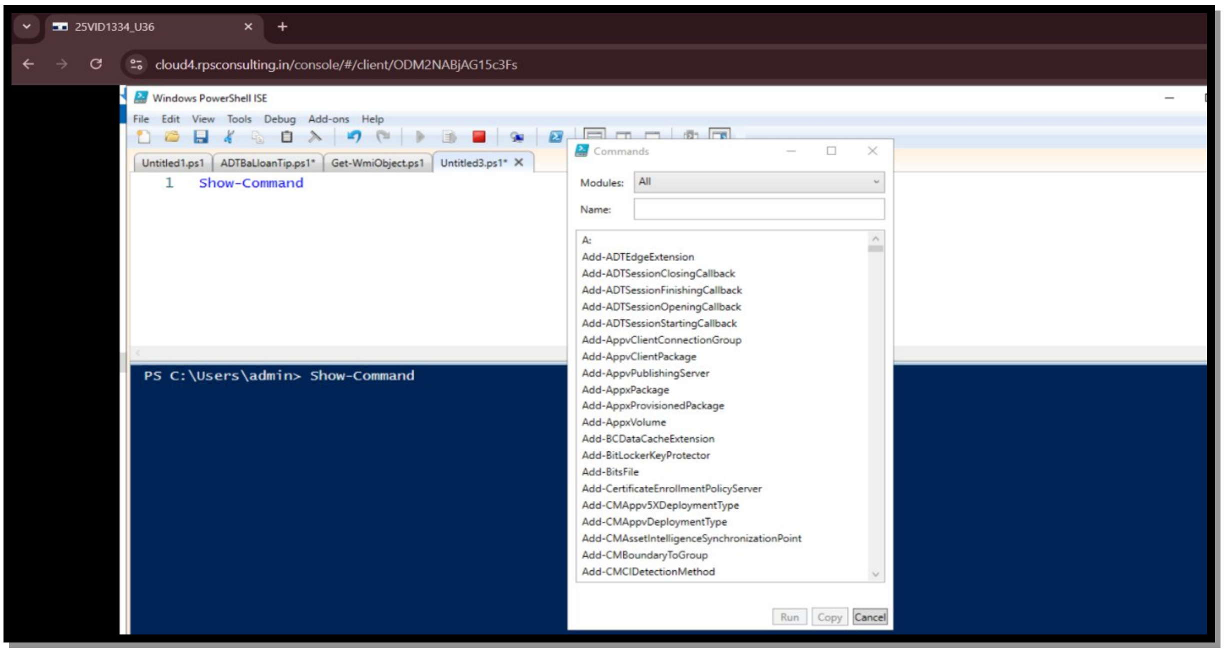
Show, Get, Start ,Restart, Out and Format Commands

List of commands

1) Show-Command

The Show-Command cmdlet lets you create a PowerShell command in a command-window.

*Show-Command is a PowerShell cmdlet that **displays a graphical window (GUI form)** to help users **interactively fill in parameters** for any PowerShell cmdlet or function.*



```
Show-Command
```

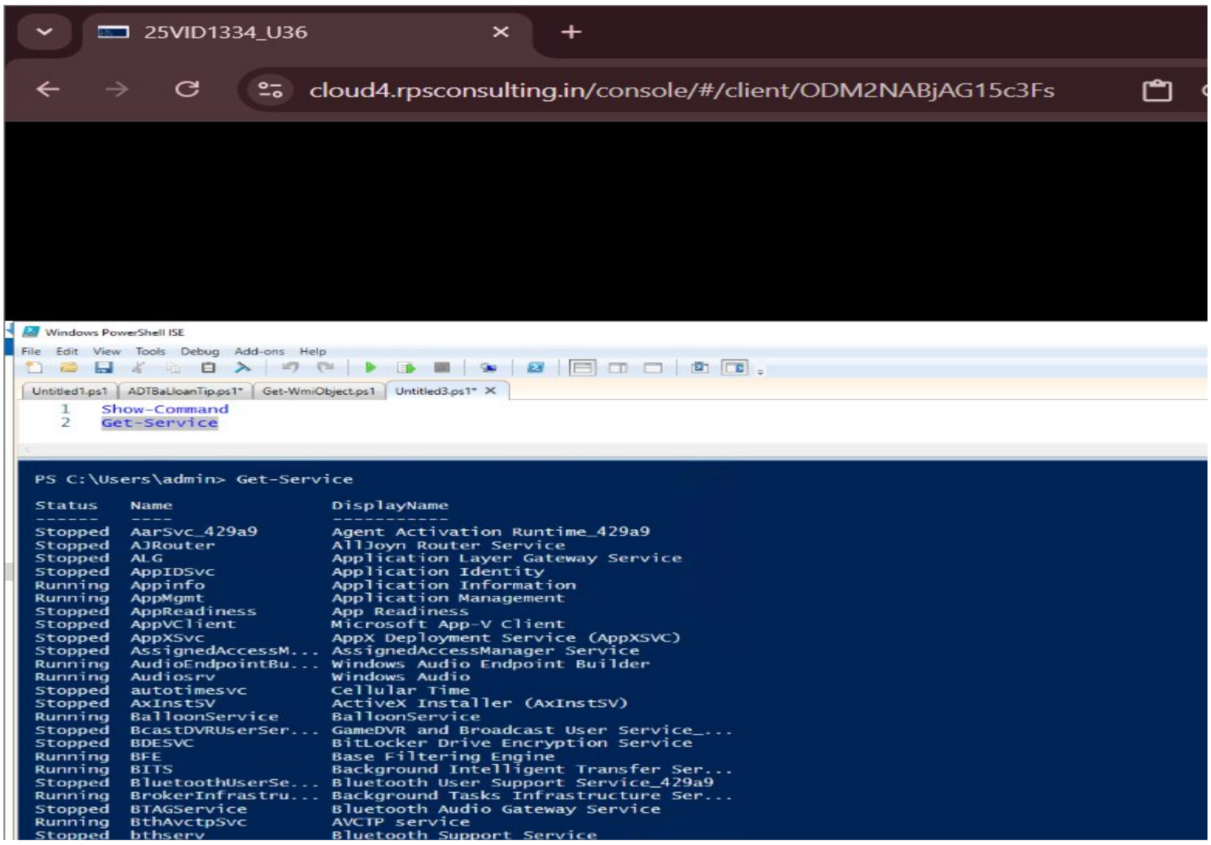
```
PS C:\Users\admin> Show-Command
```

```
A:
Add-ADTEdgeExtension
Add-ADTSessionClosingCallback
Add-ADTSessionFinishingCallback
Add-ADTSessionOpeningCallback
Add-ADTSessionStartingCallback
Add-AppvClientConnectionGroup
Add-AppvClientPackage
Add-AppvPublishingServer
Add-AppxPackage
Add-AppxProvisionedPackage
Add-AppxVolume
Add-BCDataCacheExtension
Add-BitLockerKeyProtector
Add-BitsFile
Add-CertificateEnrollmentPolicyServer
Add-CMAppv5XDeploymentType
Add-CMAppvDeploymentType
Add-CMAssetIntelligenceSynchronizationPoint
Add-CMBoundaryToGroup
Add-CMCIDetectionMethod
```

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.

Lists all services on the local machine.



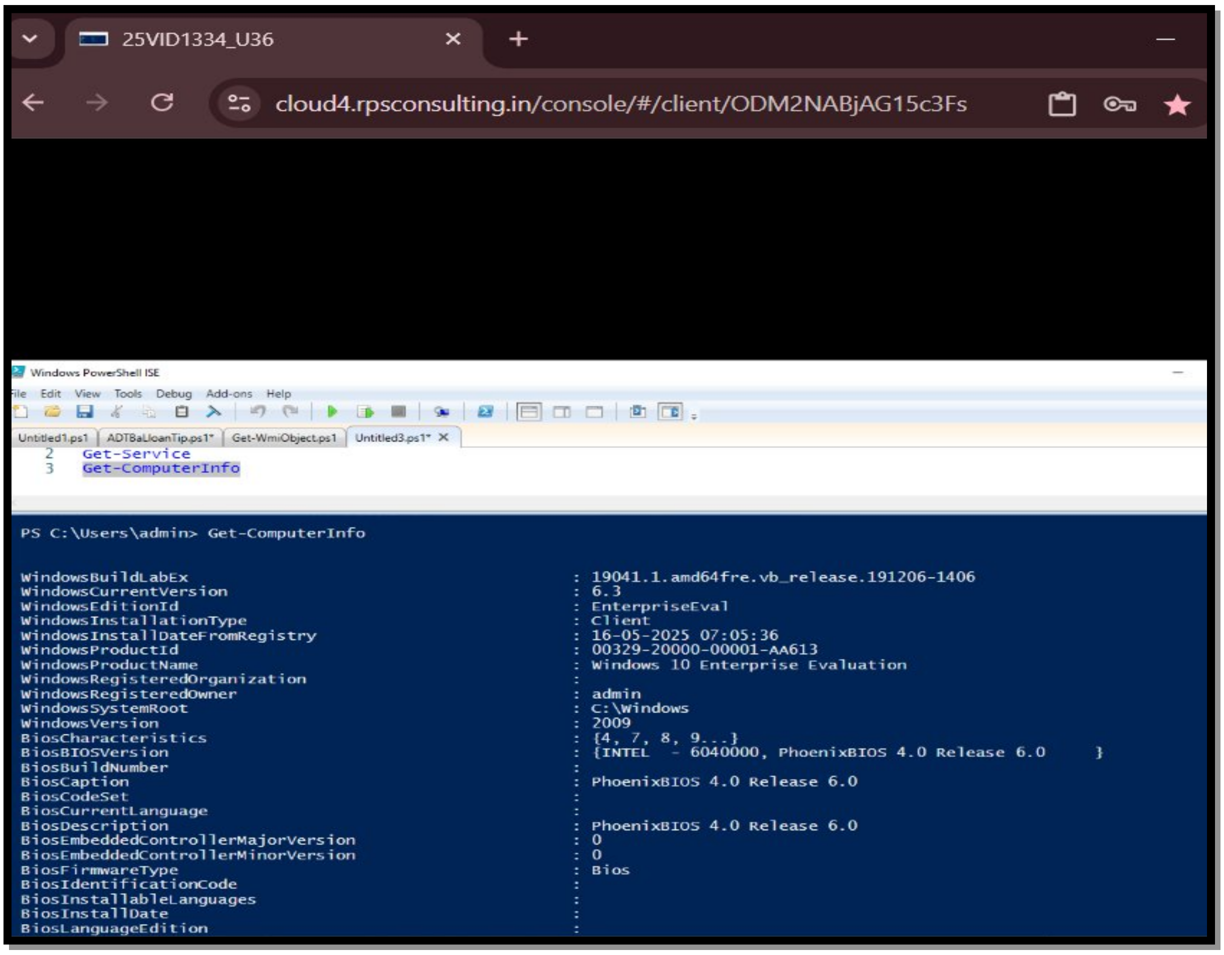
```
Show-Command
Get-Service
```

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

Status   Name               DisplayName
------   ----               -----------
Stopped  AarSvc_429a9       Agent Activation Runtime_429a9
Stopped  AJRouter           AllJoyn Router Service
Stopped  ALG                Application Layer Gateway Service
Stopped  AppIDSvc           Application Identity
Running  Appinfo            Application Information
Running  AppMgmt            Application Management
Stopped  AppReadiness       App Readiness
Stopped  AppVClient         Microsoft App-V Client
Stopped  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
Running  BITS               Background Intelligent Transfer Ser...
Stopped  BluetoothUserSe... Bluetooth User Support Service_429a9
Running  BrokerInfrastru... Background Tasks Infrastructure Ser...
Stopped  BTAGService        Bluetooth Audio Gateway Service
Running  BthAvctpSvc        AVCTP service
Stopped  bthserv            Bluetooth Support Service
```

3) Get-ComputerInfo

*Get-ComputerInfo retrieves **detailed system and operating system information** about the local computer.*

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



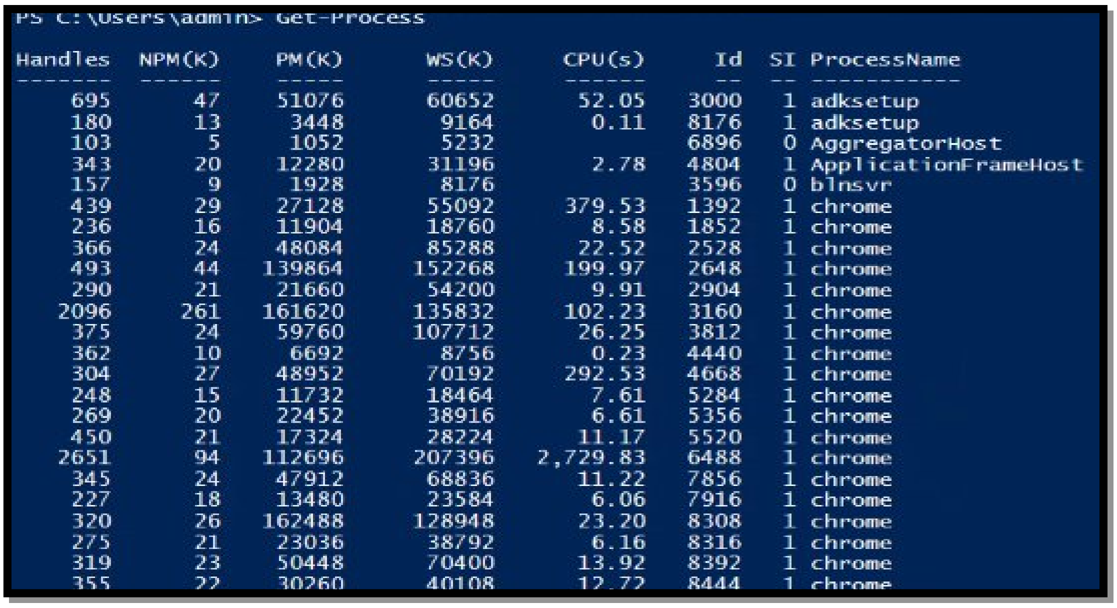
```
Get-Service
Get-ComputerInfo
```

```
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                                     :
```

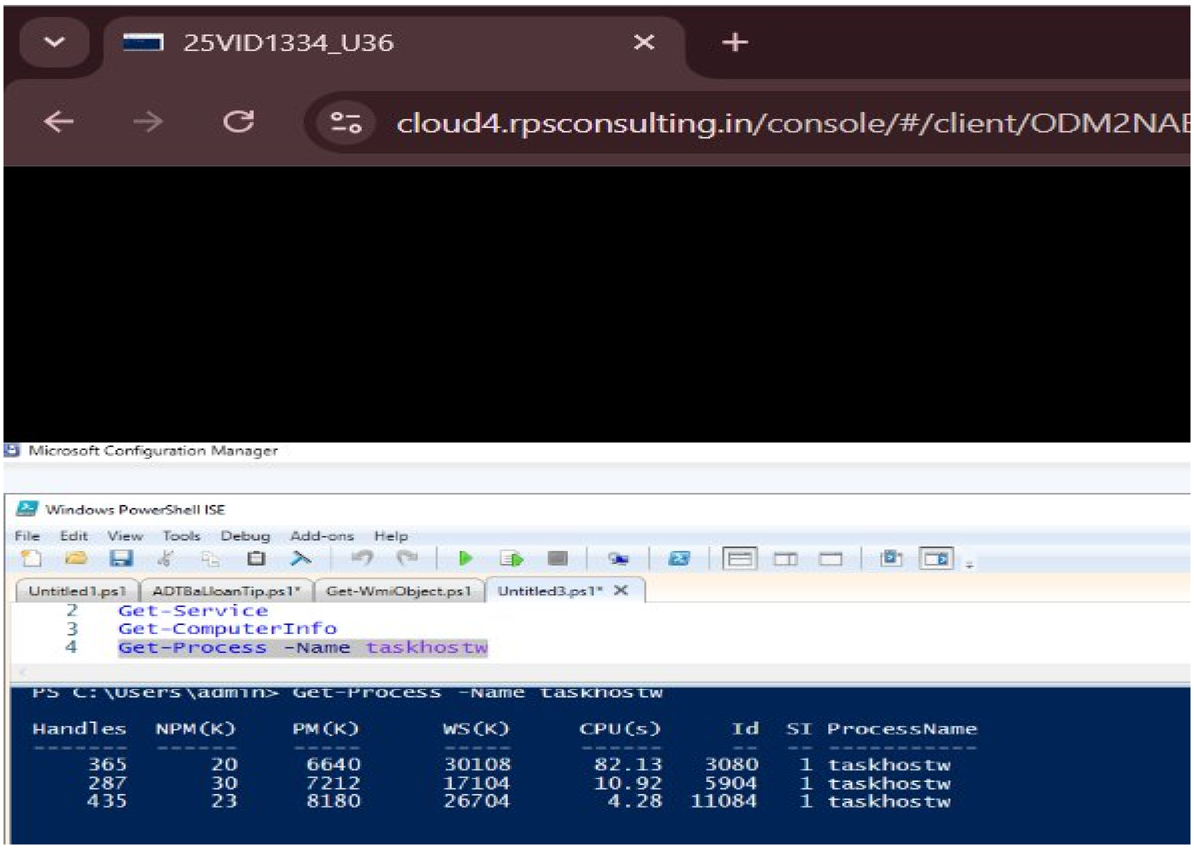
4) Get-Process

Retrieves information about running processes on the system.
*It also lists all **running processes** on the local computer, similar to **Task Manager**.*

- *Get-Process -Name taskhostw*



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

Handles  NPM(K)    PM(K)      WS(K)     CPU(s)     Id  SI ProcessName
-------  ------    -----      -----     ------     --  -- -----------
    695      47    51076      60652      52.05   3000   1 adksetup
    180      13     3448       9164       0.11   8176   1 adksetup
    103       5     1052       5232              6896   0 AggregatorHost
    343      20    12280      31196       2.78   4804   1 ApplicationFrameHost
    157       9     1928       8176              3596   0 blnsvr
    439      29    27128      55092     379.53   1392   1 chrome
    236      16    11904      18760       8.58   1852   1 chrome
    366      24    48084      85288      22.52   2528   1 chrome
    493      44   139864     152268     199.97   2648   1 chrome
    290      21    21660      54200       9.91   2904   1 chrome
   2096     261   161620     135832     102.23   3160   1 chrome
    375      24    59760     107712      26.25   3812   1 chrome
    362      10     6692       8756       0.23   4440   1 chrome
    304      27    48952      70192     292.53   4668   1 chrome
    248      15    11732      18464       7.61   5284   1 chrome
    269      20    22452      38916       6.61   5356   1 chrome
    450      21    17324      28224      11.17   5520   1 chrome
   2651      94   112696     207396   2,729.83   6488   1 chrome
    345      24    47912      68836      11.22   7856   1 chrome
    227      18    13480      23584       6.06   7916   1 chrome
    320      26   162488     128948      23.20   8308   1 chrome
    275      21    23036      38792       6.16   8316   1 chrome
    319      23    50448      70400      13.92   8392   1 chrome
    355      22    30260      40108      12.72   8444   1 chrome
```


25VID1334_U36
cloud4.rpsconsulting.in/console/#/client/ODM2NAE
Microsoft Configuration Manager
Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
Untitled1.ps1
ADTBalloonTip.ps1*
Get-WmiObject.ps1
Untitled3.ps1*

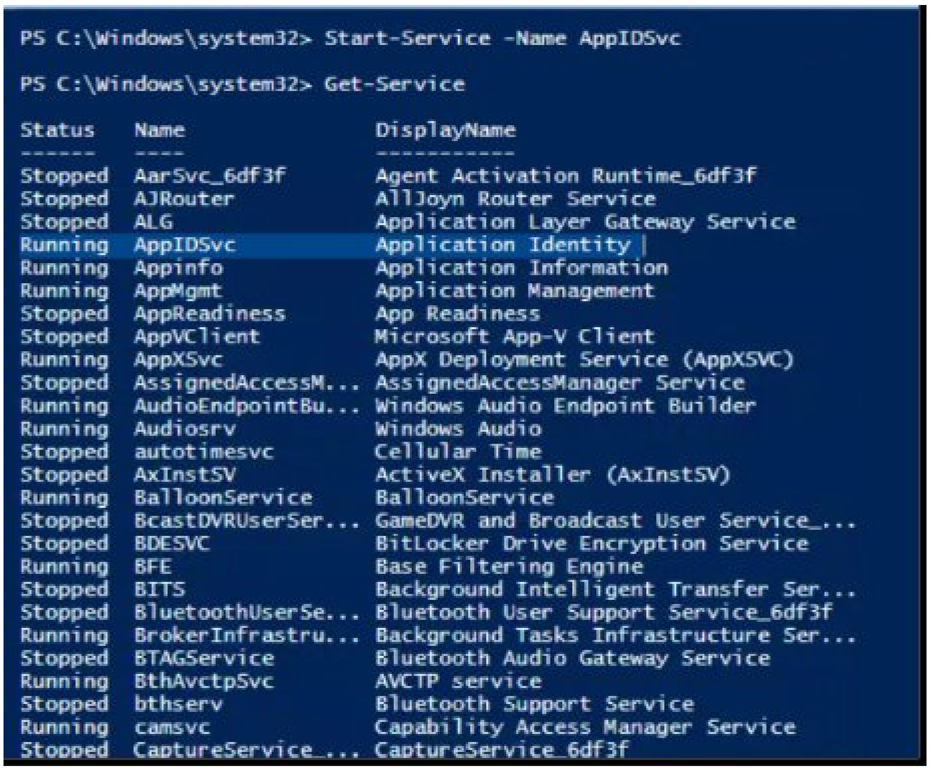
```
Get-Service
Get-ComputerInfo
Get-Process -Name taskhostw
```

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

Handles  NPM(K)    PM(K)      WS(K)     CPU(s)     Id  SI ProcessName
-------  ------    -----      -----     ------     --  -- -----------
    365      20     6640      30108      82.13   3080   1 taskhostw
    287      30     7212      17104      10.92   5904   1 taskhostw
    435      23     8180      26704       4.28  11084   1 taskhostw
```

5) Start-Service

Starts a specified service.

- *Start-Service -Name AppIDSvc*

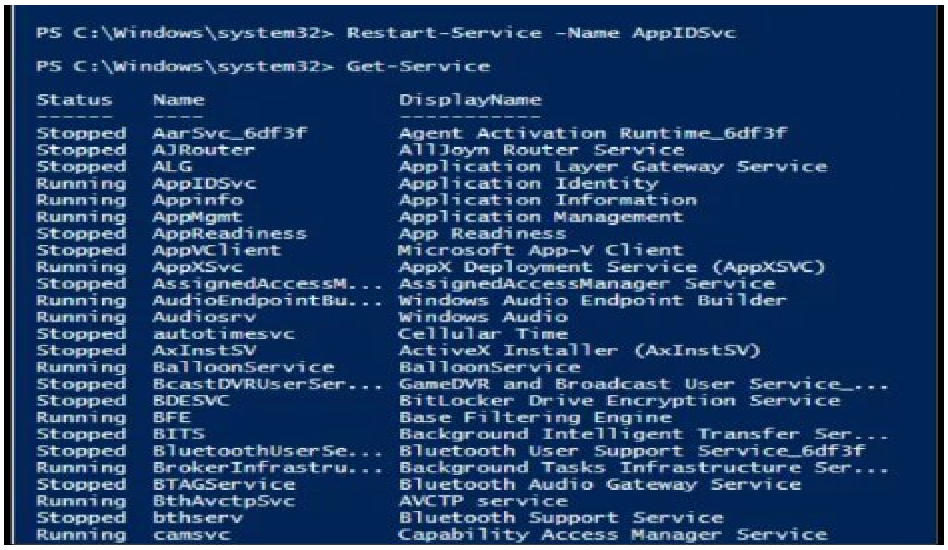


```
PS C:\Windows\system32> Start-Service -Name AppIDSvc

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*

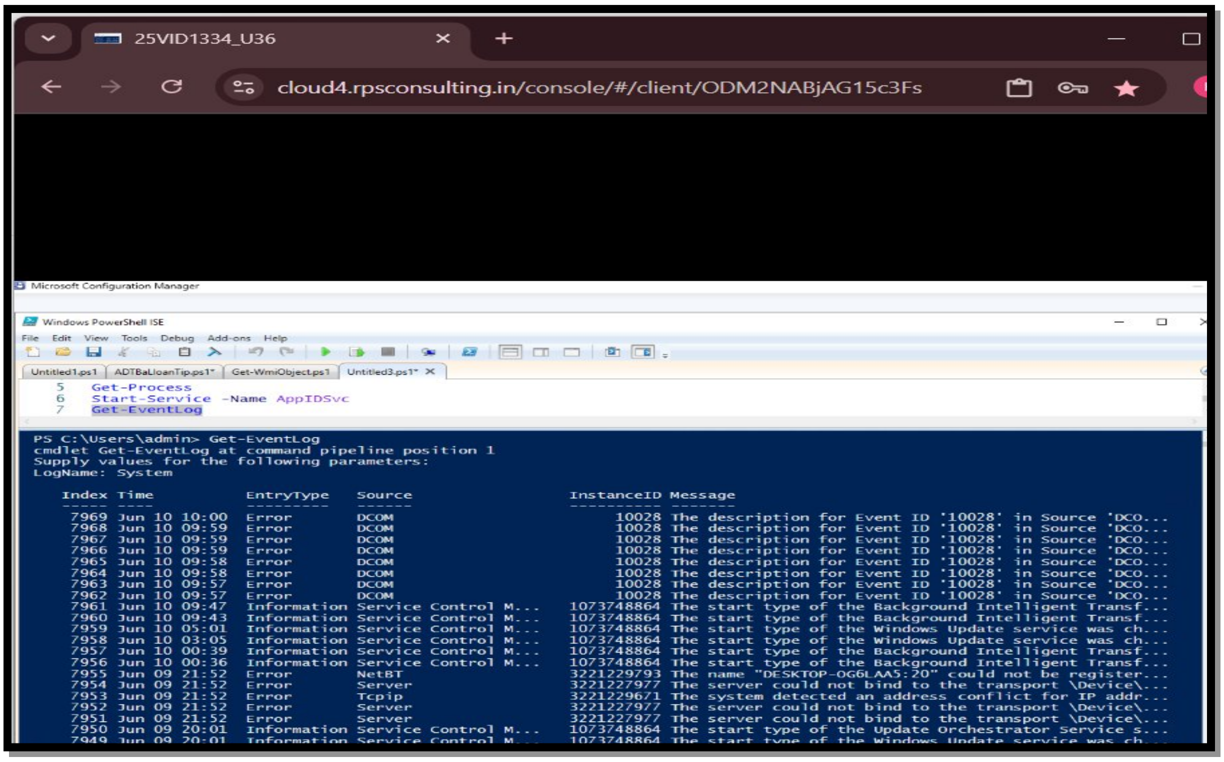


```
PS C:\Windows\system32> Restart-Service -Name AppIDSvc

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*


25VID1334_U36
cloud4.rpsconsulting.in/console/#/client/ODM2NABjAG15c3Fs
Microsoft Configuration Manager
Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
Untitled1.ps1
ADTBaLloanTip.ps1*
Get-WmiObject.ps1
Untitled3.ps1*

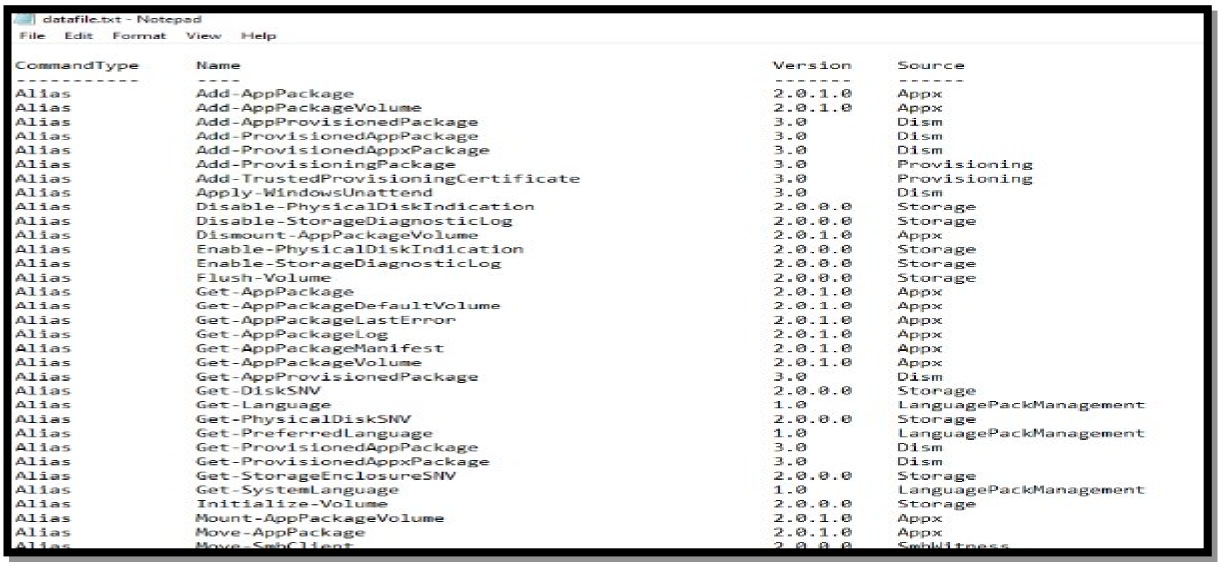
```
Get-Process
Start-Service -Name AppIDSvc
Get-EventLog
```

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

   Index Time          EntryType   Source                 InstanceID Message
   ----- ----          ---------   ------                 ---------- -------
    7969 Jun 10 10:00  Error       DCOM                        10028 The description for Event ID '10028' in Source 'DCO...
    7968 Jun 10 09:59  Error       DCOM                        10028 The description for Event ID '10028' in Source 'DCO...
    7967 Jun 10 09:59  Error       DCOM                        10028 The description for Event ID '10028' in Source 'DCO...
    7966 Jun 10 09:59  Error       DCOM                        10028 The description for Event ID '10028' in Source 'DCO...
    7965 Jun 10 09:58  Error       DCOM                        10028 The description for Event ID '10028' in Source 'DCO...
    7964 Jun 10 09:58  Error       DCOM                        10028 The description for Event ID '10028' in Source 'DCO...
    7963 Jun 10 09:57  Error       DCOM                        10028 The description for Event ID '10028' in Source 'DCO...
    7962 Jun 10 09:57  Error       DCOM                        10028 The description for Event ID '10028' in Source 'DCO...
    7961 Jun 10 09:47  Information Service Control M...   1073748864 The start type of the Background Intelligent Transf...
    7960 Jun 10 09:43  Information Service Control M...   1073748864 The start type of the Background Intelligent Transf...
    7959 Jun 10 05:01  Information Service Control M...   1073748864 The start type of the Windows Update service was ch...
    7958 Jun 10 03:05  Information Service Control M...   1073748864 The start type of the Windows Update service was ch...
    7957 Jun 10 00:39  Information Service Control M...   1073748864 The start type of the Background Intelligent Transf...
    7956 Jun 10 00:36  Information Service Control M...   1073748864 The start type of the Background Intelligent Transf...
    7955 Jun 09 21:52  Error       NetBT                  3221229793 The name "DESKTOP-OG6LAA5:20" could not be register...
    7954 Jun 09 21:52  Error       Server                 3221227977 The server could not bind to the transport \Device\...
    7953 Jun 09 21:52  Error       Tcpip                  3221229671 The system detected an address conflict for IP addr...
    7952 Jun 09 21:52  Error       Server                 3221227977 The server could not bind to the transport \Device\...
    7951 Jun 09 21:52  Error       Server                 3221227977 The server could not bind to the transport \Device\...
    7950 Jun 09 20:01  Information Service Control M...   1073748864 The start type of the Update Orchestrator Service s...
    7949 Jun 09 20:01  Information Service Control M...   1073748864 The start type of the Windows Update service was ch...
```

8) Out-File

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

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

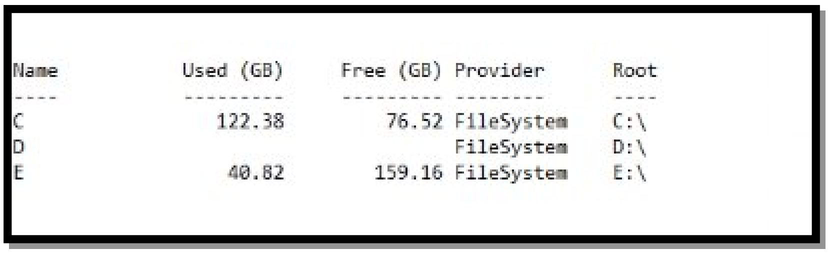


datafile.txt - Notepad

File Edit Format View Help

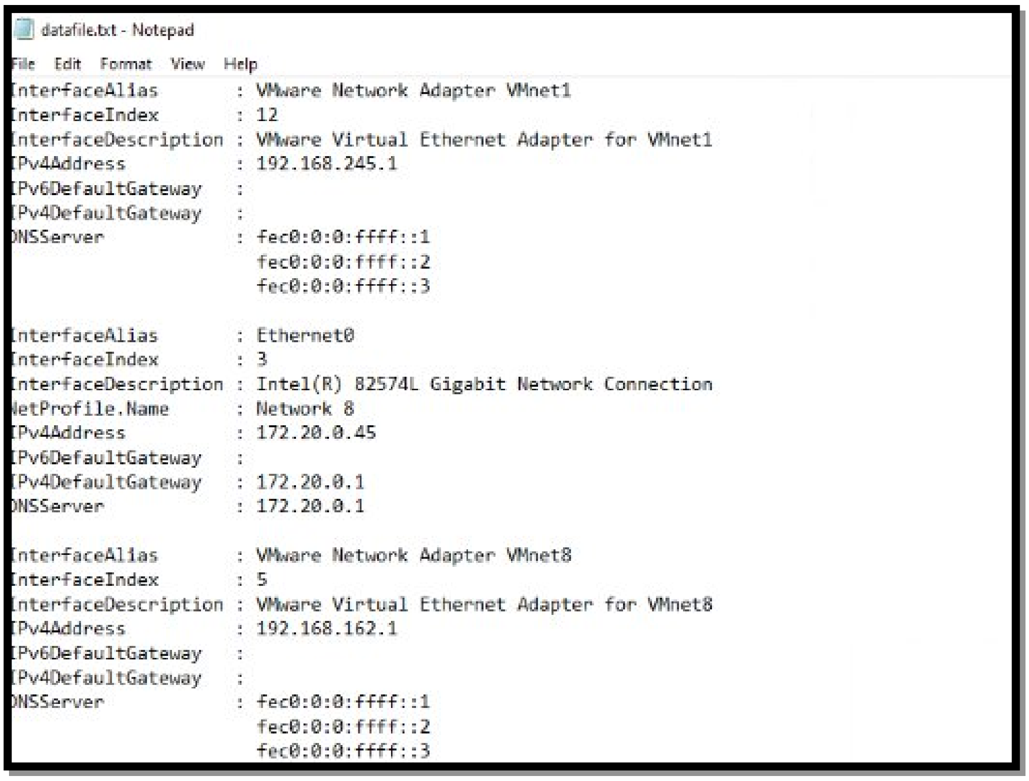
```
CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           Add-AppPackage                                     2.0.1.0    Appx
Alias           Add-AppPackageVolume                               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\datafile.txt -Append*



```
Name           Used (GB)     Free (GB) Provider      Root
----           ---------     --------- --------      ----
C                 122.38         76.52 FileSystem    C:\
D                                      FileSystem    D:\
E                  40.82        159.16 FileSystem    E:\
```

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

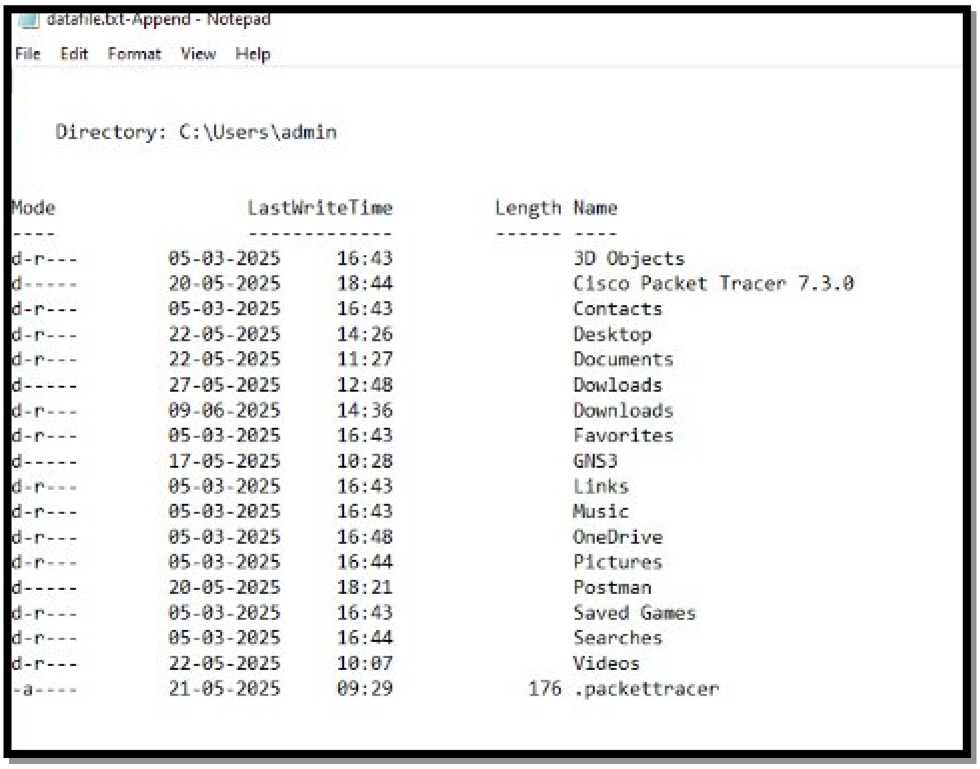


datafile.txt - Notepad

File Edit Format View Help

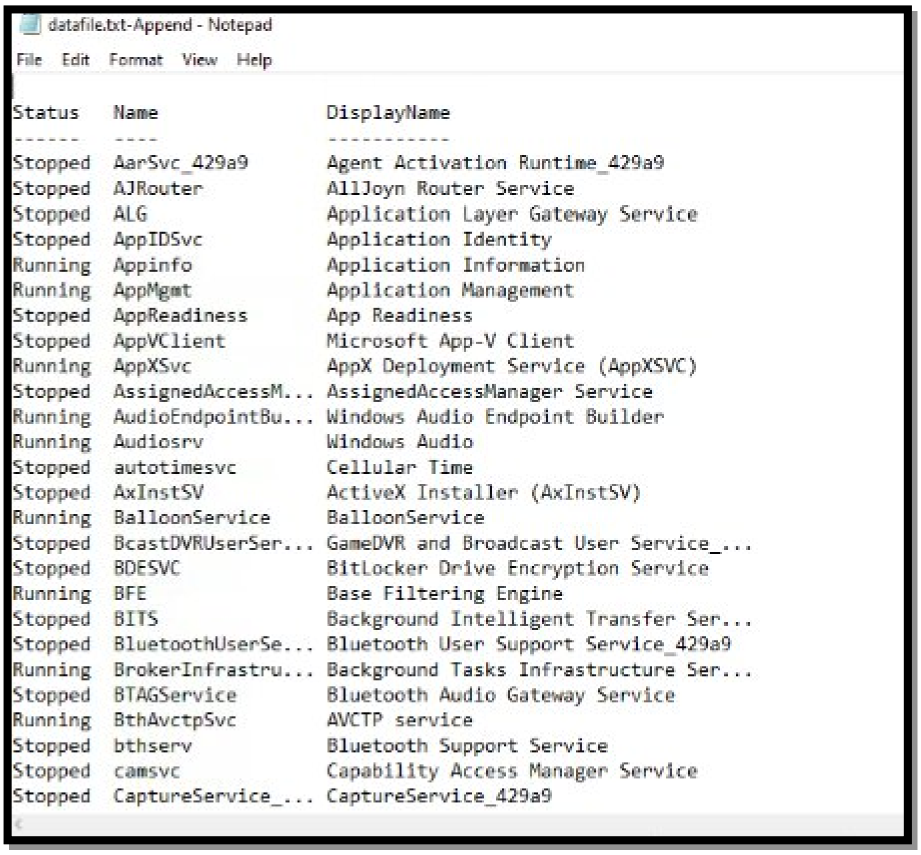
```
InterfaceAlias       : VMware Network Adapter VMnet1
InterfaceIndex       : 12
InterfaceDescription : VMware Virtual Ethernet Adapter for VMnet1
IPv4Address          : 192.168.245.1
IPv6DefaultGateway   :
IPv4DefaultGateway   :
DNSServer            : fec0:0:0:ffff::1
                       fec0:0:0:ffff::2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

InterfaceAlias       : Ethernet0
InterfaceIndex       : 3
InterfaceDescription : Intel(R) 82574L Gigabit Network Connection
NetProfile.Name      : Network 8
IPv4Address          : 172.20.0.45
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\datafile.txt -Append*



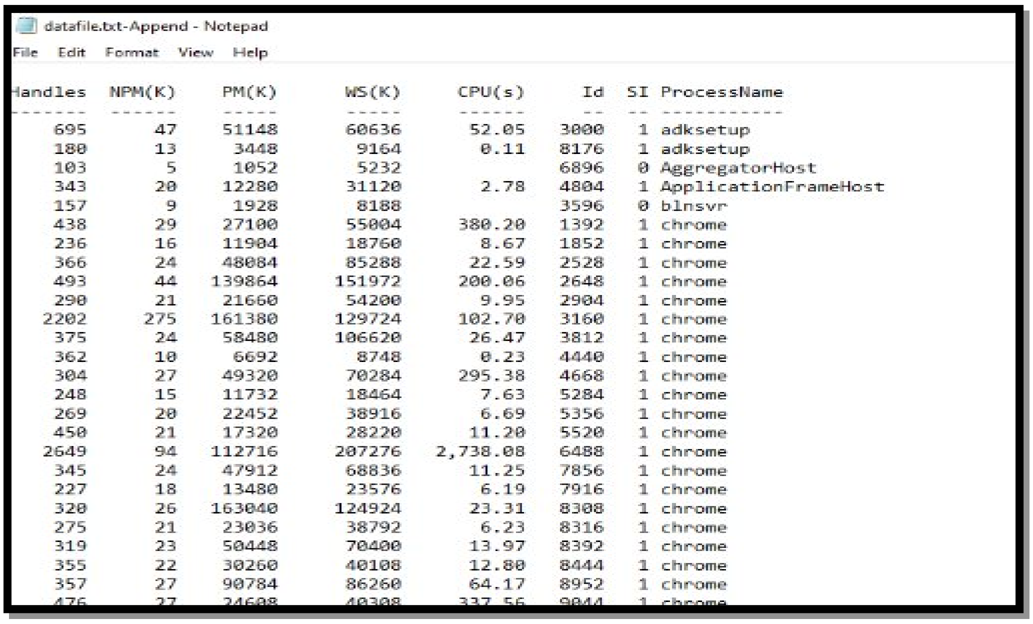
```
    Directory: C:\Users\admin


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-r---        05-03-2025     16:43                3D Objects
d-----        20-05-2025     18:44                Cisco Packet Tracer 7.3.0
d-r---        05-03-2025     16:43                Contacts
d-r---        22-05-2025     14:26                Desktop
d-r---        22-05-2025     11:27                Documents
d-----        27-05-2025     12:48                Dowloads
d-r---        09-06-2025     14:36                Downloads
d-r---        05-03-2025     16:43                Favorites
d-----        17-05-2025     10:28                GNS3
d-r---        05-03-2025     16:43                Links
d-r---        05-03-2025     16:43                Music
d-r---        05-03-2025     16:48                OneDrive
d-r---        05-03-2025     16:44                Pictures
d-----        20-05-2025     18:21                Postman
d-r---        05-03-2025     16:43                Saved Games
d-r---        05-03-2025     16:44                Searches
d-r---        22-05-2025     10:07                Videos
-a----        21-05-2025     09:29            176 .packettracer
```

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



```
Status   Name               DisplayName
------   ----               -----------
Stopped  AarSvc_429a9       Agent Activation Runtime_429a9
Stopped  AJRouter           AllJoyn Router Service
Stopped  ALG                Application Layer Gateway Service
Stopped  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_429a9
Running  BrokerInfrastru... Background Tasks Infrastructure Ser...
Stopped  BTAGService        Bluetooth Audio Gateway Service
Running  BthAvctpSvc        AVCTP service
Stopped  bthserv            Bluetooth Support Service
Stopped  camsvc             Capability Access Manager Service
Stopped  CaptureService_... CaptureService_429a9
```

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



datafile.txt-Append - Notepad

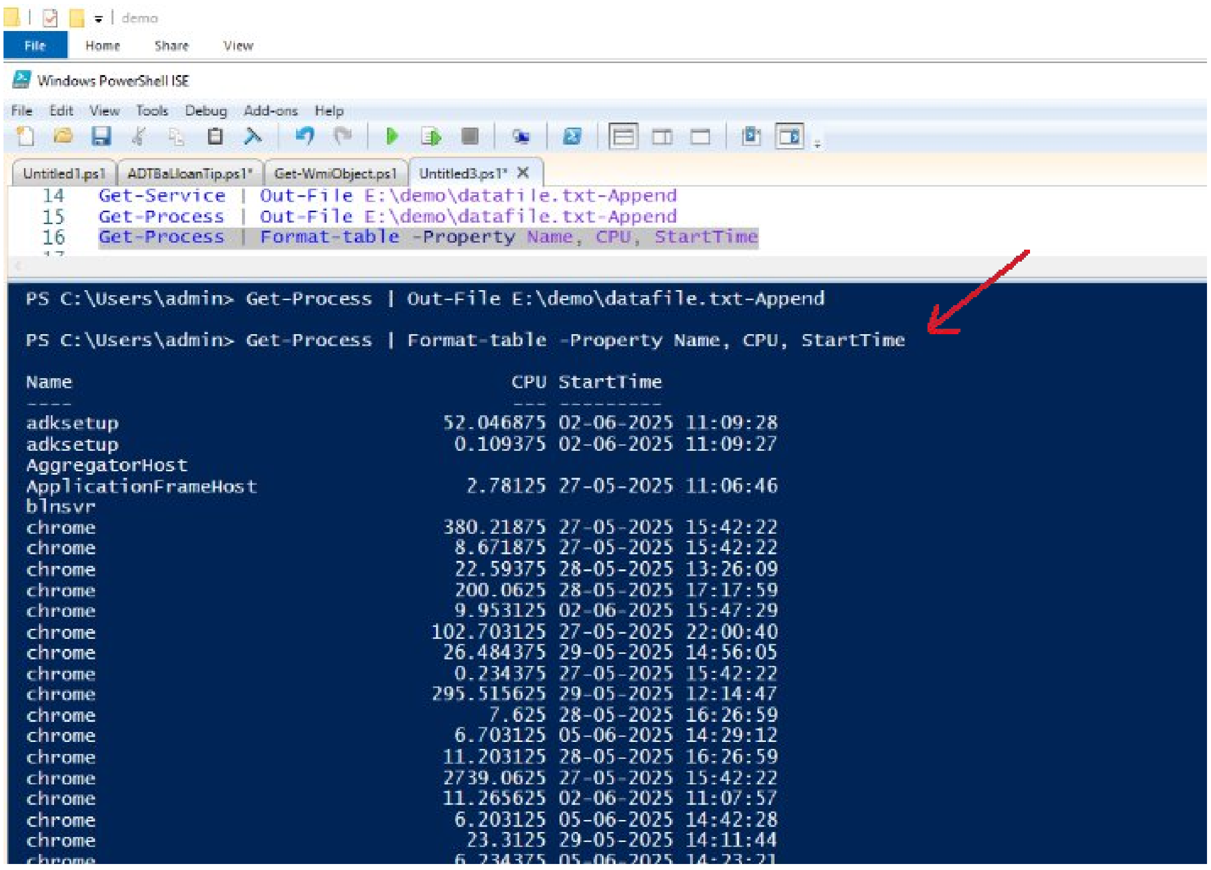
File Edit Format View Help

```
Handles  NPM(K)    PM(K)      WS(K)     CPU(s)     Id  SI ProcessName
-------  ------    -----      -----     ------     --  -- -----------
    695      47    51148      60636      52.05   3000   1 adksetup
    180      13     3448       9164       0.11   8176   1 adksetup
    103       5     1052       5232              6896   0 AggregatorHost
    343      20    12280      31120       2.78   4804   1 ApplicationFrameHost
    157       9     1928       8188              3596   0 blnsvr
    438      29    27100      55004     380.20   1392   1 chrome
    236      16    11904      18760       8.67   1852   1 chrome
    366      24    48084      85288      22.59   2528   1 chrome
    493      44   139864     151972     200.06   2648   1 chrome
    290      21    21660      54200       9.95   2904   1 chrome
   2202     275   161380     129724     102.70   3160   1 chrome
    375      24    58480     106620      26.47   3812   1 chrome
    362      10     6692       8748       0.23   4440   1 chrome
    304      27    49320      70284     295.38   4668   1 chrome
    248      15    11732      18464       7.63   5284   1 chrome
    269      20    22452      38916       6.69   5356   1 chrome
    450      21    17320      28220      11.20   5520   1 chrome
   2649      94   112716     207276   2,738.08   6488   1 chrome
    345      24    47912      68836      11.25   7856   1 chrome
    227      18    13480      23576       6.19   7916   1 chrome
    320      26   163040     124924      23.31   8308   1 chrome
    275      21    23036      38792       6.23   8316   1 chrome
    319      23    50448      70400      13.97   8392   1 chrome
    355      22    30260      40108      12.80   8444   1 chrome
    357      27    90784      86260      64.17   8952   1 chrome
    476      27    24608      40308     337.56   9044   1 chrome
```

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*



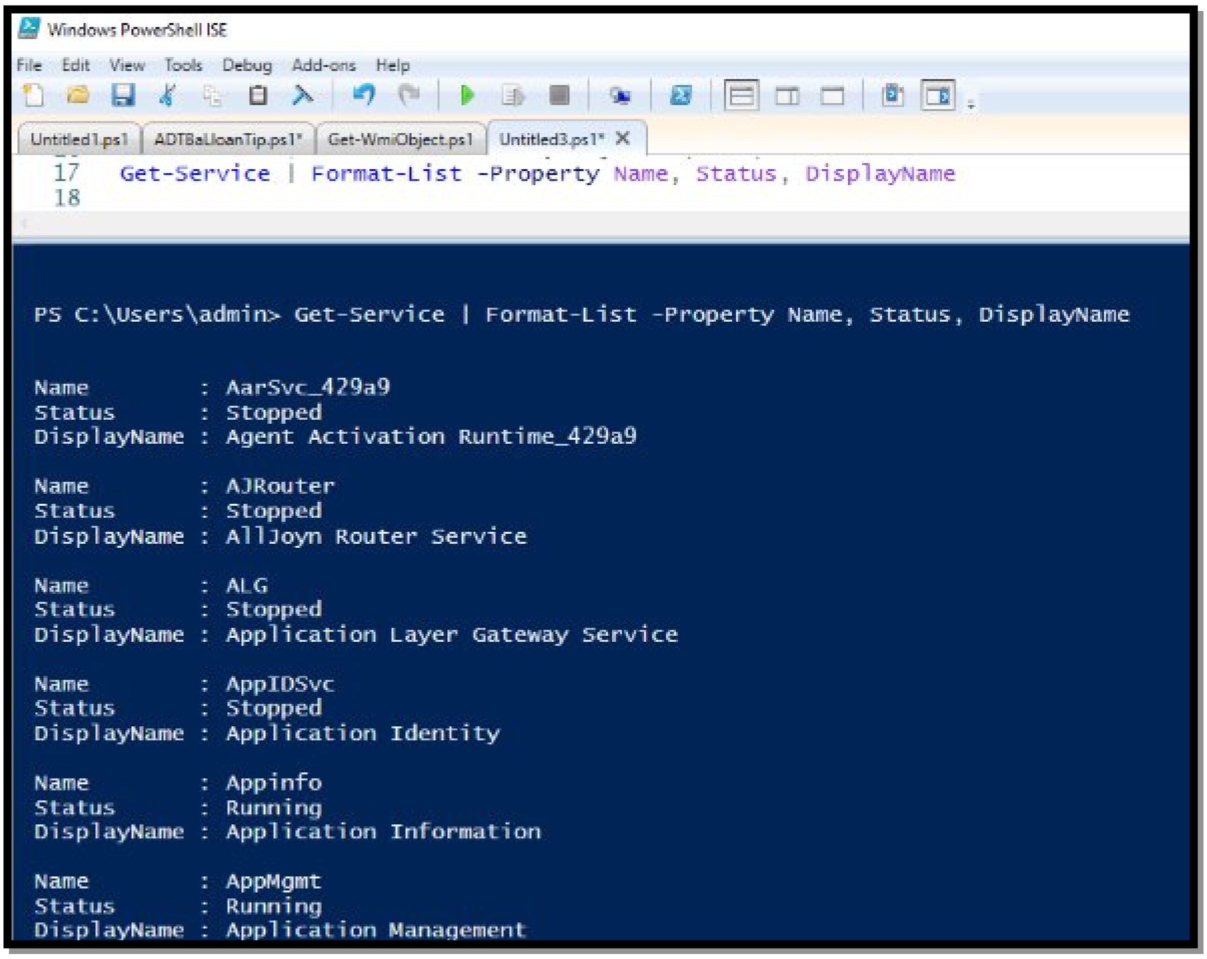
```
Get-Service | Out-File E:\demo\datafile.txt-Append
Get-Process | Out-File E:\demo\datafile.txt-Append
Get-Process | Format-table -Property Name, CPU, StartTime
```

```
PS C:\Users\admin> Get-Process | Out-File E:\demo\datafile.txt-Append

PS C:\Users\admin> Get-Process | Format-table -Property Name, CPU, StartTime

Name                                  CPU StartTime
----                                  --- ---------
adksetup                        52.046875 02-06-2025 11:09:28
adksetup                         0.109375 02-06-2025 11:09:27
AggregatorHost
ApplicationFrameHost              2.78125 27-05-2025 11:06:46
blnsvr
chrome                          380.21875 27-05-2025 15:42:22
chrome                           8.671875 27-05-2025 15:42:22
chrome                           22.59375 28-05-2025 13:26:09
chrome                           200.0625 28-05-2025 17:17:59
chrome                           9.953125 02-06-2025 15:47:29
chrome                         102.703125 27-05-2025 22:00:40
chrome                          26.484375 29-05-2025 14:56:05
chrome                           0.234375 27-05-2025 15:42:22
chrome                         295.515625 29-05-2025 12:14:47
chrome                              7.625 28-05-2025 16:26:59
chrome                           6.703125 05-06-2025 14:29:12
chrome                          11.203125 28-05-2025 16:26:59
chrome                          2739.0625 27-05-2025 15:42:22
chrome                          11.265625 02-06-2025 11:07:57
chrome                           6.203125 05-06-2025 14:42:28
chrome                            23.3125 29-05-2025 14:11:44
chrome                           6.234375 05-06-2025 14:23:21
```

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*


Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
Untitled1.ps1 ADTBaLloanTip.ps1* Get-WmiObject.ps1 Untitled3.ps1*

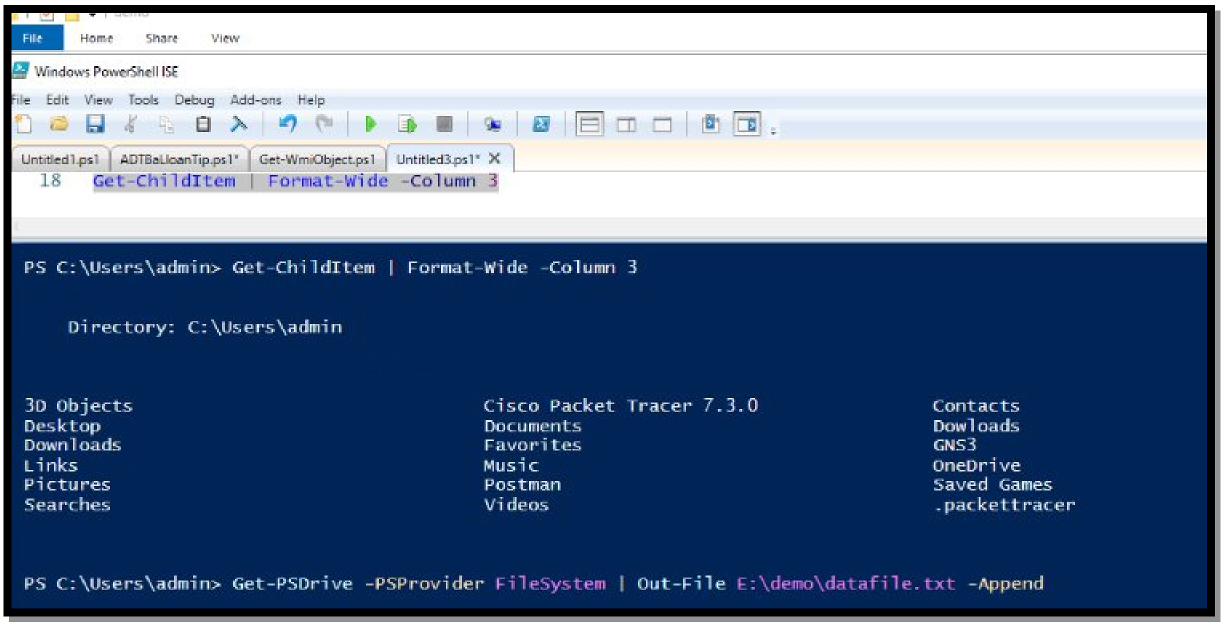
```
Get-Service | Format-List -Property Name, Status, DisplayName
```

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


Name        : AarSvc_429a9
Status      : Stopped
DisplayName : Agent Activation Runtime_429a9

Name        : AJRouter
Status      : Stopped
DisplayName : AllJoyn Router Service

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

Name        : AppIDSvc
Status      : Stopped
DisplayName : Application Identity

Name        : Appinfo
Status      : Running
DisplayName : Application Information

Name        : AppMgmt
Status      : Running
DisplayName : Application Management
```

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*


File Home Share View
Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
Untitled1.ps1 ADTBaLloanTip.ps1* Get-WmiObject.ps1 Untitled3.ps1*

```
Get-ChildItem | Format-Wide -Column 3
```

```
PS C:\Users\admin> Get-ChildItem | Format-Wide -Column 3


    Directory: C:\Users\admin


3D Objects                          Cisco Packet Tracer 7.3.0          Contacts
Desktop                             Documents                          Downloads
Downloads                           Favorites                          GNS3
Links                               Music                              OneDrive
Pictures                            Postman                            Saved Games
Searches                            Videos                             .packettracer


PS C:\Users\admin> Get-PSDrive -PSProvider FileSystem | Out-File E:\demo\datafile.txt -Append
```

Objects, Arrays, Variables, Scripting Constraints

- ***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.*
3. ***Write-Output**: Sends output to the **pipeline** (can be captured or redirected).*

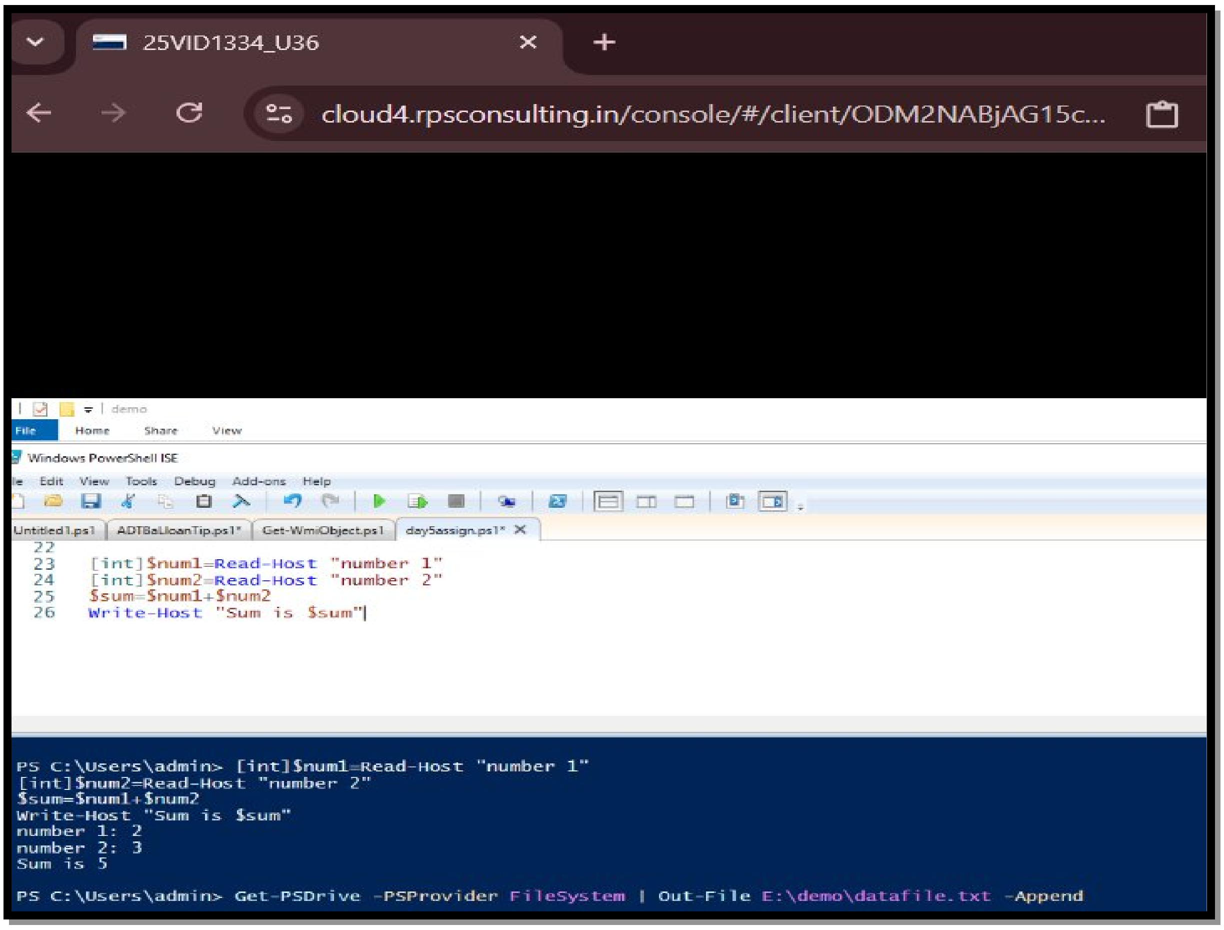
Variable
Variables are used to store values. They are denoted by $ symbol followed by name. They are not case sensitive. They can include letters, numbers and underscore. There is no need to declare datatypes of variables.

Code Example:

```
$name=”yoyo”
$age=21
$isTrue=$true
```

->Code for Addition of two variables

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



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

```
PS C:\Users\admin> [int]$num1=Read-Host "number 1"
[int]$num2=Read-Host "number 2"
$sum=$num1+$num2
Write-Host "Sum is $sum"
number 1: 2
number 2: 3
Sum is 5

PS C:\Users\admin> Get-PSDrive -PSProvider FileSystem | Out-File E:\demo\datafile.txt -Append
```

->String Formatting

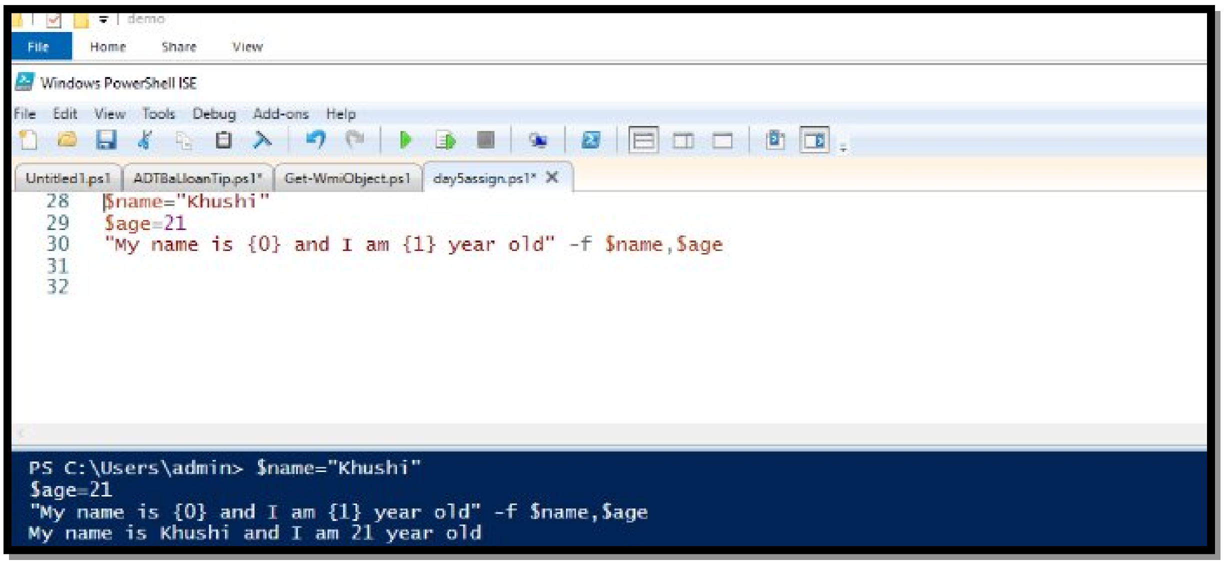
-f format operator allows for composite formatting.

Code Example:

$name = ”Khushi”

$age=21

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



```
$name="Khushi"
$age=21
"My name is {0} and I am {1} year old" -f $name,$age
```

```
PS C:\Users\admin> $name="Khushi"
$age=21
"My name is {0} and I am {1} year old" -f $name,$age
My name is Khushi and I am 21 year old
```

->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 @().

Code Example:

$arr=@(“k”,”h”,”u”,”5”,”h”,”1”)

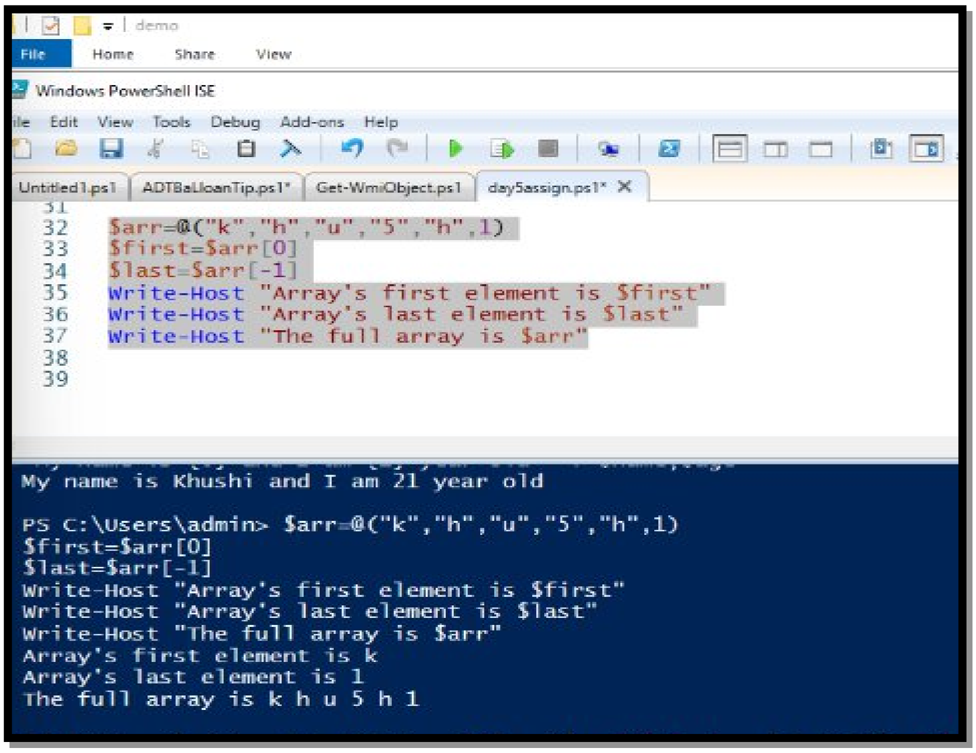
$first=$arr[0]

$last=$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"



```
$arr=@("k","h","u","5","h",1)
$first=$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"
```

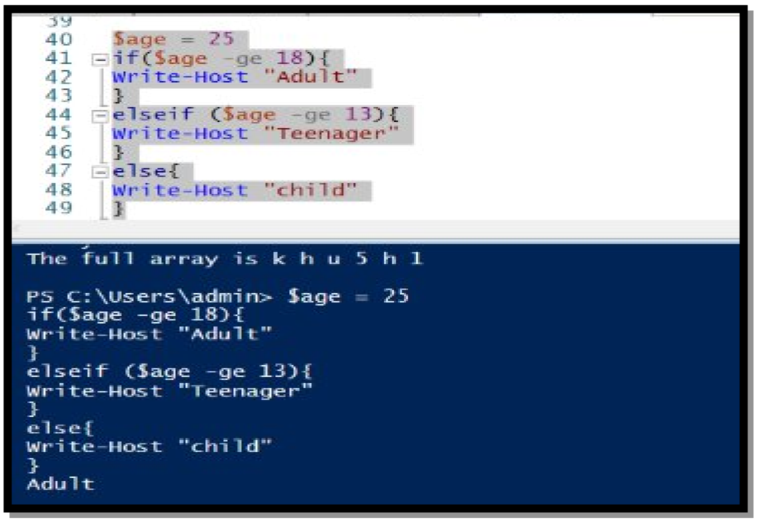
```
My name is Khushi and I am 21 year old

PS C:\Users\admin> $arr=@("k","h","u","5","h",1)
$first=$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"
Array's first element is k
Array's last element is 1
The full array is k h u 5 h 1
```

->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”
}
```



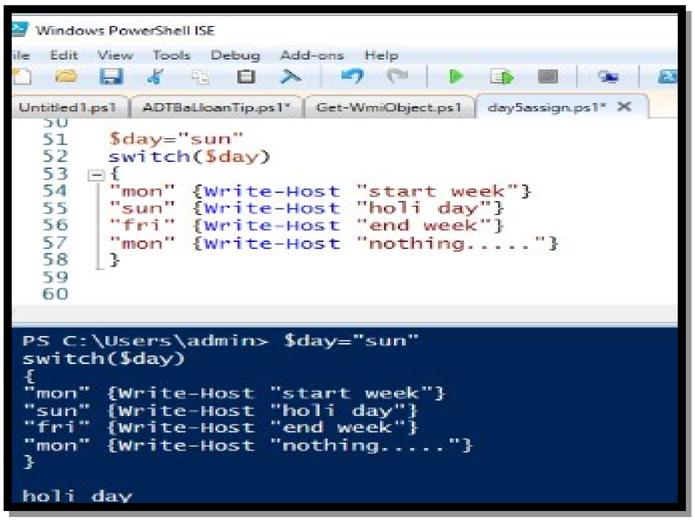
```
$age = 25
if($age -ge 18){
Write-Host "Adult"
}
elseif ($age -ge 13){
Write-Host "Teenager"
}
else{
Write-Host "child"
}
```

```
The full array is k h u 5 h 1

PS C:\Users\admin> $age = 25
if($age -ge 18){
Write-Host "Adult"
}
elseif ($age -ge 13){
Write-Host "Teenager"
}
else{
Write-Host "child"
}
Adult
```

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

Code Example:

```
$day="sun"
switch($day){
"mon" {Write-Host "Start week"}
"sun" {Write-Host "Holi day"}
"fri" {Write-Host "end week"}
default {Write-Host "nothing...."}
}
```


Windows PowerShell ISE
ile Edit View Tools Debug Add-ons Help
Untitled1.ps1 ADTBalloonTip.ps1* Get-WmiObject.ps1 day5assign.ps1*

```
$day="sun"
switch($day)
{
"mon" {Write-Host "start week"}
"sun" {Write-Host "holi day"}
"fri" {Write-Host "end week"}
"mon" {Write-Host "nothing....."}
}
```

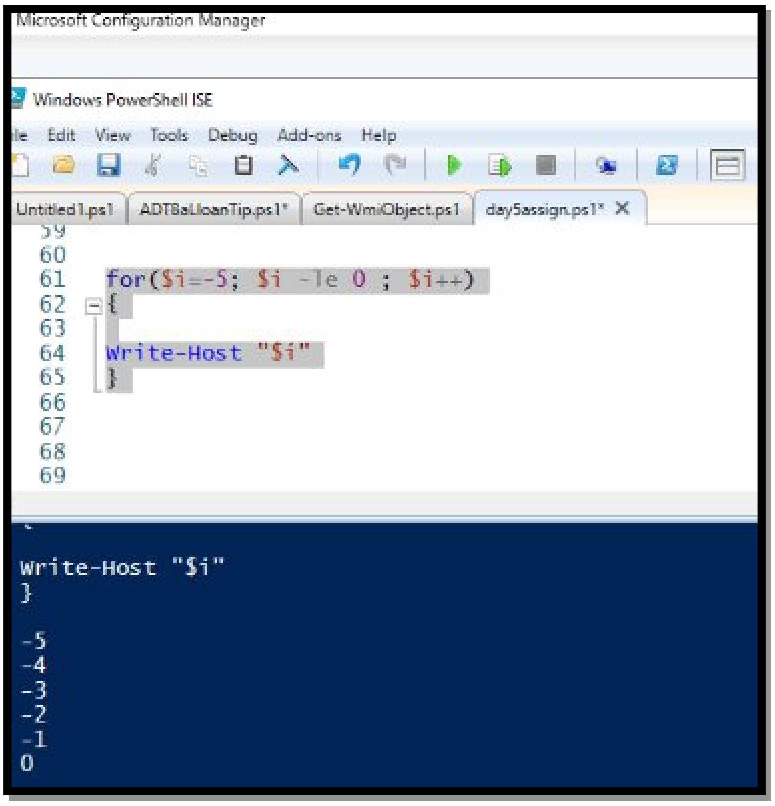
```
PS C:\Users\admin> $day="sun"
switch($day)
{
"mon" {Write-Host "start week"}
"sun" {Write-Host "holi day"}
"fri" {Write-Host "end week"}
"mon" {Write-Host "nothing....."}
}

holi day
```

->Looping Statements

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

Code Example:

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



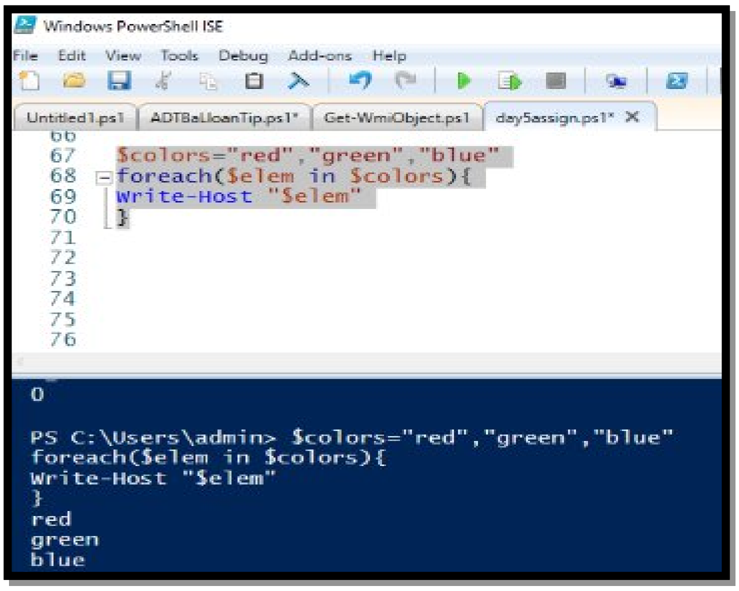
```
for($i=-5; $i -le 0 ; $i++)
{

Write-Host "$i"
}
```

```
Write-Host "$i"
}

-5
-4
-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"
}



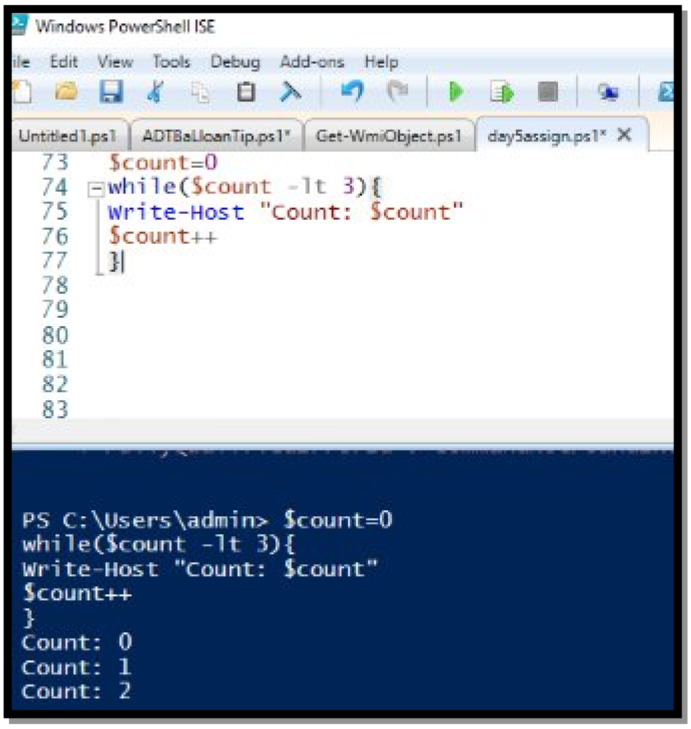
```
$colors="red","green","blue"
foreach($elem in $colors){
Write-Host "$elem"
}
```

```
0

PS C:\Users\admin> $colors="red","green","blue"
foreach($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++
}
```



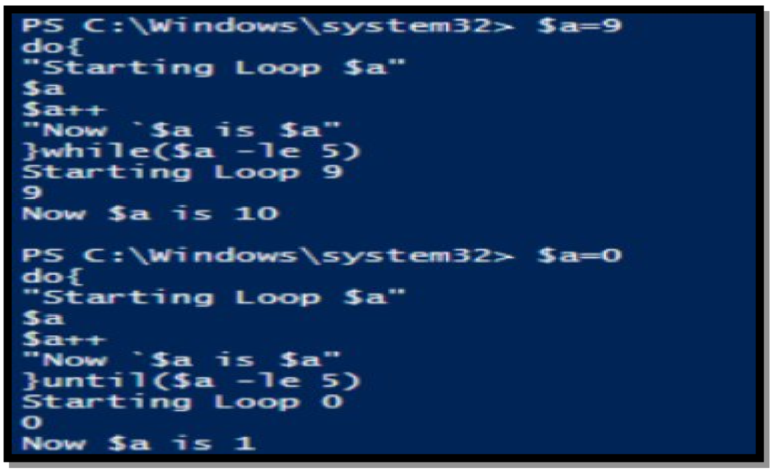
```
$count=0
while($count -lt 3){
Write-Host "Count: $count"
$count++
}
```

```
PS C:\Users\admin> $count=0
while($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)
```

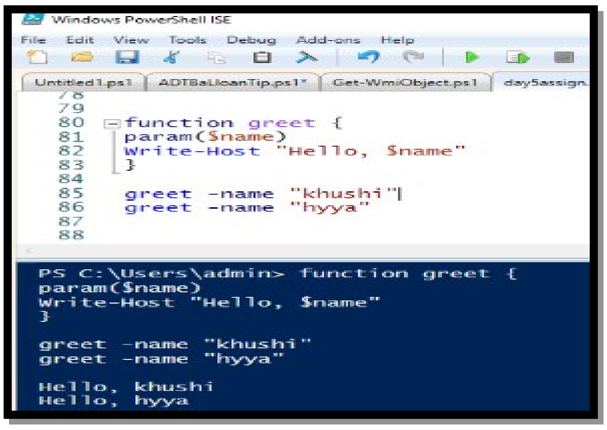


```
PS C:\Windows\system32> $a=9
do{
"Starting Loop $a"
$a
$a++
"Now `$a is $a"
}while($a -le 5)
Starting Loop 9
9
Now $a is 10

PS C:\Windows\system32> $a=0
do{
"Starting Loop $a"
$a
$a++
"Now `$a is $a"
}until($a -le 5)
Starting Loop 0
0
Now $a is 1
```

->function

A PowerShell function is a named block of code that performs a specific task. Defining functions allows you to reuse code, make your scripts more organized, and simplify complex operations.

```
Function greet{
Param($name)
Write-Host “Hello, $name”
}
greet -name “khushi”
greet -name “hyya”
```


Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
Untitled1.ps1 ADTBalloonTip.ps1* Get-WmiObject.ps1 day5assign

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

greet -name "khushi"
greet -name "hyya"
```

```
PS C:\Users\admin> function greet {
param($name)
Write-Host "Hello, $name"
}

greet -name "khushi"
greet -name "hyya"

Hello, khushi
Hello, hyya
```

->Comments in PowerShell

In PowerShell, you can add comments to your scripts using two main methods:

1. **Single-line comments:** Using the hash symbol (#)
2. **Multi-line/Block comments:** Using <# ... #>

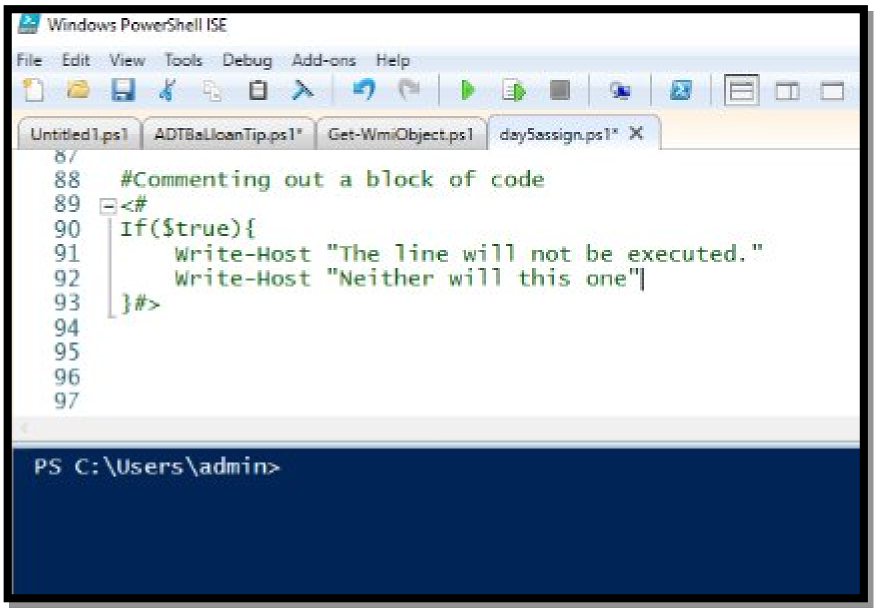
   - **Single-Line Comments (#)**
   The most common way to add comments is by starting a line with a hash symbol (#). Anything on that line after the # will be ignored by PowerShell when the script is executed.
   - **Multi-Line / Block Comments (<# ... #>)**

   For longer explanations that span multiple lines, or for commenting out larger blocks of code, you can use the <# and #> delimiters. Everything between these two delimiters will be treated as a comment.

Commenting out a block of code:

<# If ($true) { Write-Host "This line will not be executed." Write-Output "Neither will this one." }

#>



```
#Commenting out a block of code
<#
If($true){
    Write-Host "The line will not be executed."
    Write-Host "Neither will this one"
}#>
```

```
PS C:\Users\admin>
```