

Travaux Pratique N ° 3

Gestion de fichiers sous Linux partie 2

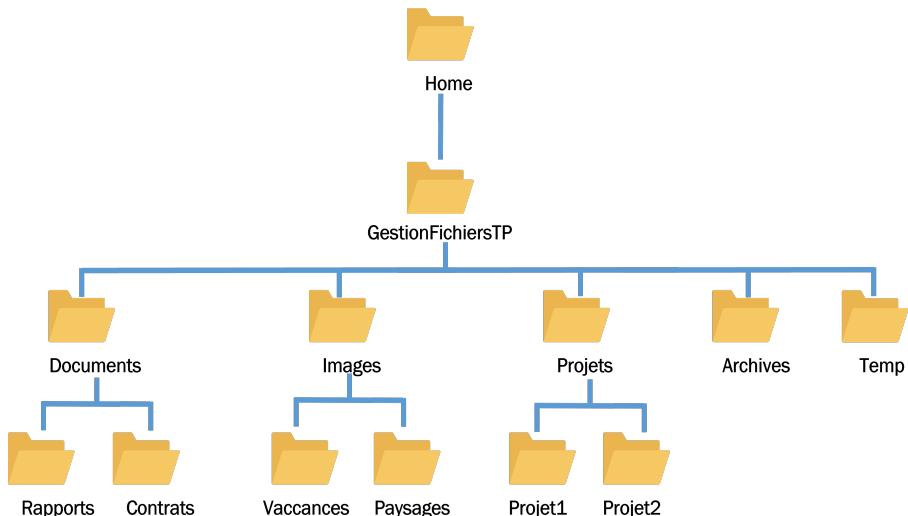
1 Objectif du TP

- Maîtriser les commandes Linux pour gérer des fichiers et des répertoires.
- Comprendre l'utilisation de commandes telles que `ls`, `mkdir`, `touch`, `cp`, `mv`, `rm`, `find`, `cat`, `grep`, `sed`, `man`, `file`, et `which`.
- Appliquer ces commandes pour effectuer des tâches courantes de gestion de fichiers sous Linux.

2 Tâches

1. Créez un répertoire nommé "GestionFichiersTP" dans votre répertoire personnel.
2. À l'intérieur de "GestionFichiersTP", créez la structure de répertoires suivante :
 - "Documents"
 - "Rapports"
 - "Contrats"
 - "Images"
 - "Vacances"
 - "Paysages"
 - "Projets"
 - "Projet1"
 - "Projet2"
 - "Archives"

— "Temp"



3. À l'intérieur du répertoire "Documents/Rapports", créez un fichier "rapport1.txt".
4. À l'intérieur du répertoire "Documents/Contrats", créez un fichier "contrat1.pdf".
5. À l'intérieur du répertoire "Images/Vacances", créez un fichier image "vacances.jpg".
6. À l'intérieur du répertoire "Images/Paysages", créez un fichier image "paysage.jpg".
7. Utilisez la commande `ls` pour afficher la liste des fichiers et répertoires dans "GestionFichiersTP".
8. Copiez le fichier "rapport1.txt" du répertoire "Documents/Rapports" dans le répertoire "Projets/Projet1".
9. Utilisez la commande `find` pour rechercher tous les fichiers ".jpg" dans l'ensemble de la structure "GestionFichiersTP".
10. Supprimez le fichier "rapport1.txt" du répertoire "Projets/Projet1".
11. Supprimez le répertoire "Images" de manière récursive.
12. Utilisez la commande `touch` pour créer un fichier "notes.txt" dans le répertoire "Documents".
13. Utilisez la commande `cat` pour afficher le contenu du fichier "notes.txt".
14. Utilisez la commande `grep` pour rechercher un mot spécifique dans "notes.txt" et afficher les lignes correspondantes.
15. Utilisez la commande `sed` pour remplacer un mot spécifique par un autre dans "notes.txt".

Exemple :

Ceci est un exemple de fichier texte.

Nous allons utiliser la commande sed pour effectuer des remplacements.
sed est un outil puissant pour la manipulation de texte.

```
sed 's/Sed/sed/g' monfichier.txt
```

Explication de la commande : sed est la commande elle-même.

- **'s/Sed/sed/g'** : C'est le motif de recherche et de remplacement délimité par des apostrophes. Il a la forme générale suivante : **'s/pattern/replacement flags'**.
- **s** : Il s'agit de l'indicateur pour effectuer une substitution (recherche et remplacement).
- **pattern** : C'est le motif que vous recherchez dans le fichier. Dans cet exemple, le motif est "Sed", et il est sensible à la casse, ce qui signifie qu'il ne correspondra qu'à "Sed" avec une majuscule initiale.
- **replacement** : C'est le texte par lequel vous souhaitez remplacer le motif trouvé. Dans cet exemple, le motif "Sed" est remplacé par "sed" (en minuscules).
- **g** : Il s'agit d'un indicateur qui signifie "global". Cela indique à sed de remplacer toutes les occurrences du motif dans le fichier. Si vous omettez "g", sed ne remplacera que la première occurrence de chaque ligne.
- **monfichier.txt** : C'est le nom du fichier sur lequel vous souhaitez appliquer la commande sed. Dans cet exemple, il s'agit du fichier "monfichier.txt".

Ainsi, la commande sed 's/Sed/sed/g' monfichier.txt recherche toutes les occurrences sensibles à la casse du mot "Sed" dans le fichier "monfichier.txt" et les remplace par "sed". Si vous souhaitez effectuer cette opération sans sensibilité à la casse (c'est-à-dire, rechercher "Sed" ou "sed" et les remplacer par "sed"), vous pouvez omettre le "g" de l'indicateur global.

16. Utilisez la commande `man` pour afficher le manuel de la commande `file` et explorez ses options.
17. Utilisez la commande `file` pour afficher le type de fichier de "notes.txt".
18. Utilisez la commande `which` pour trouver l'emplacement de l'exécutable de la commande `sed`.