status   : Disconnected

InterfaceAlias      : Local Area Connection* 1
InterfaceIndex      : 3
InterfaceDescription : Microsoft Wi-Fi Direct Virtual Adapter
NetAdapter.Status   : Disconnected
```

Step-2:- Get the IP configuration by interface index

Administrator: Command Prompt - PowerShell

```
PS C:\Users\ME> Get-NetIPConfiguration -InterfaceIndex 8
```

```
InterfaceAlias      : VMware Network Adapter VMnet1
InterfaceIndex      : 8
InterfaceDescription : VMware Virtual Ethernet Adapter for VMnet1
IPv4Address         : 192.168.58.1
IPv6DefaultGateway  :
IPv4DefaultGateway  :
DNSServer           : fec0:0:0:ffff::1
                   : fec0:0:0:ffff::2
                   : fec0:0:0:ffff::3
```

```
PS C:\Users\ME>
```

Step-3:-Get the IP configuration from pipeline input

Administrator: Command Prompt - PowerShell

```
PS C:\Users\ME> Get-NetIPConfiguration | Get-NetIPAddress
```

```
IPAddress           : fe80::b530:23ba:ed1:6b98%8
InterfaceIndex      : 8
InterfaceAlias      : VMware Network Adapter VMnet1
AddressFamily       : IPv6
Type                : Unicast
PrefixLength        : 64
PrefixOrigin        : WellKnown
SuffixOrigin        : Link
AddressState        : Preferred
ValidLifetime       : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime   : Infinite ([TimeSpan]::MaxValue)
SkipAsSource        : False
PolicyStore         : ActiveStore
```

```
IPAddress           : fe80::35ba:c676:bfe9:8ab0%13
InterfaceIndex      : 13
InterfaceAlias      : VMware Network Adapter VMnet8
AddressFamily       : IPv6
Type                : Unicast
PrefixLength        : 64
PrefixOrigin        : WellKnown
SuffixOrigin        : Link
AddressState        : Preferred
ValidLifetime       : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime   : Infinite ([TimeSpan]::MaxValue)
SkipAsSource        : False
PolicyStore         : ActiveStore
```

18.Display computers' last boot up time

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> systeminfo | Select-String "Host Name","System Boot Time"

Host Name:                DESKTOP-3JU2UCL
System Boot Time:         30-Nov-22, 9:18:12 PM

PS C:\Users\ME>
```

19.Run a command that will display all local IPv4 addresses for all network adapters.

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetIPAddress -AddressFamily IPv4

IPAddress      : 192.168.213.1
InterfaceIndex  : 13
InterfaceAlias  : VMware Network Adapter VMnet8
AddressFamily   : IPv4
Type            : Unicast
PrefixLength    : 24
PrefixOrigin    : Dhcp
SuffixOrigin     : Dhcp
AddressState     : Preferred
ValidLifetime   : 00:16:01
PreferredLifetime : 00:16:01
SkipAsSource     : False
PolicyStore      : ActiveStore

IPAddress      : 192.168.58.1
InterfaceIndex  : 8
InterfaceAlias  : VMware Network Adapter VMnet1
AddressFamily   : IPv4
Type            : Unicast
PrefixLength    : 24
PrefixOrigin    : Dhcp
SuffixOrigin     : Dhcp
AddressState     : Preferred
ValidLifetime   : 00:16:01
PreferredLifetime : 00:16:01
SkipAsSource     : False
PolicyStore      : ActiveStore
```

IPv6

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetIPAddress -AddressFamily IPv6

IPAddress      : fe80::35ba:c676:bfe9:8ab0%13
InterfaceIndex : 13
InterfaceAlias  : VMware Network Adapter VMnet8
AddressFamily   : IPv6
Type            : Unicast
PrefixLength    : 64
PrefixOrigin    : WellKnown
SuffixOrigin    : Link
AddressState    : Preferred
ValidLifetime   : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime : Infinite ([TimeSpan]::MaxValue)
SkipAsSource    : False
PolicyStore     : ActiveStore

IPAddress      : fe80::b530:23ba:ed1:6b98%8
InterfaceIndex : 8
InterfaceAlias  : VMware Network Adapter VMnet1
AddressFamily   : IPv6
Type            : Unicast
PrefixLength    : 64
PrefixOrigin    : WellKnown
SuffixOrigin    : Link
AddressState    : Preferred
ValidLifetime   : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime : Infinite ([TimeSpan]::MaxValue)
SkipAsSource    : False
PolicyStore     : ActiveStore
```

20.Run a command that will display all IPv4 routes that are defined on the local computer.

Step-1:- Get all route


```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetRoute | Format-List -Property *

Publish                : No
Protocol               : Local
Store                  : ActiveStore
AddressFamily           : IPv4
State                   : Alive
ifIndex                : 16
Caption                :
Description             :
ElementName             :
InstanceID              : <??8<??8<??8<??9=<55;@55:8:8:8:55;
AdminDistance          :
DestinationAddress      :
IsStatic                :
RouteMetric             : 256
TypeOfRoute             : 3
CompartmentId          : 1
DestinationPrefix       : 255.255.255.255/32
InterfaceAlias          : Local Area Connection* 2
InterfaceIndex          : 16
InterfaceMetric         : 25
NextHop                 : 0.0.0.0
PreferredLifetime       : 10675199.02:48:05.4775807
ValidLifetime           : 10675199.02:48:05.4775807
PSComputerName          :
CimClass                : ROOT/StandardCimv2:MSFT_NetRoute
CimInstanceProperties   : {Caption, Description, ElementName, InstanceID...}
CimSystemProperties     : Microsoft.Management.Infrastructure.CimSystemProperties

Publish                : No
Protocol               : Local
Store                  : ActiveStore
AddressFamily           : IPv4
State                   : Alive
ifIndex                : 3
Caption                :
```

Step-2:- IPv4 Route

```

Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetRoute -AddressFamily IPv4

```

ifIndex	DestinationPrefix	NextHop	RouteMetric	ifMetric	PolicyStore
16	255.255.255.255/32	0.0.0.0	256	25	ActiveStore
3	255.255.255.255/32	0.0.0.0	256	25	ActiveStore
15	255.255.255.255/32	0.0.0.0	256	55	ActiveStore
20	255.255.255.255/32	0.0.0.0	256	65	ActiveStore
6	255.255.255.255/32	0.0.0.0	256	5	ActiveStore
13	255.255.255.255/32	0.0.0.0	256	35	ActiveStore
8	255.255.255.255/32	0.0.0.0	256	35	ActiveStore
1	255.255.255.255/32	0.0.0.0	256	75	ActiveStore
16	224.0.0.0/4	0.0.0.0	256	25	ActiveStore
3	224.0.0.0/4	0.0.0.0	256	25	ActiveStore
15	224.0.0.0/4	0.0.0.0	256	55	ActiveStore
20	224.0.0.0/4	0.0.0.0	256	65	ActiveStore
6	224.0.0.0/4	0.0.0.0	256	5	ActiveStore
13	224.0.0.0/4	0.0.0.0	256	35	ActiveStore
8	224.0.0.0/4	0.0.0.0	256	35	ActiveStore
1	224.0.0.0/4	0.0.0.0	256	75	ActiveStore
13	192.168.213.255/32	0.0.0.0	256	35	ActiveStore
13	192.168.213.1/32	0.0.0.0	256	35	ActiveStore
13	192.168.213.0/24	0.0.0.0	256	35	ActiveStore
8	192.168.58.255/32	0.0.0.0	256	35	ActiveStore
8	192.168.58.1/32	0.0.0.0	256	35	ActiveStore
8	192.168.58.0/24	0.0.0.0	256	35	ActiveStore
15	192.168.1.255/32	0.0.0.0	256	55	ActiveStore
15	192.168.1.8/32	0.0.0.0	256	55	ActiveStore
15	192.168.1.0/24	0.0.0.0	256	55	ActiveStore
1	127.255.255.255/32	0.0.0.0	256	75	ActiveStore
1	127.0.0.1/32	0.0.0.0	256	75	ActiveStore
1	127.0.0.0/8	0.0.0.0	256	75	ActiveStore
15	0.0.0.0/0	192.168.1.1	0	55	ActiveStore

```

PS C:\Users\ME>

```

Step-3:- IPv6 Route

```

Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-NetRoute -AddressFamily IPv6

```

ifIndex	DestinationPrefix	NextHop	RouteMetric	ifMetric	PolicyStore
16	ff00::/8	::	256	25	ActiveStore
3	ff00::/8	::	256	25	ActiveStore
15	ff00::/8	::	256	55	ActiveStore
20	ff00::/8	::	256	65	ActiveStore
6	ff00::/8	::	256	5	ActiveStore
13	ff00::/8	::	256	35	ActiveStore
8	ff00::/8	::	256	35	ActiveStore
1	ff00::/8	::	256	75	ActiveStore
6	fe80::f163:db8d:cec1:4a63/128	::	256	5	ActiveStore
8	fe80::b530:23ba:ed1:6b98/128	::	256	35	ActiveStore
20	fe80::8464:de19:f3ad:731d/128	::	256	65	ActiveStore
15	fe80::65d1:4182:e449:5dbd/128	::	256	55	ActiveStore
16	fe80::508e:8fb4:1a9c:eb2/128	::	256	25	ActiveStore
13	fe80::35ba:c676:bfe9:8ab0/128	::	256	35	ActiveStore
3	fe80::1c99:fadc:b7b1:779d/128	::	256	25	ActiveStore
16	fe80::/64	::	256	25	ActiveStore
3	fe80::/64	::	256	25	ActiveStore
15	fe80::/64	::	256	55	ActiveStore
20	fe80::/64	::	256	65	ActiveStore
6	fe80::/64	::	256	5	ActiveStore
13	fe80::/64	::	256	35	ActiveStore
8	fe80::/64	::	256	35	ActiveStore
15	2401:4900:1cb8:ae58:d8b2:9a46:69bf:f98c/128	::	256	55	ActiveStore
15	2401:4900:1cb8:ae58:65d1:4182:e449:5dbd/128	::	256	55	ActiveStore
15	2401:4900:1cb8:ae58::/64	fe80::1	16	55	ActiveStore
15	2401:4900:1cb8:ae58::/64	::	4096	55	ActiveStore
1	::1/128	::	256	75	ActiveStore
15	::/0	fe80::1	4096	55	ActiveStore

```

PS C:\Users\ME>

```

21.Run a command that displays all services. Then, run a second set of commands that will put the exact same information, in the same format, into a text file

```
Administrator: Command Prompt - PowerShell
PS C:\Users\ME> get-service

Status Name                DisplayName
-----
Running AarSvc_4f05d         Agent Activation Runtime_4f05d
Running AdobeARMService Adobe Acrobat Update Service
Stopped AJRouter          AllJoyn Router Service
Stopped ALG            Application Layer Gateway Service
Running AnyDesk        AnyDesk Service
Stopped AppIDSvc        Application Identity
Running Appinfo         Application Information
Stopped AppMgmt         Application Management
Stopped AppReadiness    App Readiness
Stopped AppVClient      Microsoft App-V Client
Running AppXSvc         AppX Deployment Service (AppXSVC)
Running arwsrv          Realtime Behavior Detection
Stopped AssignedAccessManager AssignedAccessManager Service
```

```
PS C:\Users\ME> get-service -name anydesk

Status Name                DisplayName
-----
Running anydesk         AnyDesk Service

PS C:\Users\ME>
```

22. Display the total amount of VM, PM, and CPU for all currently running processes. Also include a count of the number of processes.

Administrator: Command Prompt - PowerShell

PS C:\Users\ME> Get-Process

Handles	NPM(K)	PM(K)	WS(K)	CPU(s)	Id	SI	ProcessName
443	35	34440	43808	75.91	2460	1	AnyDesk
345	22	23508	30988	4.94	4664	0	AnyDesk
326	20	23048	20104	1.47	12128	1	AnyDesk
402	23	10456	31832	13.20	12352	1	ApplicationFrameHost
127	8	1596	4628	0.02	4852	0	armsvc
753	26	9060	22564	25.91	2092	0	ARWSRVC
237	14	7524	16808	0.64	7028	0	audiodg
644	32	45312	36632	85.16	4824	0	bdssvc
90	9	1292	13832	0.13	2712	0	BSSISS
184	10	1684	7032	1.03	11008	1	CAudioFilterAgent64
175	11	4932	10764	83.83	7384	0	CFRUTIL
312	17	44404	89704	2.73	1764	1	chrome
265	16	17260	47764	0.52	2124	1	chrome
270	16	19660	50172	0.66	3036	1	chrome
261	16	16972	44608	0.56	3056	1	chrome
2542	88	149272	259464	522.06	3292	1	chrome
603	24	198368	238700	452.53	3780	1	chrome
312	9	2072	9632	0.11	4292	1	chrome
210	15	13812	30012	0.14	4428	1	chrome
476	18	35100	78416	2.08	5176	1	chrome
249	16	7624	20828	2.06	7064	1	chrome
281	16	22784	63840	2.06	7360	1	chrome
362	24	28088	52784	189.42	7480	1	chrome
261	16	17132	45132	0.56	8316	1	chrome
288	16	22636	52812	0.63	8380	1	chrome
376	20	53568	110832	5.30	8516	1	chrome
263	16	17140	46312	0.48	9024	1	chrome
273	16	21804	46064	0.55	9044	1	chrome
395	17	41912	82684	5.34	10176	1	chrome
264	16	18020	51736	0.97	10360	1	chrome
720	46	225276	233400	743.91	10604	1	chrome
637	24	160084	181596	33.36	10676	1	chrome
292	16	18328	47368	0.73	10836	1	chrome
353	17	20628	48260	0.64	12100	1	chrome
301	16	23424	54028	0.61	12148	1	chrome
263	16	22588	51100	0.72	12508	1	chrome
322	17	32736	72764	2.69	12700	1	chrome
548	17	57044	105264	14.75	12860	1	chrome
259	15	16820	42504	0.47	13272	1	chrome
490	21	64500	126700	15.64	13880	1	chrome

23. Write a power shell script to print name of all storage devices attached to your system.


```
Select Administrator: Command Prompt - PowerShell
PS C:\Users\ME> Get-PSDrive

Name      Used (GB)  Free (GB)  Provider  Root
-----
Alias      124.20     168.20     FileSystem C:\
Cert       47.80     136.17     Certificate \
D          0.10      58.49      FileSystem D:\
E          86.20     50.52      FileSystem F:\
Env        110.19    26.53      Function  G:\
Function   108.92    24.81      FileSystem H:\
H          HKCU       Registry   HKEY_CURRENT_USER
H          HKLM       Registry   HKEY_LOCAL_MACHINE
Variable   WSMAN      Variable   WSMAN

CurrentLocation
-----
Users\ME

PS C:\Users\ME>
```

24. Loop through a collection of the numbers(till 10), echo each number unless the number is 2

```
File Edit view tools Debug Add-ons Help
Untitled1.ps1* loop.ps1 X Untitled3.ps1

1 $a=$args[0]
2 for($i=1;$i -le 10;$i++)
3 {
4     $cal = $a * $i
5     echo $cal
6 }

PS C:\> .\loop.ps1

PS C:\> .\loop.ps1 2
2
4
6
8
10
12
14
16
18
20

PS C:\>
```

25. Write a power shell script to list the all the .text file in current directory.

```
Administrator: Command Prompt - PowerShell
PS C:\> Get-ChildItem -Path C:\PowerShell\
PS C:\> ls

Directory: C:\

Mode                LastWriteTime         Length Name
----                -
d-----          01-Dec-22    1:58 AM             cfrbackup-KXIZZQKI
d-----          27-Jul-22    1:59 PM             drivers
d-----          27-Jul-22    1:08 PM             Intel
d-----          13-Sep-22    4:30 PM             logs
d-----          07-Dec-19    2:44 PM             PerfLogs
d-----          01-Dec-22    2:17 AM             PowerShell
d-r-----        22-Oct-22    6:48 PM             Program Files
d-r-----        20-Oct-22   11:04 PM             Program Files (x86)
d-----          27-Jul-22    1:22 PM             soft
d-r-----        28-Jul-22    1:25 AM             Users
d-----          30-Nov-22    9:18 PM             Windows
-a-----        28-Jul-22    1:25 AM           8821 MRP_Debug1-[CY22M04D02-R144.BL]-1252pm.log
-a-----        28-Jul-22    1:23 AM           2518 MRP_EarlyRun_2983.log
-a-----        28-Jul-22    1:25 AM          15450 MRP_Project-[CY22M04D02-R144.BL]-1252pm.log

PS C:\> cd ./PowerShell
PS C:\PowerShell> ls

Directory: C:\PowerShell

Mode                LastWriteTime         Length Name
----                -
-a-----          01-Dec-22    2:20 AM             8 prithvi.txt

PS C:\PowerShell>
```

26. Write a power shell script to rename all .text files to .ps1 files.

```
Administrator: Command Prompt - PowerShell
PS C:\PowerShell> Rename-Item -Path "c:\PowerShell\prithvi.txt" -NewName "prithvi.ps1"
PS C:\PowerShell> ls

Directory: C:\PowerShell

Mode                LastWriteTime         Length Name
----                -
-a-----         01-Dec-22    2:20 AM             8 prithvi.ps1

PS C:\PowerShell>
```

27. Write a power shell script to check a file exists or not in your current working directory.

```
Administrator: Command Prompt - PowerShell
PS C:\PowerShell> Test-Path -path C:\PowerShell\prithvi.ps1
True
PS C:\PowerShell> Test-Path -path C:\PowerShell\prithvi.txt
False
PS C:\PowerShell> Test-Path -path C:\PowerShell\ram.txt
True
PS C:\PowerShell>
```

28. Write a power shell program to displays those files that are greater than 20K in size in given directory.

