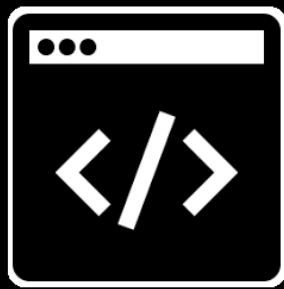


# Scripting Batch



## Sommaire

### Les Bases

- Les basiques de l'invite de commande: .....	3
- Afficher du texte: .....	3
- Les variables: .....	4
- Saisie utilisateur: .....	4
- Les opérateurs: .....	6
- Les boucles: .....	6
→ For: .....	6
→ Goto : .....	7
- Les conditions: .....	8
- De la couleur avec Batbox / spritebox : .....	8

### Créer son application

- Les commandes pratiques : .....	9
- Explication d'un bout de code du projet : .....	11
→ Fonction ajout d'utilisateur dans l'AD: .....	12

## Les Bases

### Les basiques de l'invite de commande

Commandes	Détail
cd	Changer de répertoire
cd ..	Retourner au répertoire précédent
cd /	Aller au répertoire racine
dir	Afficher le contenu du répertoire
dir * [nom de doc ou de répertoire]*	Rechercher tout ce les docs ou répertoire qui contiennent ce nom ou caractère
md ou mkdir	Créer un répertoire
rd [nom de répertoire]	Supprimer un répertoire
tree	Afficher l'arborescence
md ou mkdir	Créer un répertoire
COPY [NomDeFichier] [Destination]	Copier des fichiers
cls	effacer le texte affiché

### Afficher du texte

afficher du texte avec la commande echo :

```
INPUT
c:\> echo bonjour

OUTPUT
Bonjour
```

cette commande peut servir à plusieurs choses

- Afficher du texte : `echo [texte]`
- faire un retour à la ligne : `echo.`
- masquer l'affichage des commandes : `@echo off`
- écrire du texte dans un document : `echo bonjour > mondoc.txt`  
ou : `echo bonjour >> mondoc.txt`

## Les variables

Pour définir une variable en batch contrairement à d'autres langages il suffit de décrire simplement son nom sans symbole avant comme en PHP par exemple ou on utilise le '\$'

syntaxe : `set mavariable=valeur`

Pour l'affichage par contre on utilise le `%mavariable%` avec la commande `echo`.

Exemple:

```
INPUT
c:\> set var=bonjour
c:\> echo %var%
```

```
OUTPUT
Bonjour
```

**Set peut être utilisé avec des options comme :**

pour définir un calcul, on utilisera `set /A`

```
set /A num=num + 1
```

Syntaxe: `set /A [variable] = [calcul]`

## Saisie utilisateur

### Set

pour définir une variable par l'utilisateur `set /P`

```
set /P age= quel est ton âge ?
```

Syntaxe: `set /P [variable]= [indication de la valeur à entrer]`

## Choice



/n → définit les choix possibles par l'utilisateur (on) ou (123456789)  
/t → définit un temps d'attente de réponse de l'utilisateur en secondes  
/d → définit un choix par défaut appliqué à la fin du temps définit par /t

```
if %errorlevel%==1 [commande]
```

Choice permet de récupérer le choix de l'utilisateur, on peut en faire un menu on place toujours les choix par ordre décroissant ce qui donne pour 3 choix

```
@echo off
echo Qu'as-tu mangé ce midi ?
echo 1 : chocolat
echo 2 : cookie
echo 3 : rien
choice /c 123 /n

if %errorlevel%==3 [commande]      correspond au choix rien
if %errorlevel%==2 [commande]      correspond au choix
chocolat
if %errorlevel%==1 [commande]      correspond au choix cookie
```

On utilise aussi choice pour répondre par oui ou non à une question (seul o et n seront des entrées valables)

```
echo Aimes-tu le chocolat ?
echo o : oui
echo n : non
choice /c on /n

if %errorlevel%==2 commande      correspond au choix non
if %errorlevel%==1 commande      correspond au choix oui
```

## Les opérateurs

Les opérateurs en batch sont les mêmes que dans beaucoup de langages comme le C par exemple

opérateur	signe	autre syntaxe
addition	+	
soustraction	-	
multiplication	*	
division	/	
modulo	%	
égal	==	equ
supérieur stricte	>	gtr
supérieur ou égal	>=	geq
inférieur stricte	<	lss
inférieur ou égal	<=	leq
différent de	!=	neq
incrémentement	+=	set /A [var] += [var ou nombre]
décrémentement	-=	set /A [var] -= [var ou nombre]

## Les boucles

### Goto

goto n'est pas vraiment une boucle mais peut être utilisé pour en créer, cette commande permet de se rendre à un endroit précis dans le code sans effectuer les lignes qu'il saute.

```
@echo off
:debut
echo boucle
goto debut
```

```
echo on ne verra jamais ce message
```

## For



/f → chercher dans un document (f pour file )  
/l → boucler une série de valeur  
Tokens → décomposer le résultat en deux blocs distincts  
delims → délimitation des données exemple "delims= :"

syntaxe : For /f ["options"] %%paramètre IN ( doc.txt ) DO ( commande )

chercher dans un fichier

```
for %%f in (doc.txt) do (  
[commande]  
)
```

Exemple créer un compte à rebour

```
@echo off  
for %%v in (5 4 3 2 1) do (  
cls  
echo Debut du compte à rebour  
echo %%v  
)  
echo Fin du compte à rebour
```

exemple d'utilisation pour rechercher dans un document "ip.txt"

```
for /f "tokens=1-2 delims= :" %%a in (ip.txt) do ()
```

document ip.tkt

```
ip routeur1 : 172.16.10.5  
ip routeur2 : 172.16.20.6
```

## Les conditions

syntaxe :

```
if [condition] == [valeur] (  
  commandes..  
) else (  
  commandes..  
)
```

if peut aussi être utilisé pour récupérer une saisie utilisateur notamment avec choice comme vue précédemment

```
if %errorlevel%==1 commande
```

Vérifier si une variable existe

```
if exist  
if no exist
```

## De la couleur avec Batbox / spritebox

La couleur en batch est initialement apportée par la commande “color” mais pour une application on se retrouve vite limité notamment pour mettre certains mots ou caractères d’une couleur et pas le reste.

la commande color change la couleur de toute la page ou de tout le texte affiché sur la page

syntaxe :

```
color fond/texte  
exemple :  
color a2
```

pour remettre les couleurs par défaut utiliser :



```
color f
```

Batbox est une commande externe créé par DarkBatcher disponible sur le site suivant :  
<https://batch.xoo.it/t2243-Commande-Externe-Batbox-v1-0.htm>

#### quelques commandes batbox :

l'équivalent du echo → /d

```
batbox /d "text"
```

définir une position pour le texte dans la fenêtre. → /g

```
batbox /g x y          colonnes lignes
```

définir la couleur du texte → c au format 0xf [lettre ou chiffre de la couleur]

```
batbox /g 15 5 /c 0xfa /d texte affiché en vert
```

## Créer son application

### Les commandes pratiques

cacher l'exécution de n'importe quelle commande, mettre '>nul'

```
echo ce message ne se verra pas >nul  
pause >nul
```

définir un titre pour la fenêtre

```
title [letitre]
```

créer un document texte et stocker du texte, à savoir le '>' écrase le texte dans le document si on exécute la commande, on ne peut donc pas ajouter de texte au document mais seulement le créer avec ce texte dedans

```
set texte= bonjour, je m'appelle florent
```

```
echo %texte% > cv.txt
```

Pour ajouter du texte il faudrait utiliser le '>>' qui cette fois ci ajoute le texte à la suite dans le document.

```
set /P texte2= entrez el texte à ajouter dans ce doc.  
echo %texte2% >> cv.txt
```

Définir la taille de la fenêtre

```
mode con cols=50 lines=20
```

faire attendre 10 secondes

```
timeout /t 10 /nobreak
```

faire attendre tant que l'utilisateur n'appuie pas sur une touche

```
pause
```

définir un nombre aléatoire entre 0 et 9

```
set /a x=%random%%10
```

utiliser une commande powershell dans un .bat, à savoir que les fichiers powershell sont en extension .ps1

```
echo ajout d'utilisateur dans l'AD  
set commande= New-ADUser etc..  
echo %commande% > script.ps1 >nul  
powershell c:\chemin vers le fichier .ps1 >nul  
del c:\chemin vers le fichier .ps1 >nul  
cls  
echo script terminé...  
pause >nul  
exit
```

## Explication d'un bout de code du projet

```
:AD2
cls
echo ajout d'un utilisateur
set /p ou= nom de l'uo :
set /p prenom= prenom :
set /p nom= nom :
set ouverture=%prenom:~0,1%%nom%@m21.lan
set nomcomplet=%prenom% %nom%
set displayname=%nom% %prenom%
set sam=%prenom%.%nom%
set AD=New-ADUser -Name "%nomcomplet%" -DisplayName "%displayname%" -GivenName "%prenom%" -Surname "%nom%"
echo %AD% > ADutil.ps1
powershell c:\users\administrateur\desktop\ADutil.ps1
del c:\users\administrateur\desktop\ADutil.ps1
echo.
echo configuration terminée ...
pause>nul
cls
goto menu
```

Cet extrait traite de l'ajout d'un utilisateur dans l'active directory

ce script est pensé de sorte à ce qu'il y ait le minimum d'information à rentrer pour créer un utilisateur, ainsi on ne collecte que le nom de l'uo de destination, le prénom et le nom

au début on retrouve la fonction AD2 qui fait référence à un goto du menu

```
set ouverture=%prenom:~0,1%%nom%@m21.lan
```

on définit la variable ouverture qui va nous servir à créer le nom d'utilisateur automatiquement à partir du nom et prénom en prenant la 1ere lettre du prénom (%prenom:~0,1%) et en concaténant le nom

ce qui donne pour florent buteux un nom d'utilisateur : fbuteux

une fois les informations collectées et les valeurs assignées aux différentes variables, on crée la commande powershell que l'on va stocker dans une variable AD

```
set AD=New-ADUser -Name "%nomcomplet%" -DisplayName "%displayname%" -GivenName "%prenom%" -Surname "%nom%" -SamAccountName "%sam%" -UserPrincipalName "%ouverture%" -EmailAddress "" -Path "OU=%ou%,DC=M2L,DC=LAN" -AccountPassword (ConvertTo-SecureString "ServESP201808" -AsPlainText -Force) -ChangePasswordAtLogon $false -CannotChangePassword $true
```

```
-PasswordNeverExpires    $true    -Enabled    $true    -Server  
SRVligue.m21.lan
```

voir doc script powershell pour le détail de la commande.

Une fois la requête assemblée on crée le fichier ADutil.ps1 qui est stocké dans le même répertoire que le script et qui est supprimé quelques secondes après exécution de la commande powershell.

```
echo %AD% > ADutil.ps1  
powershell c:\users\administrateur\desktop\ADutil.ps1  
del c:\users\administrateur\desktop\ADutil.ps1
```

une fois le script terminé on retourne au menu de démarrage avec goto menu, en effaçant auparavant le texte affiché sur l'écran avec la commande cls (clear screen)

## Sources

- <https://ss64.com/>
- [https://www.tutorialspoint.com/batch\\_script/](https://www.tutorialspoint.com/batch_script/)
- <https://learn.microsoft.com/fr-fr/powershell/scripting/overview?view=powershell-7.3>
- <https://www.youtube.com/@FRSpeedCookie>