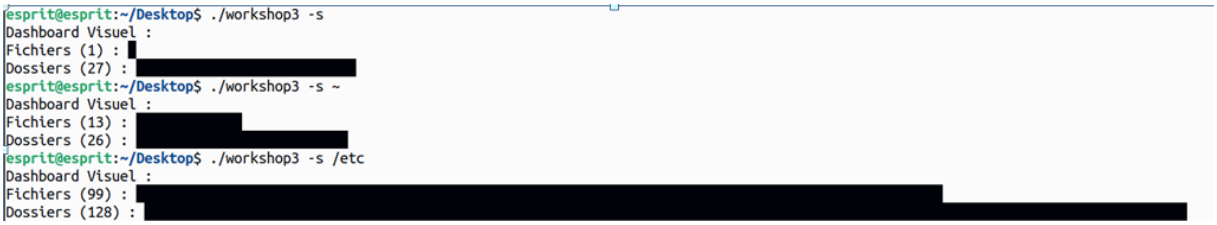
Atelier Système et Scripting

Objectifs

On se propose de développer un script Bash robuste, modulaire et interactif, nommé journal.sh, qui permet d'analyser des dossiers donnés en paramètres et de produire des rapports complets sur leur contenu et leur activité.
Ce script devra gérer de manière automatique, sécurisée et personnalisable l'analyse des répertoires, tout en fournissant des tableaux de bord et des historiques détaillés.
*Le script doit être **exécuté avec des droits suffisants** pour accéder aux fichiers cachés ou protégés*.

Fonctionnalités

1. Ecrire la fonction show_usage qui affiche sur la sortie standard le message :
   journal.sh: [-h] [-T] [-t] [-n] [-N] [-d] [-m] [-s] chemin.
2. Le script doit tester la présence d'au moins un argument, sinon il affiche l'usage sur la sortie d'erreur et échoue.
3. Ecrire une fonction nommée HELP qui permet d'afficher le help à partir d'un fichier texte help.txt contenant une description bien détaillée de l'application.
4. Ecrire une fonction **AfficheFile** qui permet de lister tous les fichiers (y compris les fichiers cachés) d'un dossier donné en paramètre en indiquant leur nom, taille en octets et droits d'accès.
5. Ecrire une fonction **AfficheDirectory** qui permet de lister uniquement les dossiers présents dans le dossier donné en paramètre, visibles et cachés en indiquant leurs nom, l'utilisateur et le groupe propriétaire (UID/GID).
6. Ecrire une fonction **NB** qui permet de compter le nombre total de fichiers, le nombre total de dossiers, le nombre de fichiers cachés et le nombre de fichiers par extension (.sh, .txt, .c, .csv). Le résultat de comptage est stocké dans un fichier count.txt.
7. Ecrire une fonction **dateAccess** qui affiche et sauvegarde la date du dernier accès pour chaque fichier/dossier dans date_journal.txt.
8. Ecrire une fonction **datemodif** qui affiche et sauvegarde la date de dernière modification pour chaque fichier/dossier dans modif_journal.txt.
9. Ecrire une fonction **Dashboard** qui permet d'afficher un résumé visuel :



10. Ecrire une fonction **LogHistory** qui permet de journaliser chaque opération effectuée dans historique_journal.log avec date, heure, et action.

Options

-h : Afficher l'aide détaillée (HELP)
-T : Affiche tous les fichiers d'un dossier (AfficheFile)
-t : Affiche tous les dossiers d'un dossier (AfficheDirectory)
-n : Compter fichiers/dossiers/extensions (NB)
-N : Afficher le propriétaire (DirectoryUser)
-d : Date dernier accès (dateAccess)
-m : Date dernière modification (datemodif)
-s : Dashboard visuel (Dashboard)

Consignes

- L'application doit être modulaire et contenant des fonctions.
- Chaque option sélectionnée doit déclencher une fonction.
- L'utilisation de **getopts** est obligatoire pour gérer les options du script