

## Documentation :

### JOB 01

- **Afficher le manuel de la commande "ls"**

>> Rentrer dans le terminal la commande : `man ls`

Cela affichera le manuel de la commande `ls` avec toutes les options et les descriptions associées.

- **Afficher les fichiers cachés du home de votre utilisateur**

>> Utiliser la commande `ls` avec l'option `-a` suivie du chemin de votre répertoire personnel (home). Par défaut, le répertoire personnel de l'utilisateur est représenté par le symbole "~".

Cette commande utilise l'option `-a` pour afficher tous les fichiers, y compris les fichiers cachés (ceux dont le nom commence par un point) du répertoire personnel de l'utilisateur (~ représente le répertoire personnel).

- **Afficher les fichiers cachés plus les informations sur les droits sous forme de liste**

Pour afficher les fichiers cachés ainsi que les informations sur les droits sous forme de liste, il faut rentrer la commande `ls` avec l'option `-l` (pour le format long) et l'option `-a` (pour afficher les fichiers cachés).

Cette commande combine les options `-a` (pour afficher les fichiers cachés) et `-l` (pour afficher les informations sur les droits sous forme de liste) pour afficher tous les fichiers, y compris les fichiers cachés, avec des informations détaillées dans le répertoire personnel de l'utilisateur (~).

### Questions

- **Comment ajouter des options à une commande ?**

Pour ajouter des options à une commande en ligne de commande Linux, il faut spécifier ces options après la commande elle-même. Les options sont généralement précédées d'un tiret (-) ou de deux tirets (--), et elles modifient le comportement de la commande en lui fournissant des paramètres supplémentaires.

- **Quelles sont les deux syntaxes principales d'écriture des options pour une commande ?**

Les deux principales syntaxes d'écriture des options pour une commande en ligne de commande Linux sont :

1. Syntaxe des options courtes (ou abrégées) : Les options courtes sont précédées d'un seul tiret (-) et sont souvent des caractères uniques, par exemple, **-a**, **-l**. Elles permettent d'activer rapidement des options spécifiques et peuvent être regroupées sous un seul tiret.
2. Syntaxe des options longues (ou explicites) : Les options longues sont précédées de deux tirets (--). Elles sont généralement des noms complets ou des phrases descriptives. Elles sont plus explicites et ne peuvent pas être regroupées sous un seul tiret.

Les options courtes sont couramment utilisées pour des commandes fréquentes et pour économiser de la frappe, tandis que les options longues sont plus explicites et facilitent la compréhension de la commande, en particulier pour les utilisateurs qui ne sont pas familiers avec la commande en question.

---

## JOB 02

- **Lire un fichier en utilisant une commande qui permet seulement de lire :**

Commande en ligne de commande Linux :

```
cat nom_du_fichier
```

Cette commande affiche le contenu complet du fichier spécifié.

- **Afficher les 10 premières lignes du fichier ".bashrc"**

Commande en ligne de commande Linux :

```
head -n 10 ~/.bashrc
```

Cette commande utilise la commande "head" pour afficher les premières 10 lignes du fichier "~/.bashrc" (le fichier de configuration de bash pour l'utilisateur courant).

- **Afficher les 10 dernières lignes du fichier ".bashrc" :**

Commande en ligne de commande Linux :

```
tail -n 10 ~/.bashrc
```

Cette commande utilise la commande "tail" pour afficher les dernières 10 lignes du fichier "~/.bashrc".

- **Afficher les 20 premières lignes du fichier ".bashrc" :**

Commande en ligne de commande Linux :

```
head -n 20 ~/.bashrc
```

Cette commande affiche les premières 20 lignes du fichier "~/.bashrc".

- **Afficher les 20 dernières lignes du fichier ".bashrc" :**

Commande en ligne de commande Linux :

```
tail -n 20 ~/.bashrc
```

Cette commande affiche les dernières 20 lignes du fichier "~/.bashrc".

---

### **JOB 03**

- **Installer le paquet "cmatrix" :**

Commande en ligne de commande Linux (en utilisant apt, un gestionnaire de paquets courant sur les distributions basées sur Debian, comme Ubuntu) :

```
sudo apt install cmatrix
```

Cette commande installe le paquet "cmatrix" sur le système.

- **Lancer le paquet que vous venez d'installer :**

Une fois installé "cmatrix", il est possible de le lancer en entrant simplement dans le terminal :

```
cmatrix
```

Cela démarrera l'effet de matrice (Matrix) sur le terminal.

- **Mettre à jour son gestionnaire de paquets :**

Commande en ligne de commande Linux (avec apt) pour mettre à jour la liste des paquets disponibles :

```
sudo apt update
```

Cette commande met à jour la liste des paquets disponibles dans les référentiels du système.

- **Mettre à jour ses différents logiciels :**

Commande en ligne de commande Linux (avec apt) pour mettre à jour tous les logiciels installés sur le système :

```
sudo apt upgrade
```

Cette commande met à jour tous les logiciels installés sur le système.

- **Télécharger les internets : Google :**

Il est possible d'utiliser un outil comme "wget" pour télécharger un site web spécifique. Pour télécharger la page d'accueil de Google, utiliser la commande suivante :

```
wget https://www.google.com
```

Cela téléchargera la page web sur le système de l'utilisateur.

- **Redémarrer votre machine :**

Commande en ligne de commande Linux pour redémarrer la machine :

```
sudo reboot
```

Cette commande redémarre le système.

- **Éteindre votre machine :**

Commande en ligne de commande Linux pour éteindre la machine :

```
sudo shutdown -h now
```

Cette commande arrête proprement le système.

---

## **JOB 04**

Pour créer un fichier "users.txt" contenant les noms d'utilisateurs "User1" et "User2" séparés par un retour à la ligne sous Linux, il est possible d'utiliser les commandes de redirection et d'écho :

Utiliser la commande suivante pour créer le fichier "users.txt" avec les noms d'utilisateurs et les retours à la ligne :

```
echo -e "User1\nUser2" > users.txt
```

Cette commande utilise l'option **-e** avec **echo** pour interpréter les séquences d'échappement telles que **\n** comme des retours à la ligne. Le **>** redirige la sortie vers un fichier appelé "users.txt".

La commande `cat users.txt` permet d'afficher le contenu du groupe "users.txt" pour vérifier le résultat de l'opération :

```
GNU nano 7.2                                users.txt
User1
User2
```

- **Créer un groupe appelé "Plateformeurs" :**

La première étape consiste à créer un groupe nommé "Plateformeurs" à l'aide de la commande `groupadd` avec le nom du groupe comme argument. Vous devez exécuter cette commande en tant qu'administrateur (ou avec les droits sudo) dans un terminal :

```
sudo groupadd Plateformeurs
```

Cette commande crée un groupe appelé "Plateformeurs".

- **Créer un utilisateur appelé "User1" :**

Pour créer un nouvel utilisateur nommé "User1", utilisez la commande `useradd` avec l'option `-m` pour créer un répertoire personnel pour l'utilisateur. Exécutez cette commande en tant qu'administrateur :

```
sudo useradd -m User1
```

Cela crée un utilisateur nommé "User1" avec son propre répertoire personnel.

- **Créer un utilisateur appelé "User2" :**

De même, vous pouvez créer un deuxième utilisateur nommé "User2" de la même manière :

```
sudo useradd -m User2
```

- **Ajouter "User2" au groupe "Plateformeurs" :**

Utilisez la commande `usermod` pour ajouter "User2" au groupe "Plateformeurs". L'option `-a` ajoute l'utilisateur au groupe existant :

```
sudo usermod -a -G Plateformeurs User2
```

Cela permet à "User2" de faire partie du groupe "Plateformeurs".

- **Copier "users.txt" dans "droits.txt" :**

Pour copier le contenu du fichier "users.txt" dans un nouveau fichier "droits.txt", utilisez la commande `cp` :

```
cp users.txt droits.txt
```

- **Copier "users.txt" dans "groupes.txt" :**

De manière similaire, copiez le contenu de "users.txt" dans un nouveau fichier "groupes.txt" avec la même commande `cp` :

```
cp users.txt groupes.txt
```

- **Changer le propriétaire du fichier "droits.txt" pour mettre "User1" :**

Pour changer le propriétaire du fichier "droits.txt" en "User1", il faut utiliser la commande `chown` :

```
sudo chown User1 droits.txt
```

Cela fait de "User1" le propriétaire du fichier "droits.txt".

- **Changer les droits du fichier "droits.txt" pour que "User2" ait accès en lecture seulement :**

Il faut utiliser la commande `chmod` pour définir les permissions du fichier "droits.txt" de manière à ce que "User2" puisse uniquement le lire :

```
chmod 644 droits.txt
```

Cela donne au propriétaire (User1) le droit de lecture et autorise également le groupe et les autres à lire le fichier.

- **Changer les droits du fichier "groupes.txt" pour que les utilisateurs puissent accéder au fichier en lecture uniquement :**

De manière similaire, il faut utiliser la commande `chmod` pour définir les permissions du fichier "groupes.txt" de manière à ce que tous les utilisateurs puissent seulement le lire :

```
chmod 644 groupes.txt
```

- **Changer les droits du fichier pour que le groupe "Plateformeurs" puisse y accéder en lecture/écriture :**

Pour permettre au groupe "Plateformeurs" d'accéder au fichier "groupes.txt" en lecture et écriture, il faut utiliser la commande `chown` pour définir le groupe propriétaire, puis `chmod` pour définir les permissions :

```
sudo chown :Plateformeurs groupes.txt
```

```
chmod 664 groupes.txt
```

Cela attribue le groupe "Plateformeurs" comme propriétaire du fichier "groupes.txt" et accorde des permissions de lecture/écriture au groupe.

## Note :

**chmod** : C'est la commande qui permet de modifier les permissions (droits d'accès) des fichiers ou répertoires.

**644** : C'est un code numérique qui représente les permissions spécifiques accordées au fichier. Les permissions sont généralement représentées par trois chiffres, chacun pouvant varier de 0 à 7. Chacun de ces chiffres représente un ensemble spécifique de droits : le premier chiffre pour le propriétaire du fichier, le deuxième pour le groupe, et le troisième pour les autres (utilisateurs).

- Le chiffre "6" en position du premier chiffre signifie que le propriétaire du fichier a les droits de lecture (4) et d'écriture (2), car  $4 + 2 = 6$ .
  - Le chiffre "4" en position du deuxième chiffre signifie que le groupe auquel appartient le fichier a seulement le droit de lecture (4).
  - Le chiffre "4" en position du troisième chiffre signifie que les autres utilisateurs (non propriétaires et non membres du groupe) ont également seulement le droit de lecture (4).
- 

## JOB 05

### 1. Ajouter un alias pour "ls -la" en tapant "la" :

*Un alias permet de créer un raccourci pour une commande longue.*

Il faut éditer le fichier de configuration du shell. Le fichier de configuration dépend du shell utilisé. Pour le shell Bash, c'est généralement `~/.bashrc`. Il faut l'ouvrir avec un éditeur de texte (par exemple, `nano`) en utilisant la commande suivante :

```
nano ~/.bashrc
```

Cette commande affiche le contenu du fichier `.bashrc`. Aller à la fin du fichier en utilisant les touches fléchées.

Ajouter l'alias en utilisant la syntaxe suivante. Cela crée un alias nommé `la` qui exécute la commande `ls -la` lorsque l'on tape `la` dans le terminal :

```
alias la='ls -la'
```

Enregistrer les modifications en appuyant sur `Ctrl + O`, puis appuyer sur "Entrée". Ensuite, il faut quitter l'éditeur en appuyant sur `Ctrl + X`. Pour activer l'alias sans avoir à fermer et rouvrir le terminal, rechargez le fichier de configuration du shell avec la commande suivante :

```
source ~/.bashrc
```

Désormais, il est possible de simplement taper `la` dans le terminal pour exécuter `ls -la` et pour afficher une liste détaillée des fichiers et répertoires dans le répertoire courant.

**- Ajouter un alias pour "apt-get update" en tapant "update" :**

Cet alias simplifie la mise à jour des paquets sur le système.

Éditer le fichier de configuration du shell, qui est généralement `~/.bashrc` pour Bash, en utilisant la commande suivante :

```
nano ~/.bashrc
```

Aller à la fin du fichier.

Ajouter l'alias pour "apt-get update" en utilisant la syntaxe suivante (cela crée un alias nommé "update" qui exécute la commande "sudo apt-get update" lorsque l'on tape "update" dans le terminal) :

```
alias update='sudo apt-get update'
```

Enregistrez les modifications en appuyant sur `Ctrl + O`, puis appuyer sur "Entrée". Ensuite, quitter l'éditeur en appuyant sur `Ctrl + X`.

Pour activer l'alias sans avoir à fermer et rouvrir le terminal, recharger le fichier de configuration du shell avec la commande suivante :

```
source ~/.bashrc
```

Désormais, vous pouvez simplement taper "update" dans le terminal pour exécuter "sudo apt-get update" et mettre à jour la liste des paquets disponibles sur votre système.

**- Ajouter un alias pour "apt-get upgrade" en tapant "upgrade" :**

Cet alias simplifie la mise à niveau des paquets sur le système.

Éditer le fichier de configuration de votre shell, qui est généralement `~/.bashrc` pour Bash, en utilisant la commande suivante :

```
nano ~/.bashrc
```

Aller à la fin du fichier.

Ajouter l'alias pour "apt-get upgrade" en utilisant la syntaxe suivante. Cela crée un alias nommé "upgrade" qui exécute la commande "sudo apt-get upgrade" lorsque l'on tape "upgrade" dans le terminal :

```
alias upgrade='sudo apt-get upgrade'
```

Enregistrer les modifications en appuyant sur `Ctrl + O`, puis appuyer sur "Entrée". Ensuite, quittez l'éditeur en appuyant sur `Ctrl + X`.



Pour activer l'alias sans avoir à fermer et rouvrir le terminal, recharger le fichier de configuration du shell avec la commande suivante :

```
source ~/.bashrc
```

Désormais, il est possible de simplement taper "upgrade" dans le terminal pour exécuter "sudo apt-get upgrade" et mettre à niveau les paquets installés sur le système.

- **Ajouter une variable d'environnement nommée "USER" égale à votre nom d'utilisateur :**

*Une variable d'environnement est une valeur stockée dans le système d'exploitation qui peut être utilisée par les programmes.*

Il est possible de définir une variable d'environnement en utilisant la commande `export`. Pour définir une variable "USER" égale au nom d'utilisateur, utiliser la commande suivante, en remplaçant `utilisateur` par le nom d'utilisateur réel (ici `david`):

```
export USER='david'
```

Cette commande définit la variable "USER" avec la valeur du nom d'utilisateur.

- **Mettre à jour les modifications de votre `.bashrc` dans votre shell actuel :**

Il n'est pas nécessaire de mettre à jour manuellement le shell actuel si le fichier `.bashrc` a déjà été modifié. Les modifications apportées à ce fichier sont automatiquement prises en compte chaque fois que l'on ouvre le terminal. Cependant, pour appliquer les modifications immédiatement sans ouvrir un nouveau terminal, il faut exécuter la commande suivante :

```
source ~/.bashrc
```

Cette commande recharge le fichier `.bashrc` dans la session actuelle, ce qui permet d'appliquer les modifications immédiatement.

- **Afficher les variables d'environnement :**

Pour afficher les variables d'environnement, il faut utiliser la commande `env` ou `printenv`. Par exemple, il est possible d'entrer l'une des deux commandes suivantes dans le terminal :

```
env
```

ou

```
printenv
```

Cela affichera une liste de toutes les variables d'environnement, y compris la variable "USER" qui a été définie.

- **Ajouter à votre PATH le chemin `"/home/votre_utilisateur/Bureau"` :**

*Le chemin "PATH" est une variable d'environnement qui spécifie les répertoires dans lesquels le système recherche les programmes exécutables.*

Pour ajouter un répertoire à votre variable d'environnement PATH, il faut suivre ces étapes :

Utiliser la commande `export` pour ajouter le chemin `"/home/votre_utilisateur/Bureau"` à votre variable PATH. Remplacer `votre_utilisateur` par le nom d'utilisateur (ici `david`) .

Dans mon cas, la commande est :

```
export PATH="/home/david/Bureau:$PATH"
```

Cette commande ajoute le chemin `"/home/david/Bureau"` à la variable PATH, tout en préservant les valeurs existantes de PATH grâce à `:$PATH`.

Pour rendre cette modification permanente, il est possible de l'ajouter au fichier `.bashrc` en utilisant un éditeur de texte, de la même manière que pour l'alias créé précédemment. Cela garantira que le chemin soit ajouté au PATH chaque fois que l'on ouvrira le terminal, et il sera donc disponible en permanence pour exécuter des programmes situés dans ce répertoire.

Pour vérifier la valeur actuelle de la variable PATH, il faut entrer la commande suivante :

```
echo $PATH
```

Cela affichera la valeur actuelle de votre variable PATH. Il faut s'assurer que le chemin `"/home/votre_utilisateur/Bureau"` que vous avez ajouté est présent dans la liste.

---

## Le Fantôme dans la Coquille

- **Créer un fichier `"une_commande.txt"` avec le texte `"Je suis votre fichier texte"`:**

Pour accomplir cette tâche, il faut utiliser la commande `echo` pour générer le contenu du fichier, suivi de la commande `>` pour rediriger ce contenu vers le fichier `"une_commande.txt"`.

```
echo "Je suis votre fichier texte" > une_commande.txt
```

### Explication :

- `echo`: Cette commande permet d'afficher du texte dans le terminal.
- `"Je suis votre fichier texte"` : C'est le texte qui sera écrit dans le fichier.

- **>** : Cet opérateur de redirection enregistre la sortie de la commande **echo** dans le fichier "une\_commande.txt".

- **Compter le nombre de lignes présentes dans le fichier source apt et les enregistrer dans un fichier nommé "nb\_lignes.txt":**

Pour compter le nombre de lignes dans un fichier, il faut utiliser la commande **wc** (word count) avec l'option **-l** pour compter les lignes.

```
wc -l < /etc/apt/sources.list > nb_lignes.txt
```

**Explication :**

- **wc -l**: Cette commande compte les lignes dans le fichier.
- **< /etc/apt/sources.list**: Cet opérateur de redirection **<** permet de prendre le contenu du fichier "/etc/apt/sources.list" comme entrée pour la commande **wc**.
- **> nb\_lignes.txt**: Cet opérateur de redirection enregistre la sortie de la commande **wc** (le nombre de lignes) dans le fichier "nb\_lignes.txt".

- **Afficher le contenu du fichier source apt et l'enregistrer dans un autre fichier appelé "save\_sources":**

Pour afficher le contenu du fichier source apt et l'enregistrer dans un autre fichier, il faut utiliser la commande **cat** pour afficher le contenu, suivi de l'opérateur **>** pour rediriger la sortie vers un fichier.

```
cat /etc/apt/sources.list > save_sources
```

**Explication :**

- **cat**: Cette commande permet d'afficher le contenu d'un fichier.
- **/etc/apt/sources.list**: C'est le chemin du fichier source apt dont le contenu sera affiché.
- **> save\_sources**: Cet opérateur de redirection enregistre la sortie de la commande **cat** (le contenu du fichier) dans le fichier "save\_sources".

- **Faites une recherche des fichiers commençant par "." tout en cherchant le mot "alias" qui sera utilisé depuis un fichier:**

Pour effectuer cette recherche, il faut utiliser la commande **grep** pour rechercher les fichiers commençant par "." tout en cherchant le mot "alias" dans ces fichiers. Il est nécessaire d'utiliser l'option **-r** pour effectuer une recherche récursive à partir du répertoire actuel.

```
grep -r "alias" . > fichiers_alias.txt
```

#### **Explication :**

- **grep**: Cette commande permet de rechercher des motifs dans des fichiers.
- **-r**: Cette option indique à **grep** de rechercher récursivement dans les fichiers et les répertoires. La recherche récursive signifie que **grep** va explorer non seulement les fichiers dans le répertoire spécifié, mais aussi tous les sous-répertoires et leurs fichiers.
- **"alias"**: C'est le motif recherché (le mot "alias").
- **.**: Cela spécifie le répertoire à partir duquel commencer la recherche (le répertoire actuel).
- **> fichiers\_alias.txt**: Cet opérateur de redirection enregistre la sortie de la commande **grep** (les fichiers contenant le mot "alias") dans le fichier "fichiers\_alias.txt".

Voici donc la ligne de commande complète :

```
echo "Je suis votre fichier texte" > une_commande.txt ; wc -l < /etc/apt/sources.list > nb_lignes.txt ; cat /etc/apt/sources.list > save_sources ; grep -r "alias" . > fichiers_alias.txt
```