

PR0302: Introducción a Powershell (II)

Nombre/s: Alejandro Prieto Pellitero

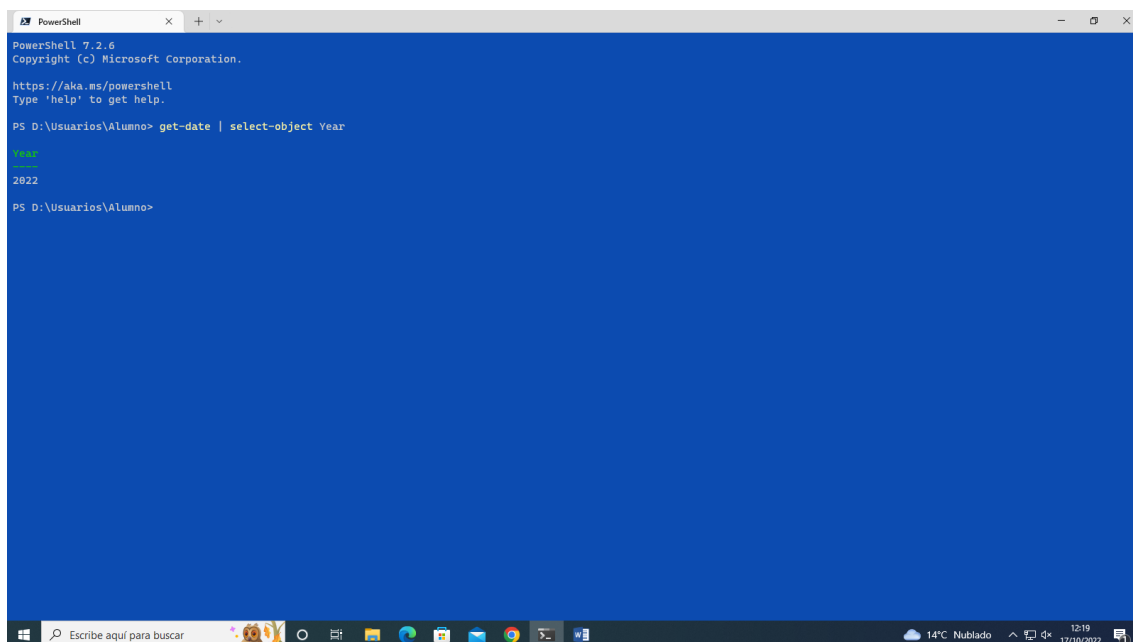
Tienes que contestar las preguntas en este mismo fichero después de cada pregunta. No te olvides de poner tu nombre en el recuadro superior.

Cuando hayas acabado todas las prácticas renombra el fichero para que se llame **{Apellido1} {Apellido2}, {Nombre} – PR0302**. En el nombre y apellidos la primera mayúscula y el resto en minúsculas. El fichero tiene que estar en formato PDF. **Cualquier fichero que no siga esta nomenclatura o no esté en PDF no será corregido.** El fichero final lo tienes que subir a la plataforma.

Ejercicio 1: Powershell

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.- El comando **Get-Date** muestra la fecha y hora actual. Muestra por pantalla únicamente el año en que estamos.



```
PowerShell 7.2.6
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

PS D:\Usuarios\Alumno> get-date | select-object Year

Year
----
2022

PS D:\Usuarios\Alumno>
```

2.- Uno de los requisitos de Windows 11 es que el procesador tenga TPM habilitado. Powershell dispone del comando **Get-TPM** que nos muestra información sobre este módulo. Muestra por pantalla, en formato tabla, las propiedades **TpmPresent**, **TpmReady**, **TpmEnabled** y **TpmActivated**.

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> get-tpm | Select-Object tpmPresent, tpmReady, tpmEnabled, tpmActivated | Format-Table
tpmPresent tpmReady tpmEnabled tpmActivated
True      True      True      True
PS D:\Usuarios\Alumno> |
```

En los siguientes ejercicios trabajaremos con los ficheros devueltos por el comando **Get-ChildItem C:\Windows\System32**.

3.- Muestra por pantalla el número de ficheros y directorios que hay en ese directorio.

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> Get-ChildItem C:\Windows\System32 | Measure-Object

Count           4631
Average
Sum
Standard
Minimum
Maximum
StandardDeviation
Property
PS D:\Usuarios\Alumno> |
```

4.- Los objetos devueltos por el comando anterior tienen una propiedad denominada **Extension**, que indica la extensión del archivo. Calcula el número de ficheros en el directorio que tienen la extensión **.dll**.

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> Get-ChildItem C:\Windows\System32 | where-object Extension -eq ".dll" | Measure-Object

Count                3448
Average
Count
Maximum
Minimum
StandardDeviation
Variance

PS D:\Usuarios\Alumno> |
```

5.- Muestra los ficheros del directorio con extensión **.exe** que tengan un tamaño superior a 50000 bytes.

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> Get-ChildItem C:\Windows\System32 | where-object Extension -eq ".exe" | where-object Length -gt "50000" | Measure-Object

Count                438
Average
Count
Maximum
Minimum
StandardDeviation
Variance

PS D:\Usuarios\Alumno> |
```

6.- Muestra los ficheros de este directorio que tengan extensión **.dll**, ordenados por fecha de creación y mostrando únicamente las propiedades de fecha de creación (*CreationTime*), último acceso (*LastAccessTime*) y nombre (*Name*).

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> Get-ChildItem C:\Windows\System32 | Where-Object extension -eq ".dll" | sort-object CreationTime | select-object CreationTime, LastAccessTime, name

CreationTime      LastAccessTime      Name
-----
02/02/2019 3:34:52 04/10/2022 14:16:34 msvcpi80.dll
02/02/2019 3:34:52 04/10/2022 14:16:34 msvcpi80.dll
27/09/2019 20:06:40 04/10/2022 14:16:11 msvcpi408_1.dll
27/09/2019 20:06:40 04/10/2022 14:16:11 vcomp1408.dll
27/09/2019 20:06:42 04/10/2022 14:16:34 mfc1408rus.dll
27/09/2019 20:06:42 04/10/2022 14:16:34 mfc1408kor.dll
27/09/2019 20:06:44 14/10/2022 12:59:20 conctrl408.dll
27/09/2019 20:06:44 04/10/2022 14:16:34 mfc1408ita.dll
27/09/2019 20:06:44 17/10/2022 12:18:42 vcruntime1408.dll
27/09/2019 20:06:46 04/10/2022 14:16:34 mfc1408fra.dll
27/09/2019 20:06:46 04/10/2022 14:16:33 mfc1408deu.dll
27/09/2019 20:06:46 04/10/2022 14:16:33 mfc1408chs.dll
27/09/2019 20:06:46 17/10/2022 12:18:42 msvcpi408.dll
27/09/2019 20:06:46 04/10/2022 14:16:11 msvcpi408_codecvt_ids.dll
27/09/2019 20:06:48 04/10/2022 14:16:33 mfc1408enu.dll
27/09/2019 20:07:10 04/10/2022 14:16:11 vccorlib1408.dll
27/09/2019 20:07:10 04/10/2022 14:16:11 vcomp1408.dll
27/09/2019 20:07:12 04/10/2022 14:16:11 msvcpi408_2.dll
27/09/2019 20:07:12 17/10/2022 10:45:25 vcruntime1408_1.dll
27/09/2019 20:07:12 04/10/2022 14:16:34 mfc1408jpn.dll
27/09/2019 20:07:12 04/10/2022 14:16:34 mfc1408esh.dll
27/09/2019 20:07:14 04/10/2022 14:16:37 mfc1408u.dll
27/09/2019 20:07:34 04/10/2022 14:16:33 mfc1408cht.dll
27/09/2019 20:07:34 04/10/2022 14:16:37 mfc1408.dll
27/09/2019 20:13:54 04/10/2022 14:16:33 mfcml408.dll
27/09/2019 20:13:54 04/10/2022 14:16:33 mfcml408u.dll
07/12/2019 10:07:50 13/10/2022 11:26:39 CFMCInst.dll
07/12/2019 10:07:53 08/09/2022 14:46:26 HalExtIntclpioDMA.dll
07/12/2019 10:07:53 08/09/2022 14:46:26 HalExtPLB88.dll
07/12/2019 10:07:56 13/10/2022 11:25:00 HidCfu.dll
07/12/2019 10:08:05 13/10/2022 12:37:57 ipxlatcfg.dll
07/12/2019 10:08:05 13/10/2022 11:25:09 luiaapi.dll
07/12/2019 10:08:05 13/10/2022 11:26:29 ApplicationControlCSP.dll
07/12/2019 10:08:05 17/10/2022 10:45:26 energyprov.dll
07/12/2019 10:08:05 13/10/2022 11:24:54 seutil.dll
07/12/2019 10:08:05 13/10/2022 11:26:13 XblGameSaveProxy.dll
07/12/2019 10:08:05 13/10/2022 11:26:13 XblGameSaveExt.dll
```

7.- Muestra el tamaño (*Length*) y nombre completo (*FullName*) de todos los ficheros del directorio ordenados por tamaño en sentido descendente.

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> Get-ChildItem C:\Windows\System32 | Select-Object Length, FullName | Sort-Object length -Descending

Length FullName
-----
147398920 C:\Windows\System32\WRT.exe
32608704 C:\Windows\System32\WindowsCodecsRaw.dll
31514312 C:\Windows\System32\nvoglv64.dll
26268672 C:\Windows\System32\edgehtml.dll
24272384 C:\Windows\System32\Hydrogen.dll
23449600 C:\Windows\System32\mshtml.dll
22992072 C:\Windows\System32\nvcompiler.dll
18970872 C:\Windows\System32\WolframWorld.dll
18634264 C:\Windows\System32\nvvgf2umx.dll
17559432 C:\Windows\System32\nvdf2umx.dll
17551872 C:\Windows\System32\Windows.UI.Xaml.dll
14208384 C:\Windows\System32\vmms.exe
13916608 C:\Windows\System32\nvopenccl.dll
13828032 C:\Windows\System32\nvcodec.dll
11445248 C:\Windows\System32\wmp.dll
10846592 C:\Windows\System32\ntoskrnl.exe
10349360 C:\Windows\System32\Windows.Media.PlayReady.dll
9893376 C:\Windows\System32\WlsLexicons008a.dll
9692160 C:\Windows\System32\WlsData008a.dll
9037312 C:\Windows\System32\wingmapc.dll
8400896 C:\Windows\System32\wstscax.dll
8233048 C:\Windows\System32\OneCoreUAPCommonProxyStub.dll
7972240 C:\Windows\System32\windows.storage.dll
7769888 C:\Windows\System32\Chakra.dll
7715328 C:\Windows\System32\ieframe.dll
7645248 C:\Windows\System32\shell32.dll
7508640 C:\Windows\System32\Windows.Media.dll
7267680 C:\Windows\System32\d3d10warp.dll
6783384 C:\Windows\System32\nvcpl.dll
6725120 C:\Windows\System32\Windows.Data.Pdf.dll
6570496 C:\Windows\System32\dbgeng.dll
6585576 C:\Windows\System32\vmchipset.dll
6430208 C:\Windows\System32\winui_peshell.dll
6191104 C:\Windows\System32\winui.dll
6070904 C:\Windows\System32\d2d1.dll
5865488 C:\Windows\System32\spwizimg.dll
5858672 C:\Windows\System32\Windows.StateRepository.dll
5813016 C:\Windows\System32\mfc1408u.dll
```

8.- Muestra el tamaño y nombre completo de todos los ficheros del directorio que tengan un tamaño superior a 10MB (10000000 bytes) ordenados por tamaño.

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> Get-ChildItem C:\Windows\System32 | Where-Object Length -gt "10000000" | Select-Object Length, FullName | Sort-Object Length

Length FullName
-----
18349368 C:\Windows\System32\Windows.Media.PlayReady.dll
18846592 C:\Windows\System32\ntoskrnl.exe
11405248 C:\Windows\System32\wmp.dll
13828832 C:\Windows\System32\nvcuda.dll
13916688 C:\Windows\System32\nvopencl.dll
14288384 C:\Windows\System32\vmms.exe
17551872 C:\Windows\System32\Windows.UI.Xaml.dll
17559432 C:\Windows\System32\nvd3dumx.dll
18634208 C:\Windows\System32\nvvpf2umx.dll
18767872 C:\Windows\System32\HologramWorld.dll
22992872 C:\Windows\System32\nvcompiler.dll
23449688 C:\Windows\System32\mshtml.dll
24272384 C:\Windows\System32\Hydrogen.dll
26268672 C:\Windows\System32\edgehtml.dll
31514312 C:\Windows\System32\nvoglv64.dll
32688704 C:\Windows\System32\WindowsCodecsRaw.dll
147398824 C:\Windows\System32\WRT.exe

PS D:\Usuarios\Alumno>
```

9.- Muestra el tamaño y nombre completo de todos los ficheros del directorio que tengan un tamaño superior a 10MB y extensión .exe ordenados por tamaño.

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> Get-ChildItem C:\Windows\System32 | Where-Object Length -gt "10000000" | Where-Object Extension -eq ".exe" | Select-Object Length, FullName | Sort-Object Length

Length FullName
-----
18846592 C:\Windows\System32\ntoskrnl.exe
14288384 C:\Windows\System32\vmms.exe
147398824 C:\Windows\System32\WRT.exe

PS D:\Usuarios\Alumno>
```

Hemos visto cómo usar el comando Where-Object para filtrar objetos con propiedades de tipo texto o numérico (por ejemplo, Where-Object CPU -gt 1 o Where-Object Name -eq "Notepad", sin embargo, hay propiedades que pueden tener otro tipo de datos. Dos de estos datos son los booleanos y los de tipo fecha.

- Las propiedades **booleanas** son las que pueden tener un valor de Verdadero o Falso, por ejemplo, la propiedad Exists del comando Get-ChildItem.

```

DirectoryName      Property      string DirectoryName {get
Exists             Property      bool Exists {get;}
Extension          Property      string Extension {get;}

```

Cuando queremos filtrar por estas propiedades y queremos poner que un valor es verdadero o falso, no podemos poner directamente True o False, ya que el sistema las interpretará como cadenas de texto en lugar de hacerlo como valores booleanos. En estos casos, es necesario utilizar dos variables del sistema que representaremos de la forma **\$True** y **\$False**.

- Otro tipo de propiedades muy común son las de fecha y hora, que podemos encontrar por ejemplo en la fecha de creación de un fichero.

```

PS C:\Users\victor> Get-ChildItem | Get-Member CreationTime

TypeName: System.IO.DirectoryInfo

Name      MemberType Definition
-----
CreationTime Property      datetime CreationTime {get;set;}

```

- Aquí encontramos el mismo problema que en el caso anterior ya que si ponemos la fecha directamente la interpretará como una cadena. En este caso, hay que utilizar el comando **Get-Date** con el parámetro **-date** que convierte una fecha en modo texto a un objeto de tipo datetime que almacena dicha fecha.

```

PS C:\Users\victor> get-date -date "2 de noviembre de 2021"

martes, 2 de noviembre de 2021 0:00:00

```

Pero ahora hay otro problema, ¿cómo hacemos para incluir el valor devuelto por este comando en el parámetro de otro comando? En este caso tenemos que recurrir a los paréntesis de la siguiente forma:

```

PS C:\Users\victor> Get-ChildItem | Where-Object CreationTime -gt (Get-Date -date "1 de octubre de 2021")

```

Los **paréntesis** hacen que en primer lugar se ejecute el comando que hay en su interior y, el valor devuelto por dicho comando reemplazará todo lo que hay entre paréntesis.

Hay diversas formas de indicar la fecha que se le pasa al comando Get-Date, tanto con fecha y hora como solo fecha. Algunos ejemplos son:

- "2 de noviembre de 2021 10:05:00"
- "02/11/2021"
- "02/11/21 10:10:30"
- "2021-02-11"

Teniendo en cuenta lo anterior, realiza los siguientes ejercicios:

10.- Muestra todos los procesos que tienen el estado Respond puesto a False, es decir, todos los procesos del sistema que se hayan colgado.

```
Administrator: PowerShell
PS D:\Usuarios\Alumno> get-process | Where-Object Responding -eq $False

Name      PID(K)  PId(M)  CPU(s)  ID  ProcessName
-----
36        34,46   1,45    0,45   1  SystemSettings

PS D:\Usuarios\Alumno>
PS D:\Usuarios\Alumno>
```

11.- Muestra todos los ficheros de C:\Windows que hayan sido creados con fecha posterior al 15 de octubre.

```
PowerShell
PS D:\Usuarios\Alumno> Get-ChildItem C:\Windows | Where-Object CreationTime -GT (Get-Date -date "15/10/2022")

Directory: C:\Windows

Name      CreationTime      Length Name
-----
d-----  18/10/2022  13:25      Nueva carpeta

PS D:\Usuarios\Alumno> |
```