

TP LINUX - Scripts Shell -

Exercice 1

Créer un script shell nommé "**change**" qui affichera la date de dernière modification d'un fichier puis la modifiera avec l'heure actuelle et enfin réaffichera la date de dernière modification du fichier.

Cette procédure acceptera 1 paramètre qui sera le nom du fichier.

Lorsque vous exécuterez "**change mon_fic**", le 10 Janvier à 11 heures 00, vous obtiendrez le résultat:

avant : -rw-rw-r-- 1 user group 40 Feb 3 2001 mon_fic

après : -r--r--r-- 1 user group 40 Jan 10 11:00 mon_fic

Exercice 2

Créer un script shell nommé "**nombreJours**" qui affichera le nombre de jours du mois courant.

"**nombreJours**" affichera pour février 2010 le message "**28 jours en février 2010**".

Exercice 3

Créer un script shell réalisant la création d'un répertoire "Exo3" contenant 10 fichiers nommés "Un" à "Dix". Chaque fichier contient une seule ligne:

- "Un" contient "Première ligne"
 - "Deux" contient "Deuxième ligne"
 - ...
 - "Dix" contient "Dixième ligne"
- i. *Vérifier que le répertoire à créer n'existe pas déjà ; auquel cas il ne sera pas recréé mais les fichiers si.*

Exercice 4

Créer un script shell qui réalise les opérations suivantes:

- Création sous votre répertoire "TP2" d'un sous répertoire nommé "annéemoisjour" (20110110 pour le 10 Janvier 2011).
 - Copie des fichiers de "Exo3" sous ce répertoire puis effacement de ces mêmes fichiers de "Exo3".
 - Création de deux fichiers sous le répertoire d'accueil (\$HOME) de la personne qui a lancé le shell:
 - un fichier nommé "Gros_fichier.pid_du_shell" dans lequel se trouvera le contenu concaténé des fichiers traités
 - un fichier nommé "Nom_du_script.pid_du_shell" dans lequel se trouvera le nom des fichiers traités.
- i. *Vérifier que le répertoire à créer n'existe pas déjà.*

Exercice 5

- Créer un script permettant d'afficher la liste des fichiers du répertoire /etc accessibles en lecture.
- Créer un script permettant d'afficher la liste des fichiers du répertoire /etc accessibles en écriture.

Exercice 6

Créer un script nommé "**table**" permettant d'afficher des tables de multiplication.

"**table 5 10**" aura pour résultat l'affichage:

```
0 x 5 = 0
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50
```

TP LINUX
Corrigé

Exercice 1

```
AVANT=$(ls -l $1)
touch $1
APRES=$(ls -l $1)
echo "Avant : "$AVANT
echo "Après : "$APRES
```

Exercice 2

```
MOIS=$(date +"%m")
case $MOIS in
    01) NOMF="janvier";;
    02) NOMF="février";;
    03) NOMF="mars";;
    04) NOMF="avril";;
    05) NOMF="mai";;
    06) NOMF="juin";;
    07) NOMF="juillet";;
    08) NOMF="août";;
    09) NOMF="septembre";;
    10) NOMF="octobre";;
    11) NOMF="novembre";;
    12) NOMF="décembre";;
esac
ANNEE=$(date +"%Y")
case $MOIS in
    01|03|05|07|08|10|12)  nb=31 ;;
    04|06|09|11)          nb=30 ;;
    02)
        QUATRE=$(expr $ANNEE % 4)
        CENT=$(expr $ANNEE % 100)
        QUATCE=$(expr $ANNEE % 400)

        if [ $QUATCE == 0 -o $QUATRE == 0 -a $CENT -ne 0 ]
        then
            nb=29
        else
            nb=28
        fi
        ;;
esac
echo $nb jours en $NOMF $ANNEE
```

Exercice 3

```
test -e Exo3
if [ $? -eq 0 ]
then
    if [ ! -d Exo3 ]
    then
        rm -f Exo3; mkdir Exo3
    fi
else
    mkdir Exo3
fi

echo "Premiere ligne" >Exo3/Un
echo "Deuxieme ligne" >Exo3/Deux
echo "Troisieme ligne" >Exo3/Trois
echo "Quatrieme ligne" >Exo3/Quatre
echo "Cinquieme ligne" >Exo3/Cinq
echo "Sixieme ligne" >Exo3/Six
echo "Septieme ligne" >Exo3/Sept
echo "Huitieme ligne" >Exo3/Huit
echo "Neuvieme ligne" >Exo3/Neuf
echo "Dixieme ligne" >Exo3/Dix
```

Exercice 4

```
REP=$(date +"%Y%m%d")
test -e $REP
if [ $? != 0 ]
then
    mkdir $REP
fi
cp Exo3/* $REP
rm Exo3/*
for file in $REP/*
do
    cat $file >> ~/Gros_fichier.$$
    echo $file >> ~/$0.$$
done
```

Exercice 5

<u>Script1</u>	<u>script2</u>
for FILE in /etc/*	for FILE in /etc/*
do	do
if [-r \$FILE]	if test -w \$FILE

```
    then
    echo $FILE
fi
done
```

```
    then
    echo $FILE
fi
done
```

Exercise 6

```
for V in $(seq 0 $2)
do
echo $V" x "$1" = "$(expr $V "*" $1 )
done
```