

1. Introducción a Powershell (II)

- Realiza las siguientes tareas que se te piden utilizando Powershell. Para contestar lo mejor es que hagas una captura de pantalla donde se vea el comando que has introducido y las primeras líneas de la salida de este.

1. Visualiza las últimas cinco entradas del historial, mostrando para cada una el comando, la hora en que finalizó su ejecución y el estado de ejecución.

```
- Get-History | Select-Object -Last 5 | Format-Table -Property CommandLine, EndExecutionTime, ExecutionStatus
```

-Captura

2. Ejecuta el comando Get-Command (que muestra todos los comandos disponibles en Powershell) e interrúmpelo antes de que finalice su ejecución pulsando las teclas Ctrl-C. A continuación, ejecútalo dejando que finalice correctamente.

```
- Get-Command
```

-Captura

3. Vuelve a ejecutar el comando del punto 1 y comprueba las diferentes salidas de finalización de estado de ejecución.

- Esto permite observar la diferencia en ExecutionStatus.

4. Muestra todos los procesos con el nombre msedge mostrando para cada uno el identificador, el consumo de CPU y los hilos (threads)

```
- Get-Process msedge | Select-Object Id, CPU, Threads
```

-Captura

5. Averigua para qué sirve el parámetro -Delimiter del comando Export-CSV

```
- Get-Help Export-CSV -Parameter Delimiter
```

-Captura

6. Muestra en una ventana la ayuda del comando Get-History

```
-Get-Help Get-History -ShowWindow
```

-Captura

7. Muestra un listado con todos los comandos que tengan el verbo Update.

```
-Get-Command -Verb Update
```

-Captura

8. Ejecuta la herramienta Recortes y localízala usando el comando Get-Process teniendo en cuenta que el proceso se llama SnippingTool.exe

```
-Get-Process -Name SnippingTool
```

-Captura

9. Averigua qué propiedades tienen los procesos devueltos con el comando Get-Process.

```
-Get-Process | Get-Member
```

-Captura

10. Busca en la ayuda para qué sirve el parámetro -MemberType del comando Get-Member.

```
Get-Help Get-Member -Parameter MemberType
```

-Captura

11. Desde la línea de comandos, finaliza la ejecución de la herramienta Recortes.

```
-Stop-Process -Name SnippingTool
```

-Captura

12. Muestra todos los procesos que tienen el nombre svchost.

```
-Get-Process -Name svchost
```

-Captura

13. Muestra por pantalla el número de instancias del proceso svchost.

```
-(Get-Process -Name svchost).Count
```

-Captura

14. Muestra por pantalla todos los procesos con el nombre svchost mostrando para cada uno: nombre, identificador, hora de inicio, tiempo total de procesador y clase de prioridad. Se deben mostrar de forma tabular.

```
-Get-Process -Name svchost | Select-Object Name, Id, StartTime, TotalProcessorTime, PriorityClass | Format-Table
```

-Captura

15. Repite la búsqueda anterior, pero ordenando por el campo tiempo total de procesador en sentido descendente.

```
-Get-Process -Name svchost | Sort-Object TotalProcessorTime -Descending | Select-Object Name, Id, StartTime, TotalProcessorTime, PriorityClass | Format-Table
```

-Captura

16. Muestra los usuarios que hay en el sistema agrupándolos por la propiedad Enabled.

```
-Get-LocalUser | Group-Object -Property Enabled
```

-Captura

17. Muestra los usuarios que hay en el sistema con la cuenta habilitada (propiedad Enabled puesta a True). Utiliza el filtrado con el comando Where-Object

```
-Get-LocalUser | Where-Object {$_.Enabled -eq $true}
```

-Captura

18. Muestra un listado de todos los usuarios del sistema con el nombre y la fecha de la última vez que iniciaron sesión (tienes que buscar la propiedad que indique último inicio de sesión o last logon)

```
-Get-LocalUser | Select-Object Name, LastLogon
```

-Captura