**NAME:-** Talari Vyshnavi

**USER ID:-** 34725

**EMAIL:-** [talari.vyshu@gmail.com](mailto:talari.vyshu@gmail.com)

**BATCH ID:-** 25VID2550

**BATCH NO:-** 5

**ASSIGNMENT**

**TOPICS:**-

* Introducing to Cmdlets
* The PowerShell Pipeline
* Key Cmdlets
* WMI & PowerShell
* Pipeline Filtering & Operators
* Input, Output & Formatting
* Scripting Overview
* **Introducing to Cmdlets** :-

Cmdlets are lightweighted commands used in windows PowerShell to perform specific tasks or operations. They are the building blocks of PowerShell scripting and automation.

* Key Features of Cmdlets:
* Cmdlets can pass data to one another using the pipeline.
* PowerShell comes with hundreds of built-in cmdlets.
* You can create custom cmdlets PowerShell or .NET.
* All cmdlets follow the Verb-Noun naming for readability and consistency.
* How cmdlets works:
* Input: Takes user input via parameters or pipeline.

Example- Get-Service -Name spooler.

* Processing: Executes logic using .NET

Example- Internally uses ProcessRecord ( ).

* Output: Returns .NET objects.

Example- Results can be piped or saved.

* **The PowerShell Pipeline :-**

The PowerShell pipeline (|) allows you to pass the ***output*** of one cmdlet as the ***input*** to another. It is a powerful feature that supports automations and scripting by ***chaining*** commands together.

Sample example- ***Get-process | Where -object { $\_.CPU -gt 100 } | Sort-object CPU -Descending***

A computer screen with a black screen

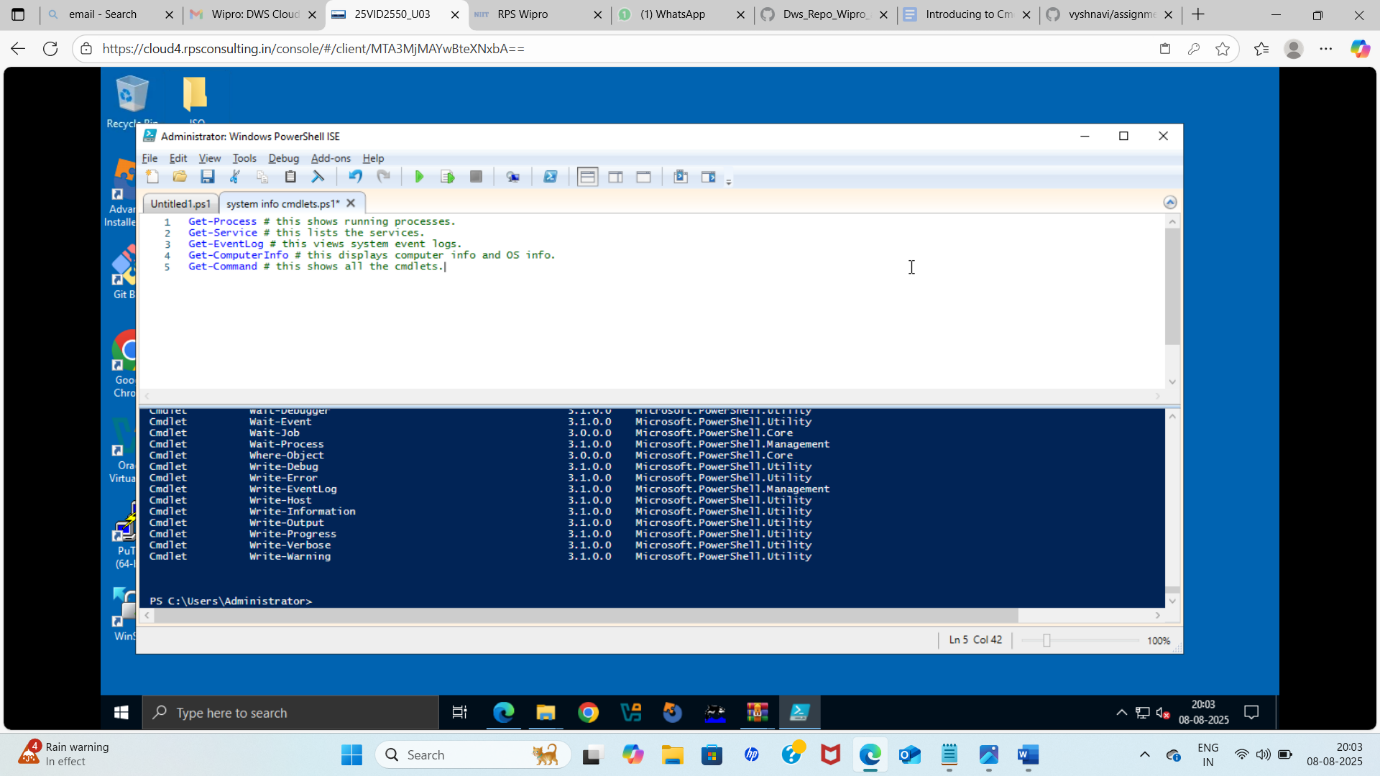
AI-generated content may be incorrect.

Result for the above example.

* **Key Cmdlets :-**

**System information :**

* + - Get-Process :- Shows running processes
    - Get-Service :- Lists service on the system
    - Get-EventLog :- Views system event logs
    - Get-ComputerInfo :-Displays system hardware & Os info
    - Get-Command :- Lists all available cmdlets



* + - Get-Childerns
    - Start-Services
    - Stop-Services
    - Set-Location
    - Set-Content
    - Copy-Item
  + Move-Item
  + Set-Content
  + Get-Help
  + Get-ADUser
  + Select-ADUser
  + Where-Object
  + Set-Object
  + Set-Date