

# Que es powershell?

PowerShell es un lenguaje de scripting y una shell de línea de comandos desarrollado por Microsoft para automatizar tareas, administrar sistemas y controlar aplicaciones.

Se puede considerar como el equivalente moderno del Command Prompt (CMD), pero mucho más potente. ¿Qué puede hacer?

Con PowerShell puedes:

- Administrar Windows, macOS y Linux.
- Automatizar tareas repetitivas.
- Administrar usuarios, discos, servicios y procesos.
- Trabajar con archivos y carpetas.
- Consultar bases de datos.
- Leer y escribir archivos CSV, XML y JSON.
- Administrar servicios en la nube como Microsoft 365 y Azure.
- Controlar VMware, Hyper-V, NetBackup, Active Directory, Exchange, etc., mediante módulos específicos.

Por ejemplo, listar todos los procesos:

Get-Process

Listar todos los servicios:

Get-Service

Mostrar los archivos de una carpeta:

Get-ChildItem ¿Por qué es diferente de CMD?

CMD trabaja principalmente con texto.

PowerShell trabaja con objetos.

Por ejemplo:

Get-Process

Cada proceso es un objeto con propiedades como:

Puedes acceder directamente a ellas:

Get-Process notepad | Select-Object Name, Id, CPU

No tienes que analizar texto como en CMD.

Los cmdlets

Los comandos de PowerShell se llaman cmdlets y siguen la forma:

## Verbo-Sustantivo

Ejemplos:

```
Get-Service
Stop-Service
Start-Service
New-Item
Remove-Item
Copy-Item
Move-Item
```

Esto hace que sean fáciles de recordar.

Variables \$nombre = "Adonis"

Write-Host \$nombre

También puedes guardar listas:

\$archivos = Get-ChildItem C:\Temp Tuberías (Pipelines)

Una de las características más potentes.

Get-Process | Where-Object CPU -gt 100

Hace lo siguiente:

Obtiene todos los procesos. Envía cada objeto por la tubería (|). Filtra solo los que usan más de 100 segundos de CPU.

También puedes ordenar:

Get-Process | Sort-Object CPU -Descending | Select-Object -First 10 Automatización

Puedes crear scripts .ps1.

Ejemplo:

\$fecha = Get-Date

Write-Host "Hola" Write-Host "Hoy es \$fecha"

Y ejecutarlo:

.\script.ps1 Funciones

Puedes crear tus propios comandos

```
function Saludar {
    param($Nombre)

    Write-Host "Hola $Nombre"
```

```
}
```

Saludar "Carlos" Ejemplo práctico

Supón que quieres saber cuánto espacio libre tienen todos los discos:

```
Get-CimInstance Win32_LogicalDisk |  
Select-Object DeviceID,  
@{Name="Libre(GB)";Expression={[math]::Round($_.FreeSpace/1GB,2)}}
```

¿En qué sistemas funciona?

**PowerShell Core (PowerShell 7) funciona en:**

- ☐ Windows
- ☐ Linux
- ☐ macOS

Con prácticamente el mismo lenguaje.

¿Por qué es tan popular entre administradores de sistemas?

Porque permite automatizar prácticamente cualquier tarea. Por ejemplo:

```
Revisar cientos de servidores.  
Copiar archivos automáticamente.  
Enviar correos cuando ocurre un error.  
Comprobar el estado de servicios.  
Leer registros (logs).  
Administrar máquinas virtuales.  
Crear informes en CSV o Excel.  
Programar tareas para que se ejecuten solas.
```

Fuente: chatGPT

From:

<https://wiki.morkey.ch/> - **Morkey-Wiki**

Permanent link:

[https://wiki.morkey.ch/doku.php?id=powershell:00\\_info&rev=1786045499](https://wiki.morkey.ch/doku.php?id=powershell:00_info&rev=1786045499)

Last update: **2026/08/06 21:44**

