BTS 1 - Services Informatiques aux Organisations

Scripts : Introduction à PowerShell – Module Active Directory

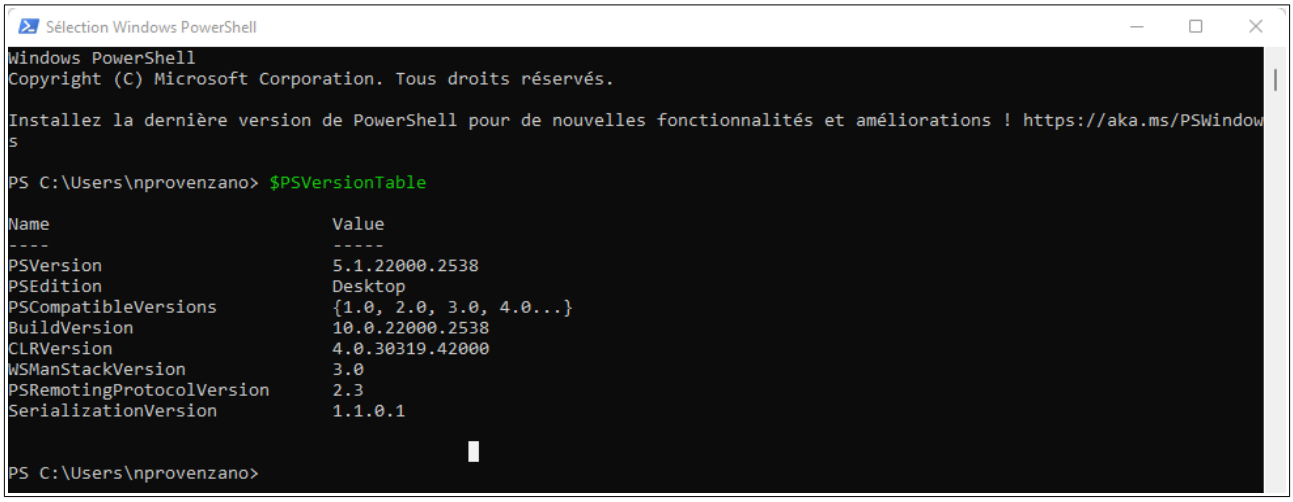
Table des matières

1. Prise en main

1.1 Découverte de la console ligne de commandes

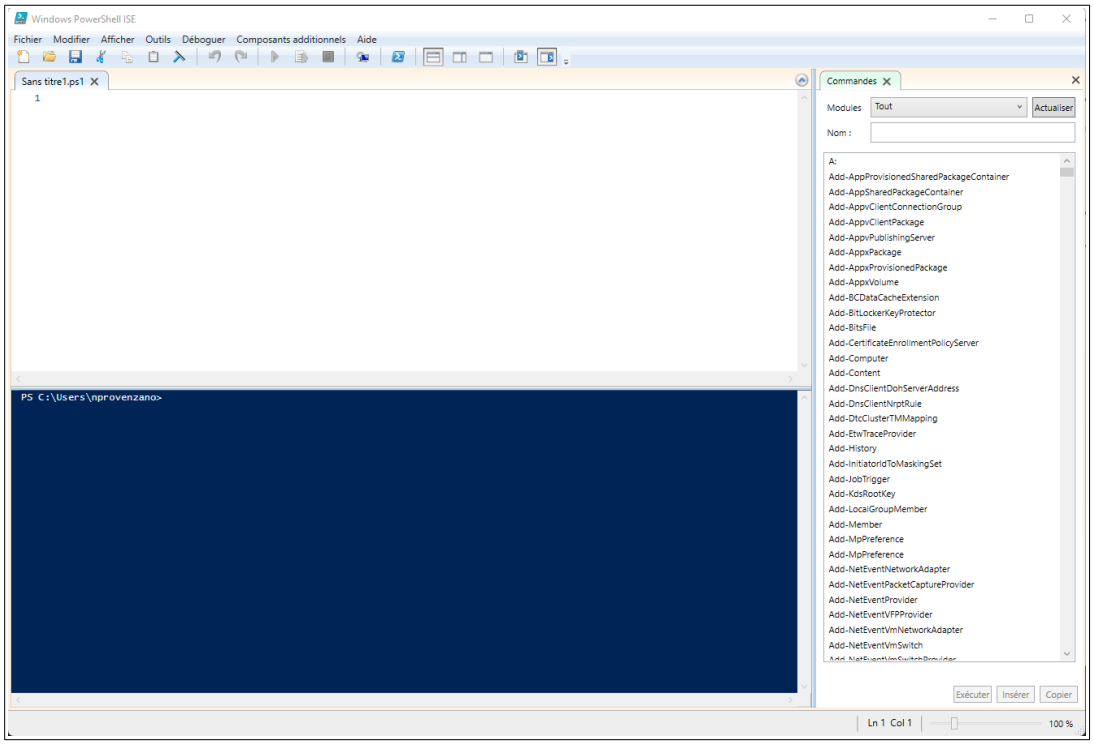
Touche [F7] qui permet en un coup d'œil de retrouver une commande dans l'historique.

→ Effectuer la commande **$PSVersionTable** dans la console pour savoir quelle version est installée sur votre poste de travail ou serveur.



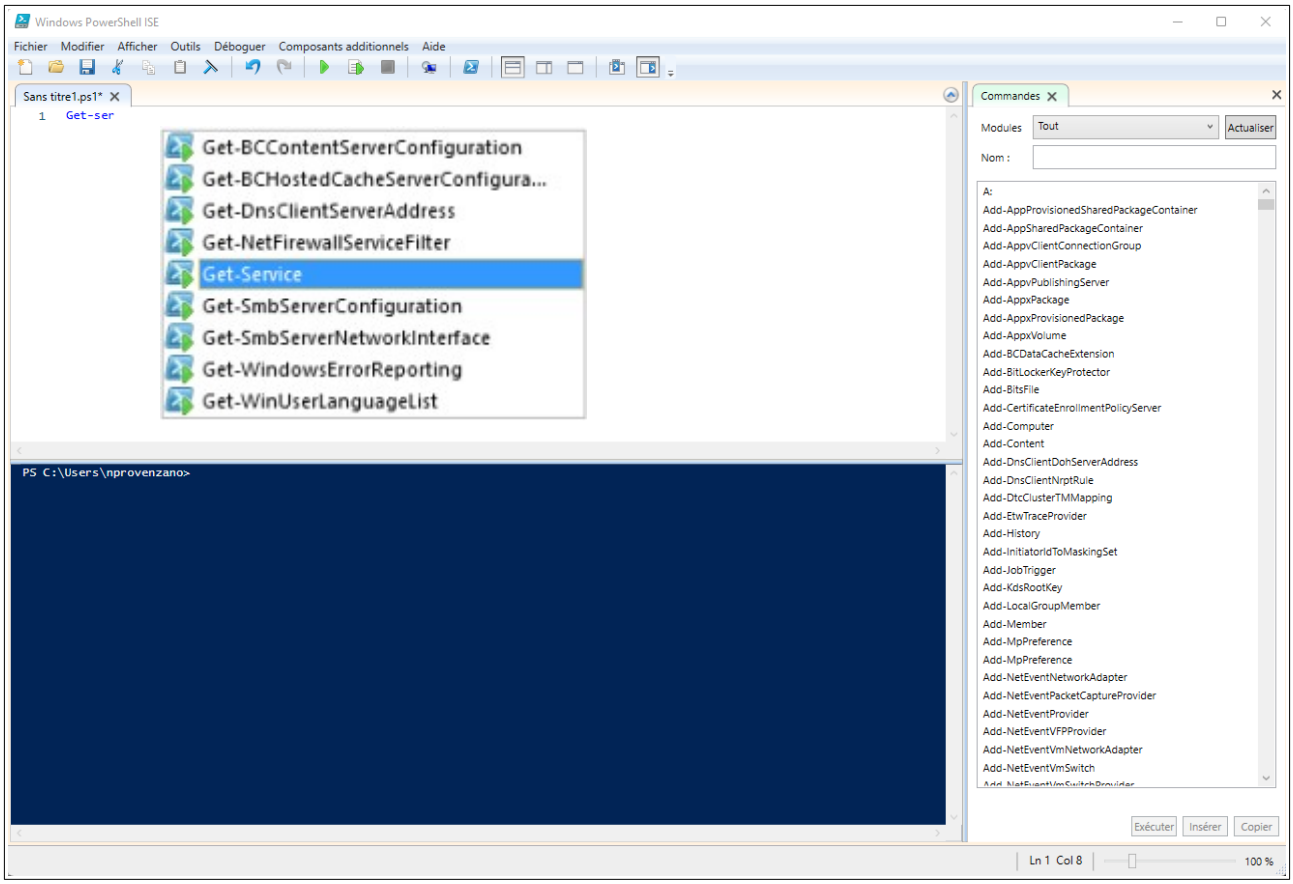
1.2 L'environnement d'écriture de scripts intégré (interface ISE)

Un éditeur de scripts est disponible depuis PowerShell v2 avec la coloration syntaxique, l'affichage des numéros de lignes, le débogueur intégré, l'aide en ligne en mode graphique, etc. On a également la possibilité d'ouvrir une console PowerShell sur une machine distance directement dans l'éditeur.



Dans le volet supérieur gauche se trouve l'éditeur de script dans lequel vont se loger des onglets. Cela permet de travailler sur plusieurs scripts à la fois. Celui du dessous quant à lui permet de saisir directement des commandes interactives, comme dans la console bleue originelle.

Fonctionnalité « **IntelliSense** »

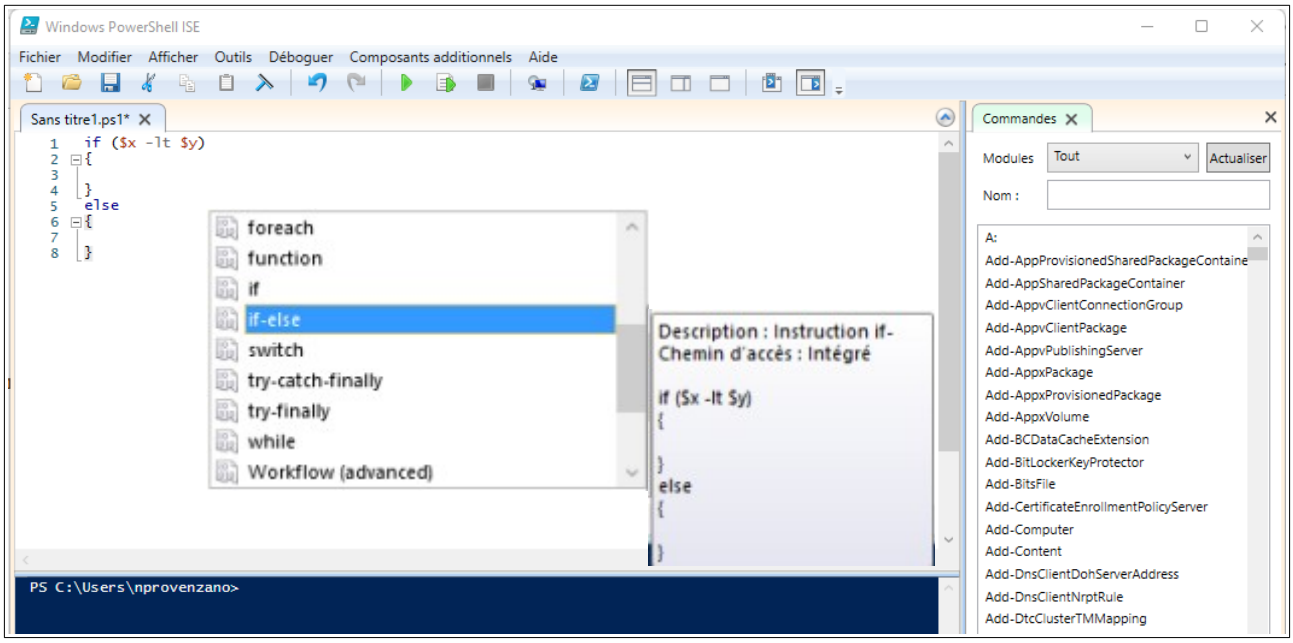


La fonctionnalité IntelliSense permet de gérer un ensemble de propositions qui sont déclenchées lorsque l'on utilise notamment les caractères suivants :

- Tiret - derrière le verbe d'une commande comme Get- ou avant de saisir le nom d'un paramètre de commande, comme dans Get-Command ;

- Point . après un nom de variable, exemple $profile.

Fonctionnalité « **Snippers** »



Les Snippers permettent de coller du code ou des structures de code PowerShell directement dans l'éditeur. Pour les utiliser, il faut saisir la combinaison **[Ctrl] + J** et une liste déroulante apparaît pour choisir un élément de code.

1.3 Transition avec le passé

Toutes les commandes qui étaient incluses dans CMD le sont aussi dans PowerShell. Certaines le sont sous forme d'alias ou de fonctions.

La commande **dir** permet de lister le contenu d'un répertoire. Ici elle est exécutée dans le PowerShell.

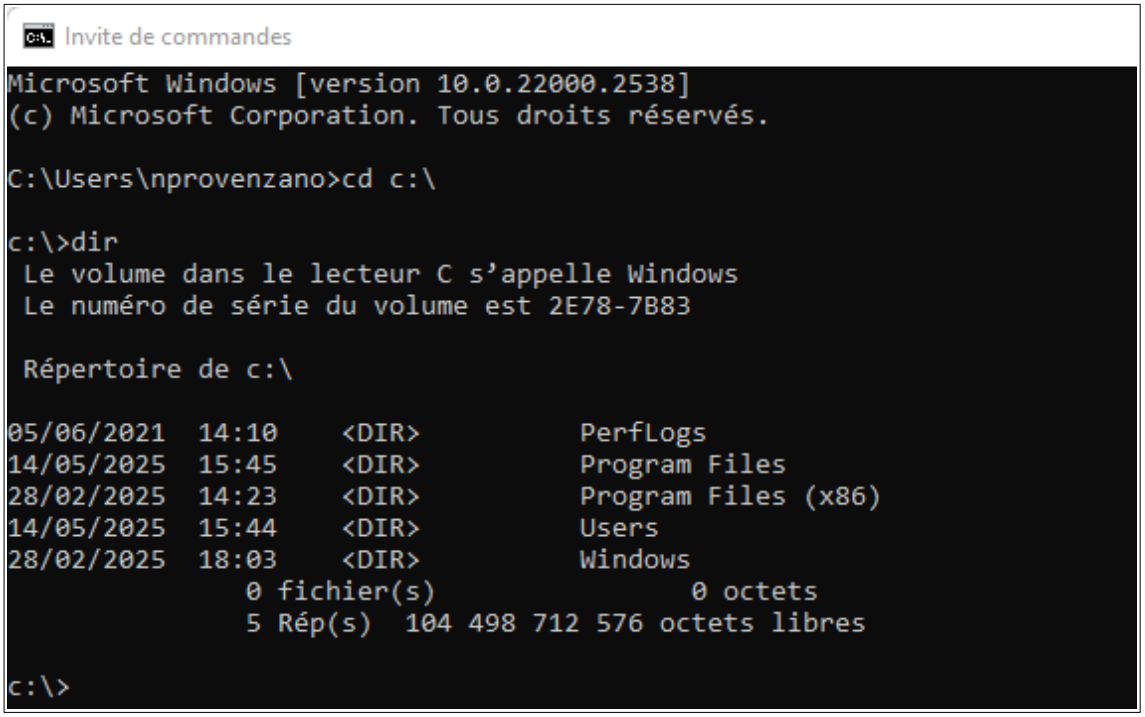
```
PS C:\Users\nprovenzano> dir


    Répertoire : C:\Users\nprovenzano


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        07/05/2025     14:44                .VirtualBox
d-----        02/04/2025     18:12                Cisco Packet Tracer 8.2.2
d-r---        14/03/2025     14:36                Contacts
d-r---        14/03/2025     14:36                Desktop
d-r---        14/03/2025     14:36                Documents
d-r---        21/03/2025     15:42                Downloads
d-r---        14/03/2025     14:36                Favorites
d-r---        14/03/2025     14:36                Links
d-r---        14/03/2025     14:36                Music
d-r---        14/03/2025     14:37                OneDrive
d-r---        02/05/2025     15:35                Pictures
d-r---        14/03/2025     14:36                Saved Games
d-r---        27/03/2025     10:57                Searches
d-r---        02/05/2025     15:35                Videos
d-----        07/05/2025     13:06                VirtualBox VMs
-a----        02/04/2025     13:59            188 .packettracer


PS C:\Users\nprovenzano>
```

Exécution de la même commande dans l'invite de commande CMD.



La commande **Get-Command -CommandType function** permet d'afficher la liste complète des fonctions disponibles. (La commande **Get-Alias** permet d'afficher la liste complètes des alias).

```
PS C:\> Get-Command -CommandType function

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Function        A:
Function        Add-BCDataCacheExtension                           1.0.0.0    BranchCache
Function        Add-BitLockerKeyProtector                          1.0.0.0    BitLocker
Function        Add-DnsClientDohServerAddress                      1.0.0.0    DnsClient
Function        Add-DnsClientNrptRule                              1.0.0.0    DnsClient
Function        Add-DtcClusterTMMapping                            1.0.0.0    MsDtc
Function        Add-EtwTraceProvider                               1.0.0.0    EventTracingMan...
Function        Add-InitiatorIdToMaskingSet                        2.0.0.0    Storage
Function        Add-MpPreference                                   1.0        ConfigDefender
Function        Add-MpPreference                                   1.0        Defender
Function        Add-NetEventNetworkAdapter                         1.0.0.0    NetEventPacketC...
Function        Add-NetEventPacketCaptureProvider                  1.0.0.0    NetEventPacketC...
Function        Add-NetEventProvider                               1.0.0.0    NetEventPacketC...
Function        Add-NetEventVFPProvider                            1.0.0.0    NetEventPacketC...
Function        Add-NetEventVmNetworkAdapter                       1.0.0.0    NetEventPacketC...
Function        Add-NetEventVmSwitch                               1.0.0.0    NetEventPacketC...
Function        Add-NetEventVmSwitchProvider                       1.0.0.0    NetEventPacketC...
Function        Add-NetEventWFPCaptureProvider                     1.0.0.0    NetEventPacketC...
Function        Add-NetIPHttpsCertBinding                          1.0.0.0    NetworkTransition
Function        Add-NetLbfoTeamMember                              2.0.0.0    NetLbfo
Function        Add-NetLbfoTeamNic                                 2.0.0.0    NetLbfo
Function        Add-NetNatExternalAddress                          1.0.0.0    NetNat
Function        Add-NetNatStaticMapping                            1.0.0.0    NetNat
Function        Add-NetSwitchTeamMember                            1.0.0.0    NetSwitchTeam
```

La commande **Get-Verb** permet de retrouver la liste complète des verbes officiels et leurs groupes d'applications.

```
PS C:\Users\nprovenzano> Get-Verb

Verb        Group
----        -----
Add         Common
Clear       Common
Close       Common
Copy        Common
Enter       Common
Exit        Common
Find        Common
Format      Common
Get         Common
Hide        Common
Join        Common
Lock        Common
Move        Common
New         Common
Open        Common
Optimize    Common
Pop         Common
Push        Common
Redo        Common
Remove      Common
Rename      Common
Reset       Common
Resize      Common
Search      Common
Select      Common
```

```
Resize      Common
Search      Common
Select      Common
Set         Common
Show        Common
Skip        Common
Split       Common
Step        Common
Switch      Common
Undo        Common
Unlock      Common
Watch       Common
Backup      Data
Checkpoint  Data
Compare     Data
Compress    Data
Convert     Data
ConvertFrom Data
ConvertTo   Data
Dismount    Data
Edit        Data
Expand      Data
Export      Data
Group       Data
Import      Data
Initialize  Data
Limit       Data
Merge       Data
Mount       Data
```

```
Merge       Data
Mount       Data
Out         Data
Publish     Data
Restore     Data
Save        Data
Sync        Data
Unpublish   Data
Update      Data
Approve     Lifecycle
Assert      Lifecycle
Complete    Lifecycle
Confirm     Lifecycle
Deny        Lifecycle
Disable     Lifecycle
Enable      Lifecycle
Install     Lifecycle
Invoke      Lifecycle
Register    Lifecycle
Request     Lifecycle
Restart     Lifecycle
Resume      Lifecycle
Start       Lifecycle
Stop        Lifecycle
Submit      Lifecycle
Suspend     Lifecycle
Uninstall   Lifecycle
Unregister  Lifecycle
Wait        Lifecycle
```

```
Unregister  Lifecycle
Wait        Lifecycle
Debug       Diagnostic
Measure     Diagnostic
Ping        Diagnostic
Repair      Diagnostic
Resolve     Diagnostic
Test        Diagnostic
Trace       Diagnostic
Connect     Communications
Disconnect  Communications
Read        Communications
Receive     Communications
Send        Communications
Write       Communications
Block       Security
Grant       Security
Protect     Security
Revoke      Security
Unblock     Security
Unprotect   Security
Use         Other


PS C:\Users\nprovenzano>
```

3. Les commandes de base

3.2 Get-Command

Les commandes de PowerShell sont appelées « cmdlets » (pour command-applets) ou commandelettes en français. Elles sont constituées de la manière suivante : un verbe et un nom séparés par un tiret : verbe-nom. Le verbe décrit l'action à appliquer sur le nom.

Pour voir les commandelettes on ajoute le paramètre **-CommandType** suivi du type de commandes choisi, à savoir **cmd**.

```
PS C:\Users\nprovenzano> Get-Command -CommandTpe cmdlet

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Function        Add-BCDataCacheExtension                           1.0.0.0    BranchCache
Function        Add-BitLockerKeyProtector                          1.0.0.0    BitLocker
Function        Add-DnsClientDohServerAddress                      1.0.0.0    DnsClient
Function        Add-DnsClientNrptRule                              1.0.0.0    DnsClient
Function        Add-DtcClusterTMMapping                            1.0.0.0    MsDtc
Function        Add-EtwTraceProvider                               1.0.0.0    EventTracingMan...
Function        Add-InitiatorIdToMaskingSet                        2.0.0.0    Storage
Function        Add-MpPreference                                   1.0        ConfigDefender
Function        Add-MpPreference                                   1.0        Defender
Function        Add-NetEventNetworkAdapter                         1.0.0.0    NetEventPacketC...
Function        Add-NetEventPacketCaptureProvider                  1.0.0.0    NetEventPacketC...
Function        Add-NetEventProvider                               1.0.0.0    NetEventPacketC...
Function        Add-NetEventVFPProvider                            1.0.0.0    NetEventPacketC...
Function        Add-NetEventVmNetworkAdapter                       1.0.0.0    NetEventPacketC...
Function        Add-NetEventVmSwitch                               1.0.0.0    NetEventPacketC...
Function        Add-NetEventVmSwitchProvider                       1.0.0.0    NetEventPacketC...
Function        Add-NetEventWFPCaptureProvider                     1.0.0.0    NetEventPacketC...
Function        Add-NetIPHttpsCertBinding                          1.0.0.0    NetworkTransition
Function        Add-NetLbfoTeamMember                              2.0.0.0    NetLbfo
```

Dans la première version de PowerShell, les commandes de base étaient au nombre de 129. Dans la version 5.1, leur nombre beaucoup plus important varie en fonction du type de système (client ou serveur), du nombre de rôles. Pour les compter, utiliser la commande
(Get-Command -CommandType cmdlet).count

```
PS C:\> (Get-Command -CommandType cmdlet).count
700

PS C:\>
```

Get-Command possède le paramètre **-verb** et ce dernier permet de retourner toutes les commandes commençant par un verbe donné. **Get-Command** possède le paramètre **-verb** et ce dernier permet de retourner toutes les commandes commençant par un verbe donné. Pour obtenit toutes les commandes dont le verbe commence par **write**, saisissez la commande **Get-Command -Verb write**

```
PS C:\> Get-Command -Verb write

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           Write-FileSystemCache                              2.0.0.0    Storage
Alias           Write-FileSystemCache                              1.0.0.0    VMDirectStorage
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.Power...
Cmdlet          Write-Error                                        3.1.0.0    Microsoft.Power...
Cmdlet          Write-EventLog                                     3.1.0.0    Microsoft.Power...
Cmdlet          Write-Host                                         3.1.0.0    Microsoft.Power...
Cmdlet          Write-Information                                  3.1.0.0    Microsoft.Power...
Cmdlet          Write-Output                                       3.1.0.0    Microsoft.Power...
Cmdlet          Write-Progress                                     3.1.0.0    Microsoft.Power...
Cmdlet          Write-Verbose                                      3.1.0.0    Microsoft.Power...
Cmdlet          Write-Warning                                      3.1.0.0    Microsoft.Power...


PS C:\>
```

De manière similaire, avec cette fois-ci le paramètre -noun, pour afficher les commandes qui s'appliquent, par exemple, aux objets, c'est-à-dire celles dont la partie nom est object avec la commande **Get-Command -Noun object**

```
PS C:\> Get-Command -Noun object

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Cmdlet          Compare-Object                                     3.1.0.0    Microsoft.Power...
Cmdlet          ForEach-Object                                     3.0.0.0    Microsoft.Power...
Cmdlet          Group-Object                                       3.1.0.0    Microsoft.Power...
Cmdlet          Measure-Object                                     3.1.0.0    Microsoft.Power...
Cmdlet          New-Object                                         3.1.0.0    Microsoft.Power...
Cmdlet          Select-Object                                      3.1.0.0    Microsoft.Power...
Cmdlet          Sort-Object                                        3.1.0.0    Microsoft.Power...
Cmdlet          Tee-Object                                         3.1.0.0    Microsoft.Power...
Cmdlet          Where-Object                                       3.0.0.0    Microsoft.Power...


PS C:\>
```

Pour afficher les commandes de type alias, utiliser la commande **Get-Command -Commadtype alias**

```
PS C:\> Get-Command -Commandtype alias

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           % -> ForEach-Object
Alias           ? -> Where-Object
Alias           ac -> Add-Content
Alias           Add-AppPackage                                     2.0.1.0    Appx
Alias           Add-AppPackageVolume                               2.0.1.0    Appx
Alias           Add-AppProvisionedPackage                          3.0        Dism
Alias           Add-ProvisionedAppPackage                          3.0        Dism
Alias           Add-ProvisionedAppSharedPackageContainer           3.0        Dism
Alias           Add-ProvisionedAppxPackage                         3.0        Dism
Alias           Add-ProvisioningPackage                            3.0        Provisioning
Alias           Add-TrustedProvisioningCertificate                 3.0        Provisioning
Alias           algm ->                                            1.0.0.0    Microsoft.Power...
Alias           Apply-WindowsUnattend                              3.0        Dism
Alias           asnp -> Add-PSSnapin
Alias           blsmba ->                                          2.0.0.0    SmbShare
Alias           cat -> Get-Content
Alias           cd -> Set-Location
Alias           CFS -> ConvertFrom-String                          3.1.0.0    Microsoft.Power...
Alias           chdir -> Set-Location
Alias           clc -> Clear-Content
Alias           clear -> Clear-Host
Alias           Clear-WUJob                                        2.2.1.5    PSWindowsUpdate
```

Pour rechercher une commande dont on ignore le nom mais sachant qu'elle commence par le verbe **Get**, saisissez la commande **Get-Command Get-***

```
PS C:\> Get-Command Get-*

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           Get-AppPackage                                     2.0.1.0    Appx
Alias           Get-AppPackageAutoUpdateSettings                   2.0.1.0    Appx
Alias           Get-AppPackageDefaultVolume                        2.0.1.0    Appx
Alias           Get-AppPackageLastError                            2.0.1.0    Appx
Alias           Get-AppPackageLog                                  2.0.1.0    Appx
Alias           Get-AppPackageManifest                             2.0.1.0    Appx
Alias           Get-AppPackageVolume                               2.0.1.0    Appx
Alias           Get-AppProvisionedPackage                          3.0        Dism
Alias           Get-DiskSNV                                        2.0.0.0    Storage
Alias           Get-DiskSNV                                        1.0.0.0    VMDirectStorage
Alias           Get-Language                                       1.0        LanguagePackMan...
Alias           Get-PhysicalDiskSNV                                2.0.0.0    Storage
Alias           Get-PhysicalDiskSNV                                1.0.0.0    VMDirectStorage
Alias           Get-PreferredLanguage                              1.0        LanguagePackMan...
Alias           Get-ProvisionedAppPackage                          3.0        Dism
Alias           Get-ProvisionedAppSharedPackageContainer           3.0        Dism
Alias           Get-ProvisionedAppxPackage                         3.0        Dism
Alias           Get-StorageEnclosureSNV                            2.0.0.0    Storage
Alias           Get-StorageEnclosureSNV                            1.0.0.0    VMDirectStorage
Alias           Get-SystemLanguage                                 1.0        LanguagePackMan...
Alias           Get-WUInstall                                      2.2.1.5    PSWindowsUpdate
Alias           Get-WUList                                         2.2.1.5    PSWindowsUpdate
Function        Get-AppBackgroundTask                              1.0.0.0    AppBackgroundTask
Function        Get-AppvVirtualProcess                             1.0.0.0    AppvClient
```

3.3 Get-Member et le concept des objets

La commande **Get-Member** retourne toutes les propriétés et méthodes d'un objet ainsi que son type.

Créer la variable **$maVariable** et affectez-lui une valeur de type chaîne (String).

```
PS C:\> $maVariable = 'Bonjour tout le monde !'

PS C:\>
```

Taper la commande **$maVariable | get-member**

```
PS C:\> $maVariable | gm

   TypeName : System.String

Name             MemberType            Definition
----             ----------            ----------
Clone            Method                System.Object Clone(), System.Object ICloneable.Clone()
CompareTo        Method                int CompareTo(System.Object value), int CompareTo(stri...
Contains         Method                bool Contains(string value)
CopyTo           Method                void CopyTo(int sourceIndex, char[] destination, int d...
EndsWith         Method                bool EndsWith(string value), bool EndsWith(string valu...
Equals           Method                bool Equals(System.Object obj), bool Equals(string val...
GetEnumerator    Method                System.CharEnumerator GetEnumerator(), System.Collecti...
GetHashCode      Method                int GetHashCode()
GetType          Method                type GetType()
GetTypeCode      Method                System.TypeCode GetTypeCode(), System.TypeCode IConver...
```

```
Substring        Method                string Substring(int startIndex), string Substring(int...
ToBoolean        Method                bool IConvertible.ToBoolean(System.IFormatProvider pro...
ToByte           Method                byte IConvertible.ToByte(System.IFormatProvider provider)
ToChar           Method                char IConvertible.ToChar(System.IFormatProvider provider)
ToCharArray      Method                char[] ToCharArray(), char[] ToCharArray(int startInde...
ToDateTime       Method                datetime IConvertible.ToDateTime(System.IFormatProvide...
ToDecimal        Method                decimal IConvertible.ToDecimal(System.IFormatProvider ...
ToDouble         Method                double IConvertible.ToDouble(System.IFormatProvider pr...
ToInt16          Method                int16 IConvertible.ToInt16(System.IFormatProvider prov...
ToInt32          Method                int IConvertible.ToInt32(System.IFormatProvider provider)
ToInt64          Method                long IConvertible.ToInt64(System.IFormatProvider provi...
ToLower          Method                string ToLower(), string ToLower(cultureinfo culture)
ToLowerInvariant Method                string ToLowerInvariant()
ToSByte          Method                sbyte IConvertible.ToSByte(System.IFormatProvider prov...
ToSingle         Method                float IConvertible.ToSingle(System.IFormatProvider pro...
ToString         Method                string ToString(), string ToString(System.IFormatProvi...
ToType           Method                System.Object IConvertible.ToType(type conversionType,...
ToUInt16         Method                uint16 IConvertible.ToUInt16(System.IFormatProvider pr...
ToUInt32         Method                uint32 IConvertible.ToUInt32(System.IFormatProvider pr...
ToUInt64         Method                uint64 IConvertible.ToUInt64(System.IFormatProvider pr...
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:\>
```

Utiliser la méthode **ToUpper** afin de retourner la chaîne contenue dans $maVariable en majuscules.

```
PS C:\> $maVariable.ToUpper()
BONJOUR TOUT LE MONDE !

PS C:\>
```

De la même façon, utiliser la propriété **Length** afin d'obtenir le nombre de caractères contenus dans notre chaîne.

```
PS C:\> $maVariable.Length
23

PS C:\>
```

4. Navigation dans les répertoires et les fichiers

4.1 Les nouvelles commandes

Lorsqu'on exécute la commande **dir** dans la console PowerShell, on fait appel à un alias qui exécute la commande **Get-ChildItem**. Pour le vérifier, tapez la commande suivante **Get-Alias dir**

```
PS C:\> Get-Alias dir

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           dir -> Get-ChildItem


PS C:\>
```

4.2 Get-ChildItem (Alias : gci, ls dir)

Cette commandelette permet d'obtenir les fichiers et dossiers présents dans le système de fichiers. Observer le résultat de la commande suivante : **gci c:**

```
PS C:\> cd C:\Windows\System32

PS C:\Windows\System32> gci c:\


    Répertoire : C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        05/06/2021     14:10                PerfLogs
d-r---        14/05/2025     15:45                Program Files
d-r---        28/02/2025     13:23                Program Files (x86)
d-r---        14/05/2025     15:44                Users
d-----        28/02/2025     17:03                Windows


PS C:\Windows\System32>
```

La colonne Mode indique la nature des objets à l'intérieur du système de fichiers, voici les valeurs possibles :

- d : pour un répertoire.
- a : pour archive.
- r : pour un objet en lecture seule.
- h : pour un objet caché.
- s : pour un objet système.

Pour afficher les fichiers cachés, ajouter à la commande **Get-Childitem** le paramètre **-Force**

```
PS C:\Windows\System32> gci c:\ -Force


    Répertoire : C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d--hs-        14/05/2025     15:44                $Recycle.Bin
d--h--        12/05/2025     14:03                $WINDOWS.~BT
d--h--        28/02/2025     09:51                $WinREAgent
d--hsl        13/01/2025     17:38                Documents and Settings
d-----        05/06/2021     14:10                PerfLogs
d-r---        14/05/2025     15:45                Program Files
d-r---        28/02/2025     13:23                Program Files (x86)
d--h--        29/04/2025     16:47                ProgramData
d--hs-        28/02/2025     09:51                Recovery
d--hs-        15/05/2025     10:00                System Volume Information
d-r---        14/05/2025     15:44                Users
d-----        28/02/2025     17:03                Windows
-a-hs-        15/05/2025     09:41          12288 DumpStack.log.tmp
-a-hs-        15/05/2025     09:41    27437273088 hiberfil.sys
-a-hs-        15/05/2025     09:41    10200547328 pagefile.sys
-a-hs-        15/05/2025     09:41       16777216 swapfile.sys


PS C:\Windows\System32>
```

Le nom des colonnes indique le nom d'une propriété de l'objet fichier ou répertoire (la commandelette **gci** renvoie une collection d'objets).

Exemples :

Afficher (récursivement) tous les fichiers ayant l'extension **.log** contenus à l'intérieur d'une arborescence avec la commande **Get-ChildItem C:\Temp* -Include *.log -Recurse**

La commande **cd ** permet de se situer à la racine

```
PS C:\Windows\System32> cd \

PS C:\>
```

```
PS C:\> Get-ChildItem C:\Users\* -Include *.log -Recurse


    Répertoire : C:\Users\Administrateur\.VirtualBox


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         25/04/2025     14:12           1346 selectorwindow.log
-a----         25/04/2025     14:12          28732 VBoxSVC.log


    Répertoire : C:\Users\Administrateur\Cisco Packet Tracer 8.2.2\logs


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         06/03/2025     17:31          74124 pt_03.06.2025_15.42.32.596.log
-a----         06/03/2025     17:31           5366 pt_03.06.2025_15.42.33.950.log
-a----         07/03/2025     16:27          72188 pt_03.07.2025_15.21.35.764.log
-a----         07/03/2025     16:27           5366 pt_03.07.2025_15.21.37.224.log
-a----         07/03/2025     16:47          82806 pt_03.07.2025_16.27.55.456.log
-a----         07/03/2025     16:47           5168 pt_03.07.2025_16.27.56.796.log
-a----         13/03/2025     16:32          85090 pt_03.13.2025_10.07.08.005.log
-a----         13/03/2025     14:52           5406 pt_03.13.2025_10.07.09.352.log
-a----         13/03/2025     15:51           5200 pt_03.13.2025_14.52.57.358.log
-a----         13/03/2025     16:33           5818 pt_03.13.2025_15.51.31.966.log
-a----         13/03/2025     16:33          24432 pt_03.13.2025_16.32.55.425.log
-a----         13/03/2025     17:42          66244 pt_03.13.2025_16.33.32.225.log
-a----         13/03/2025     17:42           5406 pt_03.13.2025_16.33.33.577.log
```

Afficher le nom des fichiers dont la taille est supérieure à **32 Ko** avec la commande **Get-ChildItem | Where-Object {$_.Length -gt 32KB}**

```
PS C:\> Get-ChildItem | Where-Object {$_.Length -gt 32KB}

PS C:\> cd C:\Windows

PS C:\Windows> Get-ChildItem | Where-Object {$_.Length -gt 32KB}


    Répertoire : C:\Windows


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         08/06/2022     11:41         102400 bfsvc.exe
-a--s-         15/05/2025     09:43          67584 bootstat.dat
-a----         28/02/2025     09:53        5092056 explorer.exe
-a----         28/02/2025     09:54        1101824 HelpPane.exe
-a----         05/06/2021     14:06          36864 hh.exe
-a----         05/06/2021     14:05          43131 mib.bin
-a----         28/02/2025     09:54         352256 notepad.exe
-a----         08/06/2022     11:43         397312 regedit.exe
-a----         28/02/2025     09:53         192512 splwow64.exe
-a----         05/06/2021     14:06          68608 twain_32.dll
-a----         05/06/2021     20:23         316640 WMSysPr9.prx


PS C:\Windows>
```

Afficher les fichiers dont la date de dernier enregistrement est **postérieure au 31/01/2018** avec la commande **Get-ChildItem | Where-Object {$_.LastWriteTime -gt '01/31/2018'}**

```
PS C:\> Get-ChildItem | Where-Object {$_.LastWriteTime -gt '01/31/2018'}


    Répertoire : C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        05/06/2021     14:10                PerfLogs
d-r---        14/05/2025     15:45                Program Files
d-r---        28/02/2025     13:23                Program Files (x86)
d-r---        14/05/2025     15:44                Users
d-----        28/02/2025     17:03                Windows


PS C:\>
```

→ La date est toujours au format américain, soit MM/JJ/AAAA.

Le pipe « | » permet de passer un ou plusieurs objets à la commande qui suit. Dans nos exemples, nous passons chaque objet à la commandelette **Where-Object** (appelée aussi clause, ou encore filtre). Cette dernière va analyser les propriétés **Length** ou **LastWriteTime** et retourner les objets correspondants. Le « **$_** » indique qu'on traite l'objet courant.

Afficher tous les fichiers ou répertoires cachés à la racine de la partition système avec la commande **Get-ChildItem c:\ -Attributes Hidden**

```
PS C:\> Get-ChildItem C:\ -Attributes Hidden


    Répertoire : C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d--hs-        14/05/2025     15:44                $Recycle.Bin
d--h--        12/05/2025     14:03                $WINDOWS.~BT
d--h--        28/02/2025     09:51                $WinREAgent
d--hsl        13/01/2025     17:38                Documents and Settings
d--h--        29/04/2025     16:47                ProgramData
d--hs-        28/02/2025     09:51                Recovery
d--hs-        15/05/2025     10:00                System Volume Information
-a-hs-        15/05/2025     09:41          12288 DumpStack.log.tmp
-a-hs-        15/05/2025     09:41    27437273088 hiberfil.sys
-a-hs-        15/05/2025     09:41    10200547328 pagefile.sys
-a-hs-        15/05/2025     09:41       16777216 swapfile.sys


PS C:\>
```

On peut associer des combinaisons d'attributs via les opérateurs suivants :

+ : pour signifier un ET logique.
, : pour signifier un OU logique.
! : pour signifier une négation.

Par exemple, filtrer uniquement les fichiers cachés et non les répertoires en combinant les attributs « caché » et « n'est pas un répertoire ».

```
PS C:\> Get-ChildItem C:\ -Attributes Hidden+!Directory


    Répertoire : C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a-hs-        15/05/2025     09:41          12288 DumpStack.log.tmp
-a-hs-        15/05/2025     09:41    27437273088 hiberfil.sys
-a-hs-        15/05/2025     09:41    10200547328 pagefile.sys
-a-hs-        15/05/2025     09:41       16777216 swapfile.sys


PS C:\>
```

4.3 Set-Location (Alias : sl, cd, chdir)

Elle nous permet de nous déplacer dans une arborescence de dossiers. Exécution de la commande **Set-Location D:**

```
PS C:\> Set-Location D:\

PS D:\>
```

Comme dans CMD, on peut utiliser des chemins relatifs ainsi que les raccourcis « **..** » et « \ », pour désigner respectivement le répertoire parent et le répertoire racine du disque en cours.

4.4 Get-Location (Alias : gl ; pwd)

Cette commande retourne l'emplacement actuel à l'intérieur d'une arborescence. Exécution de la commande **Get-Location** dans le répertoire **users**.

```
PS C:\> Set-Location D:\

PS D:\>
```

Afficher les propriétés et les méthodes avec la commande **Get-Location**

```
PS C:\> cd \users

PS C:\users> Get-Location

Path
----
C:\users


PS C:\users>
```

Afficher les propriétés et les méthodes avec la commande **Get-Location | Get-Member**

```
PS C:\users> Get-Location | Get-Member


   TypeName : System.Management.Automation.PathInfo

Name         MemberType Definition
----         ---------- ----------
Equals       Method     bool Equals(System.Object obj)
GetHashCode  Method     int GetHashCode()
GetType      Method     type GetType()
ToString     Method     string ToString()
Drive        Property   System.Management.Automation.PSDriveInfo Drive {get;}
Path         Property   string Path {get;}
Provider     Property   System.Management.Automation.ProviderInfo Provider {get;}
ProviderPath Property   string ProviderPath {get;}


PS C:\users>
```

Récupérer dans une variable le chemin (path) de l'emplacement courant en une seule ligne de commande. Exécution de la commande **$CheminCourant = (Get-Location).Path**

```
PS C:\> $CheminCourant = (Get-Location).Path
```

Stocker la valeur du chemin courant, en l'occurrence **C:\users**, dans la variable **$CheminCourant.** Affichez-le dans la console :

```
PS C:\> $CheminCourant
C:\

PS C:\>
```

4.5 New-Items (Alias : ni, md)

Cette commandelette permet de créer des répertoires, à l'instar de la commande **md** de CMD, mais aussi des fichiers.

4.5.1 Création d'un répertoire

Créer le répertoire **Temp**. Exécution de la commande **New-Item -ItemType directory -Name Temp**

```
PS C:\> New-Item -ItemType directory -Name Temp


    Répertoire : C:\

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        15/05/2025     11:11                Temp


PS C:\>
```

Si le dossier avait contenu un espace, il faut le préciser entre guillemets. Exécution de la commande **New-Item -Name 'Dossier Test' -ItemType directory**

```
PS C:\> New-Item -Name 'Dossier Test' -ItemType directory


    Répertoire : C:\

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        15/05/2025     11:12                Dossier Test


PS C:\>
```

4.5.2 Création d'un fichier

Créer un fichier nommé **monFichier.txt** qui contient la phrase « Madame Pasquier for ever ! »

```
PS C:\> ni -Name monFichier.txt -ItemType file -Value 'Madame Pasquier for ever !'


    Répertoire : C:\

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        15/05/2025     11:14             26 monFichier.txt


PS C:\>
```

→ Les opérateurs de redirection « > » et « >> » fonctionnent également avec PowerShell.

4.6 Remove-Items (Alias : ri, rm, rmdir, rd, erase, del)

Cette commandelette, comme son nom l'indique, permet de supprimer des fichiers ou des dossiers. Supprimez tous les fichiers **.log** contenus dans le répertoire **C:\Temp** avec la commande **Remove-Item C:\Temp*.log**

```
PS C:\> Remove-Item C:\Temp\*.log
PS C:\>
```

Supprimer sélectivement tous les fichiers contenus dans une arborescence de dossiers dont l'extension est **.txt** **Get-ChildItem C:\Temp* -Include *.txt -Recurse | Remove-Item**

```
PS C:\> Get-ChildItem C:\Temp\* -Include *.txt -Recurse | Remove-Item
PS C:\>
```

On liste d'abord les objets (fichiers à supprimer), puis on les passe via le pipe à la commande **Remove-item**.

Pour supprimer un fichier système, masqué ou en lecture seule, il suffit tout simplement d'utiliser le paramètre **-Force**. Exécution de la commande **ri fichierASupprimer.txt -Force**

```
PS C:\> ri fichierASupprimer.txt -Force
PS C:\>
```

4.7 Move-Item (Alias : mi, move, mv)

Cette commande permet de déplacer un fichier ou un répertoire d'un emplacement vers un autre emplacement. Dans le cas d'un répertoire, le contenu est également déplacé. **Move-Item** permet en outre d'effectuer le renommage de l'objet manipulé.

4.7.1 Déplacement de fichiers

Exemple : déplacement des fichiers ***.jpg** du répertoire courant vers le dossier « **mes photos** ».

```
PS C:\> Move-Item -Path *.jpg -destination 'Mes photos'
PS C:\>
```

4.7.2 Déplacement d'un répertoire

Le déplacement d'un répertoire est similaire au déplacement de fichiers.

```
PS C:\> New-Item -ItemType directory -Name Rep1


    Répertoire : C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        15/05/2025     11:24                Rep1


PS C:\>
```

```
PS C:\> New-Item -ItemType directory -Name Rep2


    Répertoire : C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        15/05/2025     11:22                Rep2
```

```
PS C:\> New-Item -ItemType directory -Name Rep3


    Répertoire : C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        15/05/2025     11:22                Rep3


PS C:\>
```

Exemple : Pour déplacer l'intégralité du répertoire Rep1 dans le répertoire Rep2.

```
PS C:\> Move-Item 'Rep1' 'Rep2'
PS C:\>
```

Exemple : Pour renommer le dossier Rep1 en Rep3

```
PS C:\> cd \Rep2
PS C:\Rep2> Move-Item 'Rep1' 'Rep3'
PS C:\Rep2>
```

4.8 Rename-Items (Alias : ren, rni)

L'objectif de cette commande est de renommer un fichier ou un dossier. Celle-ci n'est que moyennement utile dans la mesure où elle fait double emploi avec la commande **Move-Item**. Elle peut éviter de confondre renommage et déplacement comme dans l'exemple précédent.

4.8.1 Renommer un fichier

Exemple : renommage du fichier **monFichierDeLog.txt** en **ficlog.txt** avec la commande **Rename-Item**

```
PS C:\> Rename-Item -Path C:monFichierDelog.txt -NewName ficlog.txt
PS C:\>
```

4.8.2 Renommer un dossier

Exemple : renommage du répertoire **Dossier1** en **Dossier2** avec la commande **Rename-Item**

```
PS C:\> New-Item -ItemType directory -Name Dossier1


    Répertoire : C:\

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        15/05/2025     11:34                Dossier1


PS C:\>
```

```
PS C:\> Rename-Item C:\Dossier1 Dossier2
PS C:\>
```

4.9 Copy-Items (Alias : cpi, cp, copy)

Grâce à cette commande, nous pouvons copier des fichiers ou des répertoires, voire les 2 à la fois. Quelques exemples :

- Copie un fichier d'un répertoire source vers un répertoire destination avec la commande **Copy-Item**

```
PS C:\> Copy-Item -Path C:\ficLog.txt -destination D:\Logs
PS C:\>
```

- Copie d'une arborescence de répertoires (c'est-à-dire avec tous les sous-dossiers et fichiers).

```
PS > Copy-Item -Path RepSource -Destination RepDest -Recurse
```

→ Copy-Item crée automatiquement le répertoire de destination s'il n'existe pas.

6. Module Active Directory

6.1 Mise en route du module

Il permet d'administrer en ligne de commandes PowerShell le rôle « Active Directory Domain Services (AD DS) ». Il est installé en même temps que le rôle Active Directory.

Dans le cas où la fonctionnalité d'importation automatique des modules est désactivé, il faudra alors l'importer manuellement avec la commande **Import-Module ActiveDirectory**

```
PS C:\> Import-Module ActiveDirectory
```

Pour obtenir l'ensemble des commandes apportées par le module Active Directory, saisissez le commande **Get-Command -Module ActiveDirectory**

```
PS C:\> Get-Command -Module ActiveDirectory

PS C:\>
```