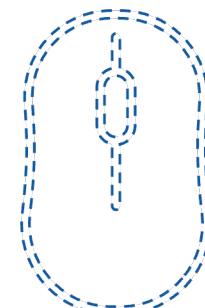




# Ligne de commandes sous Linux

420-1S6 Systèmes d'exploitation

Dans les années 80...



Mais où est la souris?

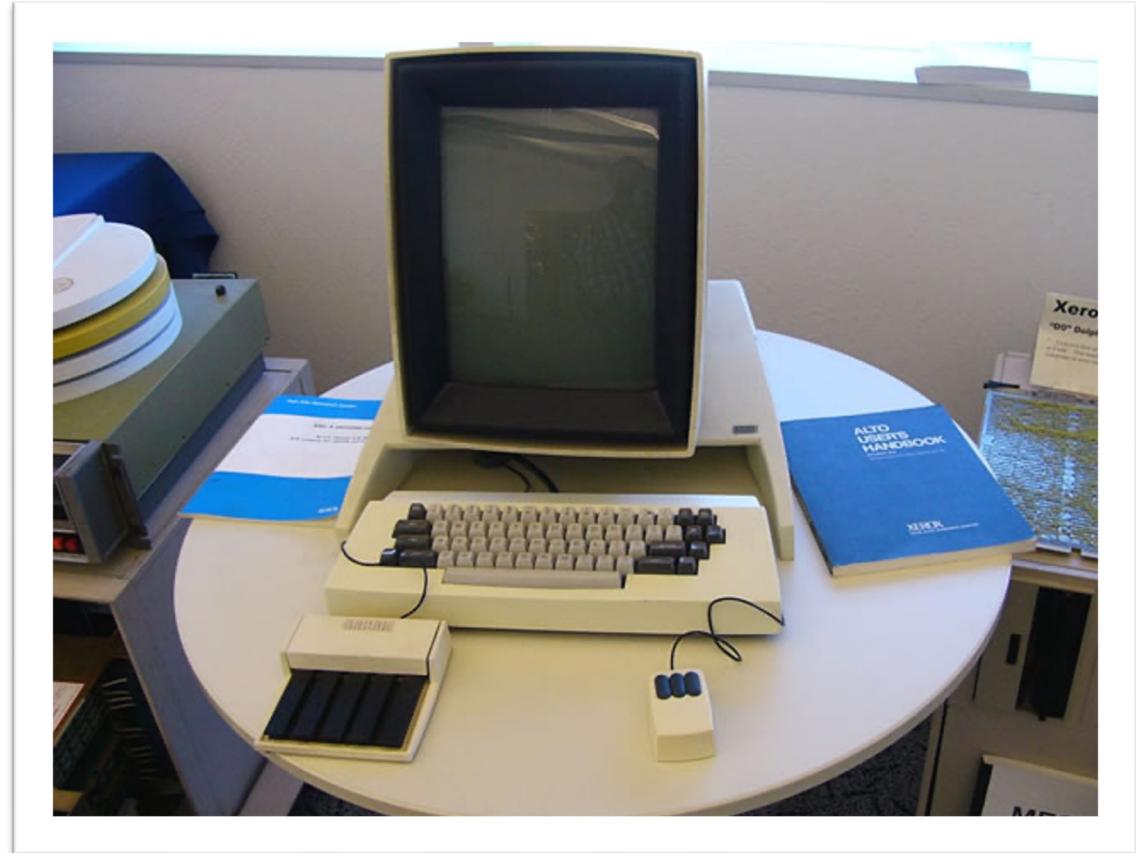
# L'interpréteur de commandes



Autrefois (dans un passé pas si lointain), la manière la plus facile pour interagir avec un ordinateur était de lui envoyer des commandes textuelles.



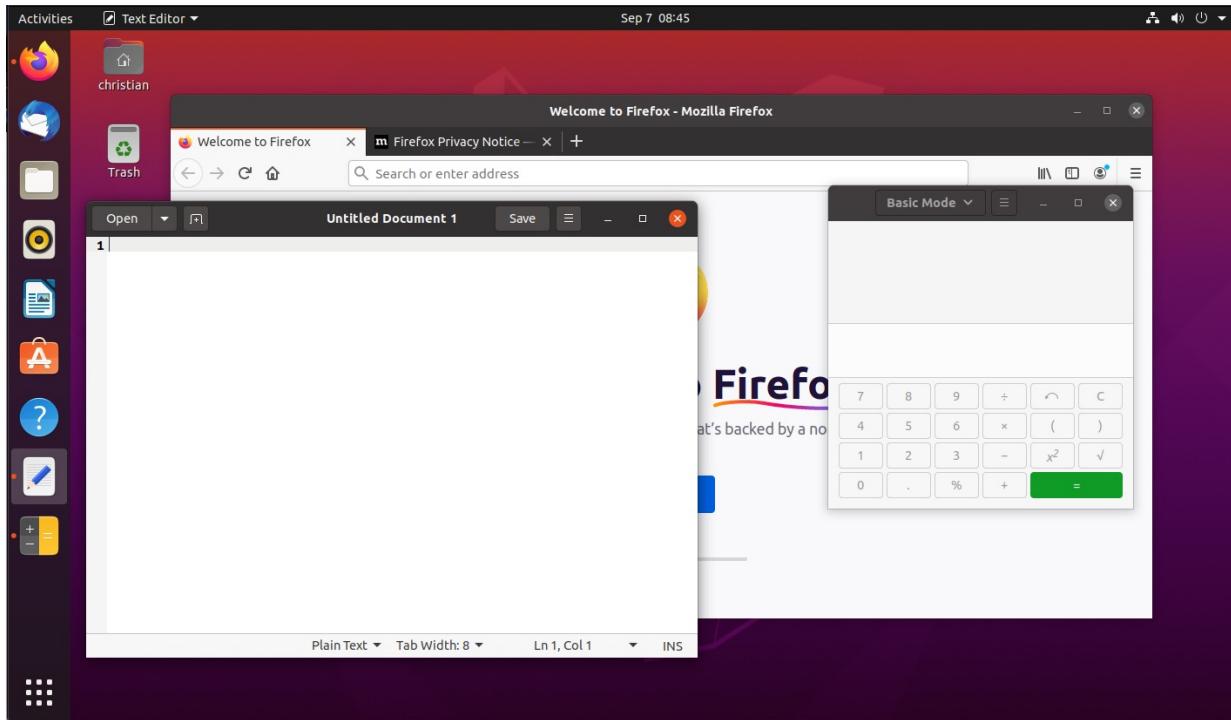
# Premier PC avec une souris



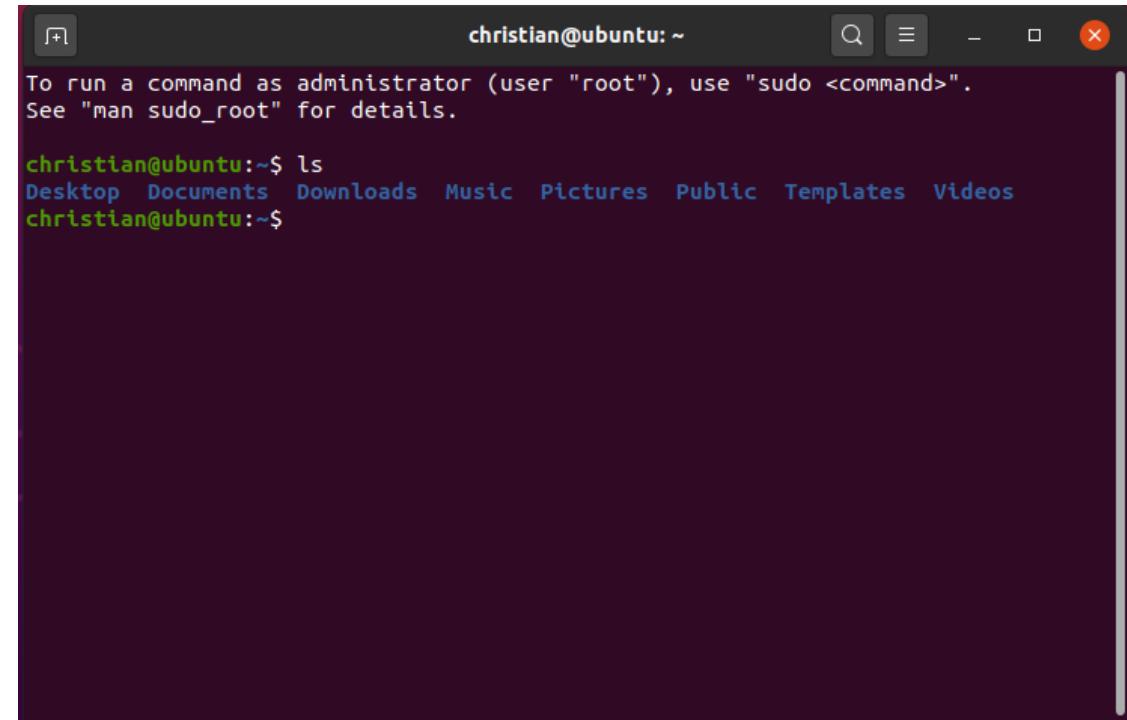
# Interface personne-machine



## Graphique (GUI)



## Ligne de commandes (CLI)



# Commandes textuelles vs. le GUI



Il y a encore des avantages à envoyer des commandes textuelles à un système d'exploitation.

- On peut poser des actions qui ne sont pas disponibles via l'interface graphique.
- Quand on connaît bien les commandes, on gagne en rapidité.
- On peut automatiser des commandes au moyen de scripts.
- On peut réduire l'usage en ressources du système en désactivant son interface graphique.
- Ça impressionne ceux qui ne connaissent pas l'informatique!

# Ligne de commande vs. le GUI



- › Sous Linux, la plupart des actions possibles en interface graphique sont aussi possibles en mode ligne de commande, aussi appelé « Terminal ».

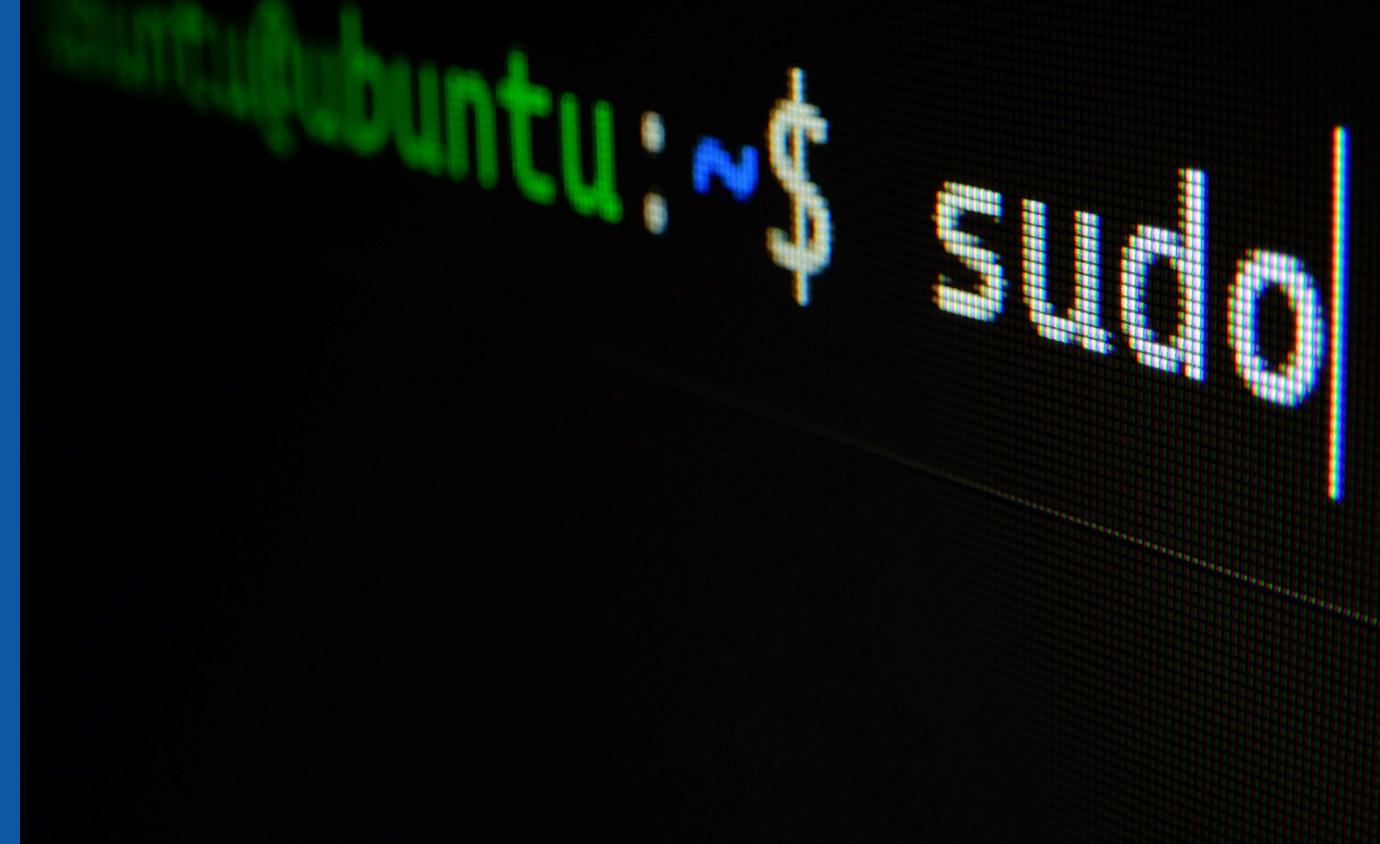
En fait, à la base, Linux fonctionne en mode ligne de commande et c'est l'interface graphique qui est en fait une couche supplémentaire.

# Interpréteurs et shells



- › En informatique, le terme shell désigne une interface personne-machine, soit un moyen pour l'utilisateur d'interagir avec le système d'exploitation. Sans shell, il serait impossible d'utiliser l'ordinateur.
- › Sous Ubuntu Linux, on a deux shells par défaut :
  - › L'interface graphique (GUI)
  - › Le terminal, qui est le shell nommé « **bash shell** »
- › Plusieurs autres shells peuvent être installés et exécutés sur un système Linux, dont sh, csh, zsh, tcsh, ect.

# Le terminal

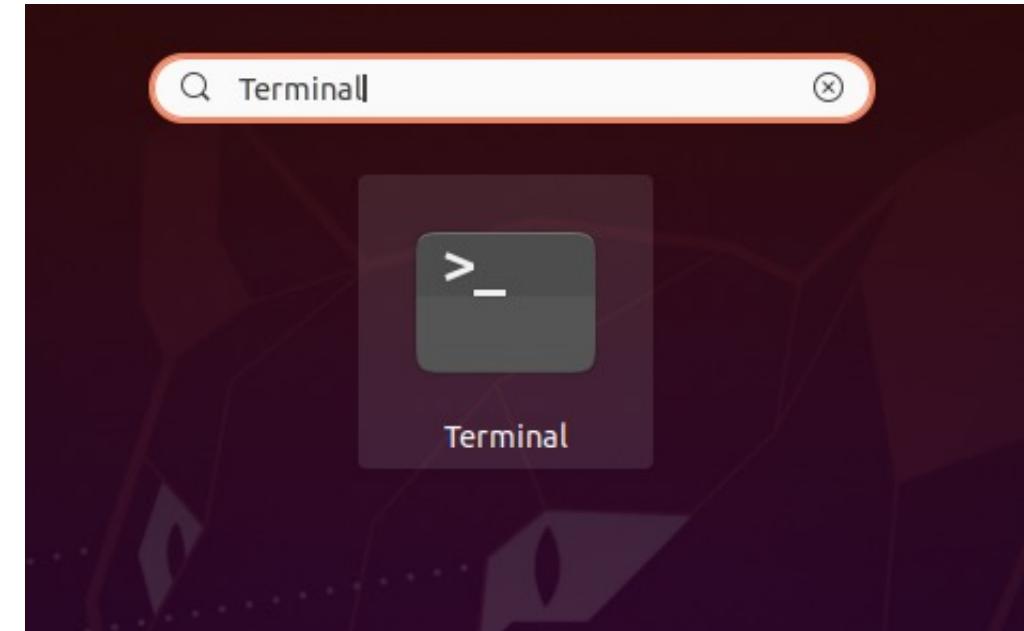
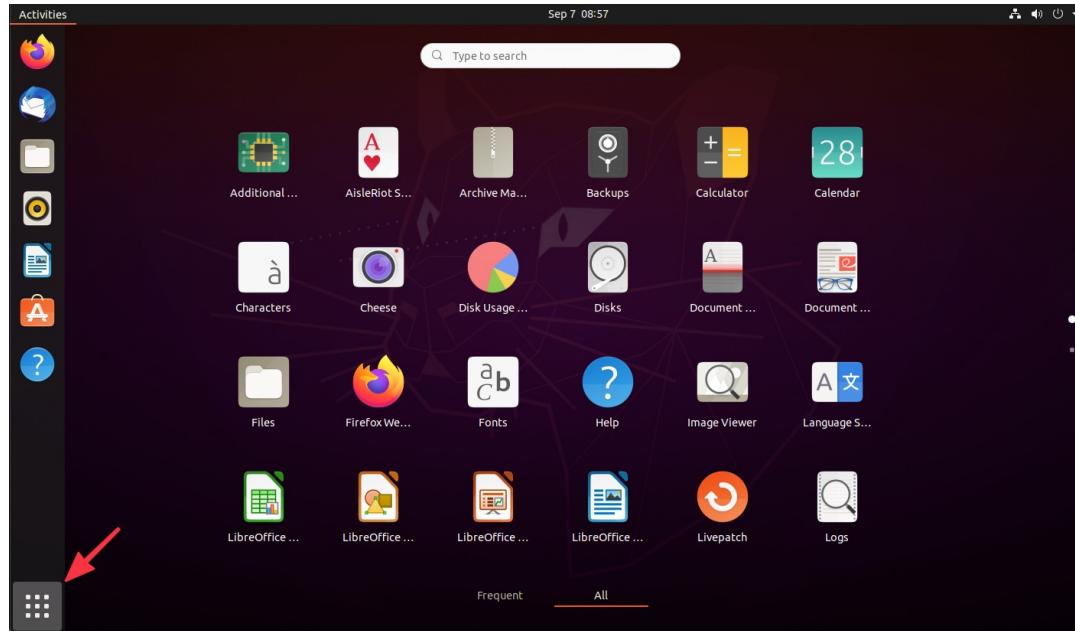


Ubuntu: ~\$ sudo |

# Le terminal



Pour accéder au terminal, il faut accéder à la liste d'applications et chercher « Terminal » :



# Le terminal



- › La console affiche une invite de commandes au début de la ligne. Cette invite rappelle votre nom d'utilisateur, le nom de la machine ainsi que le dossier dans lequel vous vous trouvez.
- › Le symbole `~` signifie que vous êtes dans votre dossier personnel « home directory ».



# Le terminal



- › Chaque commande peut être complétée de **paramètres** qui agissent comme des options pour modifier l'action de la commande.
- › Les paramètres sont généralement constitués d'une lettre précédée d'un tiret (-a) ou d'un mot précédé de deux tirets (--all).

```
etudiant@VD-Ubuntu:~$ ls
Bureau Documents Images Modèles Musique Public Téléchargements travail.txt Vidéos
etudiant@VD-Ubuntu:~$ ls -a
. .bash_logout Bureau .config .gnupg .local Musique Public travail.txt
.. .bashrc .cache Documents Images Modèles .profile Téléchargements Vidéos
etudiant@VD-Ubuntu:~$ ls --all
. .bash_logout Bureau .config .gnupg .local Musique Public travail.txt
.. .bashrc .cache Documents Images Modèles .profile Téléchargements Vidéos
etudiant@VD-Ubuntu:~$
```

# Le système de fichiers



# Le système de fichiers de Linux



- › De base, le système de fichiers de Ubuntu Linux est ext4
- › Cependant, tout comme les autres composantes de Linux, ceci peut être remplacé par plusieurs autres type de système de fichiers.

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
udev	devtmpfs	1.9G	0	1.9G	0%	/dev
tmpfs	tmpfs	201M	1.0M	200M	1%	/run
/dev/sda5	ext4	20G	6.7G	12G	37%	/
tmpfs	tmpfs	2.0G	0	2.0G	0%	/dev/shm
tmpfs	tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	tmpfs	2.0G	0	2.0G	0%	/sys/fs/cgroup
/dev/loop0	squashfs	256M	256M	0	100%	/snap/gnome-3-34-1804/36
/dev/loop1	squashfs	55M	55M	0	100%	/snap/core18/1880
/dev/loop2	squashfs	63M	63M	0	100%	/snap/gtk-common-themes/1506
/dev/loop3	squashfs	30M	30M	0	100%	/snap/snapd/8542
/dev/loop4	squashfs	50M	50M	0	100%	/snap/snap-store/467
/dev/sda1	vfat	511M	4.0K	511M	1%	/boot/efi
tmpfs	tmpfs	391M	28K	391M	1%	/run/user/1000

Nous verrons les systèmes de fichiers plus tard dans la session!

# Le système de fichiers de Linux



- > Le système de fichiers de Linux est organisé en arborescence.
- > Linux n'a qu'une seule **racine** appelé **root** et identifiée par une barre oblique ( / ).
- > Il est composé de dossiers (ou répertoires, ou directories) et de fichiers.
- > La navigation dans Linux se fait avec l'utilisation de la barre oblique ou le slash ( / ).

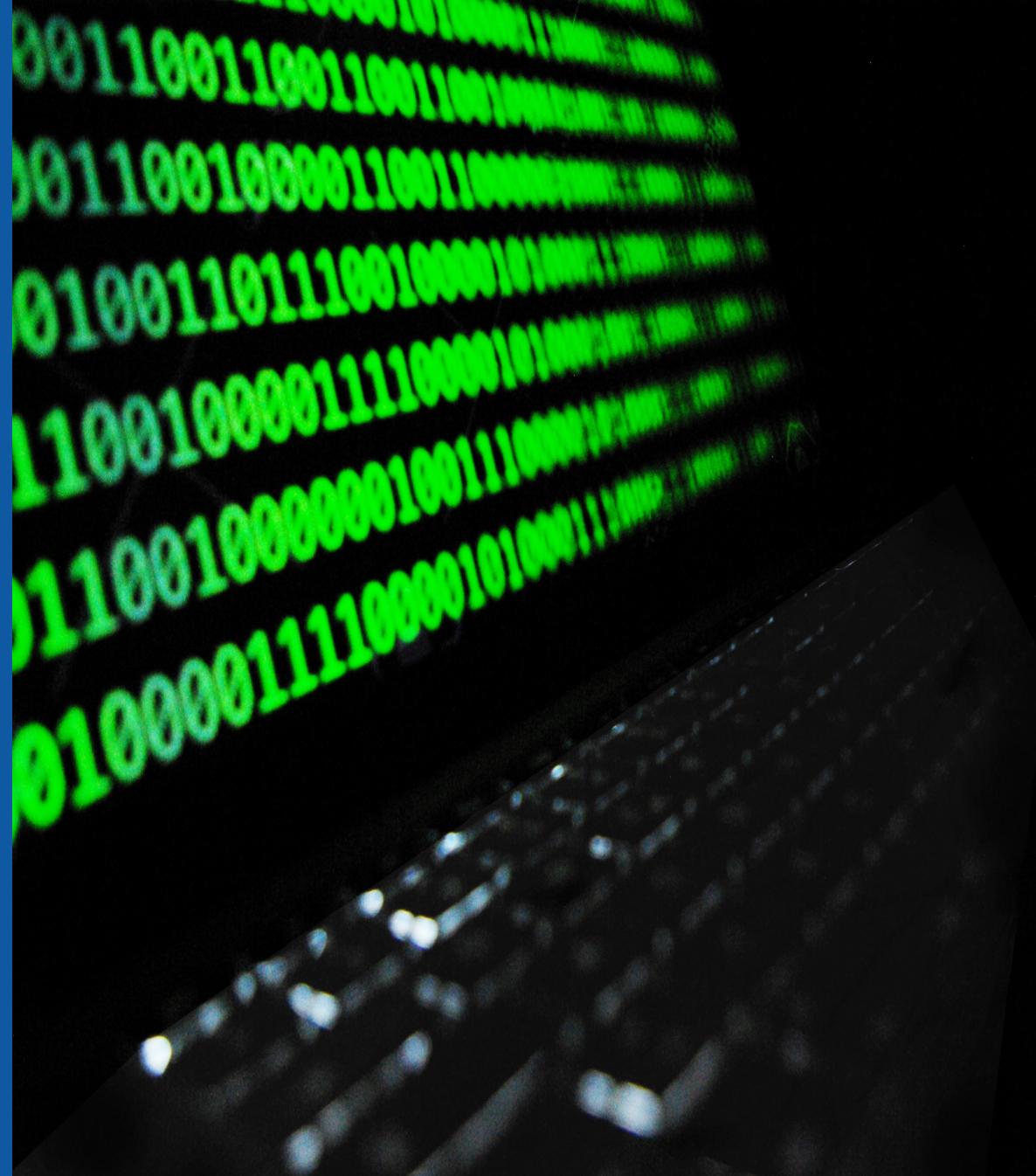
```
christian@ubuntu:~$ ls -ltr
total 945500
drwxr-xr-x  2 root root    4096 Jul 31 09:27 srv
drwxr-xr-x  2 root root    4096 Jul 31 09:27 mnt
drwxr-xr-x  3 root root    4096 Jul 31 09:27 media
drwxr-xr-x 14 root root   4096 Jul 31 09:28 usr
drwx----- 3 root root    4096 Jul 31 09:34 root
drwxr-xr-x 14 root root   4096 Jul 31 09:35 var
drwx----- 2 root root  16384 Sep  7 08:25 lost+found
-rw----- 1 root root 968110080 Sep  7 08:25 swapfile
lrwxrwxrwx  1 root root      10 Sep  7 08:25 libx32 -> usr/libx32
lrwxrwxrwx  1 root root      9 Sep  7 08:25 lib64 -> usr/lib64
lrwxrwxrwx  1 root root      9 Sep  7 08:25 lib32 -> usr/lib32
lrwxrwxrwx  1 root root      7 Sep  7 08:25 lib -> usr/lib
lrwxrwxrwx  1 root root      7 Sep  7 08:25 bin -> usr/bin
lrwxrwxrwx  1 root root      8 Sep  7 08:25 sbin -> usr/sbin
drwxrwxr-x  2 root root   4096 Sep  7 08:28 cdrom
drwxr-xr-x  3 root root   4096 Sep  7 08:28 home
drwxr-xr-x  2 root root   4096 Sep  7 08:33 opt
drwxr-xr-x  8 root root   4096 Sep  7 08:33 snap
drwxr-xr-x  4 root root   4096 Sep  7 08:37 boot
drwxr-xr-x 130 root root 12288 Sep  7 08:37 etc
dr-xr-xr-x 293 root root      0 Sep  7 08:38 proc
dr-xr-xr-x 13 root root      0 Sep  7 08:38 sys
drwxr-xr-x 19 root root   4240 Sep  7 08:38 dev
drwxr-xr-x 32 root root   840 Sep  7 08:38 run
drwxrwxrwt 19 root root  4096 Sep  7 08:48 tmp
```

# Dossiers principaux de Linux



/	Racine du système, contient les répertoires principaux
/bin	Commandes essentielles communes à tous les utilisateurs
/boot	Fichiers de démarrage du système, contient le noyau
/dev	Points d'entrée des périphériques
/etc	Fichiers de configuration
/home	Contient les répertoires personnels des différents utilisateurs
/root	Répertoire personnel de l'administrateur
/usr	Hiérarchie secondaire, applications, bibliothèques partagées
/var	Fichiers trace du système (Logs)
/proc	Système de fichier virtuel, informations en temps réel

# Les commandes de gestion des dossiers et des fichiers



# Anatomie d'une commande



Commande

Paramètres

```
rm -R repertoire "autre repertoire"
```

Switch Paramètre

Paramètre qui comprend des espaces

- › Une ligne de commande commence par le nom de la commande.
- › La commande peut être suivie d'un ou plusieurs paramètres, selon ce qu'on veut faire et ce que la commande accepte.
- › Chaque élément de la ligne est séparé par une espace.
- › Ne pas oublier que Linux est un système d'exploitation sensible à la casse, donc qui fait une différence entre les majuscules et les minuscules.



## ls

- › La commande `ls` vous permet faire la liste des fichiers et dossiers contenus dans le dossier en cours.

```
christian@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
christian@ubuntu:~$ █
```

# Aide sur une commande



- Pour obtenir de l'aide sur une commande, il suffit d'utiliser `man`, suivi en paramètre de la commande pour laquelle vous voulez de l'aide :

```
christian@ubuntu:~$ man ls
```

LS(1)

User Commands

LS(1)

NAME

`ls` - list directory contents

Nom de la commande

SYNOPSIS

`ls` [OPTION]... [FILE]...

Paramètres « switch »

Paramètre texte ou numérique

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

Information sur ce que fait la commande et comment l'utiliser

# Comprendre la syntaxe d'une commande : exemples

## SYNOPSIS

```
ls [OPTION]... [FILE]...
```

**-a**, **--all**

do not ignore entries starting with `.`

**-l**

use a long listing format

```
christian@christian-virtual-machine:/etc$ ls -la
```

```
total 1116
drwxr-xr-x 130 root root 12288 sep 11 06:05 .
drwxr-xr-x 21 root root 4096 sep 14 10:51 ..
drwxr-xr-x 3 root root 4096 avr 23 2020 acpi
-rw-r--r-- 1 root root 3028 avr 23 2020 adduser.conf
drwxr-xr-x 3 root root 4096 avr 23 2020 alsa
drwxr-xr-x 2 root root 4096 aoû 24 23:08 alternatives
-rw-r--r-- 1 root root 401 jui 16 2019 anacrontab
-rw-r--r-- 1 root root 433 oct 1 2017 apg.conf
drwxr-xr-x 5 root root 4096 avr 23 2020 apm
drwxr-xr-x 3 root root 4096 aoû 24 23:11 apparmor
drwxr-xr-x 7 root root 4096 sep 10 06:28 apparmor.d
```

La commande **ls** affiche le dossier actuel avec l'affichage format long et la présence des fichiers cachés (qui commencent par « `.` » (**-la**)).

# Comprendre la syntaxe d'une commande : exemples



## SYNOPSIS

**ls** [OPTION]... [FILE]...

```
christian@christian-virtual-machine:~$ ls -l /
total 753052
lrwxrwxrwx  1 root root      7 aoû 24 22:52 bin  -> usr/bin
drwxr-xr-x  4 root root  4096 sep  9 06:03 boot
drwxrwxr-x  2 root root  4096 aoû 24 22:54 cdrom
drwxr-xr-x 19 root root  4100 sep 14 10:49 dev
drwxr-xr-x  3 root root  4096 sep 14 10:51 dossier1
drwxr-xr-x 130 root root 12288 sep 11 06:05 etc
drwxr-xr-x  3 root root  4096 aoû 24 22:56 home
lrwxrwxrwx  1 root root      7 aoû 24 22:52 lib  -> usr/lib
lrwxrwxrwx  1 root root      9 aoû 24 22:52 lib32 -> usr/lib32
lrwxrwxrwx  1 root root      9 aoû 24 22:52 lib64 -> usr/lib64
lrwxrwxrwx  1 root root     10 aoû 24 22:52 libx32 -> usr/libx32
drwx-----  2 root root 16384 aoû 24 22:52 lost+found
drwxr-xr-x  3 root root  4096 aoû 25 14:52 media
drwxr-xr-x  2 root root  4096 avr 23 2020 mnt
drwxr-xr-x  2 root root  4096 avr 23 2020 opt
dr-xr-xr-x 350 root root      0 sep 14 10:49 proc
drwx-----  4 root root  4096 aoû 25 14:52 root
drwxr-xr-x 34 root root  880 sep 14 10:50 run
lrwxrwxrwx  1 root root      8 aoû 24 22:52 sbin -> usr/sbin
drwxr-xr-x  8 root root 4096 aoû 24 23:01 snap
```

La commande **ls** affiche le contenu du volume **/** (la racine)

# ls (suite)



- > Utiliser l'option **-a** pour afficher les fichiers cachés (fichiers avec un point devant le nom) et **-l** pour la liste détaillée

```
christian@ubuntu:~$ ls
Desktop  Downloads  Fichier2      Fichier.zip  Pictures  Templates
Documents  Fichier1  Fichier_Output  Music      Public    Videos
christian@ubuntu:~$ ls -a
.          .bashrc  Documents  Fichier_Output  .local    .profile  Videos
..         .cache   Downloads  Fichier.zip     .mozilla  Public
.bash_history  .config  Fichier1  .fontconfig   Music    .ssh
.bash_logout  Desktop  Fichier2  .gnupg      Pictures  Templates
christian@ubuntu:~$ ls -l
total 48
drwxr-xr-x 2 christian christian 4096 Sep  7 08:33 Desktop
drwxr-xr-x 2 christian christian 4096 Sep  7 10:20 Documents
drwxr-xr-x 2 christian christian 4096 Sep  7 08:33 Downloads
-rw-rw-r-- 1 christian christian   69 Sep  7 09:55 Fichier1
-rw-rw-r-- 1 christian christian   24 Sep  7 09:57 Fichier2
-rw-rw-r-- 1 christian christian  129 Sep  7 10:09 Fichier_Output
-rw-rw-r-- 1 christian christian  400 Sep  7 10:19 Fichier.zip
drwxr-xr-x 2 christian christian 4096 Sep  7 08:33 Music
drwxr-xr-x 2 christian christian 4096 Sep  7 08:33 Pictures
drwxr-xr-x 2 christian christian 4096 Sep  7 08:33 Public
drwxr-xr-x 2 christian christian 4096 Sep  7 08:33 Templates
drwxr-xr-x 2 christian christian 4096 Sep  7 08:33 Videos
```



# mkdir

- La commande `mkdir` vous permet de créer des dossiers (`mkdir` = make directory).

```
christian@ubuntu:~$ mkdir Repertoire1
christian@ubuntu:~$ ls
Desktop  Downloads  Pictures  Repertoire1  Videos
Documents  Music      Public    Templates
christian@ubuntu:~$
```



- › La commande `rmdir` permet aux utilisateurs de supprimer des dossiers du système.

```
christian@ubuntu:~$ ls
Desktop  Downloads  Pictures  Repertoire1  Videos
Documents  Music      Public    Templates
christian@ubuntu:~$ rmdir Repertoire1
christian@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
christian@ubuntu:~$ █
```

# touch



- › La commande **touch** permet de créer un fichier. Si vous faites touch sur un fichier existant, cela actualisera ses dates de dernier accès et de modification.

```
christian@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
christian@ubuntu:~$ touch Fichier1
christian@ubuntu:~$ ls
Desktop      Downloads  Music      Public      Videos
Documents  Fichier1  Pictures  Templates
christian@ubuntu:~$
```



- › La commande `rm` est utilisée pour supprimer un fichier.

```
christian@ubuntu:~$ ls
Desktop  Downloads  Music  Public  Videos
Documents  Fichier1  Pictures  Templates
christian@ubuntu:~$ rm Fichier1
christian@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
christian@ubuntu:~$ █
```

***Attention : cette commande est irréversible et peut se révéler dangereuse.***



- › La commande **cat** est utilisée pour afficher le contenu d'un fichier l'écran.

```
christian@ubuntu:~$ ls
Desktop    Downloads  Music    Public    Videos
Documents  Fichier1  Pictures  Templates
christian@ubuntu:~$ cat Fichier1
Ceci est le contenu de Fichier 1
Ligne de commande Linux (Semaine 4)
christian@ubuntu:~$ █
```

# head



- › La commande **head** affiche l'en-tête d'un fichier.
- › Par défaut, il imprime les 10 premières lignes du fichier spécifié.

```
christian@ubuntu:~$ head -2 Fichier_Output
Ceci est le contenu de Fichier 1
Ligne de commande Linux (Semaine 4)
christian@ubuntu:~$ █
```

- › Dans la capture d'écran : les 2 premières lignes du fichier Fichier\_Output sont affichées.



- › La commande **tail** affiche les N dernières lignes du fichier.

```
christian@ubuntu:~$ tail -2 Fichier_Output
G H I
J K L
christian@ubuntu:~$
```

- › Dans la capture d'écran : les 2 dernières lignes du fichier **Fichier\_Output** sont affichées.

# more



- > La commande **more** affiche le fichier page par page.
- > Elle est plus pratique pour les longs fichiers.

```
christian@ubuntu:~$ more /etc/apt/sources.list
```

```
#deb cdrom:[Ubuntu 20.04.1 LTS _Focal Fossa_ - Release amd64 (20200731)]/ focal
main restricted

# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://us.archive.ubuntu.com/ubuntu/ focal main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu/ focal main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://us.archive.ubuntu.com/ubuntu/ focal-updates main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu/ focal-updates main restricted

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team. Also, please note that software in universe WILL NOT receive any
## review or updates from the Ubuntu security team.
deb http://us.archive.ubuntu.com/ubuntu/ focal universe
# deb-src http://us.archive.ubuntu.com/ubuntu/ focal universe
deb http://us.archive.ubuntu.com/ubuntu/ focal-updates universe
# deb-src http://us.archive.ubuntu.com/ubuntu/ focal-updates universe

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
--More--(37%)
```



- › La commande **cd** signifie changer de dossier et est utilisée pour changer le dossier actuel dans lequel l'utilisateur opère via Terminal. Elle permet de naviguer dans l'arborescence.

```
christian@ubuntu:~$ cd Documents
christian@ubuntu:~/Documents$
```

- › Cette commande, sans paramètre, vous ramène au dossier home (ou ~)

```
christian@ubuntu:~/Documents$ cd
christian@ubuntu:~$ █
```

# Navigation dans la console



La touche de tabulation (« TAB ») permet de **compléter automatiquement** les commandes et les noms des fichiers et dossiers.

Les touches **haut** et **bas** permettent de naviguer dans l'historique de commandes.

CTRL+C est pratique si vous voulez interrompre l'exécution d'une commande.