

TP3

Faire les manipulations demandées à l'aide uniquement de commandes (utiliser le 'man' pour découvrir le fonctionnement des commandes).

Exercice1 :

Créer un script qui effectue à la suite les opérations suivantes :

1. crée un répertoire "Essai_Script"
2. crée un fichier vide "toto.txt" dans ce répertoire
3. crée une copie de "toto" nommée "toto2", toujours dans ce répertoire
4. stocke une liste des fichiers de votre répertoire d'accueil, toujours dans "Essai_Script"
5. affiche le texte "voilà, c'est fait !"

Exercice2 :

Réaliser un script qui affiche un message de bienvenue, qui lit une phrase de quatre mots au clavier (en utilisant la commande read) et qui l'affiche à l'envers.

Exercice3 :

Refaire l'exercice précédent sans utiliser la commande read.

Exercice4 :

Écrire un script qui compte à rebours de 5 à 1 et qui affiche ensuite "décollage".

Exercice5 :

Écrire un script qui crée 5 fichiers "fic1.txt" à "fic5.txt" dans un répertoire passé en paramètre.

Exercice6 :

1. Écrire un script qui affiche la liste des fichiers présents dans un répertoire passé en paramètre en ajoutant un "a" au début du nom de tous les fichiers.
2. Ce script peut-il être réutilisé par un administrateur système qui voudrait renommer tous les fichiers d'un répertoire donné ? Si oui, expliquer comment.

Exercice7 :

En utilisant le fichier "annuaire.txt" du TP précédent, écrire un script "existcopain" permettant de déterminer si le nom passé en paramètre à la commande correspond à une personne de l'annuaire.

Exercice8 :

Écrire un script qui donne l'écriture en anglais d'un chiffre passé en paramètre (les chiffres sont compris entre 0 et 5). Le script répondra "inconnu" en cas d'erreur.
Exemple : 1 donnera "one".

Exercice9 :

Écrire un script qui se déplace dans le répertoire passé en paramètre et y crée un fichier vide uniquement si ce répertoire existe et que l'utilisateur est autorisé à se déplacer dedans. Un message d'erreur devra être affiché le cas échéant.

Exercice10 :

Écrire un script shell qui affiche le nom de tous les fichiers du répertoire `"/usr/include"` dont le nom se termine par `".h"` et ayant plus de 100 lignes.

Exercice11 :

Écrire un programme permettant de changer facilement l'extension d'une série de fichiers (penser à la commande `basename`).

Exemple : `"renomme htm html ~/mydir"` renomme tous les fichiers `".htm"` du répertoire `"mydir"` en `".html"`.

Exercice12 :

Écrire un script qui affiche un menu donnant le choix entre 3 commandes :

1. Affichage de la date
2. Addition de deux nombres
3. Quitter

Exercice13 :

1. Écrire un script `"trash"` qui déplace dans un répertoire `"poubelle"` (à créer s'il n'existe pas encore) les fichiers dont les noms sont passés en paramètres.
2. Ajouter une option `"-c"` à la commande permettant de connaître la taille de la poubelle (indice : utiliser la commande `du`).
3. Ajouter une option `"-e"` permettant de vider la poubelle
4. Ajouter une option `"-h"` permettant d'afficher l'utilisation de la commande

Exercice14 :

Écrire une commande shell avec 3 arguments. Elle indiquera si les 3 chaînes sont identiques ou non par le code de retour suivant :

1. 0 si les 3 chaînes sont égales
2. 1, 2 ou 3 si la chaîne différente des autres est à la (1/2/3)ième position
3. 4 si les trois chaînes sont différentes
4. 5 si le nombre de paramètres est incorrect