

Automatiser des procédures à l'aide de scripts

Module 122



Table des matières

- ▶ Parcourir les dossiers
- ▶ Test si c'est un dossier
- ▶ Parcourir les fichiers
- ▶ Test si c'est un fichier
- ▶ Bash les arguments
- ▶ Exercices

Parcourir un dossier :

- ▶ Il existe plusieurs solutions pour parcourir des répertoires dans un dossier, nous allons voir une méthode simple avec une boucle for :
 - ▶ **for d in */ ; do**
 - ▶ Votre code ici d correspond à la valeur de chaque dossier dans le dossier courant
 - ▶ **done**

```
##on parcourt les tous les dossiers
for d in */ ; do
    #affiche le nom de chaque dossier
    echo "dossier : "$d
#fin de la boucle
done
```

Test si c'est un dossier :

- ▶ Pour tester si c'est un dossier, on utilise une condition avec un `if` et on utilise l'option `-d`.
 - ▶ Il ne faut pas oublier de mettre le mot clé **then** après la condition et **fi**; à la fin de la condition.

```
#Test si c'est un dossier avec $d comme valeur de dossier
if [ -d "$d" ]
then
    # le code vient ici si c'est un dossier
fi;
```

Parcourir les fichiers :

- ▶ Pour parcourir les fichiers dans un dossier, nous pouvons utiliser la même stratégie que avec les dossier à l'exception d'enlever le / ce qui va avoir comme but de parcourir tous les éléments contenu dans le dossier :
 - ▶ **for f in * ; do**
 - ▶ Votre code ici f correspond à la valeur de chaque fichier dans le dossier courant
 - ▶ **done**

```
# on parcourt les fichiers dans le répertoire courant
for f in *; do
    # on affiche le nom du fichier
    echo "Nom du fichier : " $f
done
```

Test si c'est un fichier :

- Pour tester si c'est un fichier, on utilise une condition avec un `if` et on utilise l'option `-f`.
 - Il ne faut pas oublier de mettre le mot clé **then** après la condition et **fi** à la fin de la condition.

```
#test si f est un fichier avec l'option -f
if [ -f "$f" ]
then
    #affiche le nom de chaque fichier
    echo "file : "$f
fi
```

Bash les arguments :

Bash gère nativement très facilement les arguments (paramètres envoyés à la commande).

Ces paramètres ou arguments sont simplement stockés dans \$numéroArguments.

Exemple :

commande **arg1** **arg2** **arg3** **arg4** seront stockés dans **\$1**, **\$2**, **\$3**, **\$4**

Bash les arguments :

```
GNU nano 4.8 arguments.sh
#!/bin/bash
echo "Ceci montre le premier arg : "$1
echo "Ici le deuxième : "$2
echo "Troisième : "$3
echo "Quatrième et ainsi de suite : "$4
```

```
youn@LHOSTE:~/Scripts$ ./arguments.sh param1 leDeuxCEstEZ pomme rouge
Ceci montre le premier arg : param1
Ici le deuxième : leDeuxCEstEZ
Troisième : pomme
Quatrième et ainsi de suite : rouge
```


Exercice 1 :

Télécharger et dézipper le exe4.zip. Le but de l'exercice, va être de créer un script qui supprime tous les fichiers dans chaque dossier.

[illegible]

Questions :

