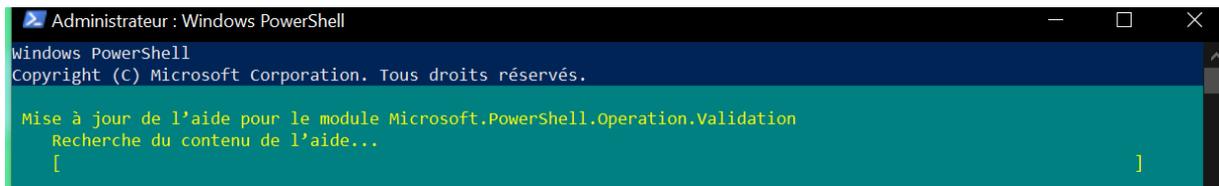


1<sup>ère</sup> partie les commandelettes de base :

- 1) Pour mettre à jour notre fichier d'aide on utilise la commande : Update-Help

Image ci-dessous :

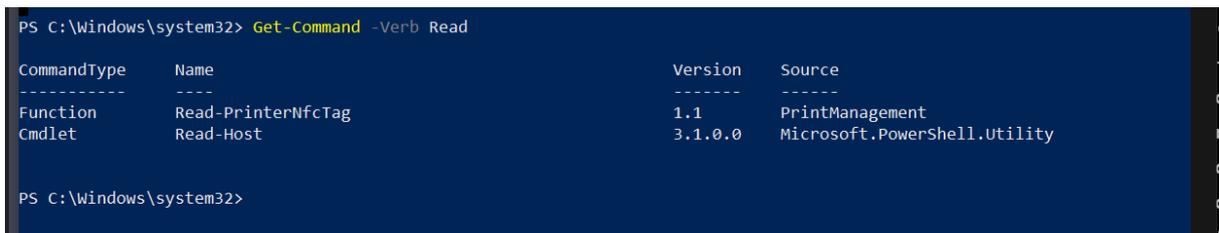


```
Administrateur : Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Mise à jour de l'aide pour le module Microsoft.PowerShell.Operation.Validation
Recherche du contenu de l'aide..
[ ]
```

- 2) Les commandes relatives au verbe read est : Get-Command -Verb Read

Image ci-dessous :



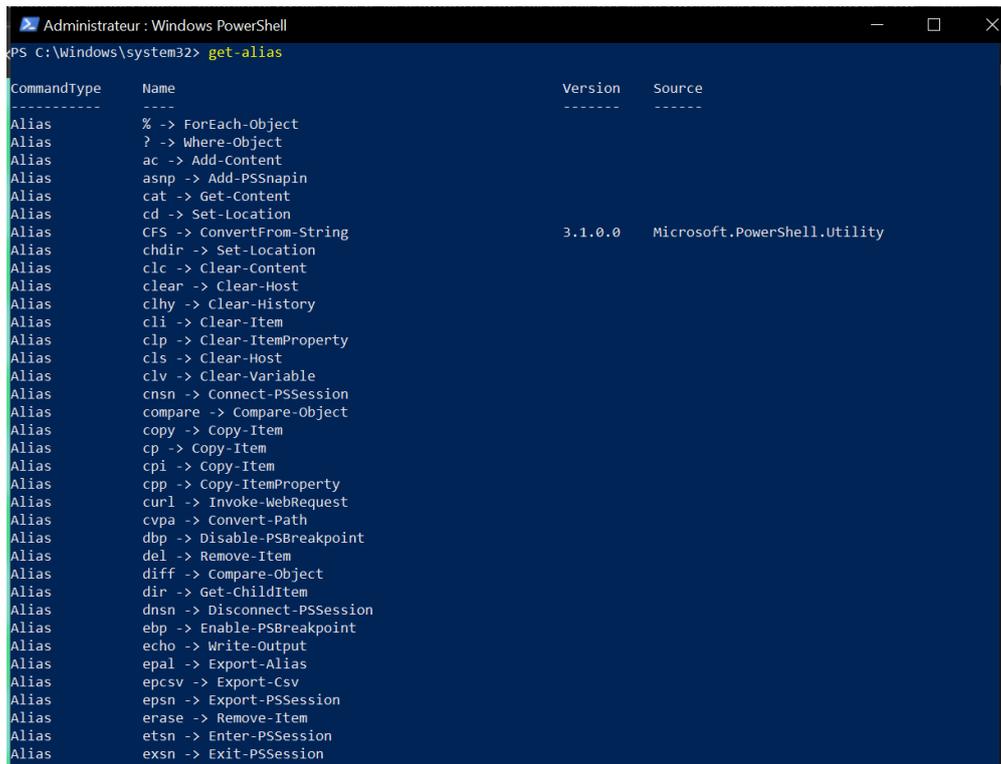
```
PS C:\Windows\system32> Get-Command -Verb Read

CommandType      Name                               Version      Source
-----
Function         Read-PrinterNfcTag                1.1         PrintManagement
Cmdlet           Read-Host                          3.1.0.0     Microsoft.PowerShell.Utility

PS C:\Windows\system32>
```

- 3) Les commandes qui sont associées à alias est : get-alias

Image ci-dessous :

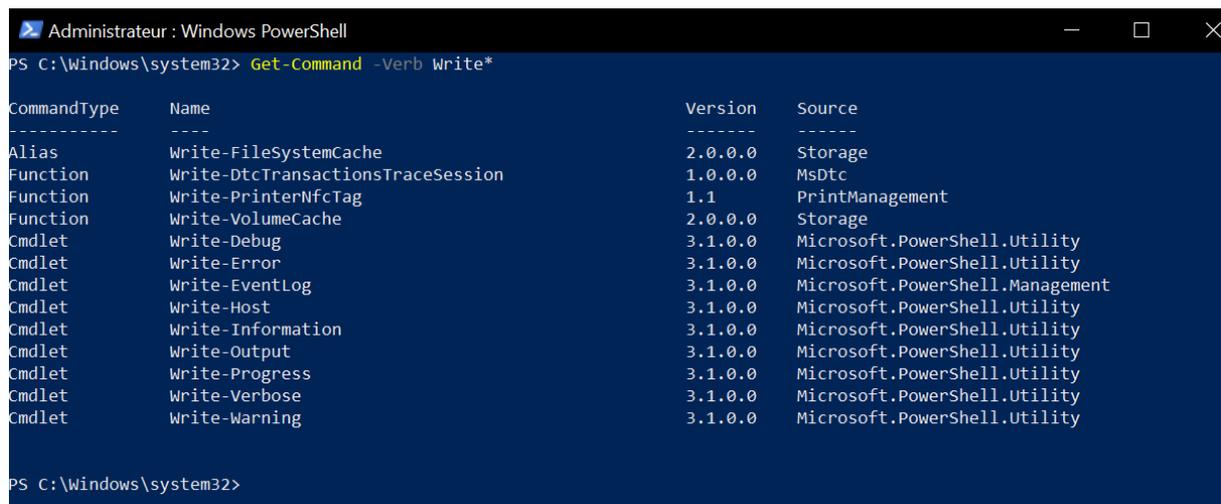


```
Administrateur : Windows PowerShell
PS C:\Windows\system32> get-alias

CommandType      Name                                     Version      Source
-----
Alias             % -> ForEach-Object
Alias             ? -> Where-Object
Alias             ac -> Add-Content
Alias             asnp -> Add-PSSnapin
Alias             cat -> Get-Content
Alias             cd -> Set-Location
Alias             CFS -> ConvertFrom-String              3.1.0.0     Microsoft.PowerShell.Utility
Alias             chdir -> Set-Location
Alias             clc -> Clear-Content
Alias             clear -> Clear-Host
Alias             clhy -> Clear-History
Alias             cli -> Clear-Item
Alias             clp -> Clear-ItemProperty
Alias             cls -> Clear-Host
Alias             clv -> Clear-Variable
Alias             cnsn -> Connect-PSession
Alias             compare -> Compare-Object
Alias             copy -> Copy-Item
Alias             cp -> Copy-Item
Alias             cpi -> Copy-Item
Alias             cpp -> Copy-ItemProperty
Alias             curl -> Invoke-WebRequest
Alias             cvpa -> Convert-Path
Alias             dbp -> Disable-PSBreakpoint
Alias             del -> Remove-Item
Alias             diff -> Compare-Object
Alias             dir -> Get-ChildItem
Alias             dnsn -> Disconnect-PSession
Alias             ebp -> Enable-PSBreakpoint
Alias             echo -> Write-Output
Alias             epal -> Export-Alias
Alias             epcsv -> Export-Csv
Alias             epsn -> Export-PSession
Alias             erase -> Remove-Item
Alias             etsn -> Enter-PSession
Alias             exsn -> Exit-PSession
```

- 4) Pour effectuer une recherche générique pour le verbe write, on utilise la commande :  
Get-Command -Verb Write\*

Image ci-dessous :



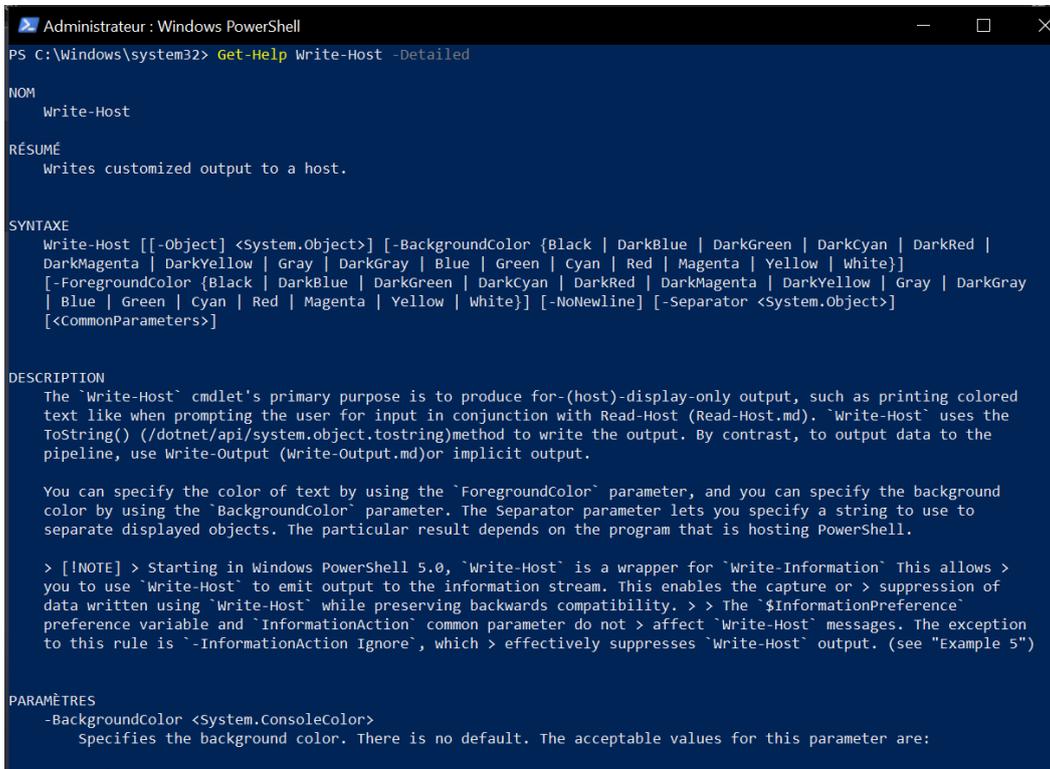
```
Administrateur : Windows PowerShell
PS C:\Windows\system32> Get-Command -Verb Write*

CommandType      Name                                     Version      Source
-----
Alias             Write-FileSystemCache                  2.0.0.0     Storage
Function          Write-DtcTransactionsTraceSession     1.0.0.0     MsDtc
Function          Write-PrinterNfcTag                   1.1         PrintManagement
Function          Write-VolumeCache                     2.0.0.0     Storage
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

PS C:\Windows\system32>
```

- 5) Pour demander de l'aide détaillée avec write-host, la commande est :  
Get-Help Write-Host -Detailed

Image ci-dessous :



```
Administrateur : Windows PowerShell
PS C:\Windows\system32> Get-Help Write-Host -Detailed

NOM
    Write-Host

RÉSUMÉ
    Writes customized output to a host.

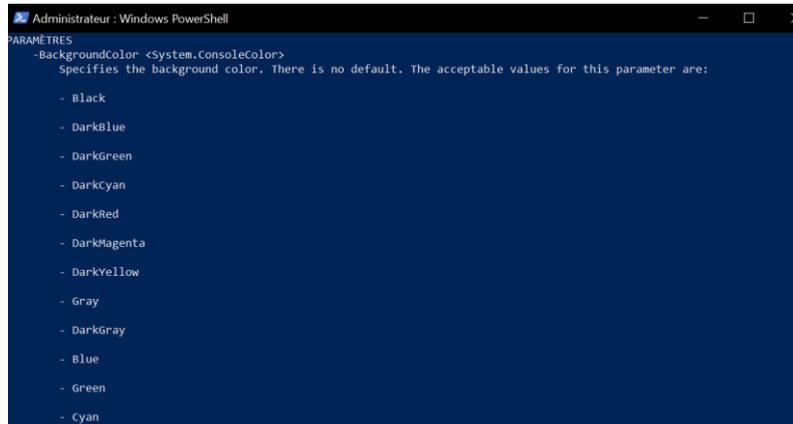
SYNTAXE
    Write-Host [[-Object] <System.Object>] [-BackgroundColor {Black | DarkBlue | DarkGreen | DarkCyan | DarkRed |
    DarkMagenta | DarkYellow | Gray | DarkGray | Blue | Green | Cyan | Red | Magenta | Yellow | White}]
    [-ForegroundColor {Black | DarkBlue | DarkGreen | DarkCyan | DarkRed | DarkMagenta | DarkYellow | Gray | DarkGray
    | Blue | Green | Cyan | Red | Magenta | Yellow | White}] [-NoNewline] [-Separator <System.Object>]
    [<CommonParameters>]

DESCRIPTION
    The `Write-Host` cmdlet's primary purpose is to produce for-(host)-display-only output, such as printing colored
    text like when prompting the user for input in conjunction with Read-Host (Read-Host.md). `Write-Host` uses the
    ToString() (/dotnet/api/system.object.tostring)method to write the output. By contrast, to output data to the
    pipeline, use Write-Output (Write-Output.md)or implicit output.

    You can specify the color of text by using the `ForegroundColor` parameter, and you can specify the background
    color by using the `BackgroundColor` parameter. The Separator parameter lets you specify a string to use to
    separate displayed objects. The particular result depends on the program that is hosting PowerShell.

    > [!NOTE] > Starting in Windows PowerShell 5.0, `Write-Host` is a wrapper for `Write-Information` This allows >
    you to use `Write-Host` to emit output to the information stream. This enables the capture or > suppression of
    data written using `Write-Host` while preserving backwards compatibility. > > The `$InformationPreference`
    preference variable and `InformationAction` common parameter do not > affect `Write-Host` messages. The exception
    to this rule is `InformationAction Ignore`, which > effectively suppresses `Write-Host` output. (see "Example 5")

PARAMÈTRES
    -BackgroundColor <System.ConsoleColor>
        Specifies the background color. There is no default. The acceptable values for this parameter are:
```



```
Administrateur : Windows PowerShell
PARAMÈTRES
    -BackgroundColor <System.ConsoleColor>
        Specifies the background color. There is no default. The acceptable values for this parameter are:

    - Black
    - DarkBlue
    - DarkGreen
    - DarkCyan
    - DarkRed
    - DarkMagenta
    - DarkYellow
    - Gray
    - DarkGray
    - Blue
    - Green
    - Cyan
```

Etc.

- 6) Pour demander de l'aide détaillée à propos des commandes commençant par w est :  
**Get-Help -Name w\* -Detailed**  
**Ou : Get-Help w\* -Detailed**

Image ci-dessous :

```

Administrateur : Windows PowerShell
REMARQUES
  Pour consulter les exemples, tapez : "get-help Write-Host -examples".
  Pour plus d'informations, tapez : "get-help Write-Host -detailed".
  Pour obtenir des informations techniques, tapez : "get-help Write-Host -full".
  Pour l'aide en ligne, tapez : "get-help Write-Host -online"

PS C:\Windows\system32> Get-Help -Name w* -Detailed

Name
----
where
write
wget
wjb
w:
Wait-Job
Where-Object
Wait-Debugger
Wait-Event
Write-Debug
Write-Error
Write-Host
Write-Information
Write-Output
Write-Progress
Write-Verbose
Write-Warning
Wait-Process
Write-EventLog
Write-DtcTransactionsTraceSession
Write-PrinterNfcTag
Write-VolumeCache

Category      Module
-----
Alias
Alias
Alias
Alias
Function
Cmdlet
Function
Function
Function

Synopsis
-----
Where-Object
Write-Output
Invoke-WebRequest
Wait-Job
...
Waits until one or all of the PowerShell jobs ...
Selects objects from a collection based on the...
Stops a script in the debugger before running ...
Waits until a particular event is raised befor...
Writes a debug message to the console.
Writes an object to the error stream.
Writes customized output to a host.
Specifies how PowerShell handles information s...
Writes the specified objects to the pipeline. ...
Displays a progress bar within a PowerShell co...
Writes text to the verbose message stream.
Writes a warning message.
Waits for the processes to be stopped before a...
Writes an event to an event log.
...
PrintManagement
Storage
  
```

Ou :

```

PS C:\Windows\system32> Get-Help w* -Detailed

Name
----
where
write
wget
wjb
w:
Wait-Job
Where-Object
Wait-Debugger
Wait-Event
Write-Debug
Write-Error
Write-Host
Write-Information
Write-Output
Write-Progress
Write-Verbose
Write-Warning
Wait-Process
Write-EventLog
Write-DtcTransactionsTraceSession
Write-PrinterNfcTag
Write-VolumeCache

Category      Module
-----
Alias
Alias
Alias
Alias
Function
Cmdlet
Function
Function
Function

Synopsis
-----
Where-Object
Write-Output
Invoke-WebRequest
Wait-Job
...
Waits until one or all of the PowerShell jobs ...
Selects objects from a collection based on the...
Stops a script in the debugger before running ...
Waits until a particular event is raised befor...
Writes a debug message to the console.
Writes an object to the error stream.
Writes customized output to a host.
Specifies how PowerShell handles information s...
Writes the specified objects to the pipeline. ...
Displays a progress bar within a PowerShell co...
Writes text to the verbose message stream.
Writes a warning message.
Waits for the processes to be stopped before a...
Writes an event to an event log.
...
PrintManagement
Storage
  
```

- 7) La commande pour créer la variable et l'initialiser à « Bienvenue à ce petit déjeuner » est :

Image ci-dessous : \$accueil = "Bienvenue à ce petit déjeuner"

```

PS C:\Windows\system32> $accueil = "Bienvenue à ce petit déjeuner"
  
```

- 8) Le contenu de la variable accueil (en taper la commande \$accueil) est : « Bienvenue à ce petit déjeuner »

Image ci-dessous :

```
PS C:\Windows\system32> $accueil
Bienvenue à ce petit déjeuner
PS C:\Windows\system32>
```

- 9) Pour afficher toutes les propriétés, méthodes et type d'un objet associé à la variable accueil est : \$accueil | Get-Member

Image ci-dessous :

```
PS C:\Windows\system32> $accueil | Get-Member

TypeName : System.String

Name      MemberType      Definition
-----
Clone     Method          System.Object Clone(), System.Object ICloneable.Clone()
CompareTo Method          int CompareTo(System.Object value), int CompareTo(string strB), int IComparab...
Contains  Method          bool Contains(string value)
CopyTo    Method          void CopyTo(int sourceIndex, char[] destination, int destinationIndex, int co...
EndsWith  Method          bool EndsWith(string value), bool EndsWith(string value, System.StringCompari...
Equals    Method          bool Equals(System.Object obj), bool Equals(string value), bool Equals(string...
Enumerate Method          System.CharEnumerator GetEnumerator(), System.Collections.IEnumerator IEnumer...
GetHashCode Method        int GetHashCode()
GetType   Method          type GetType()
GetTypeCode Method        System.TypeCode GetTypeCode(), System.TypeCode IConvertible.GetTypeCode()
IndexOf   Method          int IndexOf(char value), int IndexOf(char value, int startIndex), int IndexOf...
IndexOfAny Method        int IndexOfAny(char[] anyOf), int IndexOfAny(char[] anyOf, int startIndex), i...
Insert    Method          string Insert(int startIndex, string value)
IsNormalized Method        bool IsNormalized(), bool IsNormalized(System.Text.NormalizationForm normaliz...
LastIndexOf Method      int LastIndexOf(char value), int LastIndexOf(char value, int startIndex), int...
LastIndexOfAny Method    int LastIndexOfAny(char[] anyOf), int LastIndexOfAny(char[] anyOf, int startI...
Normalize Method          string Normalize(), string Normalize(System.Text.NormalizationForm normalizat...
PadLeft   Method          string PadLeft(int totalWidth), string PadLeft(int totalWidth, char paddingChar)
PadRight  Method          string PadRight(int totalWidth), string PadRight(int totalWidth, char padding...
Remove    Method          string Remove(int startIndex, int count), string Remove(int startIndex)
Replace   Method          string Replace(char oldChar, char newChar), string Replace(string oldValue, s...
Split     Method          string[] Split(Params char[] separator), string[] Split(char[] separator, int...
StartsWith Method        bool StartsWith(string value), bool StartsWith(string value, System.StringCom...
Substring Method        string Substring(int startIndex), string Substring(int startIndex, int length)
ToBoolean Method        bool IConvertible.ToBoolean(System.IFormatProvider provider)
ToByte    Method          byte IConvertible.ToByte(System.IFormatProvider provider)
ToChar    Method          char IConvertible.ToChar(System.IFormatProvider provider)
ToCharArray Method      char[] ToCharArray(), char[] ToCharArray(int startIndex, int length)
ToDateTime Method      dateTime IConvertible.ToDateTime(System.IFormatProvider provider)
ToDecimal Method        decimal IConvertible.ToDecimal(System.IFormatProvider provider)
ToDouble  Method          double IConvertible.ToDouble(System.IFormatProvider provider)
ToInt16   Method          int16 IConvertible.ToInt16(System.IFormatProvider provider)
ToInt32   Method          int IConvertible.ToInt32(System.IFormatProvider provider)
ToInt64   Method          long IConvertible.ToInt64(System.IFormatProvider provider)
ToLower   Method          string ToLower(), string ToLower(cultureinfo culture)
ToLowerInvariant Method    string ToLowerInvariant()

StartsWith Method        bool StartsWith(string value), bool StartsWith(string value, System.StringCom...
Substring  Method          string Substring(int startIndex), string Substring(int startIndex, int length)
ToBoolean  Method          bool IConvertible.ToBoolean(System.IFormatProvider provider)
ToByte     Method          byte IConvertible.ToByte(System.IFormatProvider provider)
ToChar     Method          char IConvertible.ToChar(System.IFormatProvider provider)
ToCharArray Method      char[] ToCharArray(), char[] ToCharArray(int startIndex, int length)
ToDateTime Method      dateTime IConvertible.ToDateTime(System.IFormatProvider provider)
ToDecimal  Method          decimal IConvertible.ToDecimal(System.IFormatProvider provider)
ToDouble   Method          double IConvertible.ToDouble(System.IFormatProvider provider)
ToInt16    Method          int16 IConvertible.ToInt16(System.IFormatProvider provider)
ToInt32    Method          int IConvertible.ToInt32(System.IFormatProvider provider)
ToInt64    Method          long IConvertible.ToInt64(System.IFormatProvider provider)
ToLower    Method          string ToLower(), string ToLower(cultureinfo culture)
ToLowerInvariant Method    string ToLowerInvariant()
ToByte     Method          sbyte IConvertible.ToSByte(System.IFormatProvider provider)
ToSingle   Method          float IConvertible.ToSingle(System.IFormatProvider provider)
ToString   Method          string ToString(), string ToString(System.IFormatProvider provider), string I...
ToType     Method          System.Object IConvertible.ToType(type conversionType, System.IFormatProvider...
ToInt16    Method          uint16 IConvertible.ToUInt16(System.IFormatProvider provider)
ToInt32    Method          uint32 IConvertible.ToUInt32(System.IFormatProvider provider)
ToInt64    Method          uint64 IConvertible.ToUInt64(System.IFormatProvider provider)
ToUpper    Method          string ToUpper(), string ToUpper(cultureinfo culture)
ToUpperInvariant Method    string ToUpperInvariant()
Trim       Method          string Trim(Params char[] trimChars), string Trim()
TrimEnd   Method          string TrimEnd(Params char[] trimChars)
TrimStart Method          string TrimStart(Params char[] trimChars)
Chars     ParameterizedProperty char Chars(int index) {get;}
Length    Property        int Length {get;}

PS C:\Windows\system32>
```

- 10) La longueur de la variable (c'est-à-dire la longueur de la chaîne de caractères) accueil est de 29 (lettre et espace qui va avec) et la commande est : \$accueil.length

```
PS C:\Windows\system32> $accueil.length
29
PS C:\Windows\system32>
```

- 11) La commande qui permet de mettre la variable accueil en lettres majuscules est : \$accueil.ToUpper()

```
PS C:\Windows\system32> $accueil.ToUpper()
BIENVENUE À CE PETIT DÉJEUNER
PS C:\Windows\system32>
```

- 12) Les alias permettent d'associer une commande PowerShell a une commande Dos

- 13) La commande pour trouver tous les alias existants est : get-alias

Image ci-dessous :

```

PS C:\Windows\system32> get-alias

CommandType      Name                                     Version      Source
-----
Alias             % -> ForEach-Object
Alias             ? -> Where-Object
Alias             ac -> Add-Content
Alias             asnp -> Add-PSSnapin
Alias             cat -> Get-Content
Alias             cd -> Set-Location
Alias             CFS -> ConvertFrom-String              3.1.0.0     Microsoft.PowerShell.Utility
Alias             chdir -> Set-Location
Alias             clc -> Clear-Content
Alias             clear -> Clear-Host
Alias             clhy -> Clear-History
Alias             cli -> Clear-Item
Alias             clp -> Clear-ItemProperty
Alias             cls -> Clear-Host
Alias             clv -> Clear-Variable
Alias             cnsn -> Connect-PSession
Alias             compare -> Compare-Object
Alias             copy -> Copy-Item
Alias             cp -> Copy-Item
Alias             cpi -> Copy-Item
Alias             cpp -> Copy-ItemProperty
Alias             curl -> Invoke-WebRequest
Alias             cvpa -> Convert-Path
Alias             dbp -> Disable-PSBreakpoint
Alias             del -> Remove-Item
Alias             diff -> Compare-Object
Alias             dir -> Get-ChildItem
Alias             dnsn -> Disconnect-PSession
Alias             ebp -> Enable-PSBreakpoint
Alias             echo -> Write-Output
Alias             epal -> Export-Alias
Alias             epcsv -> Export-Csv
Alias             epsn -> Export-PSession
Alias             erase -> Remove-Item
Alias             etsn -> Enter-PSession
Alias             exsn -> Exit-PSession
Alias             fc -> Format-Custom
Alias             fhx -> Format-Hex                        3.1.0.0     Microsoft.PowerShell.Utility
Alias             fl -> Format-List
Alias             foreach -> ForEach-Object

```

Etc.

- 14) L'alias correspondant à rmdir est la commande « rd », c'est une alternative à la commande « rmdir » pour supprimer des répertoires dans PowerShell.  
Voici un exemple : rd c:\temp\test

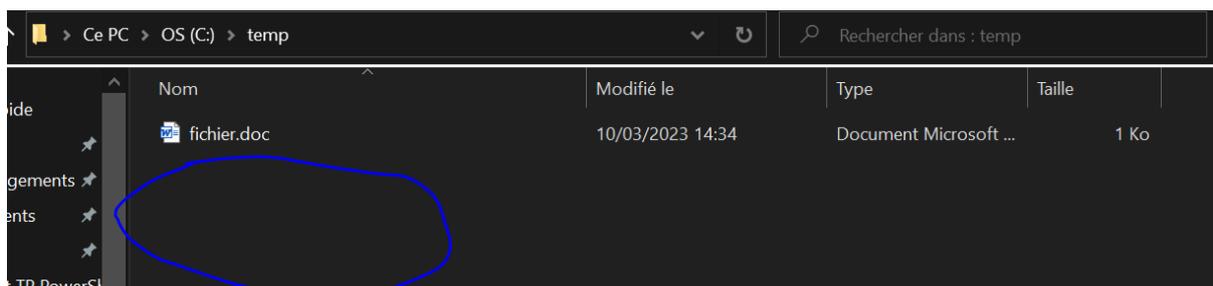
```

Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\cleme> rd c:\temp\test
PS C:\Users\cleme>

```



15) La commande pour afficher tous les fichiers et dossiers se trouvant dans le répertoire c:\Windows est : get-childitem c:\Windows

Image ci-dessous :

```
PS C:\Windows\system32> get-childitem c:\Windows

Répertoire : C:\Windows

Mode                LastWriteTime         Length Name
----                -
d-----          07/12/2019      10:50         addins
d-----          21/10/2021      12:14         appcompat
d-----          03/03/2023      15:51         apppatch
d-----          07/03/2023      21:06         AppReadiness
d-r---          01/11/2022      12:16         assembly
d-----          16/09/2021      11:25         ASUS
d-----          03/03/2023      15:51         bcastdvr
d-----          07/12/2019      10:31         Boot
d-----          07/12/2019      10:14         Branding
d-----          05/03/2023      20:45         CbsTemp
d-----          03/03/2023      13:58         Containers
d-----          07/12/2019      10:14         Cursors
d-----          18/10/2021      20:17         debug
d-----          03/03/2023      13:48         diagnostics
d-----          03/03/2023      13:58         DiagTrack
d-----          07/12/2019      10:49         DigitalLocker
d--s-          07/12/2019      10:14         Downloaded Program Files
d-----          17/10/2021      22:45         en-US
d-r-s-          27/11/2022      22:22         Fonts
d-----          29/12/2022      23:01         fr-FR
d-----          07/12/2019      10:14         GameBarPresenceWriter
d-----          07/12/2019      10:31         Globalization
d-----          07/12/2019      10:49         Help
d-----          07/12/2019      10:31         IdentityCRL
d-----          03/03/2023      13:58         IME
d-r---          03/03/2023      15:51         ImmersiveControlPanel
d-----          09/03/2023      15:21         INF
d-----          07/12/2019      10:14         InputMethod
d-----          07/12/2019      10:14         L2Schemas
d-----          20/10/2022      19:42         LiveKernelReports
d-----          17/10/2021      18:38         Log
d-----          03/03/2023      13:40         Logs
d-r-s-          07/12/2019      10:31         Media
d-r---          07/03/2023      21:07         Microsoft.NET
d-----          07/12/2019      10:14         Migration
d-----          07/12/2019      10:14         ModemLogs
```

d-----	07/12/2019	10:14	ModemLogs
d-----	17/10/2021	22:45	OCR
d-----	16/09/2021	11:25	OEM
d-r---	07/12/2019	10:14	Offline Web Pages
d-----	11/01/2022	15:48	Panther
d-----	07/12/2019	10:14	Performance
d-----	07/12/2019	10:31	PLA
d-----	14/10/2022	18:31	PolicyDefinitions
d-----	10/03/2023	09:03	Prefetch
d-r---	03/03/2023	13:58	PrintDialog
d-----	03/03/2023	13:58	Provisioning
d-----	03/03/2023	13:49	registration
d-----	07/12/2019	10:14	rescache
d-----	07/12/2019	10:31	Resources
d-----	07/12/2019	10:14	SchCache
d-----	07/12/2019	10:31	schemas
d-----	07/12/2019	10:31	security
d-----	21/11/2020	13:38	ServiceProfiles
d-----	06/03/2023	09:18	ServiceState
d-----	03/03/2023	13:58	servicing
d-----	07/12/2019	10:18	Setup
d-----	03/03/2023	13:58	ShellComponents
d-----	03/03/2023	13:58	ShellExperiences
d-----	07/12/2019	10:31	SKB
d-----	18/10/2021	09:27	SoftwareDistribution
d-----	07/12/2019	10:14	Speech
d-----	07/12/2019	10:14	Speech_OneCore
d-----	07/12/2019	10:14	System
da----	06/03/2023	14:18	System32
d-----	29/12/2022	23:01	SystemApps
d-----	03/03/2023	15:51	SystemResources
d-----	17/12/2021	22:08	SystemTemp
d-----	03/03/2023	15:51	SysWOW64
d-----	07/12/2019	10:14	TAPI
d-----	03/03/2023	13:58	Tasks
d-----	10/03/2023	10:42	Temp
d-----	07/12/2019	10:14	tracing
d-----	20/04/2020	22:57	twain_32
d-----	07/12/2019	10:14	Vss
d-----	07/12/2019	10:14	WaaS
d-----	07/12/2019	10:31	Web
d-----	05/03/2023	21:57	WinSxS
-a----	08/07/2021	11:55	24 AsDCDVer.txt
-a----	28/04/2020	07:46	68 AsEDNVer.txt

```

-a---- 16/09/2021 11:42 98 AsPToolVer.txt
-a---- 12/07/2021 03:05 64 AsProcKitVer.txt
-a---- 12/07/2021 03:09 49 AsToolCDVer.txt
-a---- 17/10/2021 22:59 81408 bfvsc.exe
-a-s- 10/03/2023 11:05 67584 bootstat.dat
-a---- 16/09/2021 11:27 1278 comsetup.log
-a---- 07/12/2019 10:10 29857 Core.xml
-a---- 07/12/2019 10:10 29857 CoreSingleLanguage.xml
-a---- 21/11/2020 13:42 10 csup.txt
-a---- 16/09/2021 11:32 22863 diagerr.xml
-a---- 16/09/2021 11:32 22863 diagwrn.xml
-a---- 16/09/2021 11:33 4291 DtcInstall.log
-a---- 03/03/2023 15:49 5253864 explorer.exe
-a---- 14/10/2022 14:08 1075712 HelpPane.exe
-a---- 07/12/2019 10:09 18432 hh.exe
-a---- 16/09/2021 11:23 3521522 Inst.log
-a---- 16/09/2021 11:23 696 Inst_AsModelCopy.log
-a---- 16/09/2021 11:23 25868 Inst_CMD.log
-a---- 16/09/2021 11:17 2499214 Inst_Device.log
-a---- 21/11/2020 13:38 1380 lsasetup.log
-a---- 07/12/2019 10:08 43131 mib.bin
-a---- 12/09/2022 16:17 201216 notepad.exe
-a---- 05/03/2023 21:56 5429282 PFR0.log
-a---- 07/11/2021 17:44 5780 PidVid_list
-a---- 04/10/2021 19:12 720112 py.exe
-a---- 04/10/2021 19:12 45296 pyshellext.amd64.dll
-a---- 04/10/2021 19:12 720104 pyw.exe
-a---- 17/10/2021 23:00 370176 regedit.exe
-a---- 07/11/2021 17:54 765936 RtkBtManServ.exe
-a---- 07/11/2021 17:44 63752 rtl8723d_mp_chip_bt40_fw_asic_rom_patch_new
-a---- 07/11/2021 17:44 55460 rtl8821c_mp_chip_bt40_fw_asic_rom_patch_new
-a---- 07/11/2021 17:44 50584 rtl8822b_mp_chip_bt40_fw_asic_rom_patch_new
-a---- 07/11/2021 17:44 68264 rtl8822c_mp_chip_bt40_fw_asic_rom_patch_new
-a---- 05/03/2023 21:56 9362 setupact.log
-a---- 16/09/2021 11:16 0 setuperr.log
-a---- 03/03/2023 15:49 163840 splwow64.exe
-a---- 07/12/2019 10:12 219 system.ini
-a---- 07/12/2019 10:10 65024 twain_32.dll
-a---- 07/12/2019 10:12 92 win.ini
-a---- 07/12/2019 10:09 670 WindowsShell.Manifest
-a---- 10/03/2023 10:41 276 WindowsUpdate.log
-a---- 07/12/2019 10:10 11776 winhlp32.exe
-a---- 07/12/2019 10:52 316640 WMSysPr9.prx
-a---- 06/12/2019 22:29 11264 write.exe

```

Etc.

16) Les modes des fichiers suivants sont :

- Pour le fichier Fonts : On voit que c'est un répertoire (lettre « d » qui apparait), que c'est un objet en lecture seule (lettre « r » qui apparait) et que c'est un objet système (lettre « s » qui apparait).
- Pour le fichier WinSxS : On voit que c'est juste un répertoire (lettre « d » qui apparait).
- Pour le fichier bootstat.dat : On voit la lettre « a » qui est un archive et la lettre « s » qui est un objet système.

17) Pour afficher les fichiers ou les répertoires cachés dans Windows ils existent 2 commandes : Get-ChildItem -Hidden **Ou alors** Get-ChildItem -Force

On utilise aussi le chemin : C:\Windows\

Voici pour Get-ChildItem -Force :

```
Administrateur : Windows PowerShell

PS C:\Windows\system32> Get-ChildItem -Force C:\Windows\

Répertoire : C:\Windows

Mode                LastWriteTime         Length Name
----                -
d-----            07/12/2019      10:50      addins
d-----            21/10/2021      12:14      appcompat
d-----            03/03/2023      15:51      apppatch
d-----            10/03/2023      12:07      AppReadiness
d-r---            01/11/2022      12:16      assembly
d-----            16/09/2021      11:25      ASUS
d-----            03/03/2023      15:51      bcastdvr
d-----            07/12/2019      10:31      Boot
d-----            07/12/2019      10:14      Branding
d-----            05/03/2023      20:45      CbsTemp
d-----            03/03/2023      13:58      Containers
d-----            07/12/2019      10:14      Cursors
d-----            18/10/2021      20:17      debug
d-----            03/03/2023      13:48      diagnostics
d-----            03/03/2023      13:58      DiagTrack
d-----            07/12/2019      10:49      DigitalLocker
d---s-            07/12/2019      10:14      Downloaded Program Files
d--h--            03/03/2023      13:58      ELAMBKUP
d-----            17/10/2021      22:45      en-US
d-r-s-            27/11/2022      22:22      Fonts
d-----            29/12/2022      23:01      fr-FR
d-----            07/12/2019      10:14      GameBarPresenceWriter
d-----            07/12/2019      10:31      Globalization
d-----            07/12/2019      10:49      Help
d-----            07/12/2019      10:31      IdentityCRL
d-----            03/03/2023      13:58      IME
d-r---            03/03/2023      15:51      ImmersiveControlPanel
d-----            09/03/2023      15:21      INF
d-----            07/12/2019      10:14      InputMethod
d--hs-            04/03/2023      16:46      Installer
d-----            07/12/2019      10:14      L2Schemas
d--h--            07/12/2019      10:14      LanguageOverlayCache
d-----            20/10/2022      19:42      LiveKernelReports
d-----            17/10/2021      22:45      ...
```

Et pour Get-ChildItem -Hidden :

```
PS C:\Windows\system32> Get-ChildItem -Hidden C:\Windows\

Répertoire : C:\Windows

Mode                LastWriteTime         Length Name
----                -
d--h--            03/03/2023      13:58      ELAMBKUP
d--hs-            04/03/2023      16:46      Installer
d--h--            07/12/2019      10:14      LanguageOverlayCache
```

18) Pour afficher les fichiers cachés mais pas les répertoires cachés dans Windows la commande est : `get-childitem c:\Windows\ -attributes hidden+directory`

Image ci-dessous :

```
PS C:\Windows\system32> get-childitem c:\Windows -attributes hidden+directory

Répertoire : C:\Windows

Mode                LastWriteTime         Length Name
----                -
d--h--            03/03/2023    13:58          ELAMBKUP
d--hs-            04/03/2023    16:46          Installer
d--h--            07/12/2019    10:14    LanguageOverlayCache
```

On peut faire aussi la commande : `get-childitem c:\ -attributes hidden+directory` (pour voir le disque) :

```
PS C:\Windows\system32> get-childitem c:\ -attributes hidden+directory

Répertoire : C:\

Mode                LastWriteTime         Length Name
----                -
d--hs-            17/10/2021    22:50          $Recycle.Bin
d--h--            11/01/2022    15:40          $Windows.~WS
d--h--            03/03/2023    15:37          $WinREAgent
d--hsl            18/10/2021    09:27    Documents and Settings
d--h--            06/03/2023    09:18          Intel
d--h--            17/10/2021    18:40          OneDriveTemp
d--h--            07/12/2019    10:14          PerfLogs
d--h--            03/03/2023    13:48          ProgramData
d--h--            16/09/2021    12:34          Recovery
d--hs-            09/03/2023    15:21    System Volume Information

PS C:\Windows\system32>
```

19) La commande qui permet d'afficher tous les fichiers de la partition C dont la taille est inférieure à 1GB est : `Get-ChildItem -Path "C:\" -File -Recurse | Where-Object Length -lt 1GB`

Image ci-dessous :

```
Administrateur : Windows PowerShell
PS C:\Windows\system32> Get-Childitem -Path "C:\\" -File -Recurse | Where-Object Length -lt 1GB
>>

Répertoire : C:\$WINDOWS.~BT\Sources\Panther

Mode                LastWriteTime         Length Name
-----
-a----          11/01/2022   15:48           1908 diagerr.xml
-a----          11/01/2022   15:48           5385 diagwrn.xml
-a----          11/01/2022   15:48           7751 setupact.log
-a----          11/01/2022   15:48              0 setuperr.log

Répertoire : C:\eSupport\eDriver

Mode                LastWriteTime         Length Name
-----
-a----          06/07/2020   09:39      4069368 AsusInswiz.exe
-a----          12/04/2012   05:35      252591  AsusInswiz.ico
-a----          30/03/2020   08:44           57  AUTORUN.INF
-a----          08/07/2021   11:55      135254  FileList.txt
-a----          08/07/2021   11:55      185344  ReleaseNote_FileList of X415JAB_2009_X64_V2.00.xls
-a----          08/07/2021   11:59      365919  X415JAB_2009_X64_V2.00.MD5

Répertoire : C:\eSupport\eDriver\LIB

Mode                LastWriteTime         Length Name
-----
-a----          18/11/2013   09:48           292  AsSetting.ini
-a----          12/05/2021   04:11      297624  AsusDriverCD.dll
-a----          06/04/2020   12:16      617464  AsusInswizMonitor.exe
-a----          08/07/2021   11:55      540946  InstallAll.ini

Répertoire : C:\eSupport\eDriver\LIB\HighImage

Mode                LastWriteTime         Length Name
-----
```

- 20) La commande qui permet d'afficher les fichiers de la partition C dont la date de dernier enregistrement est postérieure au 31/12/2018 est : `Get-ChildItem -Path "C:\" -Recurse | Where-Object { $_.LastWriteTime -gt "12/31/2018" }`

Image ci-dessous :

```
PS C:\Windows\system32> Get-ChildItem -Path "C:\" -Recurse | Where-Object { $_.LastWriteTime -gt "12/31/2018" }

Répertoire : C:\

Mode                LastWriteTime         Length Name
-----
d-----         11/01/2022         15:40      $WINDOWS.~BT
d-----         11/01/2022         15:48      ESD
d-----         16/09/2021         12:34      eSupport
d-----         31/12/2022         19:55      InitiationDocker
d-r-----        03/03/2023         22:22      Program Files
d-r-----        29/10/2022         21:29      Program Files (x86)
d-----         14/12/2021         15:47      sauve
d-r-----        17/10/2021         18:56      Users
d-----         14/12/2021         15:11      VBS
d-----         18/11/2022         14:31      VM
d-----         03/03/2023         15:51      windows
d-----         03/03/2023         13:58      xampp

Répertoire : C:\$WINDOWS.~BT

Mode                LastWriteTime         Length Name
-----
d-----         11/01/2022         15:40      Sources

Répertoire : C:\$WINDOWS.~BT\Sources
```

- 21) La commande qui permet d'afficher l'emplacement actuel est : `Get-Location`

Image ci-dessous :

```
PS C:\Windows\system32> Get-Location

Path
----
C:\Windows\system32

PS C:\Windows\system32>
```

- 22) La commande qui permet à partir de notre emplacement actuel, qu'on se déplace vers la partition C de notre environnement de travail à savoir utilisateurs à nom du compte de connexion

Image ci-dessous :

```
PS C:\> set-Location C:\Users\cleme
PS C:\Users\cleme>
```

23) La commande pour savoir notre emplacement actuel est : `get-location`

Image ci-dessous :

```
PS C:\Users\cleme> get-location

Path
----
C:\Users\cleme

PS C:\Users\cleme>
```

**Travail à effectuer : 2ème partie gestion des répertoires et fichiers :**

1) La commande pour créer le répertoire temp sur la partition C est : `new-item -path c:\ -itemtype directory -name temp`

Image ci-dessous :

```
PS C:\> new-item -path c:\ -itemtype directory -name temp

Répertoire : C:\

Mode                LastWriteTime         Length Name
----                -
d-----          10/03/2023   14:30             temp
```

2) La commande pour créer le fichier « monfichier » et initialiser (avec l'initialisation suivante : « Mon fichier contient du texte qui sera exploité par la suite ») est :

`new-item -path c:\temp -itemtype file -name monfichier -value 'Mon fichier contient du texte qui sera exploité par la suite'`

Image ci-dessous :

```
Windows PowerShell
PS C:\temp> Set-Location ..
PS C:\> new-item -path c:\temp -itemtype file -name monfichier -value 'Mon fichier contient du texte qui sera exploité par la suite'

Répertoire : C:\temp

Mode                LastWriteTime         Length Name
----                -
-a-----          10/03/2023   14:34             61 monfichier

PS C:\>
```

- 3) La commande pour afficher le contenu du fichier est : `get-content` (c'est donc : `get-content monfichier`)

Image ci-dessous :

```
PS C:\> Set-Location c:\temp
PS C:\temp> get-content monfichier
Mon fichier contient du texte qui sera exploitÃ© par la suite
PS C:\temp>
```

- 4) La commande pour renommer `monfichier` en `fichier.doc` est : `rename-item` (c'est donc : `rename-item monfichier fichier.doc`)

Image ci-dessous :

```
PS C:\temp> rename-item monfichier fichier.doc
PS C:\temp>
```

- 5) Pour vérifier que le fichier à bien été renommé est : `get-content` (donc c'est : `get-content fichier.doc`)

Image ci-dessous :

```
PS C:\temp> rename-item monfichier fichier.doc
PS C:\temp>
```

- 6) La commande pour créer les répertoires `test` et `test1` sur la partition C dans `temps` est :

`new-item -path c:\temp -itemtype directory -name test`

Et `new-item -path c:\temp -itemtype directory -name test1`

Image ci-dessous :

```
PS C:\> new-item -path c:\temp -itemtype directory -name test

Répertoire : C:\temp

Mode                LastWriteTime         Length Name
----                -
d-----            10/03/2023   14:56             test

PS C:\> new-item -path c:\temp -itemtype directory -name test1

Répertoire : C:\temp

Mode                LastWriteTime         Length Name
----                -
d-----            10/03/2023   14:56             test1

PS C:\>
```

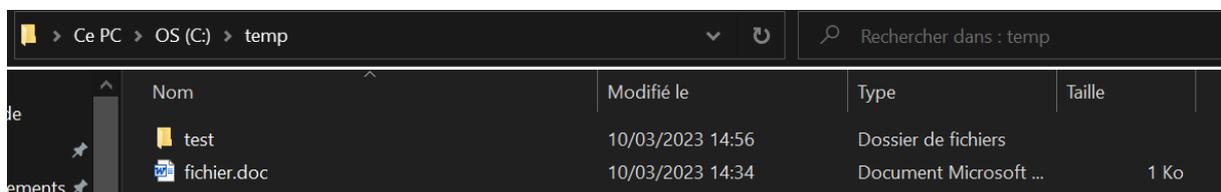
7) La commande pour renommer le répertoire test1 par jetest est : `rename-item C:\temp\test1\ -newname jetest`

Et la commande pour positionner sur la partition C est : `move-item -path C:\temp\jetest\ -destination C:\`

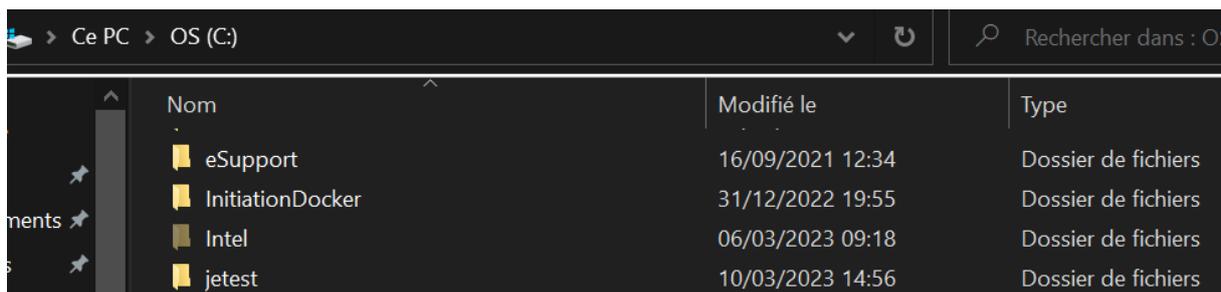
Image ci-dessous :

```
Windows PowerShell

PS C:\> rename-item C:\temp\test1\ -newname jetest
PS C:\> move-item -path C:\temp\jetest\ -destination C:\
PS C:\> get-content
```



Nom	Modifié le	Type	Taille
test	10/03/2023 14:56	Dossier de fichiers	
fichier.doc	10/03/2023 14:34	Document Microsoft ...	1 Ko



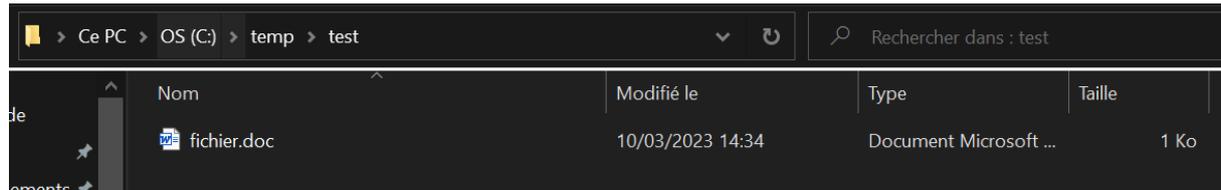
Nom	Modifié le	Type
eSupport	16/09/2021 12:34	Dossier de fichiers
InitiationDocker	31/12/2022 19:55	Dossier de fichiers
Intel	06/03/2023 09:18	Dossier de fichiers
jetest	10/03/2023 14:56	Dossier de fichiers

- 8) La commande pour copier fichier.doc (et non monfichier.doc dans l'énoncer) dans le répertoire test est : `copy-item -path C:\temp\fichier.doc -destination C:\temp\test\`

Image ci-dessous :

```
PS C:\> copy-item -path C:\temp\fichier.doc -destination C:\temp\test\  
PS C:\>
```

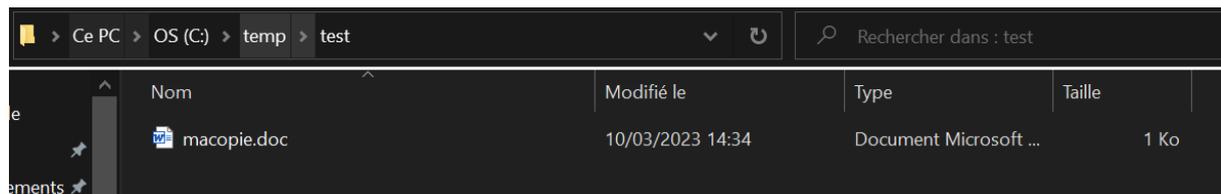
Cela à copier-coller le fichier.doc qui est dans le répertoire temp pour aussi le mettre dans le répertoire test (image ci-dessous) :



La commande pour renommer fichier.doc en macopie.doc est : `rename-item` (donc c'est : `rename-item C:\temp\test\fichier.doc macopie.doc`)

Image ci-dessous :

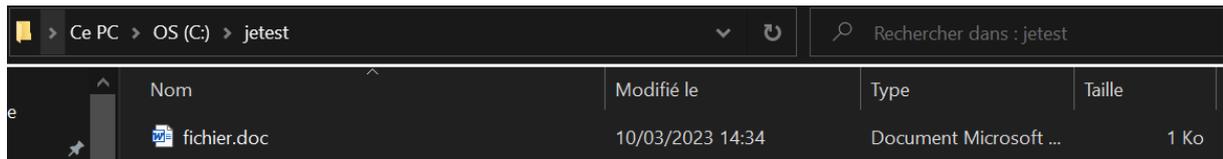
```
Windows PowerShell  
Windows PowerShell  
Copyright (C) Microsoft Corporation. Tous droits réservés.  
Testez le nouveau système multiplateforme PowerShell https://  
PS C:\Users\cleme> Set-Location ..  
PS C:\Users> Set-Location ..  
PS C:\> rename-item C:\temp\test\fichier.doc macopie.doc  
PS C:\>
```



- 9) La commande pour copier fichier.doc (et non monfichier.doc écrit dans l'énoncé) dans le répertoire jetest est : `copy-item -path C:\temp\fichier.doc -destination C:\jetest\`

Image ci-dessous :

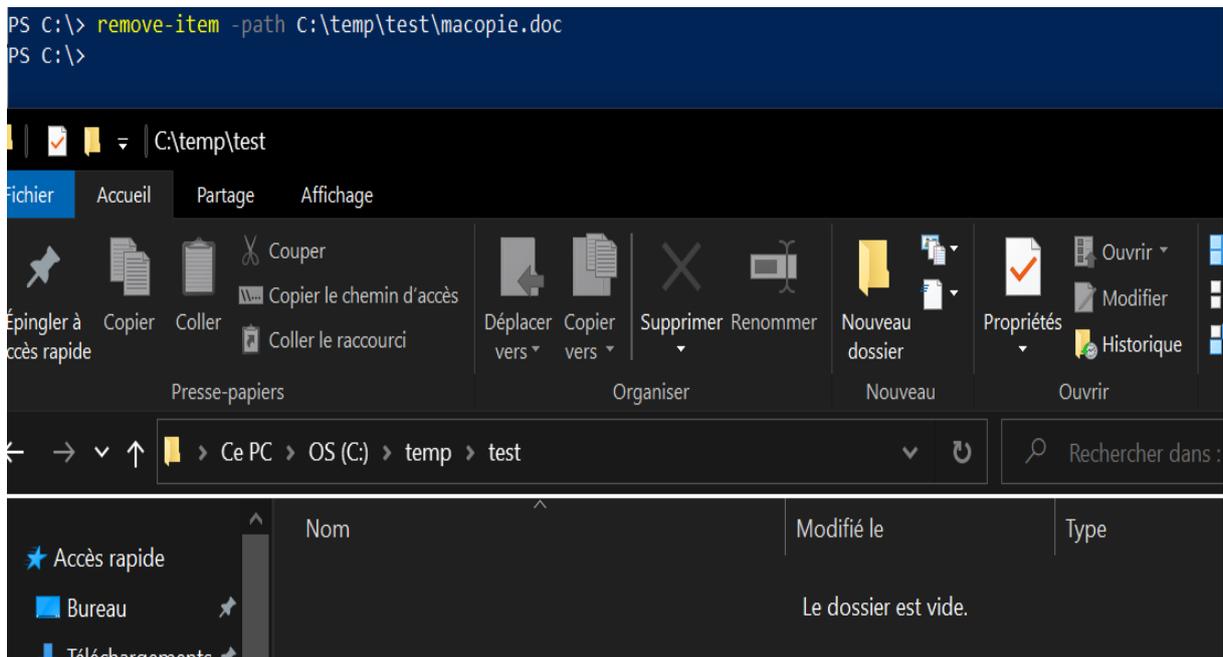
```
PS C:\> copy-item -path C:\temp\fichier.doc -destination C:\jetest\  
PS C:\>
```



La commande pour supprimer le fichier fichier.doc (qui est en vrai **macopie.doc** car on l'a renommé dans la question 8) du répertoire test est :

```
remove-item -path C:\temp\test\macopie.doc
```

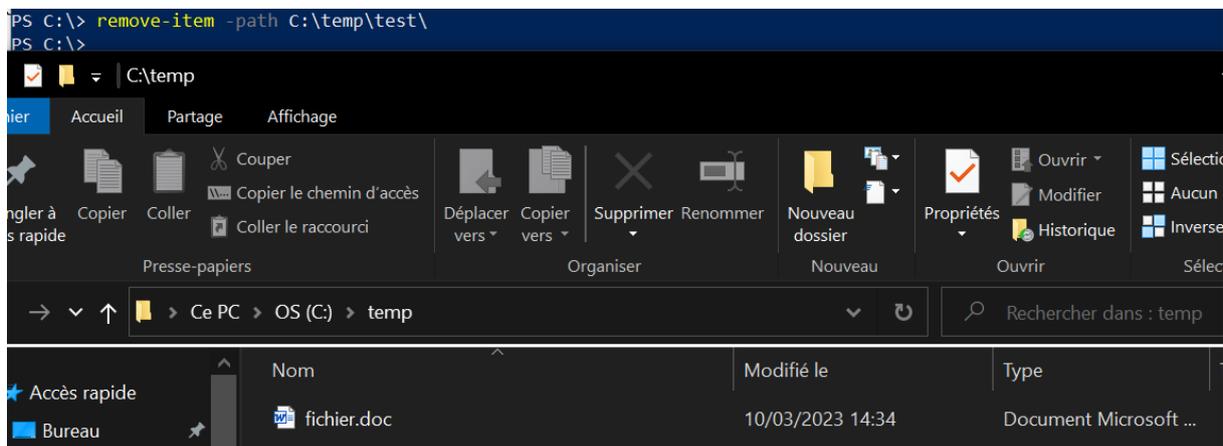
Image ci-dessous :



La commande qui permet de supprimer le répertoire test est :

```
remove-item -path C:\temp\test\
```

Image ci-dessous :



10) Les commandes pour rechercher les fichiers et répertoires commençant par T et par J sont :

Get-ChildItem -Path C:\ -Filter "T\*"

Image ci-dessous :

```
PS C:\> Get-ChildItem -Path C:\ -Filter "T*"

Répertoire : C:\

Mode                LastWriteTime         Length Name
----                -
d-----          10/03/2023   16:10             temp

PS C:\>
```

Et l'autre commande est :

Get-ChildItem -Path C:\ -Filter "J\*"

Image ci-dessous :

```
PS C:\> Get-ChildItem -Path C:\ -Filter "J*"

Répertoire : C:\

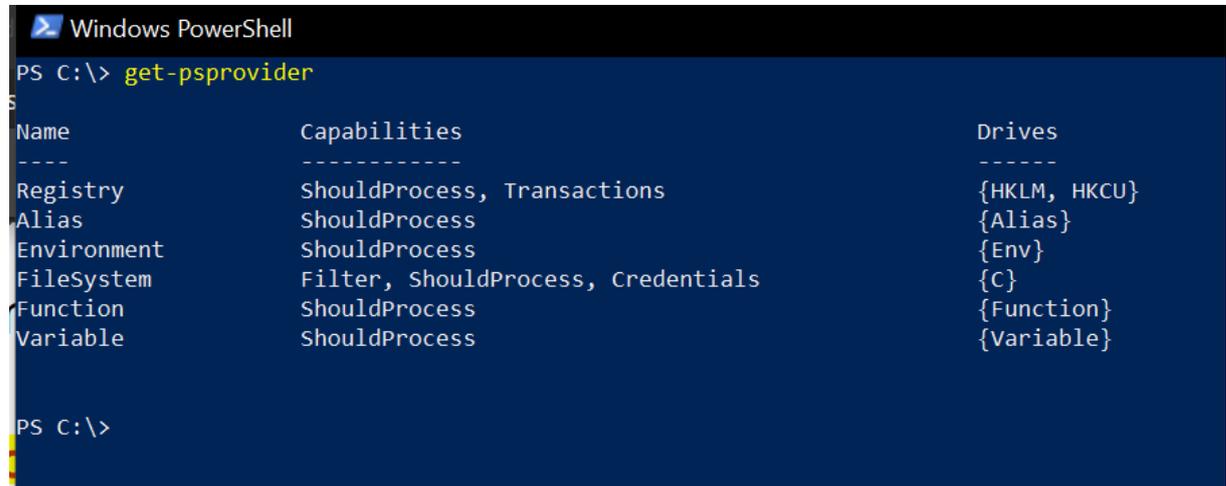
Mode                LastWriteTime         Length Name
----                -
d-----          10/03/2023   16:06             jetest

PS C:\>
```

## Travail à effectuer : 3<sup>ème</sup> partie Le provider :

1) La commande pour afficher les providers est : `get-psprovider`

Les providers disponibles sont :



```
Windows PowerShell
PS C:\> get-psprovider

Name                Capabilities                Drives
----                -
Registry            ShouldProcess, Transactions {HKLM, HKCU}
Alias               ShouldProcess              {Alias}
Environment         ShouldProcess              {Env}
FileSystem          Filter, ShouldProcess, Credentials {C}
Function            ShouldProcess              {Function}
Variable            ShouldProcess              {Variable}

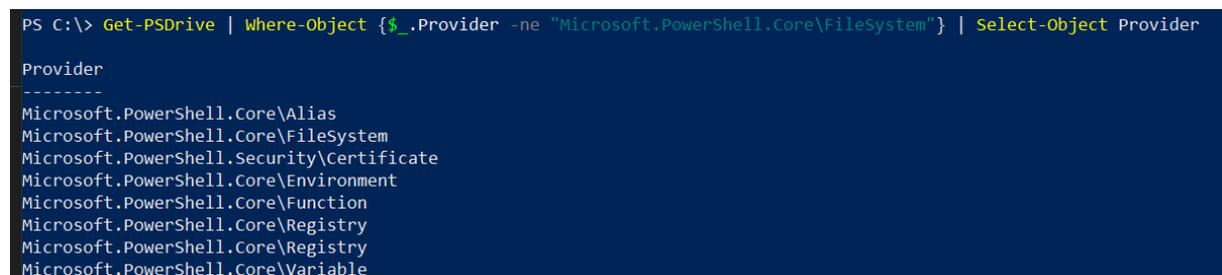
PS C:\>
```

2) On voit qu'il y a 6 noms (avec en anglais capabilities et drives) car ils sont affichés par défaut mais en réalité il y en a 8.

3) La commande pour trouver le nom des 2 providers manquants est :

```
Get-PSDrive | Where-Object {$_.Provider -ne
"Microsoft.PowerShell.Core\FileSystem"} | Select-Object Provider
```

Image ci-dessous :



```
PS C:\> Get-PSDrive | Where-Object {$_.Provider -ne "Microsoft.PowerShell.Core\FileSystem"} | Select-Object Provider

Provider
-----
Microsoft.PowerShell.Core\Alias
Microsoft.PowerShell.Core\FileSystem
Microsoft.PowerShell.Security\Certificate
Microsoft.PowerShell.Core\Environment
Microsoft.PowerShell.Core\Function
Microsoft.PowerShell.Core\Registry
Microsoft.PowerShell.Core\Registry
Microsoft.PowerShell.Core\Variable
```

On voit que les 2 providers manquants sont Registry (il y a donc 2 Registry au total) et Certificate.