

Exercice 1

Copier le script dans le fichier script1.sh :

```
#!/bin/bash
# commentaires
echo Mon premier script bash
echo Liste des fichiers :
ls -la
pwd
whoami
date
```

Ajouter la permission d'exécution au fichier "script1.sh" et exécuter le script. Que fait ce script?

Exercice 2

Ecrire un script qui prend en entrée le nom d'un fichier/répertoire, et qui retourne le résultat du test suivant :

```
Le fichier XYZ existe
Le fichier XYZ a la permission de lecture
Le fichier XYZ a la permission de modification
```

Exercice 3

Copier le script suivant dans le fichier script3.sh et exécuter le script. Que fait ce script ?

```
$ cat script3.sh
#!/bin/bash
# menu interface to simple commands
echo -e "\n COMMAND MENU\n"
echo " a. Current date and time"
echo " b. Users currently logged in"
echo " c. Name of the working directory"
echo -e " d. Contents of the working directory\n"
echo -n "Enter a, b, c, or d: "
read answer
echo

case "$answer" in
a)
    date
    ;;
b)
    who
    ;;
c)
    pwd
    ;;
d)
    ls
    ;;
*)
    echo "There is no selection: $answer"
    ;;
esac
```

Exercice 4

Ecrire un script qui affiche le menu suivant, et ensuite demande à l'utilisateur de rentrer un nombre (de 1 à 4), et qui affiche le résultat de l'exécution de la commande entre les parenthèses selon l'entrée de l'utilisateur :

1. Lister le contenu du répertoire (\$ls -l)
2. Lister les processus en cours d'exécution (\$ps aux)
3. La date d'aujourd'hui (\$date)
4. Quitter

Exercice 5

Ecrire un script qui prend comme argument un fichier et transforme toutes les lettres minuscules en majuscules en dans son contenu.

Exercice 6

Ecrire un script prend en argument le fichier */etc/passwd* et qui affiche uniquement le login et le Shell par défaut pour l'utilisateur courant uniquement.

Exercice 7

Ecrire un script qui prend le nom de deux fichiers (*filesrc* et *filedst*) comme arguments, et qui copie du fichier *filesrc* vers le fichier destination *filedst*.

Ajouter une fonction **Usage**, qui affiche une aide sur l'utilisation du script en cas où l'utilisateur rentre plus/moins que deux arguments.

Exercice 8

Ecrire un script qui prend en argument le nom d'un/plusieurs fichier(s) et qui affiche le menu ci-dessous :

Voulez-vous créer une archive

- 1) avec l'extension .tar.gz
- 2) avec l'extension .tar.bz2

Ce script doit créer une archive contenant les fichiers en arguments, avec une extension qui dépend de la réponse de l'utilisateur. N'oublier pas d'ajouter une fonction *usage*.