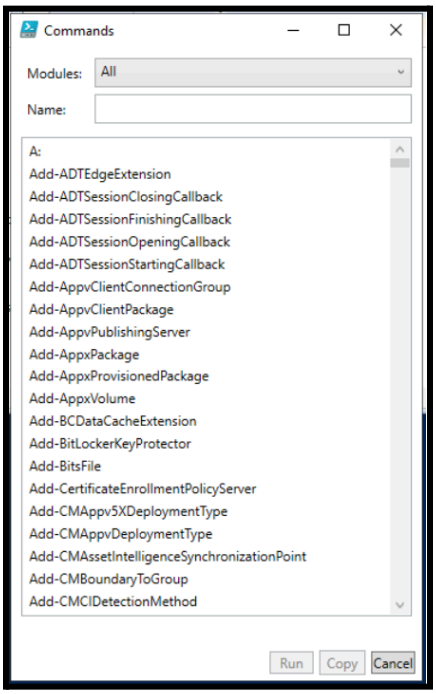
Show, Get, Start, Restart, Out and Format Commands

List of commands:

1) Show-Command

The Show-Command cmdlet opens a graphical window that displays the PowerShell cmdlets.

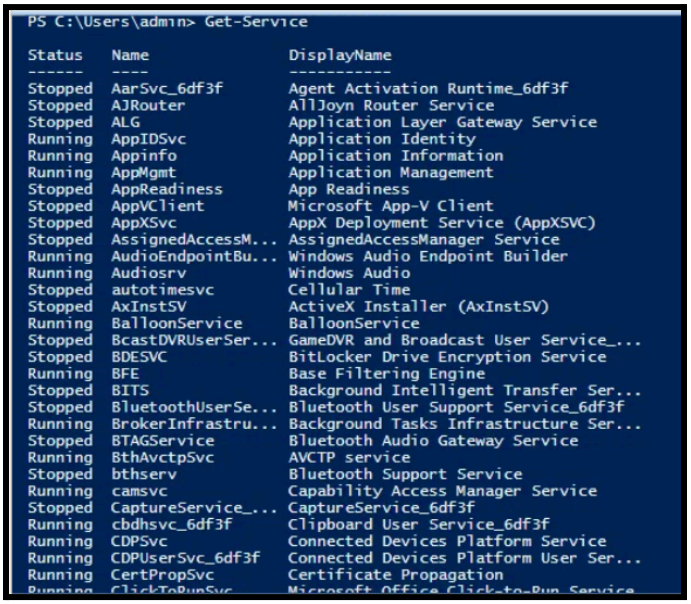
- Show-Command



2) Get-Service

The Get-Service cmdlet in PowerShell retrieves the services installed on a local or remote computer, including their status, name, and display name.

- Get-Service



3) Get-ComputerInfo

The Get-ComputerInfo cmdlet in PowerShell retrieves a consolidated object containing system and operating system properties.

- Get-ComputerInfo

```
PS C:\Users\admin> Get-ComputerInfo


WindowsBuildLabEx                           : 19041.1.amd64fre.vb_release.191206-1406
WindowsCurrentVersion                       : 6.3
WindowsEditionId                            : EnterpriseEval
WindowsInstallationType                     : Client
WindowsInstallDateFromRegistry              : 16-05-2025 07:05:36
WindowsProductId                            : 00329-20000-00001-AA613
WindowsProductName                          : Windows 10 Enterprise Evaluation
WindowsRegisteredOrganization               :
WindowsRegisteredOwner                      : admin
WindowsSystemRoot                           : C:\Windows
WindowsVersion                              : 2009
BiosCharacteristics                         : {4, 7, 8, 9...}
BiosBIOSVersion                             : {INTEL  - 6040000, PhoenixBIOS 4.0 Release 6.0     }
BiosBuildNumber                             :
BiosCaption                                 : PhoenixBIOS 4.0 Release 6.0
BiosCodeSet                                 :
BiosCurrentLanguage                         :
BiosDescription                             : PhoenixBIOS 4.0 Release 6.0
BiosEmbeddedControllerMajorVersion          : 0
BiosEmbeddedControllerMinorVersion          : 0
BiosFirmwareType                            : Bios
BiosIdentificationCode                      :
BiosInstallableLanguages                    :
BiosInstallDate                             :
BiosLanguageEdition                         :
BiosListOfLanguages                         :
BiosManufacturer                            : Phoenix Technologies LTD
BiosName                                    : PhoenixBIOS 4.0 Release 6.0
BiosOtherTargetOS                           :
BiosPrimaryBIOS                             : True
BiosReleaseDate                             : 03-07-2018 05:30:00
```

4) Get-Process

Retrieves information about running processes on the system.

- Get-Process -Name taskhostw

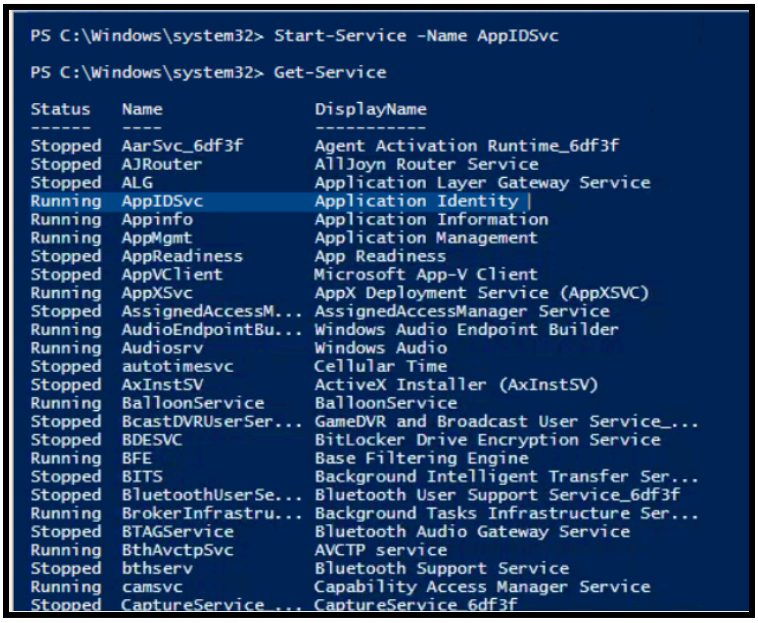
```
PS C:\Users\admin> Get-Process

Handles  NPM(K)    PM(K)      WS(K)     CPU(s)      Id  SI ProcessName
-------  ------    -----      -----     ------      --  -- -----------
    103       5     1032       5292                5636   0 AggregatorHost
    351      21    10364      32372       4.58     6068   1 ApplicationFrameHost
    158       9     1948       8400                3424   0 blnsvr
    136       9     6552       7724                4544   0 conhost
    645      24     2096       5876                 528   0 csrss
    585      22     2140       5964                 612   1 csrss
    754      18     5492      22464     185.05     6768   1 ctfmon
    361      17     9136      17576                5324   0 dasHost
    553      24     6260      15024       4.34     1728   1 dllhost
    215      18     3956      12300                4460   0 dllhost
    264      14     4048      13920                4724   0 dllhost
    943      53    47756      71832                1384   1 dwm
   2356     118   406280     246196   1,303.66     7088   1 explorer
     50       6     1516       3928                 924   0 fontdrvhost
     50       9     3968       9924                 932   1 fontdrvhost
      0       0       60          8                   0   0 Idle
    611      33     9396      36008      12.02    10120   1 IDMan
   1388      26     9396      22216                 752   0 lsass
      0       0      824      44036                2148   0 Memory Compression
    501      18    10232      22816                3608   0 MpDefenderCoreService
    230      13     2896       9372                3336   0 msdtc
    182      11     8152      21960       0.17     1108   1 msedge
    208      17    16468      32464       0.19     2464   1 msedge
    280      16    11492      29804       0.17     2524   1 msedge
    249      19    22976      55336       0.41     9012   1 msedge
    156       9     2152      10836       0.03     9252   1 msedge
    346      19    12108      38576       0.39    10444   1 msedge
   1291      45    46740     125840       2.94    10656   1 msedge
    915     226   301544     191196                3600   0 MsMpEng
```

```
PS C:\Users\admin> Get-Process -Name taskhostw

Handles  NPM(K)    PM(K)      WS(K)     CPU(s)      Id  SI ProcessName
-------  ------    -----      -----     ------      --  -- -----------
    440      24     8456      27068       2.34    6456   1 taskhostw
    283      30     6988      17296      12.16    6568   1 taskhostw
    401      22     7248      32576     103.28   10056   1 taskhostw
```

5) Start-Service

Starts a specified service.

- Start-Service -Name AppIDSvc

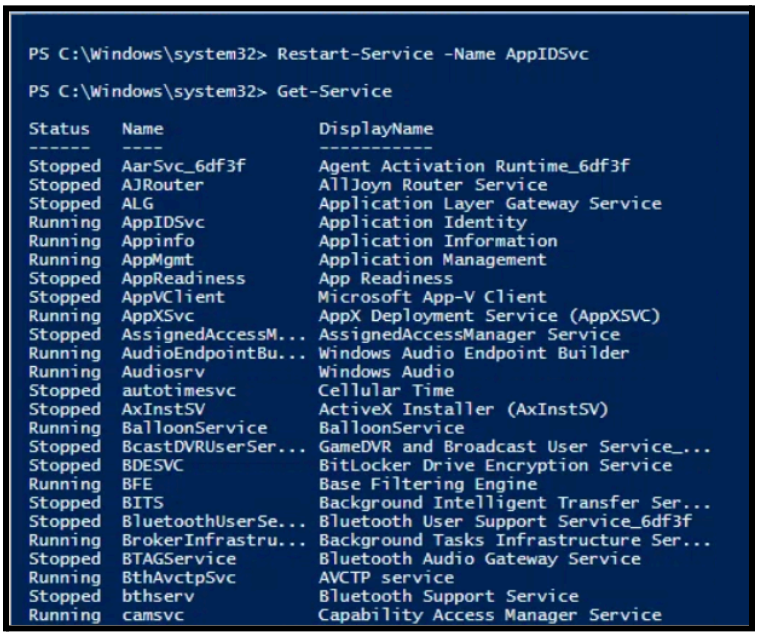


```
PS C:\Windows\system32> Start-Service -Name AppIDSvc

PS C:\Windows\system32> Get-Service

Status   Name               DisplayName
------   ----               -----------
Stopped  AarSvc_6df3f       Agent Activation Runtime_6df3f
Stopped  AJRouter           AllJoyn Router Service
Stopped  ALG                Application Layer Gateway Service
Running  AppIDSvc           Application Identity
Running  Appinfo            Application Information
Running  AppMgmt            Application Management
Stopped  AppReadiness       App Readiness
Stopped  AppVClient         Microsoft App-V Client
Running  AppXSvc            AppX Deployment Service (AppXSVC)
Stopped  AssignedAccessM... AssignedAccessManager Service
Running  AudioEndpointBu... Windows Audio Endpoint Builder
Running  Audiosrv           Windows Audio
Stopped  autotimesvc        Cellular Time
Stopped  AxInstSV           ActiveX Installer (AxInstSV)
Running  BalloonService     BalloonService
Stopped  BcastDVRUserSer... GameDVR and Broadcast User Service_...
Stopped  BDESVC             BitLocker Drive Encryption Service
Running  BFE                Base Filtering Engine
Stopped  BITS               Background Intelligent Transfer Ser...
Stopped  BluetoothUserSe... Bluetooth User Support Service_6df3f
Running  BrokerInfrastru... Background Tasks Infrastructure Ser...
Stopped  BTAGService        Bluetooth Audio Gateway Service
Running  BthAvctpSvc        AVCTP service
Stopped  bthserv            Bluetooth Support Service
Running  camsvc             Capability Access Manager Service
Stopped  CaptureService ... CaptureService_6df3f
```

6) Restart-Service

The Restart-Service cmdlet in PowerShell is used to send a stop message followed by a start message to the Windows Service Controller for a specified service.

- Restart-Service -Name AppIDSvc



```
PS C:\Windows\system32> Restart-Service -Name AppIDSvc

PS C:\Windows\system32> Get-Service

Status   Name               DisplayName
------   ----               -----------
Stopped  AarSvc_6df3f       Agent Activation Runtime_6df3f
Stopped  AJRouter           AllJoyn Router Service
Stopped  ALG                Application Layer Gateway Service
Running  AppIDSvc           Application Identity
Running  Appinfo            Application Information
Running  AppMgmt            Application Management
Stopped  AppReadiness       App Readiness
Stopped  AppVClient         Microsoft App-V Client
Running  AppXSvc            AppX Deployment Service (AppXSVC)
Stopped  AssignedAccessM... AssignedAccessManager Service
Running  AudioEndpointBu... Windows Audio Endpoint Builder
Running  Audiosrv           Windows Audio
Stopped  autotimesvc        Cellular Time
Stopped  AxInstSV           ActiveX Installer (AxInstSV)
Running  BalloonService     BalloonService
Stopped  BcastDVRUserSer... GameDVR and Broadcast User Service_...
Stopped  BDESVC             BitLocker Drive Encryption Service
Running  BFE                Base Filtering Engine
Stopped  BITS               Background Intelligent Transfer Ser...
Stopped  BluetoothUserSe... Bluetooth User Support Service_6df3f
Running  BrokerInfrastru... Background Tasks Infrastructure Ser...
Stopped  BTAGService        Bluetooth Audio Gateway Service
Running  BthAvctpSvc        AVCTP service
Stopped  bthserv            Bluetooth Support Service
Running  camsvc             Capability Access Manager Service
```

7) Get-EventLog

The Get-EventLog cmdlet in PowerShell is used to retrieve events from classic event logs on a local or remote computer.

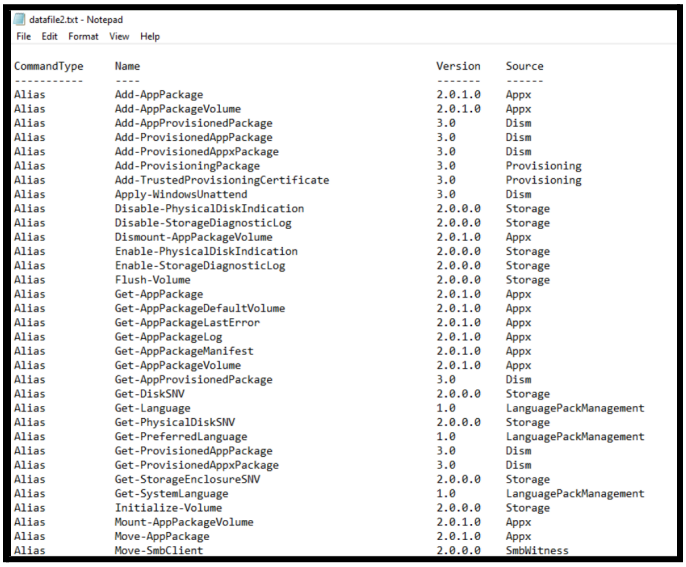
- Get-EventLog

```
PS C:\Windows\system32> Get-EventLog
cmdlet Get-EventLog at command pipeline position 1
Supply values for the following parameters:
LogName: System

   Index Time          EntryType   Source                 InstanceID Message
   ----- ----          ---------   ------                 ---------- -------
    5786 Jun 10 05:01  Information Service Control M...   1073748864 The start type of the Windows Update service was changed from demand start to disabled.
    5785 Jun 09 22:49  Information Service Control M...   1073748864 The start type of the Background Intelligent Transfer Service service was changed from auto start to demand s...
    5784 Jun 09 22:47  Information Service Control M...   1073748864 The start type of the Background Intelligent Transfer Service service was changed from demand start to auto s...
    5783 Jun 09 20:39  Information Service Control M...   1073748864 The start type of the Windows Update service was changed from disabled to demand start.
    5782 Jun 09 20:01  Information Service Control M...   1073748864 The start type of the Update Orchestrator Service service was changed from auto start to disabled.
    5781 Jun 09 20:01  Information Service Control M...   1073748864 The start type of the Windows Update service was changed from demand start to disabled.
    5780 Jun 09 15:49  Information Service Control M...   1073748864 The start type of the Background Intelligent Transfer Service service was changed from auto start to demand s...
    5779 Jun 09 15:47  Information Service Control M...   1073748864 The start type of the Background Intelligent Transfer Service service was changed from demand start to auto s...
    5778 Jun 09 13:53  Information Microsoft-Windows...          19 Installation Successful: Windows successfully installed the following update: Security Intelligence Update fo...
    5777 Jun 09 13:53  Information Microsoft-Windows...          43 Installation Started: Windows has started installing the following update: Security Intelligence Update for M...
    5776 Jun 09 13:53  Information Microsoft-Windows...          44 Windows Update started downloading an update.
    5775 Jun 09 13:44  Information Service Control M...   1073748864 The start type of the Background Intelligent Transfer Service service was changed from auto start to demand s...
    5774 Jun 09 13:42  Information Service Control M...   1073748864 The start type of the Background Intelligent Transfer Service service was changed from demand start to auto s...
    5773 Jun 09 12:00  Information EventLog               2147489661 The system uptime is 1376270 seconds.
    5772 Jun 09 07:42  Information Service Control M...   1073748864 The start type of the Update Orchestrator Service service was changed from disabled to auto start.
    5771 Jun 09 07:42  Information Service Control M...   1073748864 The start type of the Windows Update service was changed from disabled to demand start.
    5770 Jun 09 05:01  Information Service Control M...   1073748864 The start type of the Windows Update service was changed from demand start to disabled.
    5769 Jun 09 03:41  Information Service Control M...   1073748864 The start type of the Windows Update service was changed from disabled to demand start.
    5768 Jun 09 03:00  Information Microsoft-Windows...          37 The time provider NtpClient is currently receiving valid time data from time.windows.com,0x9 (ntp.m|0x9|0.0.0...
    5767 Jun 08 20:00  Information Service Control M...   1073748864 The start type of the Update Orchestrator Service service was changed from auto start to disabled.
    5766 Jun 08 20:00  Information Service Control M...   1073748864 The start type of the Windows Update service was changed from demand start to disabled.
    5765 Jun 08 18:54  Information Service Control M...   1073748864 The start type of the Background Intelligent Transfer Service service was changed from auto start to demand s...
    5764 Jun 08 18:52  Information Service Control M...   1073748864 The start type of the Background Intelligent Transfer Service service was changed from demand start to auto s...
    5763 Jun 08 18:19  Error       Server                 3221227977 The server could not bind to the transport \Device\NetBT_Tcpip_{0DA33E65-86F7-4EC9-AA69-1906821D5978} because...
```

8) Out-File

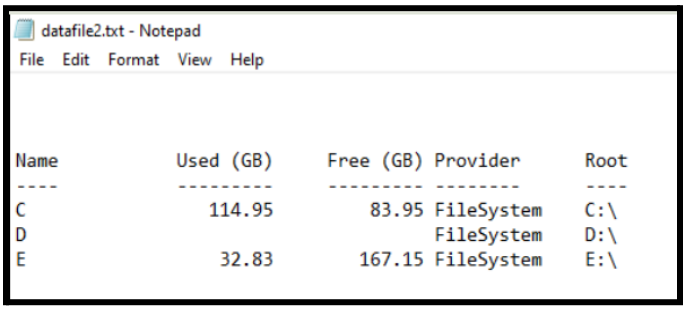
The Out-File cmdlet in PowerShell is used to send output to a file.

- Get-Command | Out-File E:\demo\datafile2.txt



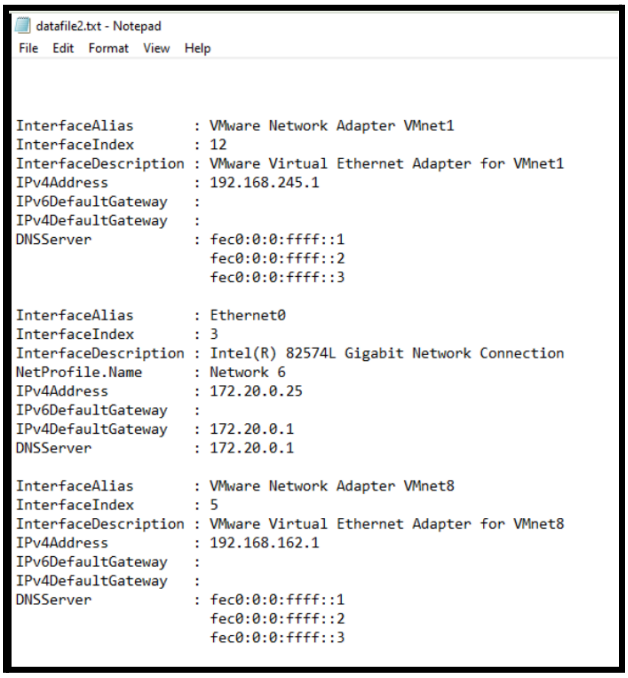
```
datafile2.txt - Notepad
File  Edit  Format  View  Help

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           Add-AppPackage                                     2.0.1.0    Appx
Alias           Add-AppPackageVolume                               2.0.1.0    Appx
Alias           Add-AppProvisionedPackage                          3.0        Dism
Alias           Add-ProvisionedAppPackage                          3.0        Dism
Alias           Add-ProvisionedAppxPackage                         3.0        Dism
Alias           Add-ProvisioningPackage                            3.0        Provisioning
Alias           Add-TrustedProvisioningCertificate                 3.0        Provisioning
Alias           Apply-WindowsUnattend                              3.0        Dism
Alias           Disable-PhysicalDiskIndication                     2.0.0.0    Storage
Alias           Disable-StorageDiagnosticLog                       2.0.0.0    Storage
Alias           Dismount-AppPackageVolume                          2.0.1.0    Appx
Alias           Enable-PhysicalDiskIndication                      2.0.0.0    Storage
Alias           Enable-StorageDiagnosticLog                        2.0.0.0    Storage
Alias           Flush-Volume                                       2.0.0.0    Storage
Alias           Get-AppPackage                                     2.0.1.0    Appx
Alias           Get-AppPackageDefaultVolume                        2.0.1.0    Appx
Alias           Get-AppPackageLastError                            2.0.1.0    Appx
Alias           Get-AppPackageLog                                  2.0.1.0    Appx
Alias           Get-AppPackageManifest                             2.0.1.0    Appx
Alias           Get-AppPackageVolume                               2.0.1.0    Appx
Alias           Get-AppProvisionedPackage                          3.0        Dism
Alias           Get-DiskSNV                                        2.0.0.0    Storage
Alias           Get-Language                                       1.0        LanguagePackManagement
Alias           Get-PhysicalDiskSNV                                2.0.0.0    Storage
Alias           Get-PreferredLanguage                              1.0        LanguagePackManagement
Alias           Get-ProvisionedAppPackage                          3.0        Dism
Alias           Get-ProvisionedAppxPackage                         3.0        Dism
Alias           Get-StorageEnclosureSNV                            2.0.0.0    Storage
Alias           Get-SystemLanguage                                 1.0        LanguagePackManagement
Alias           Initialize-Volume                                  2.0.0.0    Storage
Alias           Mount-AppPackageVolume                             2.0.1.0    Appx
Alias           Move-AppPackage                                    2.0.1.0    Appx
Alias           Move-SmbClient                                     2.0.0.0    SmbWitness
```

- Get-PSDrive -PSProvider FileSystem | Out-File E:\demo\datafile2.txt -Append



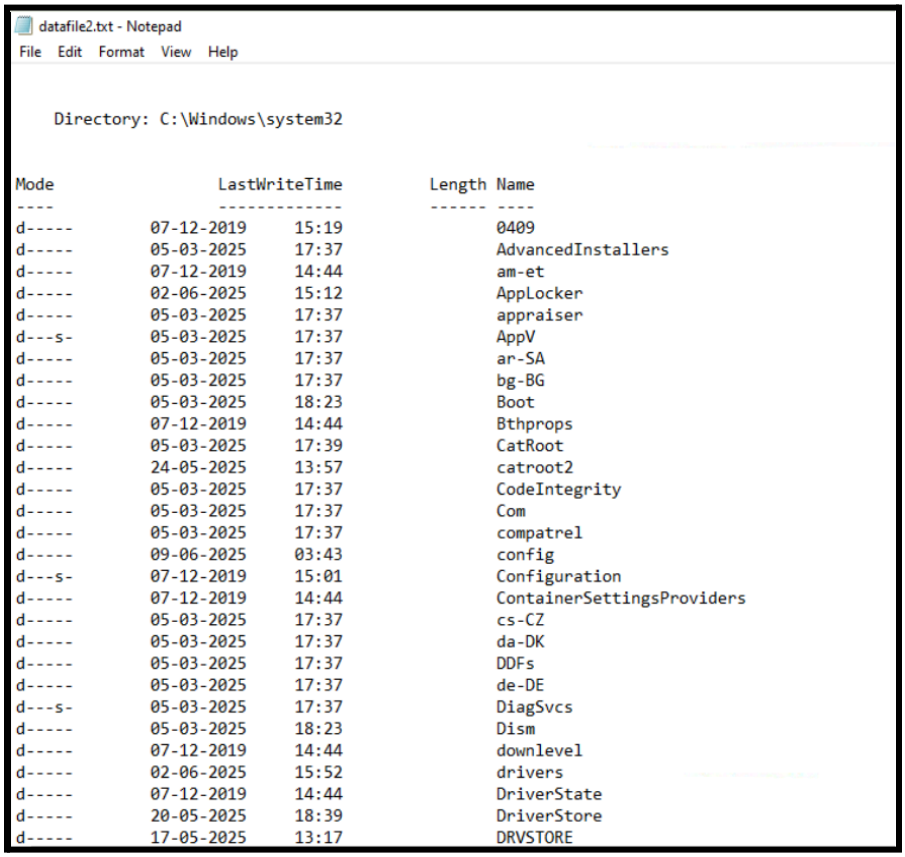
```
datafile2.txt - Notepad
File  Edit  Format  View  Help


Name           Used (GB)     Free (GB) Provider      Root
----           ---------     --------- --------      ----
C                 114.95         83.95 FileSystem    C:\
D                                       FileSystem    D:\
E                  32.83        167.15 FileSystem    E:\
```

- Get-NetIPConfiguration | Out-File E:\demo\datafile2.txt -Append



```
datafile2.txt - Notepad
File  Edit  Format  View  Help


InterfaceAlias       : VMware Network Adapter VMnet1
InterfaceIndex       : 12
InterfaceDescription : VMware Virtual Ethernet Adapter for VMnet1
IPv4Address          : 192.168.245.1
IPv6DefaultGateway   :
IPv4DefaultGateway   :
DNSServer            : fec0:0:0:ffff::1
                       fec0:0:0:ffff::2
                       fec0:0:0:ffff::3

InterfaceAlias       : Ethernet0
InterfaceIndex       : 3
InterfaceDescription : Intel(R) 82574L Gigabit Network Connection
NetProfile.Name      : Network 6
IPv4Address          : 172.20.0.25
IPv6DefaultGateway   :
IPv4DefaultGateway   : 172.20.0.1
DNSServer            : 172.20.0.1

InterfaceAlias       : VMware Network Adapter VMnet8
InterfaceIndex       : 5
InterfaceDescription : VMware Virtual Ethernet Adapter for VMnet8
IPv4Address          : 192.168.162.1
IPv6DefaultGateway   :
IPv4DefaultGateway   :
DNSServer            : fec0:0:0:ffff::1
                       fec0:0:0:ffff::2
                       fec0:0:0:ffff::3
```

- Get-ChildItem | Out-File E:\demo\datafile2.txt -Append



```
datafile2.txt - Notepad
File  Edit  Format  View  Help


    Directory: C:\Windows\system32


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        07-12-2019     15:19                0409
d-----        05-03-2025     17:37                AdvancedInstallers
d-----        07-12-2019     14:44                am-et
d-----        02-06-2025     15:12                AppLocker
d-----        05-03-2025     17:37                appraiser
d---s-        05-03-2025     17:37                AppV
d-----        05-03-2025     17:37                ar-SA
d-----        05-03-2025     17:37                bg-BG
d-----        05-03-2025     18:23                Boot
d-----        07-12-2019     14:44                Bthprops
d-----        05-03-2025     17:39                CatRoot
d-----        24-05-2025     13:57                catroot2
d-----        05-03-2025     17:37                CodeIntegrity
d-----        05-03-2025     17:37                Com
d-----        05-03-2025     17:37                compatrel
d-----        09-06-2025     03:43                config
d---s-        07-12-2019     15:01                Configuration
d-----        07-12-2019     14:44                ContainerSettingsProviders
d-----        05-03-2025     17:37                cs-CZ
d-----        05-03-2025     17:37                da-DK
d-----        05-03-2025     17:37                DDFs
d-----        05-03-2025     17:37                de-DE
d---s-        05-03-2025     17:37                DiagSvcs
d-----        05-03-2025     18:23                Dism
d-----        07-12-2019     14:44                downlevel
d-----        02-06-2025     15:52                drivers
d-----        07-12-2019     14:44                DriverState
d-----        20-05-2025     18:39                DriverStore
d-----        17-05-2025     13:17                DRVSTORE
```

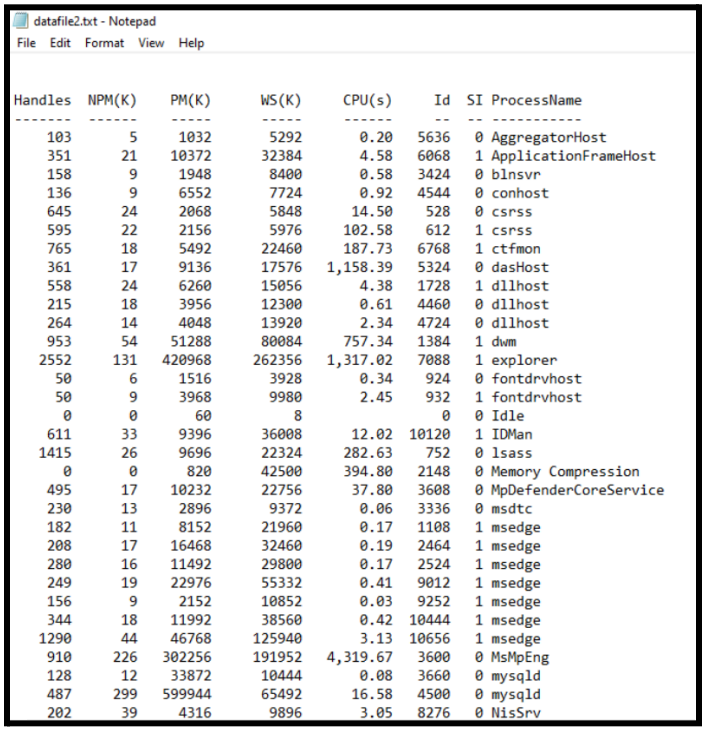
- Get-Service | Out-File E:\demo\datafile2.txt -Append

datafile2.txt - Notepad

File Edit Format View Help

```
Status   Name               DisplayName
------   ----               -----------
Stopped  AarSvc_6df3f       Agent Activation Runtime_6df3f
Stopped  AJRouter           AllJoyn Router Service
Stopped  ALG                Application Layer Gateway Service
Running  AppIDSvc           Application Identity
Running  Appinfo            Application Information
Running  AppMgmt            Application Management
Stopped  AppReadiness       App Readiness
Stopped  AppVClient         Microsoft App-V Client
Running  AppXSvc            AppX Deployment Service (AppXSVC)
Stopped  AssignedAccessM... AssignedAccessManager Service
Running  AudioEndpointBu... Windows Audio Endpoint Builder
Running  Audiosrv           Windows Audio
Stopped  autotimesvc        Cellular Time
Stopped  AxInstSV           ActiveX Installer (AxInstSV)
Running  BalloonService     BalloonService
Stopped  BcastDVRUserSer... GameDVR and Broadcast User Service_...
Stopped  BDESVC             BitLocker Drive Encryption Service
Running  BFE                Base Filtering Engine
Stopped  BITS               Background Intelligent Transfer Ser...
Stopped  BluetoothUserSe... Bluetooth User Support Service_6df3f
Running  BrokerInfrastru... Background Tasks Infrastructure Ser...
Stopped  BTAGService        Bluetooth Audio Gateway Service
Running  BthAvctpSvc        AVCTP service
Stopped  bthserv            Bluetooth Support Service
Running  camsvc             Capability Access Manager Service
Stopped  CaptureService_... CaptureService_6df3f
Running  cbdhsvc_6df3f      Clipboard User Service_6df3f
Running  CDPSvc             Connected Devices Platform Service
Running  CDPUserSvc_6df3f   Connected Devices Platform User Ser...
Running  CertPropSvc        Certificate Propagation
Running  ClickToRunSvc      Microsoft Office Click-to-Run Service
Stopped  ClipSVC            Client License Service (ClipSVC)
```

- Get-Process | Out-File E:\demo\datafile2.txt -Append



datafile2.txt - Notepad

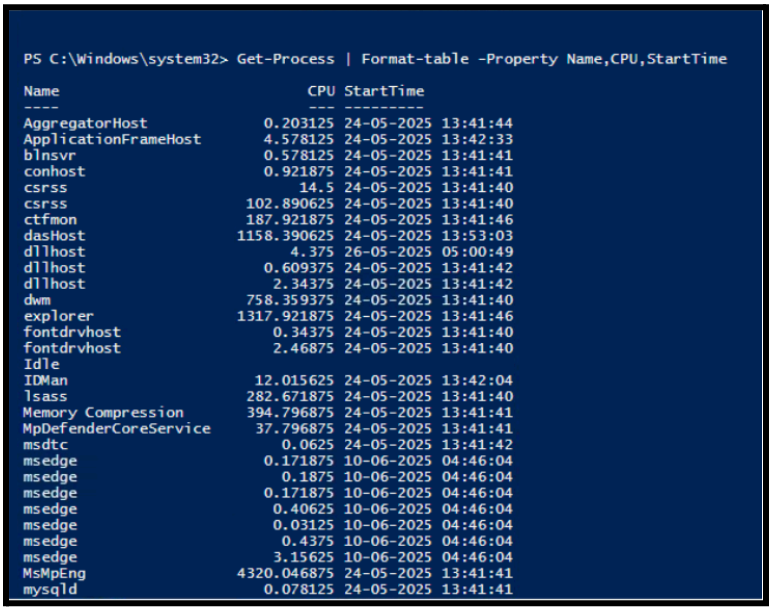
File Edit Format View Help

```
Handles  NPM(K)    PM(K)      WS(K)     CPU(s)      Id  SI ProcessName
-------  ------    -----      -----     ------      --  -- -----------
    103       5     1032       5292       0.20    5636   0 AggregatorHost
    351      21    10372      32384       4.58    6068   1 ApplicationFrameHost
    158       9     1948       8400       0.58    3424   0 blnsvr
    136       9     6552       7724       0.92    4544   0 conhost
    645      24     2068       5848      14.50     528   0 csrss
    595      22     2156       5976     102.58     612   1 csrss
    765      18     5492      22460     187.73    6768   1 ctfmon
    361      17     9136      17576   1,158.39    5324   0 dasHost
    558      24     6260      15056       4.38    1728   1 dllhost
    215      18     3956      12300       0.61    4460   0 dllhost
    264      14     4048      13920       2.34    4724   0 dllhost
    953      54    51288      80084     757.34    1384   1 dwm
   2552     131   420968     262356   1,317.02    7088   1 explorer
     50       6     1516       3928       0.34     924   0 fontdrvhost
     50       9     3968       9980       2.45     932   1 fontdrvhost
      0       0       60          8                  0   0 Idle
    611      33     9396      36008      12.02   10120   1 IDMan
   1415      26     9696      22324     282.63     752   0 lsass
      0       0      820      42500     394.80    2148   0 Memory Compression
    495      17    10232      22756      37.80    3608   0 MpDefenderCoreService
    230      13     2896       9372       0.06    3336   0 msdtc
    182      11     8152      21960       0.17    1108   1 msedge
    208      17    16468      32460       0.19    2464   1 msedge
    280      16    11492      29800       0.17    2524   1 msedge
    249      19    22976      55332       0.41    9012   1 msedge
    156       9     2152      10852       0.03    9252   1 msedge
    344      18    11992      38560       0.42   10444   1 msedge
   1290      44    46768     125940       3.13   10656   1 msedge
    910     226   302256     191952   4,319.67    3600   0 MsMpEng
    128      12    33872      10444       0.08    3660   0 mysqld
    487     299   599944      65492      16.58    4500   0 mysqld
    202      39     4316       9896       3.05    8276   0 NisSrv
```

9) Format-Table

The Format-Table cmdlet in PowerShell formats the output of a command as a table with the selected properties of the object in each column.

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

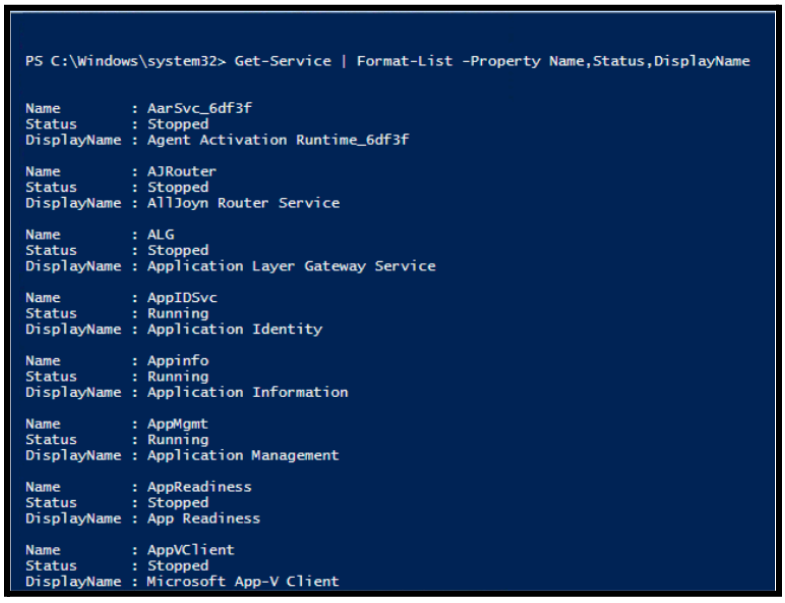


```
PS C:\Windows\system32> Get-Process | Format-table -Property Name,CPU,StartTime

Name                           CPU StartTime
----                           --- ---------
AggregatorHost            0.203125 24-05-2025 13:41:44
ApplicationFrameHost      4.578125 24-05-2025 13:42:33
blnsvr                    0.578125 24-05-2025 13:41:41
conhost                   0.921875 24-05-2025 13:41:41
csrss                         14.5 24-05-2025 13:41:40
csrss                   102.890625 24-05-2025 13:41:40
ctfmon                  187.921875 24-05-2025 13:41:46
dasHost                1158.390625 24-05-2025 13:53:03
dllhost                     4.375 26-05-2025 05:00:49
dllhost                  0.609375 24-05-2025 13:41:42
dllhost                   2.34375 24-05-2025 13:41:42
dwm                     758.359375 24-05-2025 13:41:40
explorer               1317.921875 24-05-2025 13:41:46
fontdrvhost               0.34375 24-05-2025 13:41:40
fontdrvhost               2.46875 24-05-2025 13:41:40
Idle
IDMan                    12.015625 24-05-2025 13:42:04
lsass                   282.671875 24-05-2025 13:41:40
Memory Compression      394.796875 24-05-2025 13:41:41
MpDefenderCoreService    37.796875 24-05-2025 13:41:41
msdtc                       0.0625 24-05-2025 13:41:42
msedge                   0.171875 10-06-2025 04:46:04
msedge                      0.1875 10-06-2025 04:46:04
msedge                   0.171875 10-06-2025 04:46:04
msedge                    0.40625 10-06-2025 04:46:04
msedge                    0.03125 10-06-2025 04:46:04
msedge                     0.4375 10-06-2025 04:46:04
msedge                    3.15625 10-06-2025 04:46:04
MsMpEng                4320.046875 24-05-2025 13:41:41
mysqld                   0.078125 24-05-2025 13:41:41
```

10) Format-List

The Format-List cmdlet in PowerShell formats the output of a command as a list of properties, displaying each property on a separate line.

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

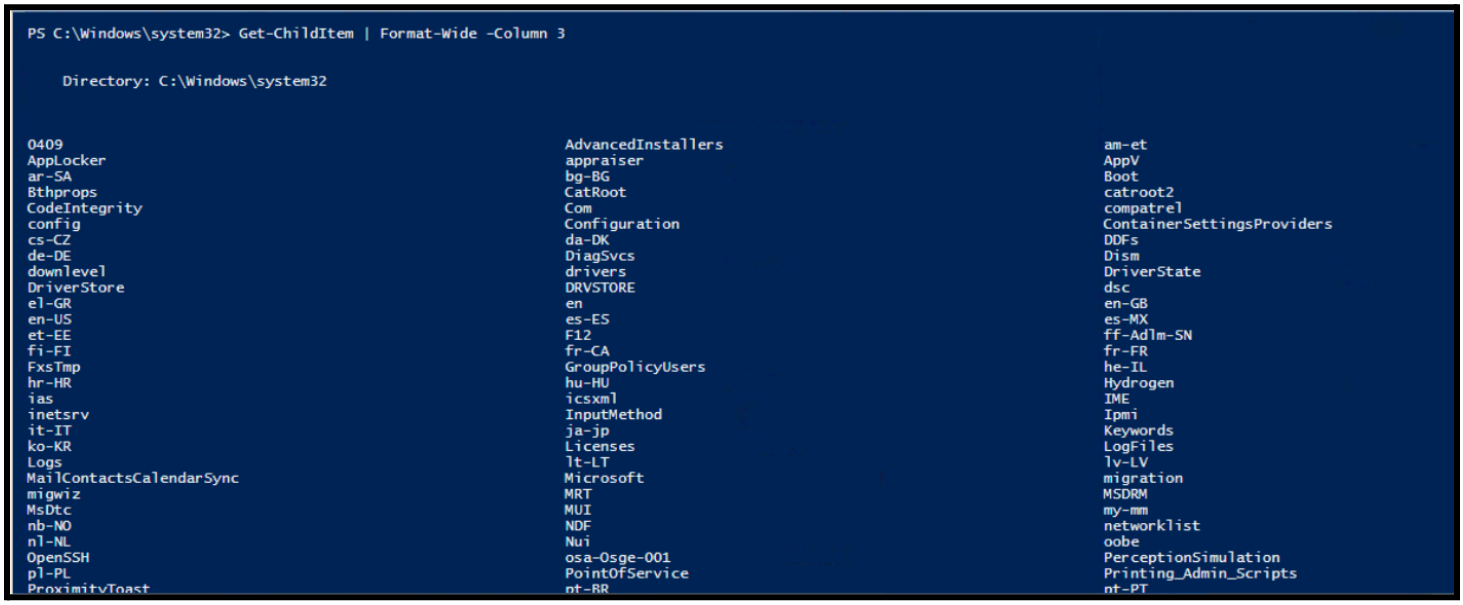


```
PS C:\Windows\system32> Get-Service | Format-List -Property Name,Status,DisplayName


Name        : AarSvc_6df3f
Status      : Stopped
DisplayName : Agent Activation Runtime_6df3f

Name        : AJRouter
Status      : Stopped
DisplayName : AllJoyn Router Service

Name        : ALG
Status      : Stopped
DisplayName : Application Layer Gateway Service

Name        : AppIDSvc
Status      : Running
DisplayName : Application Identity

Name        : Appinfo
Status      : Running
DisplayName : Application Information

Name        : AppMgmt
Status      : Running
DisplayName : Application Management

Name        : AppReadiness
Status      : Stopped
DisplayName : App Readiness

Name        : AppVClient
Status      : Stopped
DisplayName : Microsoft App-V Client
```

11) Format-Wide

The Format-Wide cmdlet in PowerShell formats objects as a wide table that displays only one property of each object.

- Get-ChildItem | Format-Wide -Column 3



Objects, Arrays, Variables, Scripting Constraints

❖ Variable

Variables are used to store values. They are denoted by $ symbol followed by name. They are not case sensitive. They can include letters, numbers and underscore. There is no need to declare datatypes of variables. However, we can use type casting if needed.

Code Example:

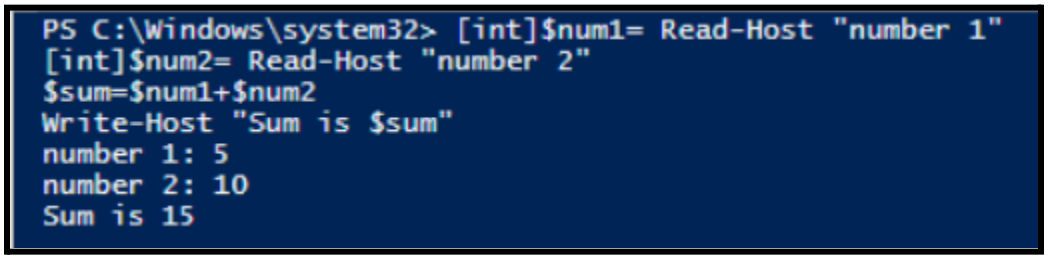
```
$name=”Siddhesh”
$age=21
$isTrue=$true
[int]$number=”123”
```

❖ Input and Output Formatting

1) **Read-Host**: This prompts the user for input and stores it as a string.
2) **Write-Host**: Displays output directly to the console.

Code Example:

```
[int]$num1= Read-Host "number 1"
[int]$num2= Read-Host "number 2"
$sum=$num1+$num2
Write-Host "Sum is $sum"
```

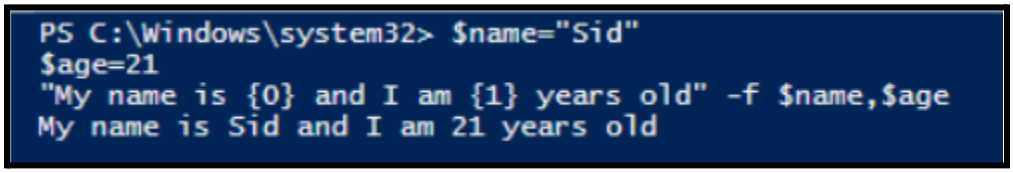


❖ String Formatting

-f format operator allows for composite formatting.

Code Example:

```
$name=”Sid”
$age=21
“My name is {0} and I am {1} years old” -f $name,$age
```

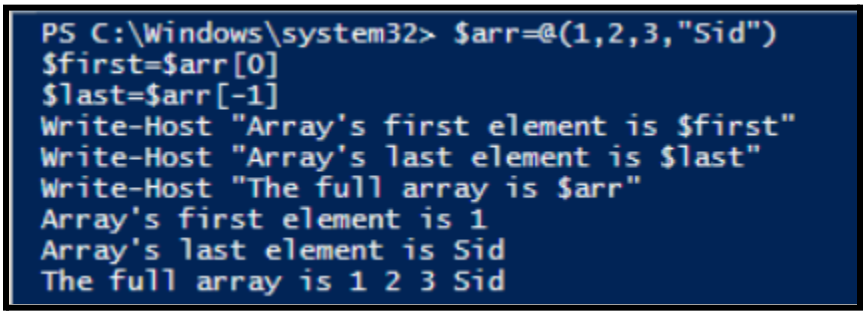


❖ Array

Arrays are used to store collections of items. An array can hold multiple objects of the same or different types. We can also declare them directly with @(elements).

Code Example:

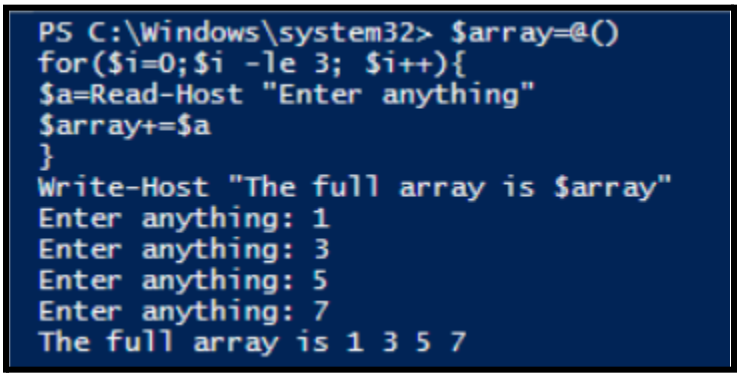
```
$arr=@(1,2,3,"Sid")
$first=$arr[0]
$last=$arr[-1]
Write-Host "Array's first element is $first"
Write-Host "Array's last element is $last"
Write-Host "The full array is $arr"
```



Size of an array can be adjusted. We can add elements using += operator.

Code Example:

```
$array=@()
for($i=0;$i -le 3; $i++){
$a=Read-Host "Enter anything"
$array+=$a
}
```

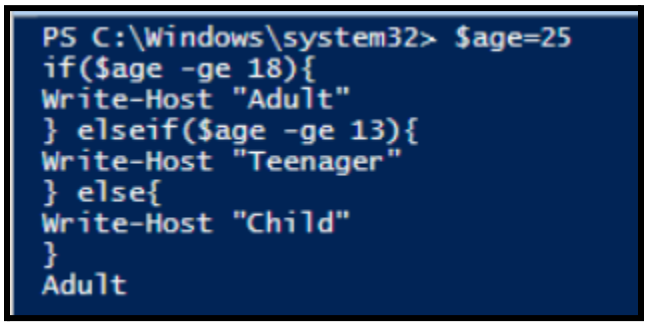


❖ Conditional Statements

1) **if, elseif, else:** These are used to execute code blocks based on conditions.

Code Example:

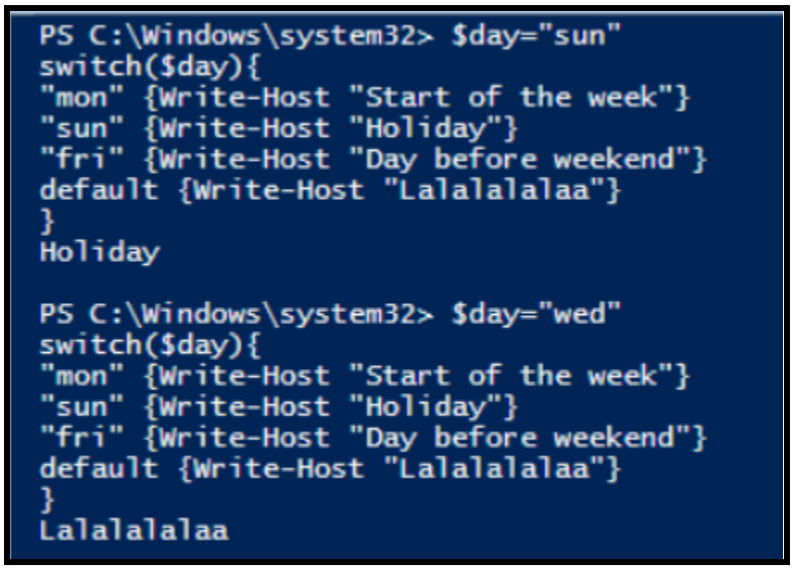
```
$age=25
if($age -ge 18){
Write-Host “Adult”
} elseif($age -ge 13){
Write-Host “Teenager”
} else{
Write-Host “Child”
}
```



2) **switch:** Switch case efficiently handles multiple conditions.

Code Example:

```
$day="sun"
switch($day){
"mon" {Write-Host "Start of the week"}
"sun" {Write-Host "Holiday"}
"fri" {Write-Host "Day before weekend"}
default {Write-Host "Lalalalalaa"}
}
```



❖ Looping Statements

1) **for:** For loop iterates a specific number of times.

Code Example:

```
for($i=-3;$i -le 0;$i++){
Write-Host "$i"
}
```

```
PS C:\Windows\system32> for($i=-3;$i -le 0;$i++){
Write-Host "$i"
}
-3
-2
-1
0
```

2) **foreach:** Foreach loop iterates through a collection.

Code Example:

```
$colors=”red”,”green”,”blue”
foreach($elem in $colors){
Write-Host "$elem"
}
```

```
PS C:\Windows\system32> $colors="red","green","blue"
foreach($elem in $colors){
Write-Host "$elem"
}
red
green
blue
```

3) **while:** While loop repeats as long as a condition is true.

Code Example:

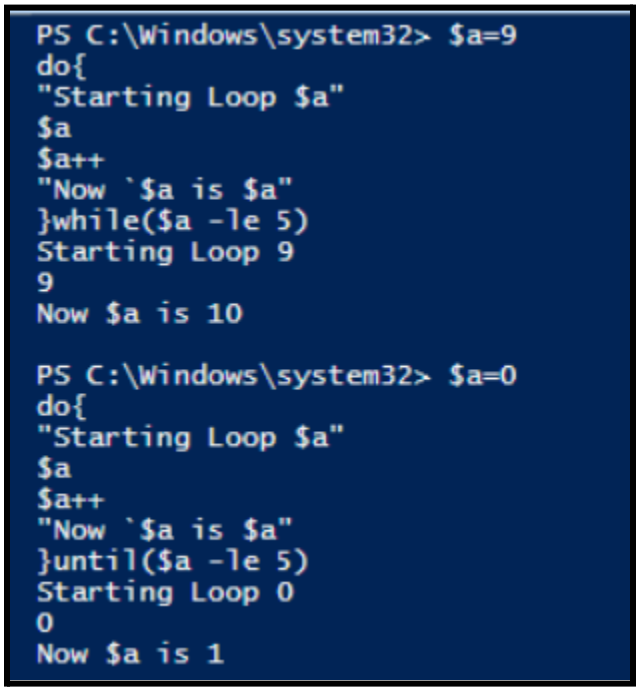
```
$count=0
while($count -lt 3){
Write-Host "Count: $count"
$count++
}
```

```
PS C:\Windows\system32> $count=0
while($count -lt 3){
Write-Host "Count: $count"
$count++
}
Count: 0
Count: 1
Count: 2
```

4) **do-while** & **do-until:** They are similar to while but the condition is checked at the end.

Code Example:

```
$a=9
do{
"Starting Loop $a"
$a
$a++
"Now `$a is $a"
}while($a -le 5)

$a=0
do{
"Starting Loop $a"
$a
$a++
"Now `$a is $a"
}until($a -le 5)
```

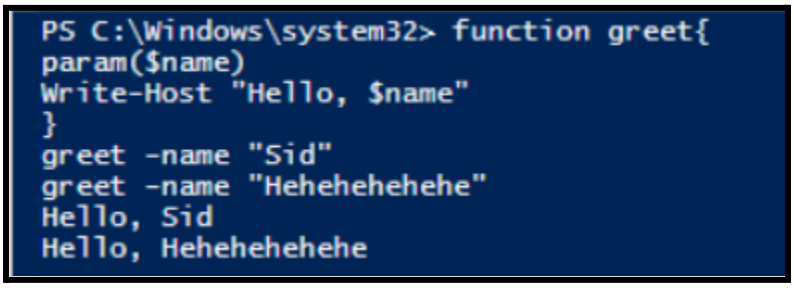


❖ Functions

PowerShell functions are blocks of code designed to perform specific tasks making scripts more modular, reusable and easier to maintain. To define a basic function, we use the function keyword followed by the function name and a block of code enclosed in curly braces { }. To invoke or call a function, simply use the function's name. Parameters can be added to the function declaration to make them more flexible. The keyword `param` is used to declare all parameters in the first function statement. To invoke a function with arguments, the values are passed to the function as parameters or list arguments after the function call.

Code Example:

```
function greet{
param($name)
Write-Host "Hello, $name"
}
greet -name "Sid"
greet -name "Hehehehehehe"
```

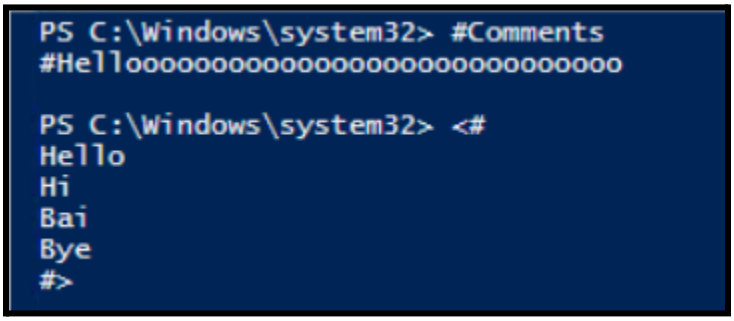


❖ Comments

PowerShell allows for two types of comments: single-line and multi-line (block) comments. Single-line comments are created using the hash symbol (#) whereas Multi-line comments are created using the <#.....#> delimiters.

Code Example:

```
#Hellooooooooooooooooooooooooooooo

<#
Hello
Hi
Bai
Bye
#>
```



❖ Error Handling

PowerShell error handling is essential for creating robust and reliable scripts. It involves identifying, managing, and responding to errors that occur during script execution. In PowerShell, there are two types of errors: terminating and non-terminating. Terminating errors stop script execution, while non-terminating errors allow the script to continue running but can affect the output.

To handle errors effectively, we can use the Try / Catch / Finally block. The Try block contains code that could throw an exception. If that happens, control is transferred to the Catch block which handles the error by taking actions such as skipping a problematic file or invalid input. The Finally block always executes, performing cleanup tasks such as closing files, releasing resources or logging information. To catch a non-terminating error with a Try / Catch block, we must convert it into a terminating error. This can be done using the -ErrorAction Stop parameter.

Code Example:

```
try{
Get-Content “nonexistent_file.txt” -ErrorAction Stop
} catch{
Write-Host “Error: $($_.Exception.Message)”
} finally{
Write-Host “Cleanup actions”
}
```