

Fiche 13 : Blocs d'instructions Begin, Process, End et les fonctions

1. Les blocs d'instructions Begin, Process, End

- **Begin** : Ce bloc d'instructions est exécuté une seule fois avant la transmission du premier objet.
- **Process** : Ce bloc d'instructions est exécuté pour chaque objet transmis.
- **End** : Ce bloc d'instructions est exécuté une seule fois après la transmission du dernier objet.

2. Utilisation dans un pipeline

Exemple 1, Compter le nombre de noms :

```
Windows PowerShell V2 (CTP2)
PS F:\> "Pierre","Daniel","Mickael" | ForEach-Object -begin {<i>$i=0</i>} -process {<i>$i++</i>} -End {write-Host "Résultat : <i>$i</i>"}
Résultat : 3
```

Exemple 2, script qui comptabilise le total de l'espace libre de tous les lecteurs d'un poste :

```
1 Get-WmiObject Win32_LogicalDisk -Filter DriveType=3 -ComputerName localhost |
2 ForEach-Object -Begin {$total=0} -process {$total=$total+[math]::Round($_.FreeSpace / 1GB,2)} -End {"Total libre : $total"}
Ici, la ligne 2 récupère les objets (lecteurs disque de localhost) transmis par la ligne 1 via le pipeline.
```

3. Utilisation dans une fonction

Exemple 1, reprise de l'exemple 1 du §2, mais avec une fonction appelée à la fin du script :

```
1 function bonjour {
2 BEGIN {
3 Write-Host "Présentation des personnes : "
4 $i=0
5 }
6 PROCESS{
7 Write-Host "Bonjour $_"
8 $i++
9 }
10 END{
11
12 Write-Host "bonjour à $i personne(s)"
13 }
14 }
15
16 "Pierre","Daniel","Mickael" | bonjour
```

```
Console PowerShell
Présentation des personnes :
Bonjour Pierre
Bonjour Daniel
Bonjour Mickael
bonjour à 3 personne(s)
```

Exemple 2, reprise de l'exemple 2 du §2, mais avec une fonction appelée à la fin du script :

```
1 function espace libre {
2 BEGIN {
3 $total=0
4 Write-Host "Espace libre : "
5 }
6 PROCESS{
7 $total=$total+[Math]::Round($_.freespace/1GB,2)
8 Write-Host ($_.DeviceID+[math]::Round($_.FreeSpace / 1GB,2))
9 }
10 END{
11 Write-Host "Total libre : $total"
12 }
13 }
14
15 Get-WmiObject Win32_LogicalDisk -Filter DriveType=3 -ComputerName localhost | espace libre
```

```
Console PowerShell
Espace libre :
C:2.02
F:4.54
G:8.69
M:29.83
Total libre : 45.08
```

4. Fonction qui retourne un objet

Exemple, toujours la fonction qui comptabilise l'espace libre de tous les lecteurs d'un poste :

```
1 function espace libre {
2 BEGIN {
3 $obj=New-Object PSObject
4 $total=0
5 }
6 PROCESS{
7 $total=$total+[Math]::Round($_.freespace/1GB,2)
8 $libre=[math]::Round($_.FreeSpace / 1GB,2)
9 $obj | add-member NoteProperty $_.DeviceID ($libre)
10 }
11 END{
12 $obj | add-member NoteProperty total ($total)
13 Write-Output $obj
14 }
15 }
```

Déclaration d'un nouvel objet..

Ajoute une propriété à l'objet et lui affecte une valeur.

Instruction qui retourne un objet.

Maintenant la fonction retourne un objet qui peut être traité via un pipeline :

Résultat affiché dans un tableau :

... | `espacelibre` | `Format-Table`

Console PowerShell									
				C:	F:	G:	M:	total	
				--	--	--	--	-----	
				2,02	4,54	8,69	29,83	45,08	

Résultat affiché dans une liste :

... | `espacelibre` | `Format-List`

```
Console PowerShell
C:      : 2,02
F:      : 4,54
G:      : 8,69
M:      : 29,83
total   : 45,08
```

Affiche que la propriété total :

... | `espacelibre` | `Select-Object -Property total`