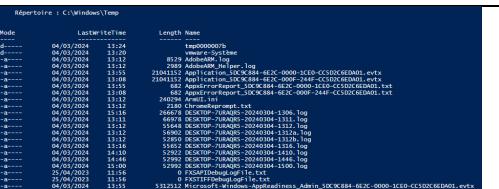
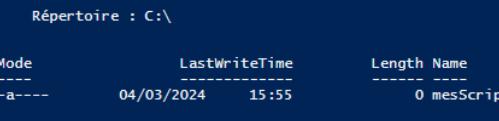


Action	Commande	Résultat
Lancer Powershell ISE	PS C:\Users\WILLIAMS> powershell_ISE.e	
Lancer PowerShell	PS C:\Users\WILLIAMS> powershell.exe	
Lancer PowerShell en mode administrateur		
Lancer une commande DOS, puis PowerShell	C:\Users\WILLIAMS>powershell.exe	
Lancer une commande dans PowerShell d'aide	PS C:\Users\WILLIAMS> get-help Get-ChildItem	
Lancer l'aide de la commande « Move-item »	PS C:\Users\WILLIAMS> get-help Move-Item	
Ouvrez l'aide de la commande « Move-Item » sur internet	PS C:\Users\WILLIAMS> get-help Move-Item -Online	
Lister le contenu d'un dossier	PS C:\Users\WILLIAMS> Get-ChildItem "c:"	

Action	Commande Powershell	Résultat																										
Lister les fichiers présent dans le dossier c:\Temp	<pre>PS C:\Users\WILLIAMS> Get-ChildItem "C:\Windows\Temp"</pre>																											
Demander le nom de votre PC	<pre>PS C:\Users\WILLIAMS> hostname</pre>	DESKTOP-7URAQR5																										
Redémarrer votre PC au bout de 30 secondes																												
Crée un dossier dans C:\ Nommer « MesScripts »	<pre>New-Item c:\ -Name mesScript</pre>																											
Demander la date année mois jours heures secondes et fuseau horaire	<pre>PS C:\WINDOWS\system32> Get-Date</pre>	Lundi 4 mars 2024 15:58:38																										
Crée un fichier sur votre bureau nommer « j'aime le powershell.txt »	<pre>New-Item "C:\Users\WILLIAMS\Desktop" -Name jaimele_powershell.txt</pre>																											
Afficher la Build Windows sur le bureau windows (modification par la base de registre)	<pre>[System.Environment]::osversion.version</pre>	<table border="1"> <thead> <tr> <th>Major</th> <th>Minor</th> <th>Build</th> <th>Revision</th> </tr> </thead> <tbody> <tr> <td>10</td> <td>0</td> <td>19045</td> <td>0</td> </tr> </tbody> </table>	Major	Minor	Build	Revision	10	0	19045	0																		
Major	Minor	Build	Revision																									
10	0	19045	0																									
Ajouter la fonctionnalité windows service d'impression en PDF	<pre>Enable-WindowsOptionalFeature -Online -FeatureName PrintingToPDFServices-Features</pre>	<table border="1"> <thead> <tr> <th>Path</th> <th>:</th> </tr> </thead> <tbody> <tr> <td>Online</td> <td>: True</td> </tr> <tr> <td>RestartNeeded</td> <td>: False</td> </tr> </tbody> </table>	Path	:	Online	: True	RestartNeeded	: False																				
Path	:																											
Online	: True																											
RestartNeeded	: False																											
Récupérer votre adresse IP v4	<pre>Get-NetIPAddress -AddressFamily IPV4</pre>	<table border="1"> <tbody> <tr> <td>IPAddress</td> <td>: 10.1.42.109</td> </tr> <tr> <td>InterfaceIndex</td> <td>: 19</td> </tr> <tr> <td>InterfaceAlias</td> <td>: Ethernet 2</td> </tr> <tr> <td>AddressFamily</td> <td>: IPv4</td> </tr> <tr> <td>Type</td> <td>: Unicast</td> </tr> <tr> <td>PrefixLength</td> <td>: 23</td> </tr> <tr> <td>PrefixOrigin</td> <td>: Dhcp</td> </tr> <tr> <td>SuffixOrigin</td> <td>: Dhcp</td> </tr> <tr> <td>AddressState</td> <td>: Preferred</td> </tr> <tr> <td>ValidLifetime</td> <td>: 4.21:16:35</td> </tr> <tr> <td>PreferredLifetime</td> <td>: 4.21:16:35</td> </tr> <tr> <td>SkipAsSource</td> <td>: False</td> </tr> <tr> <td>PolicyStore</td> <td>: ActiveStore</td> </tr> </tbody> </table>	IPAddress	: 10.1.42.109	InterfaceIndex	: 19	InterfaceAlias	: Ethernet 2	AddressFamily	: IPv4	Type	: Unicast	PrefixLength	: 23	PrefixOrigin	: Dhcp	SuffixOrigin	: Dhcp	AddressState	: Preferred	ValidLifetime	: 4.21:16:35	PreferredLifetime	: 4.21:16:35	SkipAsSource	: False	PolicyStore	: ActiveStore
IPAddress	: 10.1.42.109																											
InterfaceIndex	: 19																											
InterfaceAlias	: Ethernet 2																											
AddressFamily	: IPv4																											
Type	: Unicast																											
PrefixLength	: 23																											
PrefixOrigin	: Dhcp																											
SuffixOrigin	: Dhcp																											
AddressState	: Preferred																											
ValidLifetime	: 4.21:16:35																											
PreferredLifetime	: 4.21:16:35																											
SkipAsSource	: False																											
PolicyStore	: ActiveStore																											

Manipulation de fichiers / dossiers		
Apprendre	Commande en PS	Remarque
Crée un dossier	<code>New-Item "C:\Users\jack.williams\Desktop\TEST" -ItemType Directory</code>	J'ai créer un dossier sur le Bureau
Crée un fichier	<code>New-Item -Path "C:\Users\jack.williams\Desktop\TEST\test.txt" -ItemType File</code>	J'ai créer un fichier .txt dans le dossier TEST
Bouger un fichier	<code>Get-Item -Path C:\Users\jack.williams\Desktop\TEST\test.txt Move-Item -Destination C:\Users\jack.williams\Documents</code>	J'ai déplacer le fichier test.txt dans le dossier Documents
Bouger un dossier	<code>Get-Item -Path C:\Users\jack.williams\Desktop\TEST Move-Item -Destination C:\Users\jack.williams\Documents</code>	J'ai déplacer le dossier TEST dans le dossier Documents
Renommer un fichier	<code>Rename-Item -Path C:\Users\jack.williams\Documents\test.txt Powershell</code>	J'ai renommer le fichier test.txt en powershell.txt
Renommer un dossier	<code>Rename-Item -Path C:\Users\jack.williams\Documents\TEST Powershell_file</code>	J'ai renommer le dossier TEST en Powershell_file
Supprimer un fichier	<code>Remove-Item -Path C:\Users\jack.williams\Documents\Powershell</code>	J'ai supprimer le fichier Powershell
Supprimer un dossier	<code>Remove-Item -Path C:\Users\jack.williams\Documents\Powershell_file</code>	J'ai supprimer le dossier Powershell_file

TP POWERSHELL 2

Demande	Comma nde en PS	Résultat																														
Les commentaires	#Test	<pre>PS C:\WINDOWS\system32> #Test</pre>																														
Les Variables	\$	<pre>1 #Cette commande nous créer une variable qui contient "Hello". script lancer par Jack 2 \$mavariablen = "Hello"</pre>																														
Get-Module	Get-module	<p>#Cette commande nous montre tout les fichier dans Te repertoire C:\Program Files\windowsPowerShell\Modules et C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules.</p> <pre>PS C:\WINDOWS\system32> #Cette commande nous montre tout les fichier dans le repertoire C:\Program Files\WindowsPowerShell\Modules et C:\WINDOWS\system32> Get-module -ListAvailable Répertoire : C:\Program Files\WindowsPowerShell\Modules ModuleType Version Name ExportedCommands ----- ----- ---- ----- Script 1.0.1 Microsoft.PowerShell.Operation.V... {Get-OperationValidation, Invoke-OperationValidation} Binary 1.0.0.1 PackageManagement {Find-Package, Get-Package, Get-PackageProvider, Get-PackageSource...} Script 3.4.0 Pester {Describe, Context, It, Should...} Script 1.0.0.1 PowerShellGet {Install-Module, Find-Module, Save-Module, Update-Module...} Script 2.0.0 PSReadLine {Get-PSReadLineKeyHandler, Set-PSReadLineKeyHandler, Remove-PSReadLineKeyHandler, Ge...</pre> <p>Répertoire : C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules</p> <pre>ModuleType Version Name ExportedCommands ----- ----- ---- ----- Manifest 1.0.0.0 AppBackgroundTask {Disable-AppBackgroundTaskDiagnosticLog, Enable-AppBackgroundTaskDiagnosticLog, Set-AppBackgroundTaskDiagnosticLog...} Manifest 2.0.0.0 AppLocker {Get-AppLockerFileInformation, Get-AppLockerPolicy, New-AppLockerPolicy, Set-AppLock...} Manifest 1.0.0.0 AppvClient {Add-AppvClientConnectionGroup, Add-AppvClientPackage, Add-AppvPublishingServer, Dis...</pre> Manifest 2.0.1.0 Apps {Add-AppvPackage, Get-AppvPackage, Get-AppvPackageAndUpdateSettings, Get-Appv...																														
Clear-host	Clear-host	<p>#Cette commande nous supprime tout les commande au paravant . Script lancer par Jack</p> <pre>Clear-Host</pre>																														
Write-host vs Echo	Echo et write-host	<p>#Cette commande nous affiche ce qu'on lui demande. Script lancer par Jack</p> <pre>write-host test echo Test 2</pre>																														
Read-host	Read-host	<p>#Cette commande nous créer un prompt qui nous demande d'entrer une info. Read-host "Entrez votre age."</p>																														
Les Pipes		<pre>PS C:\WINDOWS\system32> #Cette commande nous créer un prompt qui nous demande d'entrer une info. Script lancer par Jack Get-Apppackage Select Name, PackageFullname</pre> <table border="1"> <thead> <tr> <th>Name</th> <th>PackageFullname</th> </tr> </thead> <tbody> <tr> <td>Microsoft.Windows.OOBENetworkConnectionFlow</td> <td>Microsoft.Windows.OOBENetworkConnectionFlow_10.0.21302.1000_neutral_cw5nlh2txyewy</td> </tr> <tr> <td>Microsoft.Windows.OOBENetworkCaptivePortal</td> <td>Microsoft.Windows.OOBENetworkCaptivePortal_10.0.21302.1000_neutral_cw5nlh2txyewy</td> </tr> <tr> <td>Microsoft.Windows.UndockableKit</td> <td>Microsoft.Windows.UndockableDevKit_10.0.22621.1_neutral_neutral_cw5nlh2txyewy</td> </tr> <tr> <td>Microsoft.UI.Xaml.CBS</td> <td>Microsoft.UI.Xaml.CBS_8.2303.31001.0_x64_8wekyb3d8bwbe</td> </tr> <tr> <td>Microsoft.WindowsAppRuntime.CBS</td> <td>Microsoft.WindowsAppRuntime.CBS_4000.864.2406.0_x64_8wekyb3d8bwbe</td> </tr> <tr> <td>Microsoft.Windows.Client.FileExp</td> <td>Microsoft.Windows.Client.FileExp_10.0.21302.1000_neutral_cw5nlh2txyewy</td> </tr> <tr> <td>Microsoft.Wallet</td> <td>Microsoft.Wallet_2.4.18324.0_x64_8wekyb3d8bwbe</td> </tr> <tr> <td>Microsoft.BioEnrollment</td> <td>Microsoft.BioEnrollment_10.0.19587.1000_neutral_cw5nlh2txyewy</td> </tr> </tbody> </table>	Name	PackageFullname	Microsoft.Windows.OOBENetworkConnectionFlow	Microsoft.Windows.OOBENetworkConnectionFlow_10.0.21302.1000_neutral_cw5nlh2txyewy	Microsoft.Windows.OOBENetworkCaptivePortal	Microsoft.Windows.OOBENetworkCaptivePortal_10.0.21302.1000_neutral_cw5nlh2txyewy	Microsoft.Windows.UndockableKit	Microsoft.Windows.UndockableDevKit_10.0.22621.1_neutral_neutral_cw5nlh2txyewy	Microsoft.UI.Xaml.CBS	Microsoft.UI.Xaml.CBS_8.2303.31001.0_x64_8wekyb3d8bwbe	Microsoft.WindowsAppRuntime.CBS	Microsoft.WindowsAppRuntime.CBS_4000.864.2406.0_x64_8wekyb3d8bwbe	Microsoft.Windows.Client.FileExp	Microsoft.Windows.Client.FileExp_10.0.21302.1000_neutral_cw5nlh2txyewy	Microsoft.Wallet	Microsoft.Wallet_2.4.18324.0_x64_8wekyb3d8bwbe	Microsoft.BioEnrollment	Microsoft.BioEnrollment_10.0.19587.1000_neutral_cw5nlh2txyewy												
Name	PackageFullname																															
Microsoft.Windows.OOBENetworkConnectionFlow	Microsoft.Windows.OOBENetworkConnectionFlow_10.0.21302.1000_neutral_cw5nlh2txyewy																															
Microsoft.Windows.OOBENetworkCaptivePortal	Microsoft.Windows.OOBENetworkCaptivePortal_10.0.21302.1000_neutral_cw5nlh2txyewy																															
Microsoft.Windows.UndockableKit	Microsoft.Windows.UndockableDevKit_10.0.22621.1_neutral_neutral_cw5nlh2txyewy																															
Microsoft.UI.Xaml.CBS	Microsoft.UI.Xaml.CBS_8.2303.31001.0_x64_8wekyb3d8bwbe																															
Microsoft.WindowsAppRuntime.CBS	Microsoft.WindowsAppRuntime.CBS_4000.864.2406.0_x64_8wekyb3d8bwbe																															
Microsoft.Windows.Client.FileExp	Microsoft.Windows.Client.FileExp_10.0.21302.1000_neutral_cw5nlh2txyewy																															
Microsoft.Wallet	Microsoft.Wallet_2.4.18324.0_x64_8wekyb3d8bwbe																															
Microsoft.BioEnrollment	Microsoft.BioEnrollment_10.0.19587.1000_neutral_cw5nlh2txyewy																															
Format-List	get-VM fl *	<pre>PS C:\WINDOWS\system32> #Cette commande nous affiche tout les informations sur les VM qui sont stocker sur le poste. Script lancer par Jack Get-VM fl *</pre> <table border="1"> <thead> <tr> <th>ParentCheckpointId</th> <th>: ac0e67eb-aa41-0014-9de-2e1aa3381fc6</th> </tr> <tr> <th>ParentCheckpointName</th> <th>: Fichier de contrôle automatique - ADDS01 - (19/02/2024 - 13:32:10)</th> </tr> <tr> <th>VMId</th> <th>: ADDS01</th> </tr> <tr> <th>CheckpointFileName</th> <th>: 728d34ef-9335-4c89-ab3a-245cc6a066bc</th> </tr> <tr> <th>ConfigurationLocation</th> <th>: C:\ProgramData\Microsoft\Windows\Hyper-V</th> </tr> <tr> <th>GuestMemorySize</th> <th>: 4096</th> </tr> <tr> <th>SmartPagingFileInUse</th> <th>: False</th> </tr> <tr> <th>SmartPagingFilePath</th> <th>: C:\ProgramData\Microsoft\Windows\Hyper-V</th> </tr> <tr> <th>SnapshotFileLocation</th> <th>: C:\ProgramData\Microsoft\Windows\Hyper-V</th> </tr> <tr> <th>AutomaticStartAction</th> <th>: StartIfRunning</th> </tr> <tr> <th>AutomaticStopAction</th> <th>: Off</th> </tr> <tr> <th>AutomaticStopAction</th> <th>: Save</th> </tr> <tr> <th>AutomaticCriticalErrorAction</th> <th>: Pause</th> </tr> <tr> <th>AutomaticCriticalErrorActionTimeout</th> <th>: 30</th> </tr> <tr> <th>AutomaticCheckpointsEnabled</th> <th>: True</th> </tr> </thead> </table>	ParentCheckpointId	: ac0e67eb-aa41-0014-9de-2e1aa3381fc6	ParentCheckpointName	: Fichier de contrôle automatique - ADDS01 - (19/02/2024 - 13:32:10)	VMId	: ADDS01	CheckpointFileName	: 728d34ef-9335-4c89-ab3a-245cc6a066bc	ConfigurationLocation	: C:\ProgramData\Microsoft\Windows\Hyper-V	GuestMemorySize	: 4096	SmartPagingFileInUse	: False	SmartPagingFilePath	: C:\ProgramData\Microsoft\Windows\Hyper-V	SnapshotFileLocation	: C:\ProgramData\Microsoft\Windows\Hyper-V	AutomaticStartAction	: StartIfRunning	AutomaticStopAction	: Off	AutomaticStopAction	: Save	AutomaticCriticalErrorAction	: Pause	AutomaticCriticalErrorActionTimeout	: 30	AutomaticCheckpointsEnabled	: True
ParentCheckpointId	: ac0e67eb-aa41-0014-9de-2e1aa3381fc6																															
ParentCheckpointName	: Fichier de contrôle automatique - ADDS01 - (19/02/2024 - 13:32:10)																															
VMId	: ADDS01																															
CheckpointFileName	: 728d34ef-9335-4c89-ab3a-245cc6a066bc																															
ConfigurationLocation	: C:\ProgramData\Microsoft\Windows\Hyper-V																															
GuestMemorySize	: 4096																															
SmartPagingFileInUse	: False																															
SmartPagingFilePath	: C:\ProgramData\Microsoft\Windows\Hyper-V																															
SnapshotFileLocation	: C:\ProgramData\Microsoft\Windows\Hyper-V																															
AutomaticStartAction	: StartIfRunning																															
AutomaticStopAction	: Off																															
AutomaticStopAction	: Save																															
AutomaticCriticalErrorAction	: Pause																															
AutomaticCriticalErrorActionTimeout	: 30																															
AutomaticCheckpointsEnabled	: True																															
Opérateurs de comparaison		<p>-eq = equal -ne-gt = not equal-greater than -lt-le = less than-lower-equal -ge = greater equal -like= comme -notlike = pas comme -in= dans -notin= pas dedans - contains = contient -notcontains = ne contient pas -is = est -isnot = n'est pas</p>																														

Where-Object	Get-Service Where-object	<pre>PS C:\WINDOWS\system32> #Cette commande nous affiche tout les processus qui tournent en arrière plan. Script lancer par Jack Get-Service Where-object -Property status -EQ 'Running' Status Name DisplayName ---- ---- ----------- Running AdobeARMservice Adobe Acrobat Update Service Running agent_ovpnconnect OpenVPN Agent_ovpnconnect Running AppPrint Informations d'impression Running AppXSvc Service de déploiement AppX (AppXSVC) Running AudioEndpointB... Générateur de points de terminaison... Running Audiosrv Audio Windows Running BDESVC Service de chiffrement de lecteur B... Running BFE Moteur de filtrage de base Running BluetoothUserService_4f33a BluetoothUserService_4f33a Running BluetoothUserService_4f33b Service de support des utilisateurs... Running BrEndpointSvc HP Sure Click Endpoint Service Running BrokerInfrast... Service d'infrastructure des tâches... Running Browser Explorateur d'ordinateurs</pre>
Opérateurs de redirection	>>	<pre>PS C:\WINDOWS\system32> #Cette commande nous amène tout les éléments dans fichier.txt. Script lancer par Jack Get-process >> .\Services.txt</pre>
Enumeration	Enum	
Format-Table	Format-table	<pre>PS C:\WINDOWS\system32> #Cette commande nous affiche les VM en table. Script lancer par Jack Get-VM Format-Table -AutoSize Name State CPUUsage (%) MemoryAssigned(M) Uptime Status Version ---- ---- ----- ----- ----- ----- ADD501 Running 0 3386 02:50:20.318000 Fonctionnement normal 11.0 ADRODC Running 0 3124 02:50:19.781000 Fonctionnement normal 11.0 SVRFILES Running 0 2246 02:50:19.676000 Fonctionnement normal 11.0</pre>

TP powershell 3

While :

```
1 $i = 0
2 $j = 0..99
3 while($i -lt $j.Length){
4     $j[$i]
5     $i++
6 }
7
8 Write-Host $i
```

Ce script ajoute +1 entre le 0 et 99 avec la boucle while

Do While :

```
Clear-Host
$i = 0
$j = Read-Host "Entrez un nombre"

Write-Host "Vous avez entrer le nombre $j le compte se fait de $i a $j tant que $i est plus petit que $j"
do {
    "La valeur est : $i/$j"
    $i++
} while ($i -le $j)

Write-Host "Fin de la boucle :)"
```

```
Entrez un nombre: 12
Vous avez entrer le nombre 12 le compte se fait de 0 a 12 tant que 0 est plus petit ou égal à 12
La valeur est : 0/12
La valeur est : 1/12
La valeur est : 2/12
La valeur est : 3/12
La valeur est : 4/12
La valeur est : 5/12
La valeur est : 6/12
La valeur est : 7/12
La valeur est : 8/12
La valeur est : 9/12
La valeur est : 10/12
La valeur est : 11/12
La valeur est : 12/12
Fin de la boucle :)
```

Cette commande ajoute +1 jusqu'au nombre choisi. Avec la commande do while.

Do Until :

```
Entrez un nombre pour faire le décompte: 12
i = 1
i = 2
i = 3
i = 4
i = 5
i = 6
i = 7
i = 8
i = 9
i = 10
i = 11
i = 12
La boucle est arrivée à 12, le décompte est terminé !
PS C:\Users\Jack>
```

Ce script ajoute +1 jusqu'aux nombre qu'on a choisi.

For

```
clear For ($i=0; $i -lt 10; $i++)  
{ Write-host "Ligne N° $i/10 " -ForegroundColor red -BackgroundColor white }
```

```
Ligne N° 0/10  
Ligne N° 1/10  
Ligne N° 2/10  
Ligne N° 3/10  
Ligne N° 4/10  
Ligne N° 5/10  
Ligne N° 6/10  
Ligne N° 7/10  
Ligne N° 8/10  
Ligne N° 9/10  
PS C:\Users\Jack>
```

Ce script nous choisi 10 ligne et surligne les ligne choisi.

For Each

```
Clear  
$fruits= "Pomme", "Orange", "Poire", "Peche", "Abricot"  
foreach ($fruit in $fruits)  
{  
    echo $fruit  
}
```

```
Pomme  
Orange  
Poire  
Peche  
Abricot  
PS C:\Users\Jack>
```

Ce script nous imprime chaque fruit qui est dans la liste \$fruit

Exemple if, Elseif, Then

```
Clear-Host  
$fruit = Read-Host "Entrez le nom d'un fruit"
```

```
if ($fruit -eq "Banane") {  
    Write-Host "C'est très bon la banane !"  
}  
elseif ($fruit -eq "Poire") {  
    Write-Host "C'est encore meilleur une poire !"  
}  
elseif ($fruit -eq "Kiwi") {
```

```

        Write-Host "C'est bon le kiwi !"
    }
elseif ($fruit -eq "Pêche") {
    Write-Host "C'est bon la pêche !"
}
elseif ($fruit -eq "Abricot") {
    Write-Host "C'est bon l'abricot !"
}
else {
    Write-Host "Ce fruit $fruit n'est pas dans la liste !"
}

```

```

Entrez le nom d'un fruit: banane
C'est très bon la banane !
PS C:\Users\Jack>

```

Ce script nous demande de choisir un fruit dans la liste et répond c'est très bon et si le fruit n'est pas dans la liste il répond le fruit n'est pas dans la liste.

Switch

```

$nombre = Read-Host -Prompt "Entrez un nombre entre 1 et 5"
switch ($nombre) {
    1 { Write-Host "Bravo, Vous avez entré le Nombre 1" ; Break }
    3 { Write-Host "Bravo, Vous avez entré le Nombre 3" ; Break }
    4 { Write-Host "Bravo, Vous avez entré le Nombre 4" ; Break }
    5 { Write-Host "Bravo, Vous avez entré le Nombre 5" ; Break }
    Default { Write-Host "Vous avez entré un nombre incorrect : $nombre, qui n'est pas entre 1 et 5 !" }
}
Entrez un nombre entre 1 et 5: 4
Bravo, Vous avez entré le Nombre 4
PS C:\Users\Jack>

```

Ce script nous demande de choisir un nombre entre 1 et 5 ensuite il défile la liste donné jusqu'à qu'il trouve notre nombre entrée

Fonctions :

```

PS C:\Users\Jack>
function Bonjour {
    $date = Get-Date
    Write-Host "Bonjour, nous sommes le $date"
}

Bonjour
Bonjour, nous sommes le 04/08/2024 14:22:21
PS C:\Users\Jack>

```

Ce script récupère la date d'aujourd'hui et nous l'imprime lorsqu'il est appellé.

Les fonctions avec Arguments Dans Powershell :

```

Bonjour Jack Williams, nous sommes le 04/08/2024 14:35:00
PS C:\Users\Jack>
Clear
function Bonjour
{
    $date = Get-Date
    Write-Host "Bonjour $args, nous sommes le $date"
}
Bonjour "Jack Williams"

```

Ce script est exactement comme le fonction des dates mais nous ajoute un argument « \$arg » pour ajouter notre nom.

Exemple des paramètres:

```
Titre du message: ceci est un titre de message
Un message pour vous : Bonjour a tous, ceci est un contenu de message !
PS C:\Users\Jack>

clear

function Show-Popup
{
Param ($message, $titre)
Write-Host "Titre du message:" $titre
Write-Host "Un message pour vous :" $message
}

Show-Popup -message "Bonjour a tous, ceci est un contenu de message !" -titre "ceci est un titre de m
```

Ce script nous imprime des titre avec des fonctions appeler \$titre

Script que j'ai créer qui te donne l'option d'ouvrir des applications Office 365 :

```
1 function Open-OfficeApp {
2     param (
3         [Parameter(Mandatory = $true)]
4         [string]$appName
5     )
6
7     switch ($appName) {
8         "Excel" { Start-Process "Excel" }
9         "Access" { Start-Process "msaccess" }
10        "Publisher" { Start-Process "mspub" }
11        "OneNote" { Start-Process "onenote" }
12        "Outlook" { Start-Process "outlook" }
13        "PowerPoint" { Start-Process "powerpnt" }
14        "Word" { Start-Process "winword" }
15        default { Write-Host "Application pas reconnu." }
16    }
17 }
18
19 $appToOpen = Read-Host "Quelle application Office 365 voulais vous ouvrir? (Excel, Access, Publ
20
21 Open-OfficeApp -appName $appToOpen
```