

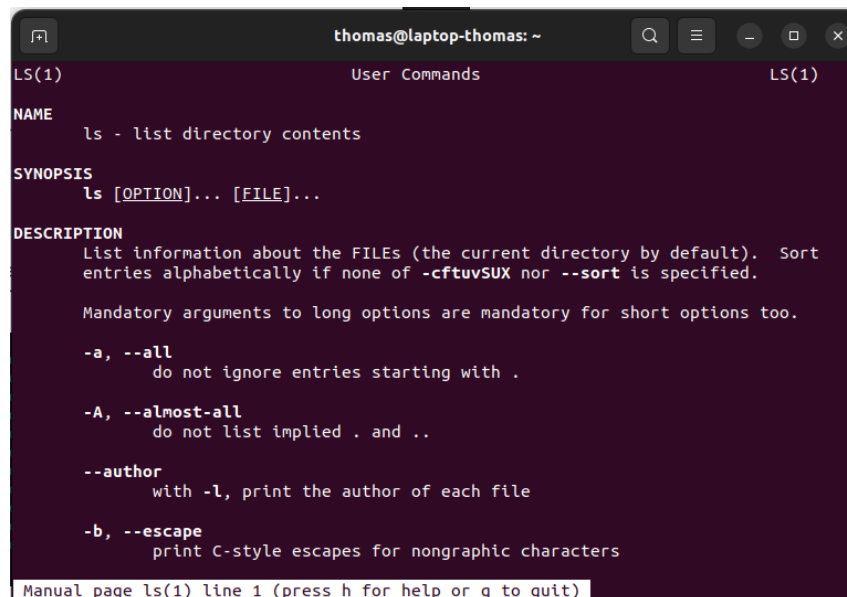
Shell

Code ASCII

Le code en haut des 2 sujets : 116 101 114 109 105 110 97 108 32 108 105 110 117 120
signifie : terminal linux

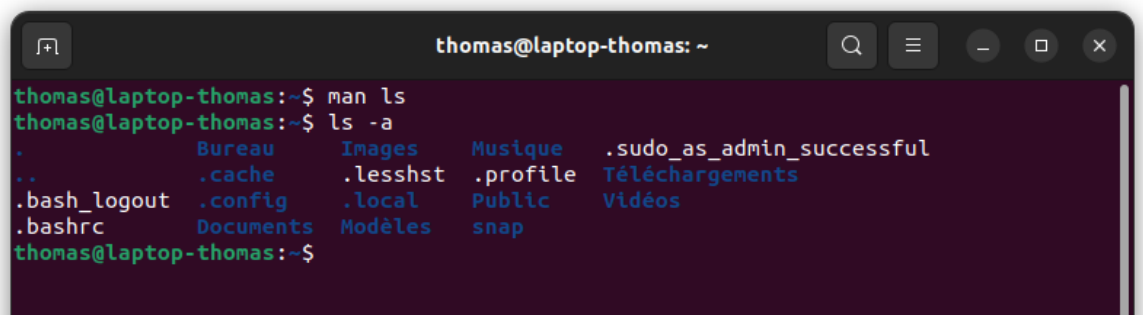
Job 1

- Afficher le manuel de la commande ls:
man ls



```
thomas@laptop-thomas: ~  
LS(1) User Commands LS(1)  
NAME  
    ls - list directory contents  
SYNOPSIS  
    ls [OPTION]... [FILE]...  
DESCRIPTION  
    List information about the FILES (the current directory by default). Sort  
    entries alphabetically if none of -cftuvSUX nor --sort is specified.  
  
    Mandatory arguments to long options are mandatory for short options too.  
  
    -a, --all  
        do not ignore entries starting with .  
  
    -A, --almost-all  
        do not list implied . and ..  
  
    --author  
        with -l, print the author of each file  
  
    -b, --escape  
        print C-style escapes for nongraphic characters  
  
Manual page ls(1) line 1 (press h for help or q to quit)
```

- Afficher les fichiers cachés du home de votre utilisateur:
ls -a



```
thomas@laptop-thomas: ~  
thomas@laptop-thomas:~$ man ls  
thomas@laptop-thomas:~$ ls -a  
.  
..  
.bash_logout  
.bashrc  
Bureau  
.cache  
.config  
Documents  
Images  
.lessht  
.local  
Modèles  
Musique  
.profile  
Public  
snap  
.sudo_as_admin_successful  
Téléchargements  
Vidéos  
thomas@laptop-thomas:~$
```

exemple de fichier caché : .profile

- Afficher les fichiers cachés plus les informations sur les droits sous forme de liste:
ls -la

```
thomas@laptop-thomas: ~  
.  
..  
.bash_logout  
.bashrc  
Bureau  
.cache  
.config  
Documents  
Images  
.lessht  
.local  
Modèles  
Musique  
.profile  
Public  
snap  
.sudo_as_admin_successful  
Téléchargements  
Vidéos  
thomas@laptop-thomas:~$ man ls  
thomas@laptop-thomas:~$ ls -la  
total 72  
drwxr-x--- 14 thomas thomas 4096 sept. 19 09:49 .  
drwxr-xr-x  3 root   root   4096 sept. 16 15:57 ..  
-rw-r--r--  1 thomas thomas  220 sept. 16 15:57 .bash_logout  
-rw-r--r--  1 thomas thomas 3771 sept. 16 15:57 .bashrc  
drwxr-xr-x  2 thomas thomas 4096 sept. 16 18:00 Bureau  
drwx----- 11 thomas thomas 4096 sept. 16 16:02 .cache  
drwx----- 11 thomas thomas 4096 sept. 16 16:04 .config  
drwxr-xr-x  2 thomas thomas 4096 sept. 16 18:00 Documents  
drwxr-xr-x  3 thomas thomas 4096 sept. 19 09:42 Images  
-rw-----  1 thomas thomas  20 sept. 19 09:49 .lessht  
drwx-----  3 thomas thomas 4096 sept. 16 18:00 .local  
drwxr-xr-x  2 thomas thomas 4096 sept. 16 18:00 Modèles  
drwxr-xr-x  2 thomas thomas 4096 sept. 16 18:00 Musique  
-rw-r--r--  1 thomas thomas  807 sept. 16 15:57 .profile  
drwxr-xr-x  2 thomas thomas 4096 sept. 16 18:00 Public  
drwx-----  4 thomas thomas 4096 sept. 19 08:46 snap  
-rw-r--r--  1 thomas thomas   0 sept. 19 09:20 .sudo_as_admin_successful  
drwxr-xr-x  2 thomas thomas 4096 sept. 16 18:00 Téléchargements  
drwxr-xr-x  2 thomas thomas 4096 sept. 16 18:00 Vidéos  
thomas@laptop-thomas:~$
```

Les droits sont regroupés par 3 types, pour 3 types d'utilisateurs différents:

- "r" pour le droit de lire un fichier
- "w" pour le droit de modifier
- "x" pour le droit d'exécuter
- "-" pour l'absence du droit correspondant

Les 3 premières lettres correspondent à l'utilisateur ("u"), les 3 suivantes au groupe ("g"), et les 3 dernières aux autres utilisateurs ("o").

Comment ajouter des options à une commande ?

Pour ajouter des options à une commande, il suffit d'écrire la commande, de sauter un espace puis de rajouter un tiret "-" avant d'écrire la lettre correspondant à l'option voulue.

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

Soit avec un tiret "-" suivi d'une lettre, soit avec un double tirets "--" suivie de l'option en toute lettre (lorsque c'est disponible).

Job 2

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

cat .bashrc

- afficher les 10 premières lignes du fichier ".bashrc":

head .bashrc

```
thomas@laptop-thomas:~$ head .bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

thomas@laptop-thomas:~$
```

- afficher les 10 dernières lignes du fichier ".bashrc":

tail .bashrc

```
thomas@laptop-thomas:~$ tail .bashrc
# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

thomas@laptop-thomas:~$
```

- afficher les 20 premières lignes du fichier :

head -20 .bashrc

```
thomas@laptop-thomas:~$ head -20 .bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

thomas@laptop-thomas:~$
```

- afficher les 20 dernières lignes du fichier :

tail -20 .bashrc

```
thomas@laptop-thomas:~$ tail -20 .bashrc

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi
thomas@laptop-thomas:~$
```

Job 3

- Installer le paquet “cmatrix”:
sudo apt install cmatrix
- Lancer le paquet que vous venez d’installer:
cmatrix
- Mettre à jour son gestionnaire de paquets:
sudo apt-get update
- Mettre à jour ses différents logiciels:
sudo apt-get upgrade
- Télécharger les internets : Google chrome:
wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
sudo apt install ./google-chrome-stable_current_amd64.deb
- Redémarrer votre machine :
sudo reboot
- éteindre votre machine:
sudo shutdown

Job 4

Créer un fichier users.txt qui contiendra User1 et User2 séparé par un retour à la ligne :

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

- Créer un groupe appelé "Plateformeurs":
sudo groupadd Plateformeurs
- Créer un utilisateur appelé "User1"
sudo useradd User1
ou **sudo adduser user1** (n'accepte pas les majuscules et est plus "complexe")
- Créer un utilisateur appelé "User2"
sudo useradd User2
- Ajouter "User2" au groupe Plateformeurs:
sudo adduser User2 Plateformeurs
- Copier votre "users.txt" dans un fichier "droits.txt"
cp users.txt droits.txt
- Copier votre "users.txt" dans un fichier "groupes.txt"
cp users.txt groupes.txt
- Changer le propriétaire du fichier "droits.txt" pour mettre "User1"
sudo chown User1 droits.txt
chown pour change owner
- Changer les droits du fichiers "droits.txt" pour que "User2" ai accès seulement en lecture
sudo chgrp Plateformeurs droits.txt
sudo chmod g+r-wx droits.txt
- changer les droits du fichier "groupes.txt" pour que les utilisateurs puissent accéder au fichier en lecture uniquement
sudo chmod o+r-wx groupes.txt
- Changer les droits du fichier (j'ai utilisé groupes.txt) pour que le groupe "Plateformeurs" puisse y accéder en lecture/écriture.
sudo chgrp Plateformeurs groupes.txt (pour que ce fichier appartient à ce groupe)
sudo chmod g+rw-x groupes.txt

Job 5

Pour créer les alias, il faut se rendre dans le fichier `.bashrc`, et prend la forme :

`alias nom_de_votre_alias='commande de votre alias'`

Pour ouvrir `.bashrc` : **gedit .bashrc** (ou un éditeur intégré au terminal comme nano ou vim)

Pour afficher les alias créés dans le terminal: **alias**

Il est aussi possible de réaliser ces alias de la même manière directement dans le terminal, mais dans ce cas elle ne sont que temporaires, sauf si l'on rajoute après la commande :

>> .bashrc

- Ajouter un alias qui permettra de lancer la commande "ls -la" en tapant "la":
alias la='ls -la'
- Ajouter un alias qui permettra de lancer la commande "apt-get update" en tapant "update":
alias update='sudo apt-get update'
- ajouter un alias qui permettra de lancer la commande "apt-get upgrade" en tapant "upgrade":
alias upgrade='sudo apt-get upgrade'
- ajouter une variable d'environnement qui se nommera "USER" et qui sera égale à votre nom d'utilisateur
USER=thomas (ne crée qu'une variable de shell, pour une variable environnement la commande suivante est nécessaire)
export USER
De manière plus concise, il est possible d'effectuer les deux commandes en une seule :
export USER=thomas
- Mettre à jour les modifications de votre bashrc dans votre shell actuel
source ~/.bashrc (ou tout simplement fermer puis réouvrir le terminal)
- afficher les variables d'environnement
printenv (affiche toutes les variables d'environnement)
printenv Nom_Variable (affiche la variable en question)
- Ajouter à votre Path le chemin "/home/'votre utilisateur'/Bureau"
export PATH=\$PATH:/home/thomas/Bureau (permet de rajouter ce chemin à la variable PATH. Les ":" permettent d'ajouter ce chemin en tant que valeurs supplémentaire, au lieu d'écraser ce qui est déjà dans la variable)

Job 6

Télécharger l'archive suivante et la désarchiver seulement avec le terminal.

<https://drive.google.com/file/d/11dSelXQuH4tih6zesbv-6OMEpr-sT77X/view?usp=sharing>

Téléchargement :

wget

<https://drive.google.com/file/d/11dSelXQuH4tih6zesbv-6OMEpr-sT77X/view?usp=sharing>

désarchiver :

tar -xzf archive.tar.gz

Job 7

Toutes les actions sont à réaliser en une seule commande

- Créer un fichier "une_commande.txt" avec le texte suivant "Je suis votre fichier texte":
echo 'Je suis votre fichier texte' > une_commande.txt
- Compter le nombre de lignes présentes dans votre fichier de source apt et les enregistrer dans un fichier nommé "nb_lignes.txt":
wc -l /etc/apt/sources.list > ~/Documents/La_plateforme/init/nb_lignes.txt
- Afficher le contenu du fichier source apt et l'enregistrer dans un autre fichier appelé "save_sources":
cat /etc/apt/sources.list > ~/Documents/init/save_sources
- Faites une recherche des fichiers commençant par "." tout en cherchant le mot "alias" dans ces fichiers :
find / -type f -name ".*"| grep -rnw "alias"
les options permettent :
 - -r permet de réaliser la recherche de manière récursive, donc dans tous les fichiers du répertoire
 - -n → permet l'affichage du numéro de ligne où se trouve le pattern dans le fichier
 - -w n'affiche que les fichiers dont le pattern est complet

Pour aller plus loin

Toutes les actions sont à réaliser en une seule ligne de commande

Premièrement étape par étape :

- Installer la commande tree :
sudo apt install tree
- Lancer la commande tree en arrière-plan qui aura pour but d'afficher toute l'arborescence de votre / en enregistrant le résultat dans un fichier "tree.save":
tree / > tree.save &
- lister les éléments présents dans le dossier courant est utilisé directement sur le résultat de votre première commande pour compter le nombre d'éléments trouvés :
ls | wc -l
- Lancer une commande pour update vos paquets, si l'update réussit alors, vous devrez lancer un upgrade de vos paquets. Si l'update échoue, votre upgrade ne se lancera pas:
update && upgrade

Deuxièmement sur une ligne (sauf l'installation de tree, on pourrait rajouter && pour la mettre sur la même) :

sudo apt install tree
tree / > tree.save & ls | wc -l & update && upgrade