

2025

# CMD y PowerShell

UNIDAD 2 – PRACTICA 2  
MARCO VALIENTE RODRÍGUEZ

## Contenido

2. CMD y Powershell. Primeros pasos y comandos básicos.....	3
Objetivos a cumplir .....	3
Instrucciones para la entrega.....	3
Nota sobre el uso de IA .....	3
Evaluación .....	4
Pregunta 1 .....	5
Pregunta 2 .....	5
Pregunta 3 .....	6
Pregunta 4 .....	10
Pregunta 5 .....	11

# Introducción

En esta práctica tenemos que realizar los primeros pasos del sistema operativo, empezando a usar tanto la CMD como la Powershell.

Como dato interesante, los conocimientos previos con el PowerShell de antemano vienen a ser nulos, pero, sin embargo, a la hora de describir y usar la consola, ya tengo más conocimientos, más allá de usar consolas como las de javascript en páginas web o como en la consola de java, o git, sino que tengo experiencia previa con la consola del propio PC para crear carpetas o tareas básicas.

De esta práctica, al menos podré aprender el uso de PowerShell y revisar mis conocimientos de CMD a medida que avance con las cinco cuestiones a responder.

## 2. CMD y Powershell. Primeros pasos y comandos básicos.

En Windows existen dos consolas de línea de comandos: CMD (Símbolo del sistema) y PowerShell. Ambas permiten interactuar con el sistema operativo de forma directa, ejecutar instrucciones y automatizar tareas. Con esta actividad aprenderás a diferenciar entre ambas, acceder a ellas de distintas formas y ejecutar comandos básicos para obtener información del sistema y practicar la navegación en consola.

### Objetivos a cumplir

- Diferenciar entre CMD y PowerShell, identificando sus usos principales.
- Conocer distintas formas de abrir ambas consolas en Windows.
- Ejecutar comandos básicos y analizar los resultados.
- Documentar de forma clara y ordenada las salidas obtenidas.

### Instrucciones para la entrega

- La actividad es **individual**.
- El documento debe seguir la estructura indicada para las actividades anteriores.
- Prepara un documento en el que incluyas:
  - Una breve explicación de las diferencias entre CMD y PowerShell.
  - Las diferentes formas de abrir cada una en Windows.
  - Las salidas de cada comando solicitado, con una breve interpretación de lo que muestran.
- Incluye al final del documento las **fuentes consultadas** (mínimo 3 distintas, además de Wikipedia).

### Nota sobre el uso de IA

Se permite el uso de herramientas de inteligencia artificial como apoyo, siempre que el contenido se reelabore y se comprenda. El copia/pega literal penalizará la nota.

Las **instrucciones** a seguir:

1. Explica brevemente qué es CMD y qué es PowerShell, señalando sus principales diferencias.
2. Investiga y describe al menos dos formas distintas de abrir CMD y PowerShell en Windows (por ejemplo, desde menú inicio, ejecutar, búsqueda, clic derecho...).
3. Abre CMD y ejecuta los siguientes comandos, en este orden, copiando en el documento el resultado (puedes usar capturas de pantalla o copiar/pegar el texto):

systeminfo

cls

dir

ipconfig

color 0a (*observa cómo cambia el color de la consola*)

tree (*ejecútalo en una carpeta con contenido para ver el resultado*)

echo Hola Mundo!

cls

exit (*para cerrar la consola*)

4. Abre PowerShell, ejecuta el comando “get-command + enter” y espera a que se muestren en pantalla los comandos disponibles. No es necesario que ejecutes ninguno, pero captura la salida o describe lo que observas. Para cerrar la consola de Powershell, escribe exit y pulsa enter.
5. Para cada comando (en CMD y en PowerShell), añade una breve interpretación de lo que hace y de la información que muestra.

## Evaluación

La calificación se repartirá entre los criterios del **RA2** y el **RA7**:

- RA2 (contenidos técnicos): 60%
- RA7 (documentación y presentación): 40% Basada en la siguiente rúbrica:

Criterio	0–4 pts (Insuficiente)	5–6 pts (Aceptable)	7–8 pts (Notable)	9–10 pts (Excelente)
<b>RA2 – Identificación de herramientas</b>	No explica qué son CMD y PowerShell o lo hace erróneamente.	Explicación básica, poco clara.	Explica correctamente ambas herramientas y sus usos.	Explicación detallada, con diferencias bien justificadas y ejemplos de uso.
<b>RA2 – Acceso a consolas</b>	No explica cómo acceder o solo menciona un método.	Menciona formas de acceso pero incompletas.	Menciona al menos dos formas de acceso para cada consola.	Explica varias formas de acceso y recomienda las más prácticas.
<b>RA2 – Ejecución e interpretación de comandos</b>	No ejecuta los comandos o resultados ilegibles.	Ejecuta pero no interpreta correctamente.	Ejecuta todos los comandos y explica su función básica.	Ejecuta, interpreta de forma detallada y relaciona con la administración del sistema.
<b>RA7 – Documentación y presentación</b>	Documento desordenado o incompleto.	Documento básico pero legible.	Documento bien estructurado y claro.	Documento impecable, bien maquetado y con redacción excelente.
<b>RA7 – Originalidad y uso de fuentes</b>	Respuestas copiadas sin citar.	Sin fuentes o muy básicas.	Al menos 2 fuentes correctas.	Fuentes variadas, fiables y respuestas reelaboradas.

## Pregunta 1

La CMD es la abreviación a la llamada consola de comandos, también llamada Command Prompt o Símbolo del Sistema en español, y que se ve dentro de un ordenador como una pantalla negra, donde se pueden escribir en esta, líneas básicas que hacer.

PowerShell es una versión más actualizada y moderna con capacidad de integrarse dentro del propio Windows/Linux/MacOS con actuación a través de scripts, gestión remota (a través de la nube) y la automatización multiplataforma. Este nivel de complejidad puede encontrarse excesivo para tareas más casuales.

Una de las mayores diferencias entre estas es que la CMD solo funciona devolviendo texto, lo cual es limitante e ideal para archivos antiguos, mientras que PowerShell trabaja sobre un .NET y es capaz de devolver objetos, ya que esta está orienta sobre estos. La amplia variedad de funciones que un solo objeto puede ofrecer permite que los ingenieros IT sean capaces de administrar el sistema de forma efectiva, así como un control más detallado.

## Pregunta 2

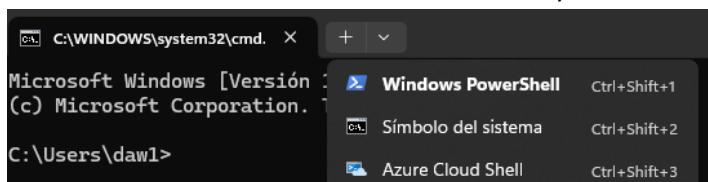
Dentro del Sistema de Windows, inmediatamente se me ocurre simplemente presionar la tecla de Windows en el teclado, que abre el buscador. Una vez que el buscador está abierto, se puede escribir CMD para obtener directamente la consola de CMD. También se puede escribir Símbolo del Sistema ya que estamos trabajando en español.

Otro método para llegar a este dentro de Windows es a través de la tecla de Windows presionada seguido de la letra R, que al soltar se puede escribir cmd dentro de este, y te ejecuta automáticamente para abrir la consola de comandos.

Para PowerShell se puede buscar de la misma manera o usando Windows + R como para la consola de comandos para abrirla.

Las ventajas de usar Ejecutar (Windows + R) es que se le da la orden al ordenador de abrir directamente la ventana con la que se quiere operar con acceso inmediato, mientras que las ventajas de buscarlo dentro del propio inicio de Windows es que para una persona más casual que no necesite de atajos de teclado y simplemente desee tomarse su tiempo y encontrar las cosas de forma sencilla, puede hacerlo sin mayores problemas.

Una última forma de hacerlo es abriendo una y usando esa para abrir la otra.



### Pregunta 3

Ahora vamos a ejecutar dentro de CMD una lista de órdenes.

```
- systeminfo  
C:\Users\daw1>systeminfo  
  
Nombre de host: COMPUS021  
Nombre del sistema operativo: Microsoft Windows 11 Pro Education  
Versión del sistema operativo: 10.0.26200 N/D Compilación 26200  
Fabricante del sistema operativo: Microsoft Corporation  
Configuración del sistema operativo: Estación de trabajo independiente  
Tipo de compilación del sistema operativo: Multiprocessor Free  
Propietario registrado: N/D  
Organización registrada: N/D  
Id. del producto: 00378-60402-40886-AA361  
Fecha de instalación original: 24/06/2025, 9:18:35  
Tiempo de arranque del sistema: 02/12/2025, 14:22:21  
Fabricante del sistema: LENOVO  
Modelo del sistema: 90RJ002VES  
Tipo de sistema: x64-based PC  
Procesadores:  
1 Procesadores instalados.  
[01]: Intel64 Family 6 Model 167 Stepping 1 GenuineIntel ~2592 Mhz  
LENOVO M3GKT29A, 16/06/2021  
Versión del BIOS:  
Directorio de Windows: C:\WINDOWS  
Directorio de sistema: C:\WINDOWS\system32  
Dispositivo de arranque: \Device\HarddiskVolume1  
Configuración regional del sistema: es;Español (internacional)  
Idioma de entrada: es;Español (tradicional)  
Zona horaria: (UTC+01:00) Bruselas, Copenhague, Madrid, París  
Memoria física total: 16.122 MB  
Memoria física disponible: 9.018 MB  
Memoria virtual: tamaño máximo: 17.146 MB  
Memoria virtual: disponible: 7.558 MB  
Memoria virtual: en uso: 9.588 MB  
  
- cls  
  
C:\WINDOWS\system32\cmd. X + ▾  
  
C:\Users\daw1>
```

```
- dir
C:\Users\daw1>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 32A8-18E4

Directorio de C:\Users\daw1

01/12/2025  08:31    <DIR>          .
01/12/2025  12:06    <DIR>          ..
26/09/2025  10:07          206 .gitconfig
29/09/2025  07:03    <DIR>          .junie
26/09/2025  07:32    <DIR>          .skiko
10/10/2025  08:03    <DIR>          .VirtualBox
17/09/2025  11:42    <DIR>          .vscode
17/09/2025  07:45    <DIR>          Contacts
02/12/2025  14:15    <DIR>          Desktop
05/12/2025  14:16    <DIR>          Documents
09/12/2025  09:58    <DIR>          Downloads
17/09/2025  07:45    <DIR>          Favorites
08/10/2025  08:38    <DIR>          IdeaSnapshots
17/09/2025  07:45    <DIR>          Links
17/09/2025  07:45    <DIR>          Music
01/12/2025  08:31    <DIR>          OneDrive
02/12/2025  14:13    <DIR>          OneDrive - Educantabria
18/09/2025  13:11    <DIR>          Pictures
17/09/2025  07:45    <DIR>          Saved Games
17/09/2025  11:04    <DIR>          Searches
19/09/2025  07:04    <DIR>          Videos
10/10/2025  08:01    <DIR>          VirtualBox VMs
                           1 archivos          206 bytes
                           21 dirs   278.228.664.320 bytes libres
```

- ipconfig

C:\Users\daw1>ipconfig

## Configuración IP de Windows

## Adaptador de Ethernet Ethernet 2:

#### **Adaptador de Ethernet Ethernet:**

Adaptador de Ethernet Conexión de red Bluetooth:

Estado de los medios. . . . . : medios desconectados  
Sufijo DNS específico para la conexión. . . :

- color 0a

C:\Users\daw1>color a@

- tree

```
C:\Users\daw1>tree "Documents\Repositorios - Marco\Sistemas Informaticos"
Listado de rutas de carpetas
El n mero de serie del volumen es 32A8-18E4
C:\USERS\DAW1\DOCUMENTS\REPOSITORIOS - MARCO\SISTEMAS INFORMATICOS
|
+-- UD1
    |
    +-- A1.2
    |
    +-- A1.3
    |
    +-- A1.4
|
+-- UD2
    |
    +-- A2.1
    |
    +-- A2.3
    |
    +-- A2.6
|
+-- UD3
    |
    +-- P1
    |
    +-- P2
```

- echo Hola Mundo!

```
C:\Users\daw1>echo Hola Mundo!  
Hola Mundo!
```

- cls

```
C:\Users\daw1>cls  
+ | v
```

```
C:\Users\daw1>
```

- exit

```
C:\Users\daw1>exit  
+ | v
```

```
C:\Users\daw1>exit|
```

## Pregunta 4

Ahora hacemos el salto a seguir las instrucciones dentro de PowerShell.

CommandType	Name	Version	Source
Alias	Add-AppPackage	2.0.1.0	Appx
Alias	Add-AppVolume	2.0.1.0	Appx
Alias	Add-ProvisionedPackage	3.0	Dism
Alias	Add-MsixPackage	2.0.1.0	Appx
Alias	Add-MsixPackageVolume	2.0.1.0	Appx
Alias	Add-MsixVolume	2.0.1.0	Appx
Alias	Add-ProvisionedAppPackage	3.0	Dism
Alias	Add-ProvisionedAppSharedPackageContainer	3.0	Dism
Alias	Add-ProvisionedPackage	3.0	Provisioning
Alias	Add-ProvisioningPackage	3.0	Provisioning
Alias	Add-TrustedProvisioningCertificate	3.0	Dism
Alias	Apply-WindowsUnattend	3.0	
Alias	Disable-PhysicalDiskIndication	2.0.0.0	Storage
Alias	Disable-PhysicalDiskIndication	1.0.0.0	VMDirectStorage
Alias	Disable-StorageDiagnosticLog	2.0.0.0	Storage
Alias	Disable-StorageDiagnosticLog	1.0.0.0	VMDirectStorage
Alias	Disable-StorageManifest	2.0.1.0	Appx
Alias	Dismount-MsixPackageVolume	2.0.1.0	Appx
Alias	Dismount-MsixVolume	2.0.1.0	Appx
Alias	Enable-PhysicalDiskIndication	2.0.0.0	Storage
Alias	Enable-PhysicalDiskIndication	1.0.0.0	VMDirectStorage
Alias	Enable-StorageDiagnosticLog	2.0.0.0	Storage
Alias	Enable-StorageDiagnosticLog	1.0.0.0	VMDirectStorage
Alias	Flush-Volume	2.0.0.0	Storage
Alias	Flushn-Volume	1.0.0.0	VMDirectStorage
Alias	Get-AppPackage	2.0.1.0	Appx
Alias	Get-AppPackageAutoupdateSettings	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-ProvisionedPackage	3.0	Dism
Alias	Get-DiskNV	2.0.0.0	Storage
Alias	Get-DiskNV	1.0.0.0	VMDirectStorage
Alias	Get-Language	1.0	LanguagePackManagement
Alias	Get-MsixBadDefaultVolume	2.0.1.0	Appx
Alias	Get-MsixBadError	2.0.1.0	Appx
Alias	Get-MsixLog	2.0.1.0	Appx
Alias	Get-MsixPackage	2.0.1.0	Appx
Alias	Get-MsixPackageAutoupdateSettings	2.0.1.0	Appx
Alias	Get-MsixPackageDefaultVolume	2.0.1.0	Appx
Alias	Get-MsixPackageLastError	2.0.1.0	Appx
Alias	Get-MsixPackageLog	2.0.1.0	Appx
Alias	Get-MsixPackageManifest	2.0.1.0	Appx
Alias	Get-MsixPackageVolume	2.0.1.0	Appx
Cmdlet	Update-WIMBootEntry	3.0	Dism
Cmdlet	Use-Transaction	3.1.0.0	Microsoft.PowerShell.Management
Cmdlet	Use-WindowsUnattend	3.0	Dism
Cmdlet	Wait-Debugger	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Wait-Event	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Wait-Job	3.0.0.0	Microsoft.PowerShell.Core
Cmdlet	Wait-Process	3.1.0.0	Microsoft.PowerShell.Management
Cmdlet	Where-Object	3.0.0.0	Microsoft.PowerShell.Core
Cmdlet	Write-Debug	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Write-Error	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Write-EventLog	3.1.0.0	Microsoft.PowerShell.Management
Cmdlet	Write-Host	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Write-Information	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Write-Output	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Write-Progress	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Write-Verbose	3.1.0.0	Microsoft.PowerShell.Utility
Cmdlet	Write-Warning	3.1.0.0	Microsoft.PowerShell.Utility

Y después de esto, solo queda:

```
PS C:\Users\daw1> exit
```

Y salió exitosamente.

## Pregunta 5

Ahora, lo que nos queda no es nada más que razonar qué han hecho los comandos que han sido proporcionados tanto PowerShell como la CMD.

- **systeminfo**

Como indica en las siglas en inglés, esto muestra la información del sistema, como el sistema operativo como los detalles físicos del propio sistema.

- **cls**

Este comando es menos evidente hasta que se ejecuta. Lo que hace es limpiar la pantalla de la consola completamente como si la acabases de abrir.

- **dir**

Este comando te habla del directorio del ordenador, así como del espacio en bits libres.

- **ipconfig**

El nombre es una gran pista, y también no recomendaría compartir ese resultado de forma pública a través de redes, pero efectivamente, se trata de la información y configuración de la IP de la conexión de red.

- **color 0a**

Es muy aparente en las imágenes, se trata de dar en código hexadecimal el color de la pantalla.

- **tree**

Abriendo una ruta específica, se pueden mostrar las ramas de carpetas desde una carpeta inicial. Si se ejecuta sin más, te mostrará todas las rutas de todas las carpetas del ordenador en los ámbitos más accesibles al usuario, pero se pueden especificar rutas usando \ y tabulando dentro de las carpetas para navegar.

- **echo Hola Mundo!**

Echo es el comando en el que la consola repite lo que viene a continuación, simplemente como texto, y al escribir "Hola Mundo!", la consola repite "Hola Mundo!"

- **cls**

Como anteriormente descrito, esto resetea la consola. Se puede ver que el color no cambió, por lo que se puede entonces deducir que no se reinicia la consola completamente sino que simplemente se limpia todo lo ejecutado.

- **exit**

Evidentemente. Se cierra la consola. El programa se cierra por sí mismo una vez ejecutado.

- **get-command**

Como te describe el propio enunciado, este comando te lista todas las acciones ejecutables que se pueden hacer dentro de PowerShell, lo cual es algo que se necesita ver más a menudo que en la consola de comandos, ya que PowerShell está orientada a objetos, que pueden tener comandos específicos.

- **exit**

Simplemente se cierra PowerShell. No es una sorpresa.

# Conclusiones

En esta práctica, se puede decir que se han usado con éxito la CMD y PowerShell, y además cerrado cualquier preocupación que existiese con PowerShell debido a la menor familiaridad con la que se ha entrado a esta práctica.

Una de las cosas que se puede confirmar que han sido aprendidas son el hecho de que la consola de comandos usa en Windows la barra \ mientras que, en Linux, esta es sin embargo /.

# Referencias

- <https://netwrix.com/es/resources/blog/powershell-vs-cmd/>