

Université Abderrahmane MIRA - Bejaia

Faculté de Technologie
Département Technologie



Logiciel libre et open source

Dr. A. OUARET



ahmed.ouaret@univ-bejaia.dz

Laboratoire de Technologie Industrielle et de l'Information, Faculté de Technologie Université de Bejaia, 06000 Bejaia, Algérie



Présentation de l'auteur



- ***Dr. Ahmed OUARET***
- **Maitre de Conférences classe A**
- **Université Abderrahmane Mira - Bejaia**
- **Contact : ahmed.ouaret@univ-bejaia.dz**
- **Disponibilité au département : Tous les jours**

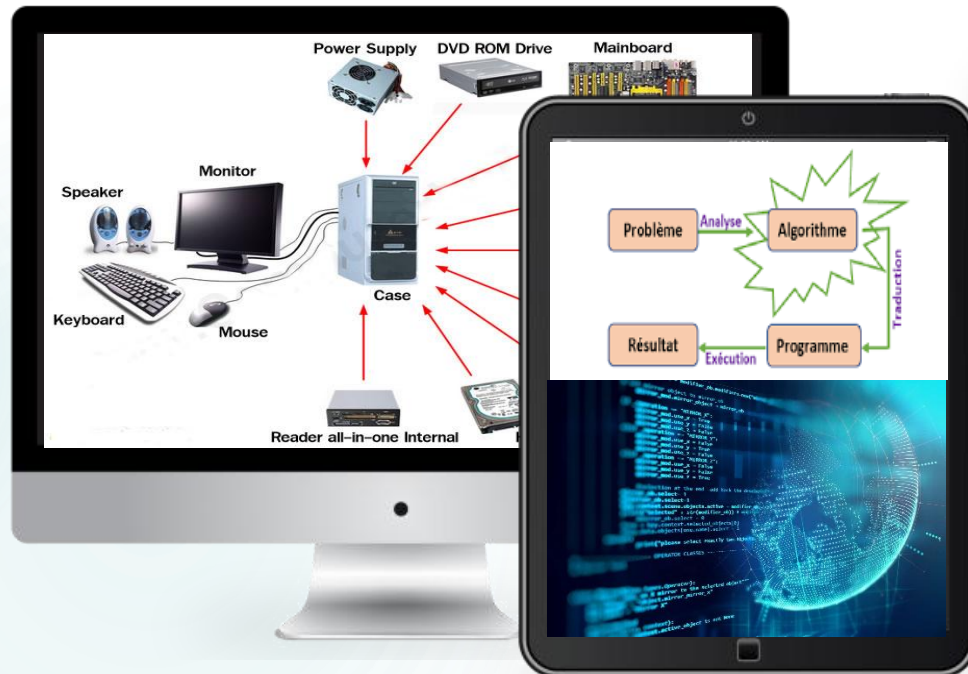
Public cible

Ce cours est destiné aux étudiants de **1^{ère} année Science et Technologie (ST)**



Informations sur le cours

4



● **Crédits : 02**

● **Coefficient : 02**

● **Volume horaire semestriel : 45h00**

● **Volume horaire hebdomadaire : 03h00 (1h30 Cours et 1h30 TP)**

● **Durée : 15 semaines**

Objectifs du cours



- 1 Distinguer entre un logiciel libre et un logiciel open source ainsi que leur importance dans l'univers numérique actuel.
- 2 Maîtriser les aspects juridiques liés aux licences des logiciels libres.
- 3 Utiliser des environnements libres comme GNU/Linux et des logiciels libres dans un cadre académique et professionnel.
- 4 Apprendre à utiliser le terminal
- 5 Maîtriser des outils bureautiques libres (Writer, Calc)

Pré-requis

Pour bien suivre ce cours, l'étudiant doit avoir acquis certaines connaissances :



Notions de base en informatique (utilisation d'un ordinateur, navigation web, manipulation de fichiers)



Connaissance générale des systèmes d'exploitation



Mode d'évaluation

Contrôle continu : 40 %

Examen : 60%



Participation (10%)



Travail personnel (15%)



Tests (15%)



Examen final sur table (60%)

Contenu du cours

Le cours de **logiciel libre et open source** contient quatre principaux chapitres :

1 Chapitre 1 : Fondements du logiciel libre

2 Chapitre 2 : Cadre juridique et licences

3 Chapitre 3 : Systèmes d'exploitation libres

4 Chapitre 4 : Solutions bureautiques libres

Chapitre 1

Fondements du logiciel libre

1. Introduction :

Le développement des technologies numériques a conduit à une large utilisation de logiciels propriétaires, dont le code source est fermé et contrôlé par des entreprises.

En réponse à ces limites, le mouvement du logiciel libre est apparu avec pour objectif de redonner aux utilisateurs le contrôle sur les logiciels qu'ils utilisent.

Les logiciels libres et open sources proposent une autre approche :

- Un accès au code source
- La possibilité de comprendre, modifier et partager les logiciels

2. Différence entre "free software" et "open source" :

La différence entre le « **free software** » (logiciel libre) et « **l'open source** » repose principalement sur la philosophie de ces deux approches, même si, dans la pratique, elles reposent souvent sur des licences similaires. Le free software, défendu par Richard Stallman et la Free Software Foundation, met l'accent sur la liberté des utilisateurs et sur une **vision éthique et sociale du logiciel**. Il insiste sur les quatre libertés fondamentales :

- Utiliser le programme
- Etudier son fonctionnement
- Modifier
- Redistribuer.

2. Différence entre "free software" et "open source" :

À l'inverse, le terme open source a été introduit pour promouvoir le partage du code sous un angle plus **économique**, en mettant en avant des avantages comme la **qualité** du code, la **sécurité**, la **fiabilité** et la **rapidité d'innovation**, sans nécessairement adopter le discours idéologique du logiciel libre.

Exemple

En open source : le code est ouvert mais la licence peut autoriser :

- qu'une entreprise prenne le code
- le modifie
- le referme
- et redistribue une version propriétaire

2. Différence entre "free software" et "open source" :

En résumé, un logiciel libre doit rester libre, tandis qu'un logiciel open source peut, dans certains cas, devenir fermé selon sa licence.

⚠ Attention

le terme free software peut prêter à confusion, car il ne signifie pas « gratuit », mais libre, c'est-à-dire le respect des libertés d'utilisation, de modification et de partage du logiciel.

💬 Remarque

un logiciel free software est toujours open source, mais un logiciel open source n'est pas forcément free software, car il peut ne pas garantir toutes les libertés des utilisateurs.

3. Philosophie de Richard Stallman et le projet GNU

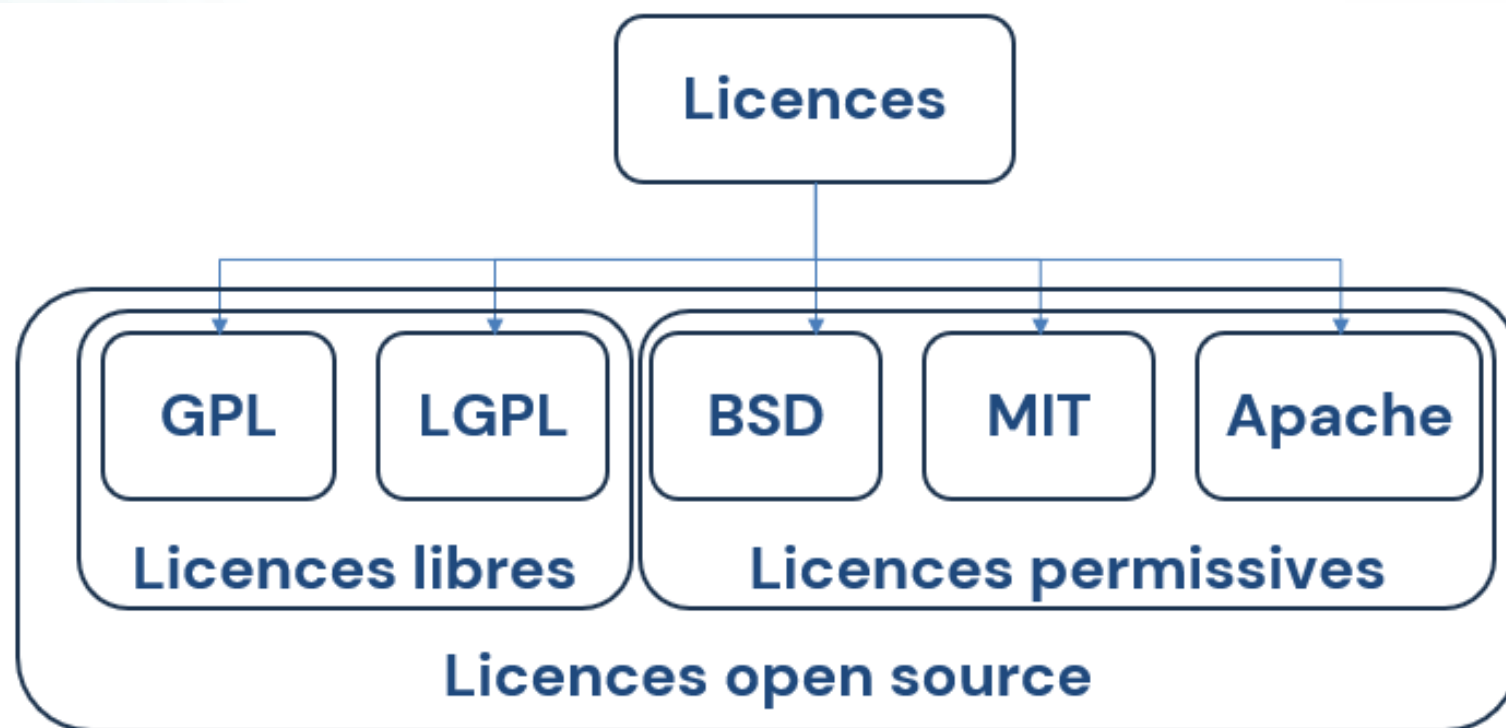
La philosophie de **Richard Stallman** défend la liberté des utilisateurs de **utiliser, modifier et partager les logiciels**. Face aux logiciels propriétaires qui limitent ces libertés, il lance en 1983 le **GNU** pour créer un système entièrement libre. Afin de garantir ces libertés dans le temps, il introduit le copyleft, notamment via la licence **GNU GPL**, qui impose que toute version modifiée reste libre.

Chapitre 2

Cadre juridique et licences

1. Cadre juridique et licences :

Les licences libres sont des instruments juridiques essentiels qui définissent précisément les droits et obligations liées à l'utilisation, la modification et la redistribution des logiciels.



2. Compatibilité entre licences

La compatibilité entre licences permet de mélanger des codes de différentes licences sans conflit juridique.

Elle existe lorsque les règles sont compatibles, sinon il y a incompatibilité si elles se contredisent.

👁 Exemple

- Un logiciel sous licence MIT peut être intégré dans un logiciel sous licence GPL. (La licence MIT est permissive : elle autorise la modification, la redistribution et l'intégration dans d'autres projets.)
- Un logiciel sous licence GPL ne peut pas être intégré dans un logiciel propriétaire. (La GPL impose que toute redistribution reste libre)

Az Définition

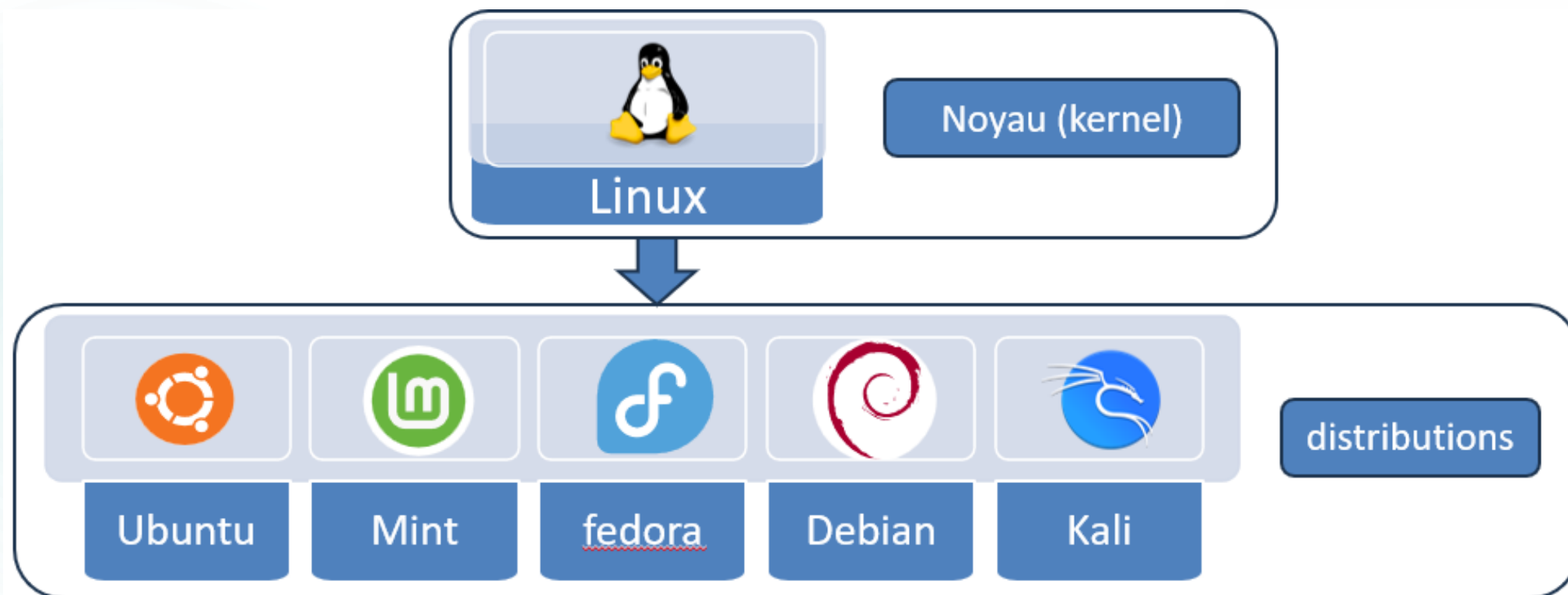
Une licence permissive (MIT, BSD) est compatible avec une licence libre stricte (GPL), mais l'inverse n'est pas possible.

Chapitre 3

Systemes d'exploitation libres

1. Introduction et présentation des distributions

Un système d'exploitation libre est un système dont le code source est accessible et qui garantit aux utilisateurs les libertés fondamentales.



■ Distributions pour usage bureautique / personnel

Ces distributions sont conçues pour un usage quotidien (navigation Internet, bureautique, multimédia).

👁 Exemple

Ubuntu, Linux Mint , Zorin OS



Les distributions présentées, de gauche à droite, sont : Ubuntu, Linux Mint et Zorin OS.

■ Distributions Linux pour l'éducation

Ces distributions sont conçues pour l'enseignement, les écoles, les lycées et parfois les universités. les Logiciels éducatifs préinstallés.

👁 Exemple

Edubuntu, Endless OS



Les distributions présentées, de gauche à droite, sont : Edubuntu, Endless OS

■ Distributions pour la sécurité et les réseaux

Ces distributions sont spécialisées dans la cybersécurité et les tests réseaux.

👁 Exemple

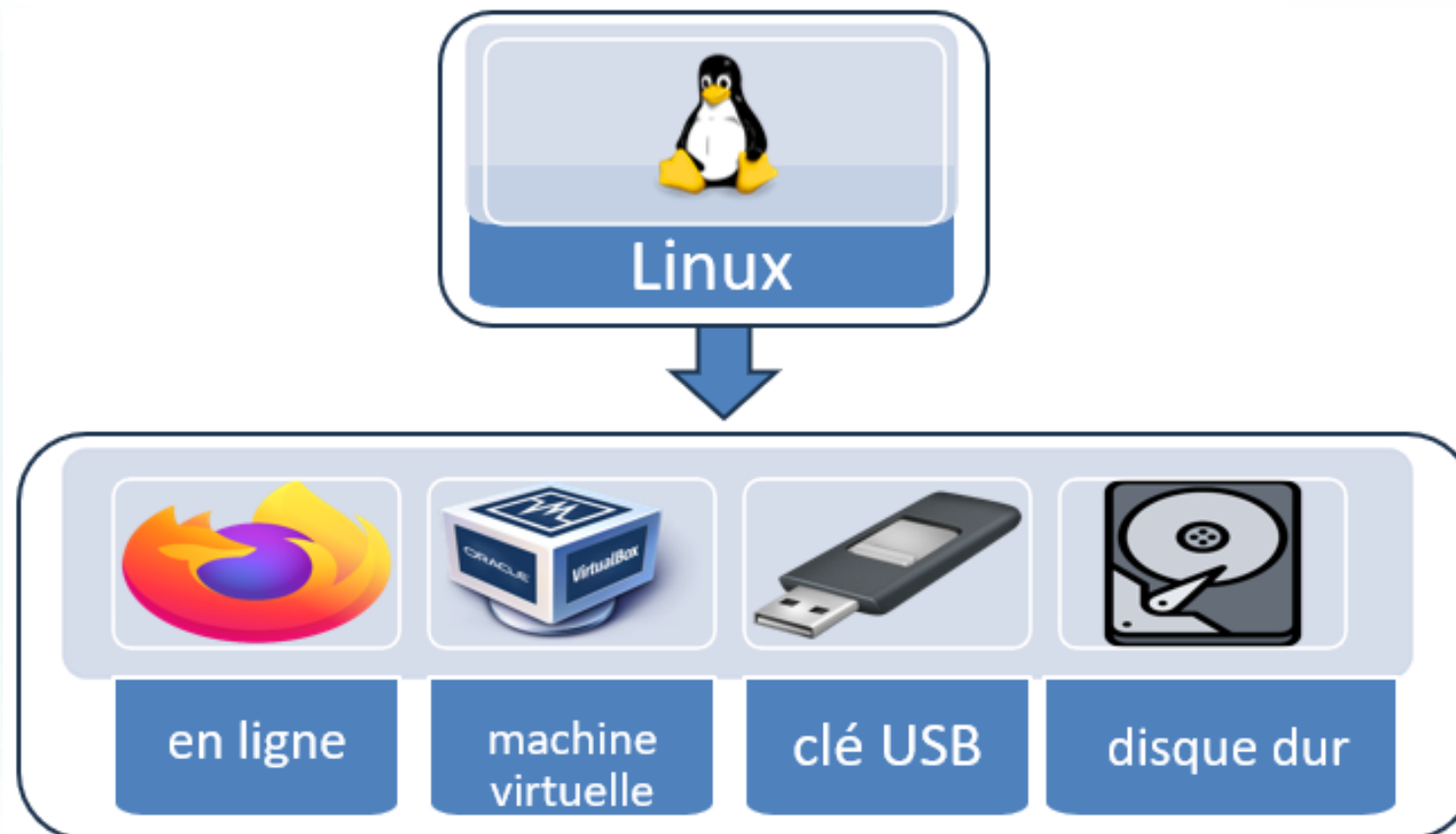
Kali Linux, Parrot OS



Les distributions présentées, de gauche à droite, sont : Kali linux, Parrot OS

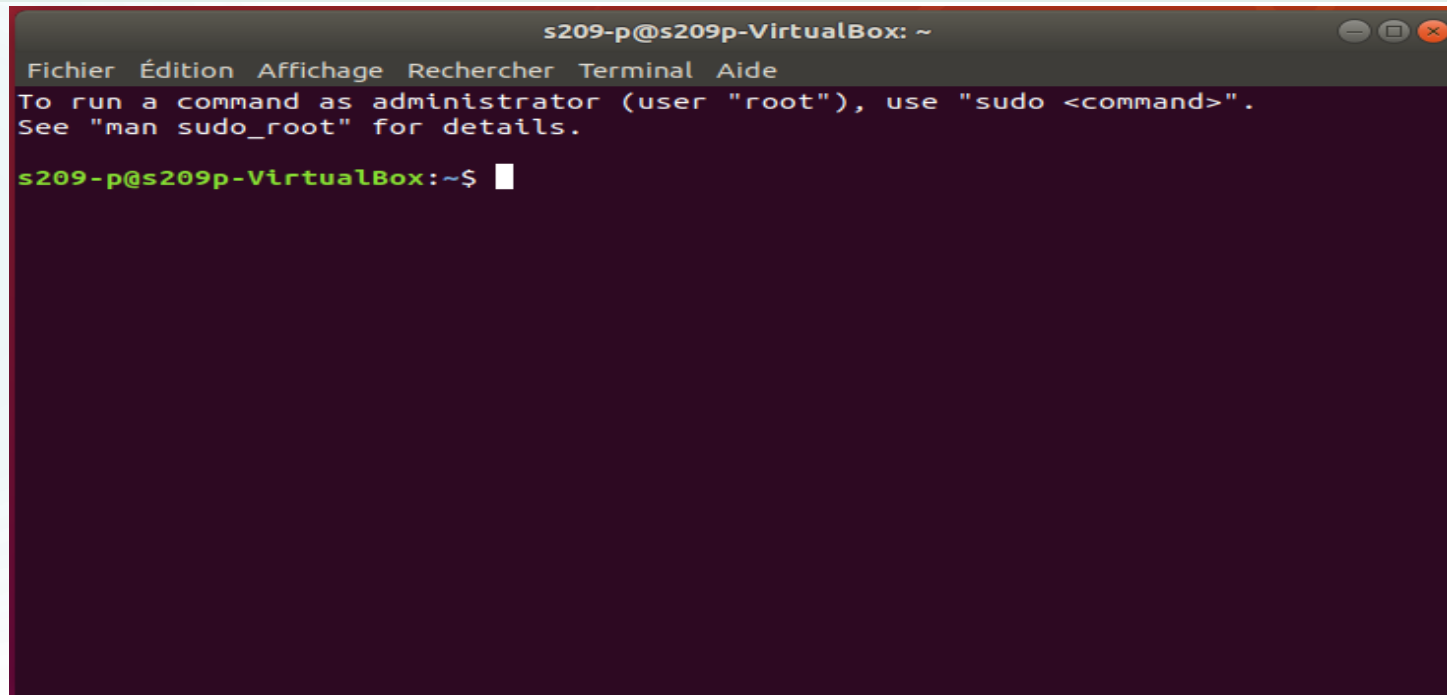
3. Principes d'installation et configuration de base

Il existe plusieurs façons d'utiliser Linux : en ligne via un navigateur, en l'installant sur une clé USB, sur le disque dur d'un ordinateur, ou en utilisant une machine virtuelle.



4. Commandes fondamentales

Le terminal est un outil essentiel dans les systèmes Linux. Il permet à l'utilisateur de communiquer directement avec le système à l'aide de commandes textuelles afin d'exécuter des tâches telles que la gestion des fichiers, l'installation de logiciels ou l'administration du système.



```
s209-p@s209p-VirtualBox: ~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
s209-p@s209p-VirtualBox:~$
```

4.1. Commandes système essentielles

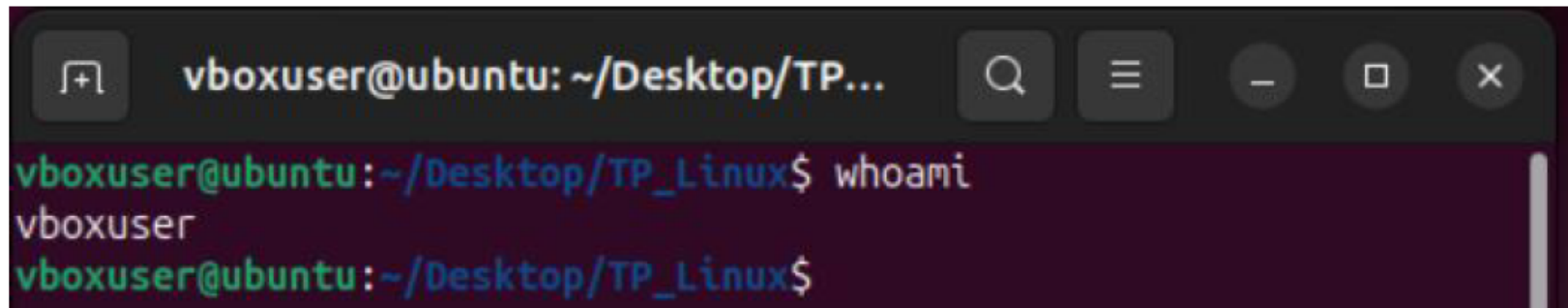
- **Afficher le nom de l'utilisateur actuellement connecté au système**

Cette commande affiche le nom de l'utilisateur actuellement connecté au système.

whoami

Syntaxe

Exemple



```
vboxuser@ubuntu: ~/Desktop/TP...  
vboxuser@ubuntu:~/Desktop/TP_Linux$ whoami  
vboxuser  
vboxuser@ubuntu:~/Desktop/TP_Linux$
```

■ Espace disque disponible

Cette commande montre l'espace disque disponible et utilisé sur l'ordinateur. L'option `df -h` permet d'afficher les tailles de manière lisible (en Go et Mo).

`df -h`

Syntaxe


Exemple

```
vboxuser@ubuntu: ~/Desktop/TP_Linux
vboxuser@ubuntu:~/Desktop/TP_Linux$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           383M  1.5M  381M   1% /run
/dev/sda2       25G   6.3G   17G  27% /
tmpfs           1.9G   0    1.9G   0% /dev/shm
tmpfs           5.0M   8.0K  5.0M   1% /run/lock
tmpfs           383M  124K  382M   1% /run/user/1000
vboxuser@ubuntu:~/Desktop/TP_Linux$
```

■ Quantité de mémoire vive (RAM)

Cette commande affiche la quantité de mémoire vive (RAM) utilisée et disponible. L'option **free -h** rend l'affichage plus clair pour l'utilisateur.

free -h

 Syntaxe

 Exemple

```
vboxuser@ubuntu:~/Desktop/TP_Linux$ free -h
              total        used         free       shared  buff/cache   available
Mem:          3.7Gi        987Mi        878Mi         28Mi         2.2Gi         2.8Gi
Swap:          0B           0B           0B
```


4.2. Navigation dans le système de fichiers

Ces commandes permettent de se repérer dans le système de fichiers et de se déplacer entre les dossiers.

- **Afficher le chemin du dossier courant**

Cette commande permet d'afficher le chemin du dossier courant.

pwd

 Syntaxe

 Exemple


```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ pwd  
/home/ubuntu  
ubuntu@ubuntu:~$
```

4.2. Navigation dans le système de fichiers

- **Afficher la liste des fichiers et dossiers**

Cette commande permet d'afficher la liste des fichiers et dossiers courant.

ls

 Syntaxe

 Exemple

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

4.2. Navigation dans le système de fichiers

- **Afficher les fichiers avec des informations détaillées**

Cette commande permet d'afficher les fichiers avec des informations détaillées.

ls -l

 Syntaxe

 Exemple


```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ ls -l  
total 0  
drwxr-xr-x 2 ubuntu ubuntu 60 Dec 28 19:51 Desktop  
drwxr-xr-x 2 ubuntu ubuntu 40 Dec 28 19:52 Documents  
drwxr-xr-x 2 ubuntu ubuntu 40 Dec 28 19:52 Downloads  
drwxr-xr-x 2 ubuntu ubuntu 40 Dec 28 19:52 Music  
drwxr-xr-x 2 ubuntu ubuntu 40 Dec 28 19:52 Pictures  
drwxr-xr-x 2 ubuntu ubuntu 40 Dec 28 19:52 Public  
drwxr-xr-x 2 ubuntu ubuntu 40 Dec 28 19:52 Templates  
drwxr-xr-x 2 ubuntu ubuntu 40 Dec 28 19:52 Videos  
drwx----- 5 ubuntu ubuntu 100 Dec 28 19:53 snap  
ubuntu@ubuntu:~$
```


4.2. Navigation dans le système de fichiers

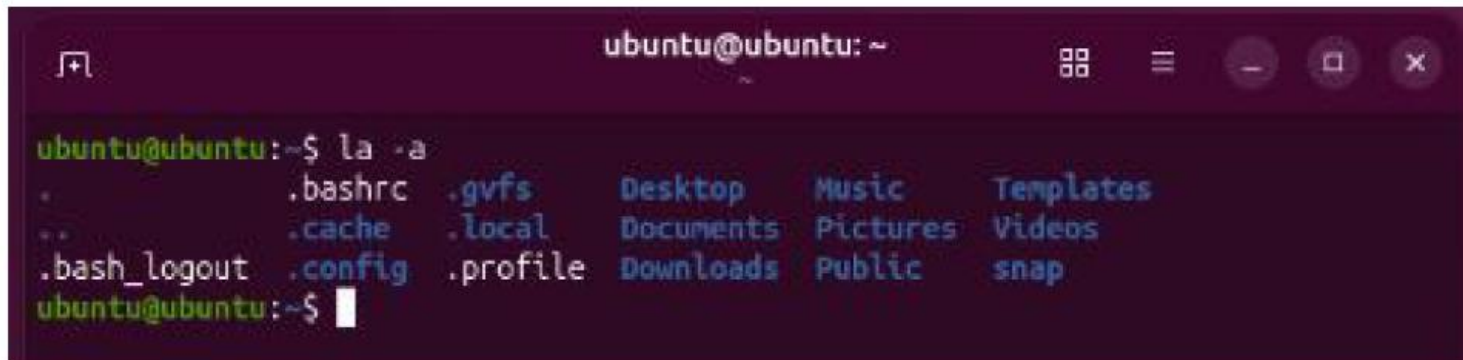
- **Afficher tous les fichiers, y compris les fichiers cachés**

Cette commande permet d'afficher tous les fichiers, y compris les fichiers cachés.

ls -a

 Syntaxe

 Exemple




```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ ls -a  
.          .bashrc   .gvfs     Desktop   Music     Templates  
..         .cache    .local    Documents Pictures   Videos  
.bash_logout .config  .profile  Downloads Public     snap  
ubuntu@ubuntu:~$
```

4.2. Navigation dans le système de fichiers

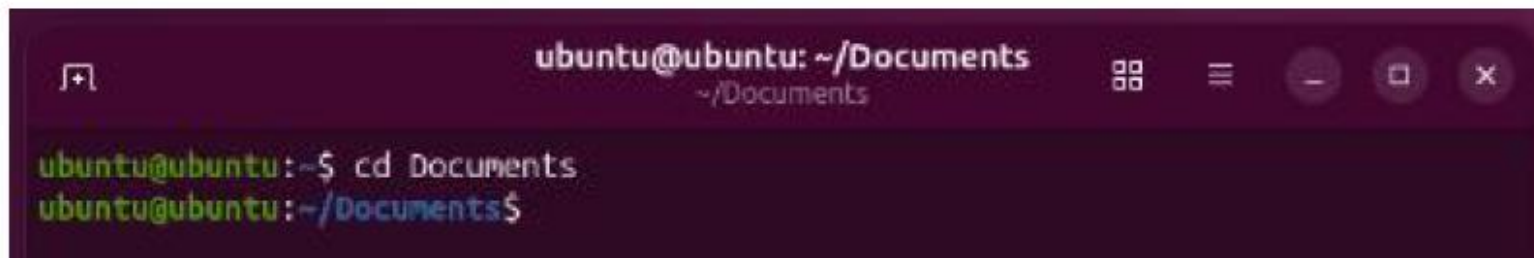
■ Accéder à un dossier

Cette commande permet d'accéder à un dossier.

```
cd nom_dossier
```

 Syntaxe

 Exemple



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

4.2. Navigation dans le système de fichiers

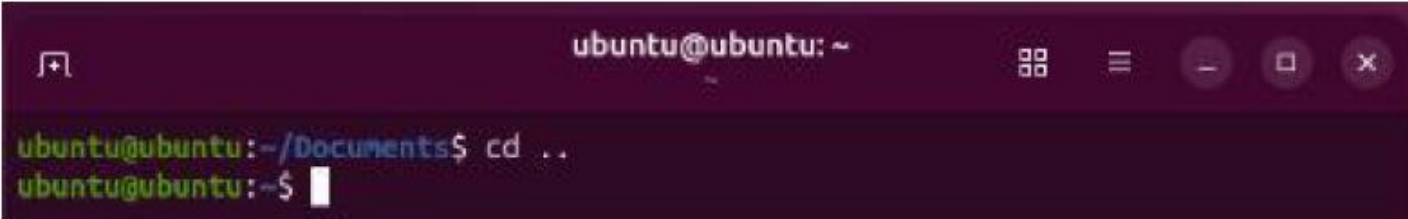
■ Revenir au dossier parent

Cette commande permet de revenir au dossier parent.

cd ..

 Syntaxe

 Exemple



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

4.2. Navigation dans le système de fichiers

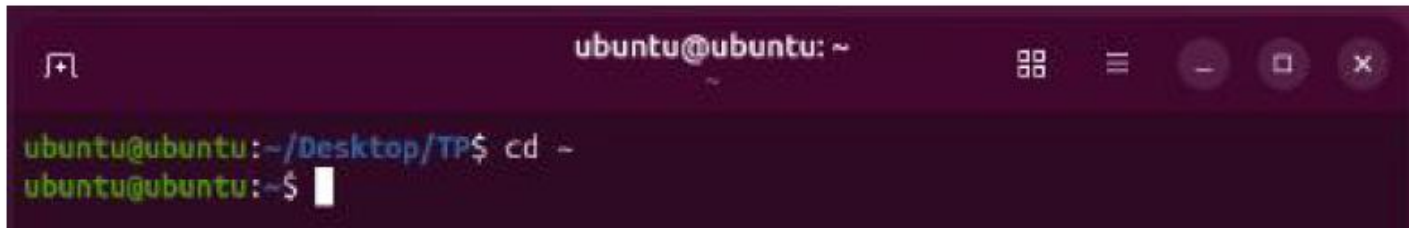
- **Accéder au dossier personnel de l'utilisateur**

Cette commande permet d'accéder au dossier personnel de l'utilisateur.

cd ~

 Syntaxe

 Exemple




```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~/Desktop/TP$ cd ~  
ubuntu@ubuntu:~$
```

4.2. Navigation dans le système de fichiers

■ Nettoyer l'écran du terminal

Cette commande permet de nettoyer l'écran du terminal.

clear

 Syntaxe

 Exemple

```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~/Desktop/TP$ cd -  
ubuntu@ubuntu:~$ ls  
Desktop  Downloads  Pictures  Templates  snap  
Documents Music      Public    Videos  
ubuntu@ubuntu:~$ clear
```


```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ █
```

4.3. Manipulation de fichiers et dossiers

■ Créer un nouveau dossier

Cette commande permet de créer un nouveau dossier.

```
mkdir nom_dossier
```

 Syntaxe

 Exemple


```
ubuntu@ubuntu: ~/Desktop
~/Desktop
ubuntu@ubuntu:~/Desktop$ mkdir TP_linux
ubuntu@ubuntu:~/Desktop$
```


4.3. Manipulation de fichiers et dossiers

■ Créer un nouveau dossier

Cette commande permet de créer un nouveau dossier.

```
mkdir nom_dossier
```

 Syntaxe

 Exemple




```
ubuntu@ubuntu: ~/Desktop
ubuntu@ubuntu:~/Desktop$ mkdir TP_linux
ubuntu@ubuntu:~/Desktop$
```

4.3. Manipulation de fichiers et dossiers

■ Créer un fichier vide

Cette commande permet de créer un fichier texte vide.

```
touch Nom_fichier.txt
```

 Syntaxe


 Exemple

```
ubuntu@ubuntu: ~/Desktop/TP_linux
~/Desktop/TP_linux
ubuntu@ubuntu:~/Desktop/TP_linux$ touch fichier1.txt
ubuntu@ubuntu:~/Desktop/TP_linux$
```

4.3. Manipulation de fichiers et dossiers

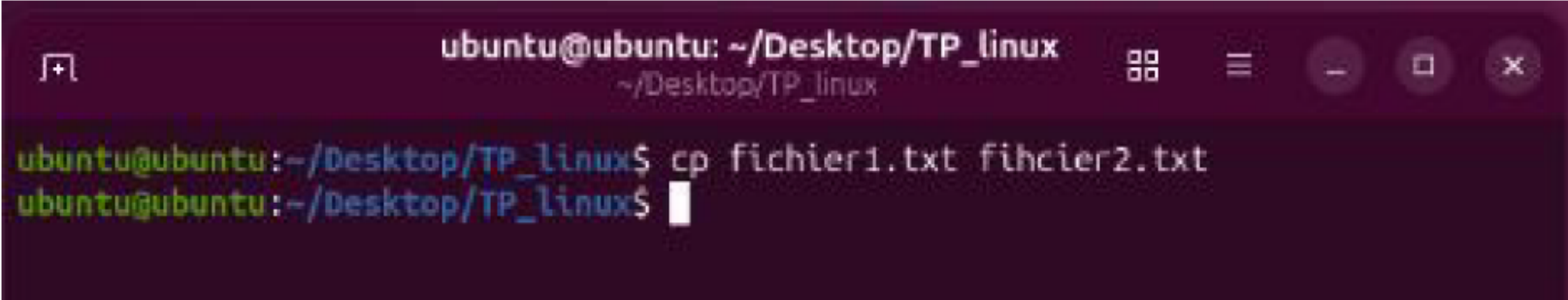
■ Copier un fichier

Cette commande permet de copier un fichier.

 Syntaxe

```
cp fichier_a_copier fichier_copie
```

 Exemple




```
ubuntu@ubuntu: ~/Desktop/TP_linux
~/Desktop/TP_linux
ubuntu@ubuntu:~/Desktop/TP_linux$ cp fichier1.txt fihcier2.txt
ubuntu@ubuntu:~/Desktop/TP_linux$
```

4.3. Manipulation de fichiers et dossiers

■ Copier un dossier avec son contenu

Cette commande permet de copier un dossier avec son contenu.

 Syntaxe

```
cp -r dossier_a_copier dossier_copie
```

 Exemple




```
ubuntu@ubuntu: ~/Desktop/TP_linux
~/Desktop/TP_linux
ubuntu@ubuntu:~/Desktop/TP_linux$ cp -r dossier1 dossier2
ubuntu@ubuntu:~/Desktop/TP_linux$
```

4.3. Manipulation de fichiers et dossiers

■ Déplacer ou de renommer un fichier

Cette commande permet de déplacer ou de renommer un fichier.

 Syntaxe

déplacer un fichier

```
mv fichier_a_deplacer dossier_destination
```

renommer un fichier

```
mv fichier_a_renommer nouveau_nom
```

4.3. Manipulation de fichiers et dossiers

- Déplacer ou de renommer un fichier

👁 Exemple

```
ubuntu@ubuntu: ~/Desktop/TP_linux
~/Desktop/TP_linux

ubuntu@ubuntu:~/Desktop/TP_linux$ mv fichier1.txt dossier1
ubuntu@ubuntu:~/Desktop/TP_linux$
```

```
vboxuser@ubuntu: ~/Desktop/TP_linux
~/Desktop/TP_linux


vboxuser@ubuntu:~/Desktop/TP_linux$ mv fichier1.txt fichier2.txt
vboxuser@ubuntu:~/Desktop/TP_linux$
```

4.3. Manipulation de fichiers et dossiers

■ Supprimer un fichier

Cette commande permet de supprimer un fichier.

rm fichier

 Syntaxe

 Exemple




```
ubuntu@ubuntu: ~/Desktop/TP_linux
~/Desktop/TP_linux
ubuntu@ubuntu:~/Desktop/TP_linux$ rm fichier1.txt
ubuntu@ubuntu:~/Desktop/TP_linux$
```


4.3. Manipulation de fichiers et dossiers

■ Supprimer un dossier vide

Cette commande permet de supprimer un dossier vide.

 Syntaxe

```
rmdir nom_dossier
```

 Exemple




```
ubuntu@ubuntu: ~/Desktop
~/Desktop
ubuntu@ubuntu:~/Desktop$ rmdir TP_linux
ubuntu@ubuntu:~/Desktop$
```

4.3. Manipulation de fichiers et dossiers

■ Supprimer un dossier et son contenu

Cette commande permet de supprimer un dossier et son contenu

`rm -r dossier`

 Syntaxe

 Exemple

```
ubuntu@ubuntu: ~/Desktop/TP_linux
~/Desktop/TP_linux


ubuntu@ubuntu:~/Desktop/TP_linux$ rm -r dossier1
ubuntu@ubuntu:~/Desktop/TP_linux$
```

4.4. Consultation et édition de fichiers texte

- Affiche le contenu d'un fichier

Cette commande affiche tout le contenu du fichier.

```
cat fichier.txt
```

 Syntaxe

```
vboxuser@ubuntu:~/Desktop/TP_Linux$ cat fichier1.txt  
TP Linux  
vboxuser@ubuntu:~/Desktop/TP_Linux$
```


 Exemple

4.4. Consultation et édition de fichiers texte

■ Ecrire du texte dans un fichier

Cette commande permet d'écrire du texte dans un fichier.

```
echo "texte" > fichier.txt
```

 Syntaxe

 Exemple

```
vboxuser@ubuntu:~/Desktop/TP_Linux$ echo "text" > fichier1.txt  
vboxuser@ubuntu:~/Desktop/TP_Linux$
```


4.4. Consultation et édition de fichiers texte

▪ Ecrire du texte dans un fichier

 Attention

Si le fichier existe, son contenu est remplacé.

Pour ajouter du texte à la fin d'un fichier existant, il suffit de remplacer le symbole `>` par `>>`.


 Syntaxe

```
echo "texte" >> fichier.txt
```


4.4. Consultation et édition de fichiers texte

- Ouvrir et modifier un fichier texte à l'aide de l'éditeur nano

```
nano fichier.txt
```

 Syntaxe

Pour enregistrer : **Ctrl + O**, pour quitter : **Ctrl + X**.

 Remarque

4.4. Consultation et édition de fichiers texte

- Ouvrir et modifier un fichier texte à l'aide de l'éditeur nano



Exemple

```
vboxuser@ubuntu:~/Desktop/TP_Linux$ nano fichier1.txt
```

GNU nano 7.2 fichier1.txt
text

[Read 1 line]


^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute
^X Exit	^R Read File	^I Replace	^U Paste	^J Justify

4.5. Aide et documentation

- **Afficher le manuel d'une commande**

Cette commande permet d'afficher le manuel d'une commande.

```
man commande
```


 Syntaxe

4.5. Aide et documentation

- **Afficher une aide rapide**

Cette commande permet d'afficher une aide rapide.

```
commande --help
```

 Syntaxe

Conclusion

- ❖ Le logiciel libre repose sur des principes de liberté et de partage.
- ❖ Il est encadré par des licences juridiques qui garantissent ces libertés.
- ❖ Linux existe en plusieurs distributions chaque distribution répond à un besoin spécifique.
- ❖ Le terminal est un outil qui permet de communiquer avec le système à l'aide de commandes textuelles.



Logiciels Libres et Open Source

Série de TP N°1– Navigation dans l'interface et utilisation des commandes de base

But de TP :

Initier l'étudiant à l'utilisation du terminal Linux pour lui permettre de : naviguer dans le système, gérer des fichiers et dossiers, et manipuler des fichiers texte de manière autonome.

Travail demandé :

Partie A : Navigation dans l'interface

- Accéder au menu des applications.
- Localiser l'application Terminal.
- Ouvrir le terminal.

Partie B : commandes de base

- Exécuter une commande permettant d'afficher le nom de l'utilisateur connecté.
- Exécuter une commande permettant d'afficher le répertoire courant.
- afficher le contenu du répertoire courant.
- Afficher le contenu du répertoire courant sous forme détaillée.
- Afficher tous les fichiers y compris les fichiers cachés.
- Se déplacer vers le répertoire Documents.
- Revenir au répertoire parent.
- Accéder au dossier Bureau
- Créer un dossier nommé TP_Linux dans le répertoire personnel
- Accéder au dossier TP_Linux.
- Créer deux fichiers texte nommés :
 - fichier1.txt
 - fichier2.txt
- Créer un dossier nommé data.
- Déplacer fichier1.txt vers le dossier data.
- Supprimer le fichier fichier2.txt.
- Supprimer le dossier data.
- Créer un fichier nommé message.txt.
- Écrire une phrase de votre choix dans ce fichier.
- Ajouter une deuxième phrase sans supprimer la première.
- Afficher le contenu du fichier dans le terminal.
- Ouvrir le fichier en mode édition (nano) et ajouter une troisième phrase.
- Revenir au répertoire parent.
- Supprimer le dossier TP_Linux

Logiciels Libres et Open Source

Rappel des commandes de base

Command	Syntaxe	Description
Commandes système essentielles		
whoami	whoami	affiche le nom de l'utilisateur actuellement connecté au système
df	df -h	montre l'espace disque disponible
free	free -h	affiche la quantité de mémoire vive (RAM) utilisée et disponible
Navigation dans le système de fichiers		
pwd	pwd	afficher le chemin du dossier courant.
ls	ls	afficher la liste des fichiers et dossiers
	ls -l	afficher les fichiers avec des informations détaillées
	ls -a	afficher tous les fichiers, y compris les fichiers cachés
cd	cd nom_dossier	accéder à un dossier
	cd ..	revenir au dossier parent
	cd ~	accéder au dossier personnel de l'utilisateur
Manipulation de fichiers et dossiers		
mkdir	mkdir nom_dossier	permet de créer un nouveau dossier
touch	touch Nom_fichier.txt	commande permet de créer un fichier texte vide
cp	cp fichier_a_copier fichier_copie	copier un fichier.
	cp -r dossier_a_copier dossier_copie	copier un dossier avec son contenu
mv	mv fichier_a_deplacer dossier_destination	déplace un fichier
	mv fichier_a_renommer nouveau_nom	renommer un fichier
rm	rm fichier	supprimer un fichier
	rm -r dossier	supprimer un dossier et son contenu
rmdir	rmdir nom_dossier	supprimer un dossier vide
Consultation et édition de fichiers texte		
cat	cat fichier.txt	affiche le contenu d'un fichier
echo	echo "texte" > fichier.txt	écrire du texte dans un fichier
	echo "texte" >>fichier.txt	Pour ajouter du texte à la fin d'un fichier existant
nano	nano fichier.txt	ouvrir et modifier un fichier texte à l'aide de l'éditeur nano