

LES VARIABLES

LES VARIABLES...

- PowerShell n'est pas un langage de programmation fortement typé, par conséquent il n'y a pas besoin de déclarer ou de spécifier le type d'une variable avant de l'utiliser.

- Exemples:

```
$variable1 = 15
```

```
$variable2 = 'A'
```

```
$variable3 = 'Bonjour'
```

- Il est possible de pouvoir noter une variable avec des caractères spéciaux, pour ce faire, il faut utiliser des accolades.

```
${B@tm@n.le-plus#fort} = 1989
```

DÉTERMINER LE TYPE D'UNE VARIABLE

- Afin de connaître le type de la variable que PowerShell a imposé, il faut utiliser la fonction suivante : `GetType()`

```
4 $variable1 = 15
5
6 $variable1.GetType()
7
```

```
PS C:\Users\Lago>
```

```
$variable1 = 15
```

```
$variable1.GetType()
```

IsPublic	IsSerial	Name	BaseType
-----	-----	----	-----
True	True	Int32	System.ValueType

ACCÉDER AU CONTENU DE LA VARIABLE

- Pour accéder au contenu de la variable, il suffit de noter le nom de la variable dans le script

```
4  $variable1 = 17
5
6  $variable1
7
8  $variable1 = $variable1 + 12
9
10 $variable1
11
12
```

PS C:\Users\Lago>

```
$variable1 = 17
```

```
$variable1
```

```
$variable1 = $variable1 + 12
```

```
$variable1
```

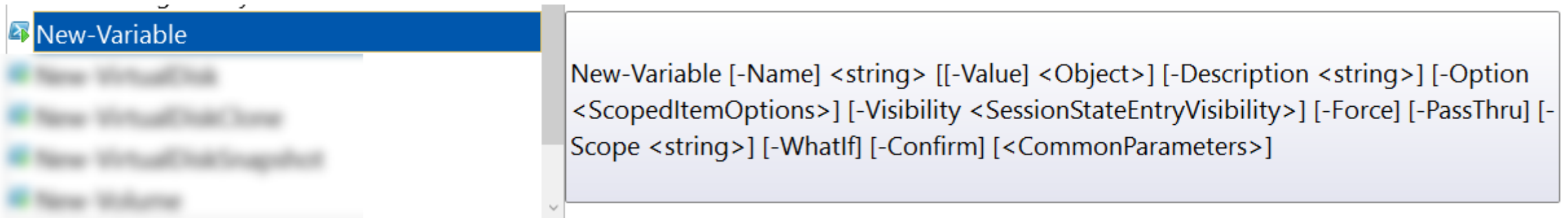
```
17
```

```
29
```

PS C:\Users\Lago>

LES CONSTANTES

- Les constantes sont des variables qui ne peuvent pas changer de valeur au cours du temps.
- Dans PowerShell, afin de s'assurer de rendre une variable **immodifiable**, il faut utiliser la fonction suivante :



- Exemple :

```
New-Variable -Name PI -Value 3.14 -Option Constant
```

- PS : La bienséance veut que les constantes soient déclarées en **MAJUSCULE** et les variables en **minuscule**

TYPES DE DONNÉES...

- Les différents types possibles en PowerShell sont les suivants:

Type	Description
Int, Int32, Int64	Ce sont des entiers entre -2'147'483'648 et 2'147'483'647
Double	Ce sont les nombres à virgule aussi grands que 10^{308} avec une précision de 15 chiffres
Char	Représente n'importe lequel des 65'536 caractères Unicode
String	Représente une chaîne sous forme de suite de caractères Unicode
Boolean	C'est une valeur booléenne (true ou false)
Array	C'est un tableau contenant plusieurs valeurs, chaque valeur peut être d'un type différent
Object, PSObject, PSCustomObject	Représente un objet qui peut être personnalisé

IMPOSER LE TYPE DE LA VARIABLE

- Il est quand même possible d'imposer le type d'une variable en indiquant le type voulu entre [] avant la variable

```
4 [int]$variable = Read-Host 'Entrez un nombre'
5
```

```
PS C:\Users\Lago>
```

```
[int]$variable = Read-Host 'Entrez un nombre'
```

```
Entrez un nombre : cinq
```

```
Impossible de convertir la valeur « cinq » en type « System.Int32 ». Erreur : « Le format de la chaîne d'entrée est incorrect. »
```

```
Au caractère Ligne:4 : 1
```

```
+ [int]$variable = Read-Host 'Entrez un nombre'
```

```
+ ~~~~~
```

```
+ CategoryInfo          : MetadataError: (:) [], ArgumentTransformationMetadataException
```

```
+ FullyQualifiedErrorId : RuntimeException
```

```
PS C:\Users\Lago>
```