

Actividad 04. Ejemplos de uso de cmdlets en PowerShell

20 ejemplos sencillos de uso de cmdlets PowerShell

1. Obtener la hora actual

```
Get-Date
```

2. Listar archivos en un directorio

```
Get-ChildItem -Path C:\Users
```

3. Ver los servicios del sistema

```
Get-Service
```

4. Detener un servicio

```
Stop-Service -Name "Spooler"
```

5. Iniciar un servicio

```
Start-Service -Name "Spooler"
```

6. Reiniciar un servicio

```
Restart-Service -Name "Spooler"
```

7. Listar procesos en ejecución

```
Get-Process
```

8. Terminar un proceso

```
Stop-Process -Name "notepad"
```

9. Obtener información del sistema

```
Get-ComputerInfo
```

10. Crear un directorio

```
New-Item -Path "C:\NewFolder" -ItemType Directory
```

11. Eliminar un archivo

```
Remove-Item -Path "C:\NewFolder\file.txt"
```

12. Copiar un archivo

```
Copy-Item -Path "C:\source\file.txt" -Destination "C:\destination\file.txt"
```

13. Mover un archivo

```
Move-Item -Path "C:\source\file.txt" -Destination "C:\destination\file.txt"
```

14. Cambiar la política de ejecución de scripts

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned
```

15. Ver la política de ejecución actual

```
Get-ExecutionPolicy
```

16. Exportar una lista de procesos a un archivo CSV

```
Get-Process | Export-Csv -Path "C:\processes.csv" -NoTypeInfoation
```

17. Importar datos de un archivo CSV

```
Import-Csv -Path "C:\processes.csv"
```

18. Crear una variable y asignarle un valor

```
$greeting = "Hello, PowerShell!"
```

19. Mostrar el valor de una variable

```
Write-Output $greeting
```

20. Filtrar servicios en ejecución

```
Get-Service | Where-Object { $_.Status -eq 'Running' }
```

20 ejemplos intermedios de uso de cmdlets PowerShell

1. Filtrar servicios por estado

```
Get-Service | Where-Object { $_.Status -eq 'Running' }
```

2. Buscar archivos grandes en una carpeta

```
Get-ChildItem -Path C:\Users\YourUsername\Documents -Recurse | Where-Object {  
    $_.Length -gt 1GB }
```

3. Modificar el contenido de un archivo de texto

```
(Get-Content -Path "C:\Users\YourUsername\Documents\example.txt") -replace  
"oldText", "newText" | Set-Content -Path  
"C:\Users\YourUsername\Documents\example.txt"
```

4. Obtener el espacio libre en todos los discos

```
Get-PSDrive -PSProvider FileSystem | Select-Object Name,  
@{Name="Free(GB)";Expression={ "{0:N2}" -f ($_.Free/1GB) }}
```

5. Enviar un correo electrónico

```
Send-MailMessage -From "you@example.com" -To "someone@example.com" -Subject  
"Test email" -Body "This is a test email." -SmtpServer "smtp.example.com"
```

6. Descargar un archivo desde Internet

```
Invoke-WebRequest -Uri "https://example.com/file.zip" -OutFile  
"C:\Users\YourUsername\Downloads\file.zip"
```

7. Crear y administrar un directorio

```
New-Item -Path "C:\Users\YourUsername\Documents\NewFolder" -ItemType Directory  
Set-Location -Path "C:\Users\YourUsername\Documents\NewFolder"
```

8. Obtener y modificar el registro

```
# Obtener una clave del registro
Get-ItemProperty -Path "HKCU:\Software\Microsoft\Windows\CurrentVersion\Explorer"

# Modificar una clave del registro
Set-ItemProperty -Path "HKCU:\Software\Microsoft\Windows\CurrentVersion\Explorer" -
Name "ExampleKey" -Value "NewValue"
```

9. Administrar procesos

```
# Listar procesos que usan más del 50% de la CPU
Get-Process | Where-Object { $_.CPU -gt 50 }

# Detener un proceso por nombre
Stop-Process -Name "notepad"
```

10. Crear y usar funciones personalizadas

```
function Get-DiskSpace {
    param (
        [string]$drive = "C"
    )
    Get-PSDrive -Name $drive | Select-Object Name,
    @{Name="FreeSpace(GB)";Expression={ "{0:N2}" -f ($_.Free/1GB) }}
}

# Usar la función
Get-DiskSpace -drive "C"
```

10 ejemplos sencillos de uso de cmdlets PowerShell

1. Hello World

```
# File: HelloWorld.ps1  
Write-Host "Hello, World!"
```

2. Listar Archivos en un Directorio

```
# File: ListFiles.ps1  
$path = "C:\Users\YourUsername\Documents"  
Get-ChildItem -Path $path
```

3. Crear un Directorio

```
# File: CreateDirectory.ps1  
$path = "C:\Users\YourUsername\Documents\NewFolder"  
New-Item -Path $path -ItemType Directory
```

4. Copiar Archivos

```
# File: CopyFiles.ps1  
$source = "C:\Users\YourUsername\Documents\source.txt"  
$destination = "C:\Users\YourUsername\Documents\backup\source.txt"  
Copy-Item -Path $source -Destination $destination
```

5. Renombrar Archivos

```
# File: RenameFiles.ps1
$path = "C:\Users\YourUsername\Documents\oldname.txt"
$newName = "C:\Users\YourUsername\Documents\newname.txt"
Rename-Item -Path $path -NewName $newName
```

6. Eliminar Archivos

```
# File: DeleteFiles.ps1
$path = "C:\Users\YourUsername\Documents\delete.txt"
Remove-Item -Path $path
```

7. Leer un Archivo de Texto

```
# File: ReadFile.ps1
$path = "C:\Users\YourUsername\Documents\readme.txt"
$content = Get-Content -Path $path
Write-Host $content
```

8. Escribir en un Archivo de Texto

```
# File: WriteFile.ps1
$path = "C:\Users\YourUsername\Documents\write.txt"
$text = "This is some text."
Set-Content -Path $path -Value $text
```

9. Obtener Información del Sistema

```
# File: SystemInfo.ps1
$os = Get-ComputerInfo | Select-Object CsName, WindowsVersion,
OsArchitecture
Write-Host "Computer Name: $($os.CsName)"
Write-Host "Windows Version: $($os.WindowsVersion)"
Write-Host "OS Architecture: $($os.OsArchitecture)"
```

10. Enviar un Correo Electrónico

```
# File: SendEmail.ps1
$smtpServer = "smtp.example.com"
$from = "you@example.com"
$to = "someone@example.com"
$subject = "Test Email"
$body = "This is a test email sent from PowerShell."
Send-MailMessage -SmtpServer $smtpServer -From $from -To $to -Subject
$subject -Body $body
```