

# Linux - Bases

lundi 12 décembre 2022 09:01

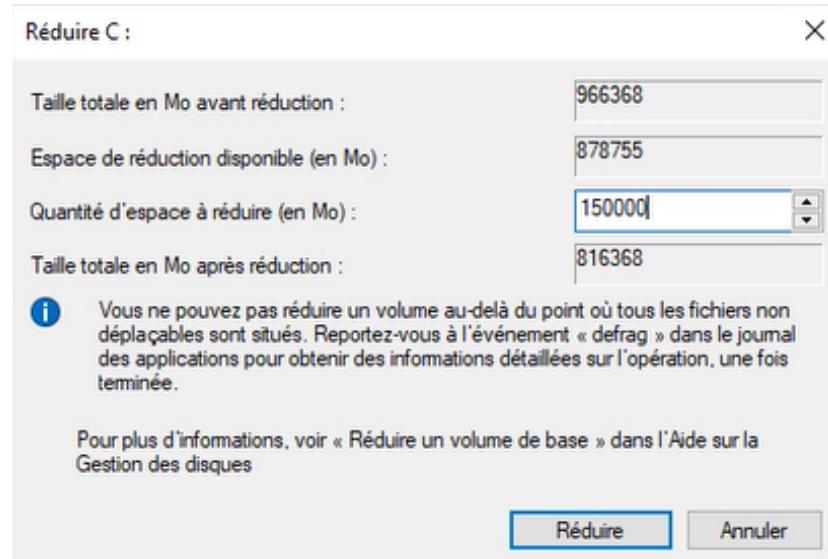
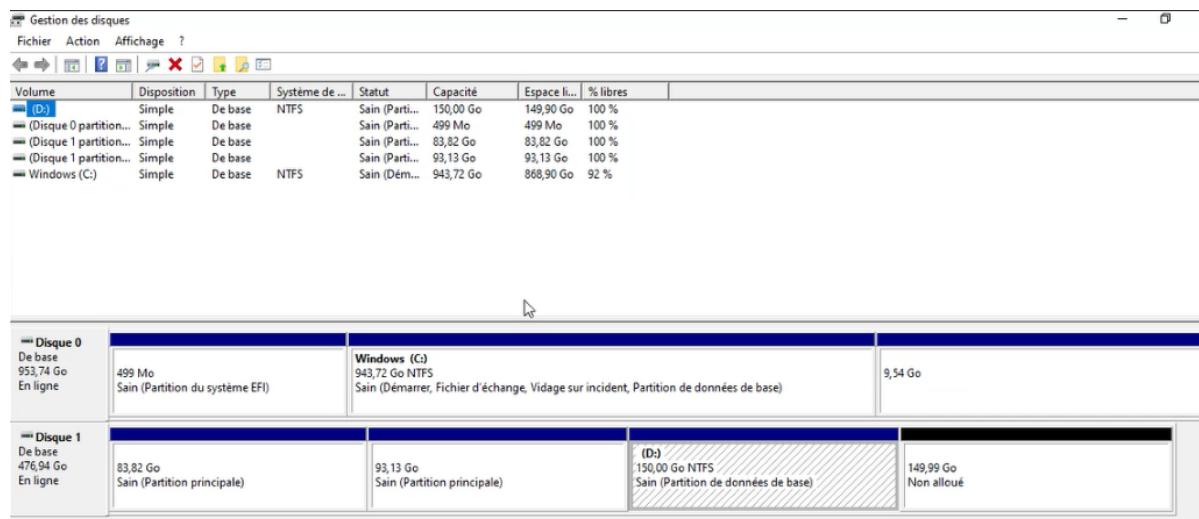
Formateur : Loup FORMENT

Lien d'utilité publique pour savoir ce que va faire une commande :

<https://explainshell.com/>

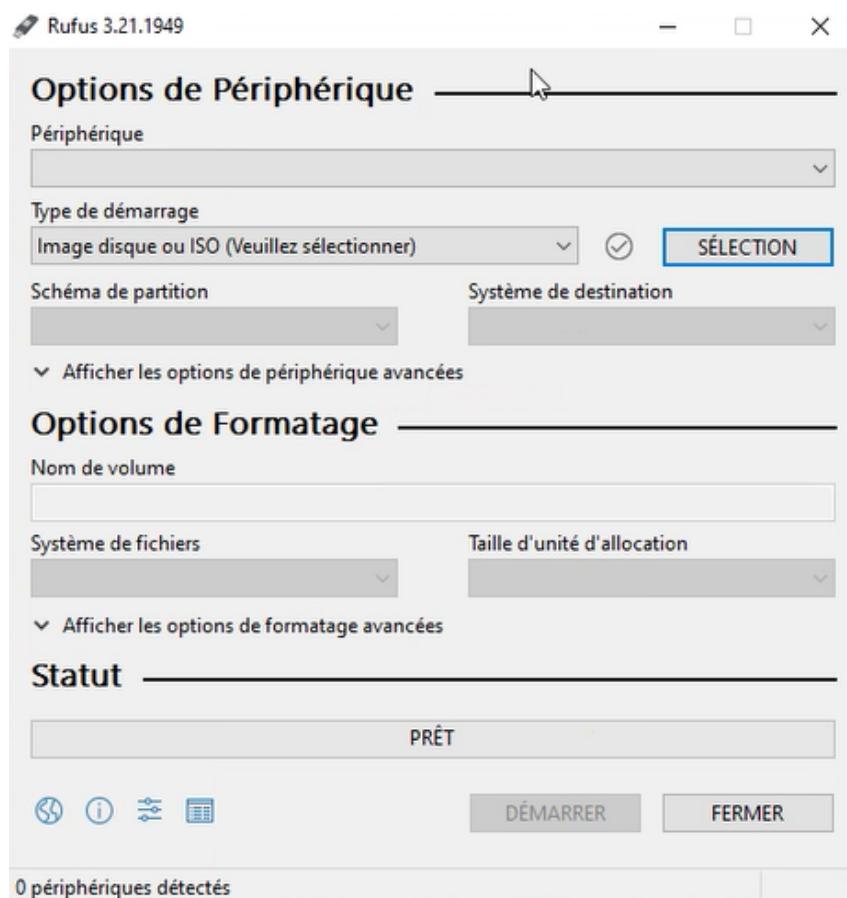
Pour installer Linux en dual boot :

1ère étape : réduire la taille du disque C: pour réserver de la place sur la machine

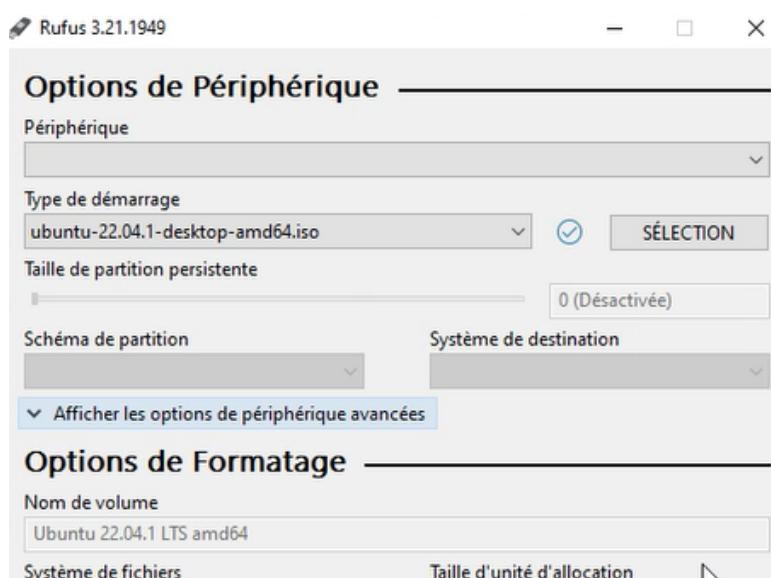


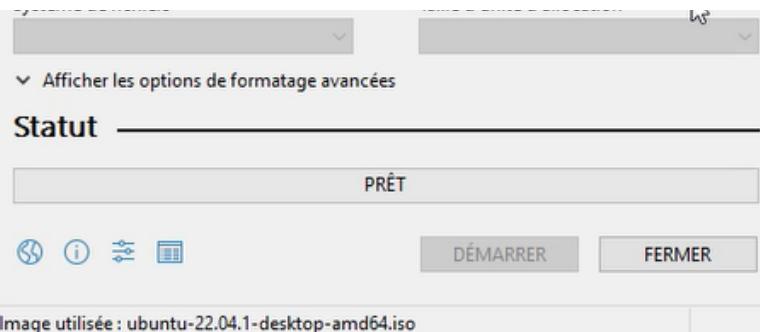
Etape 2 : télécharger Rufus sur une clé USB pour se faire un clé de boot





Une fois sur cette interface, on va chercher notre Iso :





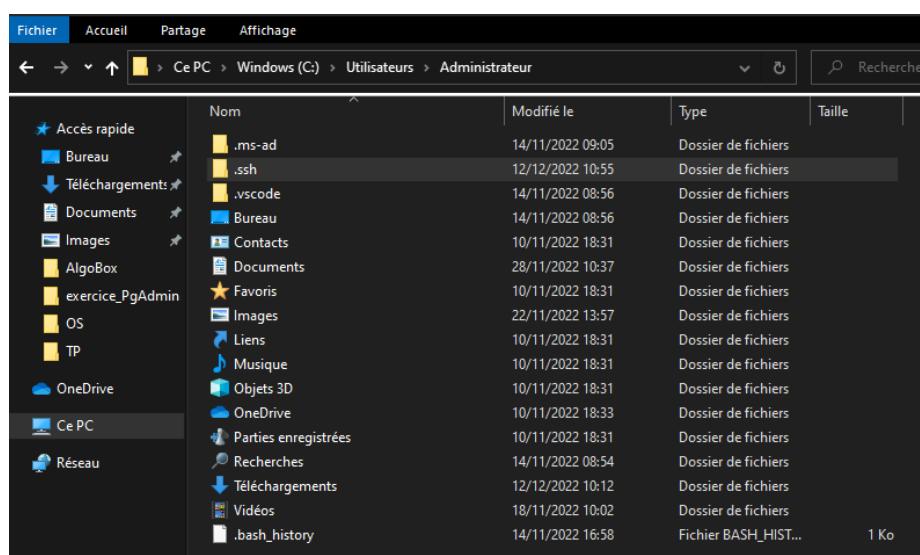
Personnellement, j'ai choisi d'installer Ubuntu en VM et d'utiliser SSH pour la ligne de commande :

Prendre la main en SSH :

Générer la clé SSH :

<https://git-scm.com/book/fr/v2/Git-sur-le-serveur-G%C3%A9n%C3%A9ration-des-cl%C3%A9s-publiques-SSH>

Vérifier qu'on possède bien un dossier SSH :



Sous GIT :

```
Administrator@LIL-JTJ3KN3 MINGW64 ~
$ cd ~/.ssh

Administrator@LIL-JTJ3KN3 MINGW64 ~/ssh
$ ls
config

Administrator@LIL-JTJ3KN3 MINGW64 ~/ssh
$ ssh-keygen -o
```

Installer open SSH sous Ubuntu : `sudo apt install open-ssh`

```
vel@vel-Virtual-Machine:~$ sudo apt install openssh-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Le paquet suivant a été installé automatiquement et n'est plus nécessaire :
  systemd-hwe-hwdb
Veuillez utiliser « sudo apt autoremove » pour le supprimer.
Les paquets suivants seront également installés :
```

les paquets supplémentaires suivants seront installés :

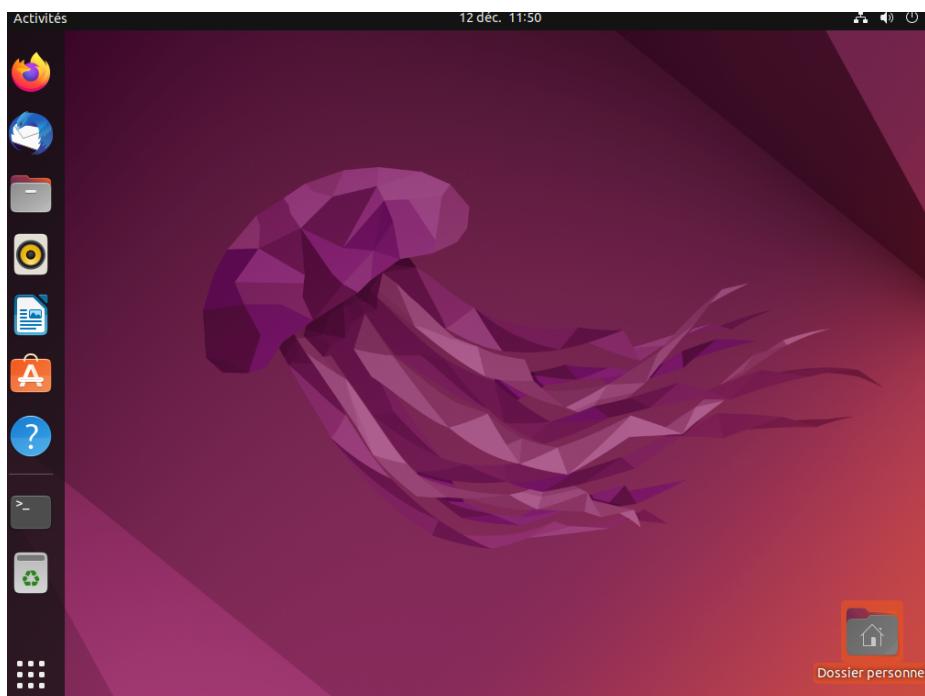
Pour envoyer la clé depuis mon windows : ssh-copy\_id (nom\_utilisateur)@(adresse ip ou nom de la machine)

```
Administrateur@LIL-JTJ3KN3 MINGW64 ~/.ssh
$ ssh-copy-id vel@172.27.25.173
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/c/Users/Administrateur/.ssh/id_rsa.pub"
The authenticity of host '172.27.25.173 (172.27.25.173)' can't be established.
ED25519 key fingerprint is SHA256:gE8K9VJ0VQR8eFowlJ8Ux6y+tM+kDrwFbR1mhkBUsVQ.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
vel@172.27.25.173's password:

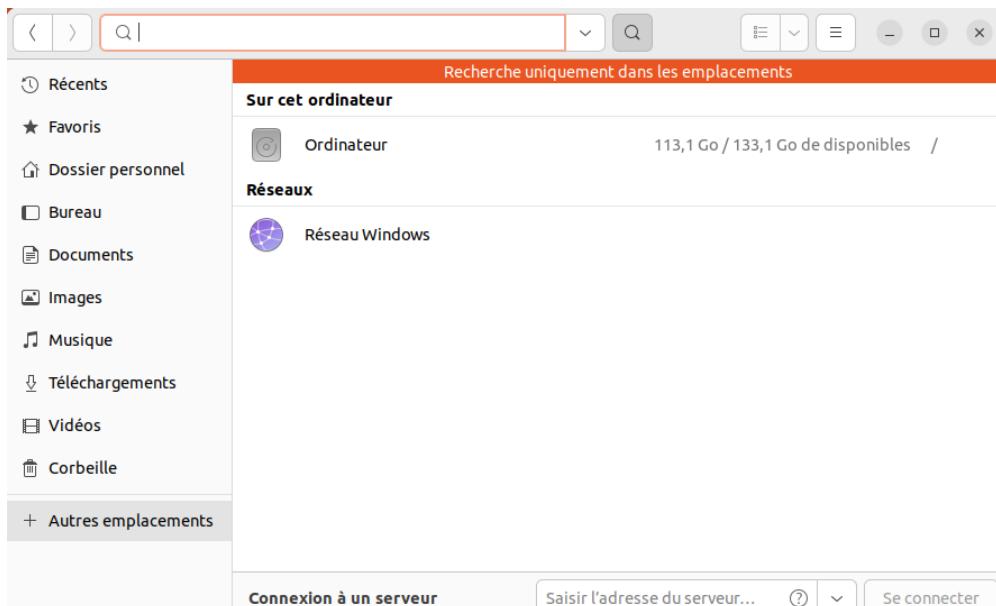
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'vel@172.27.25.173'"
and check to make sure that only the key(s) you wanted were added.
```

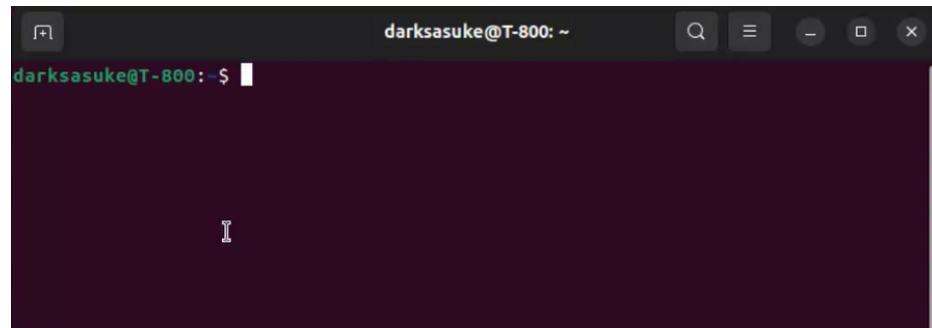
Suite à l'installation d'Ubuntu : on accède à l'interface graphique (GNOME) ou GUI (Graphical User Interface)



Pour voir les disques :



On va principalement travailler avec le shell sous Linux (Ou CLI) :



Notre shell est un interpréteur : il attend des commandes :

```
darksasuke@T-800:~$ python3
Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

```
darksasuke@T-800:~$ python3
Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> lhgdsgsd
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'lhgdsgsd' is not defined
>>> print("bonjour")
bonjour
>>> 3+5
8
>>> 
```

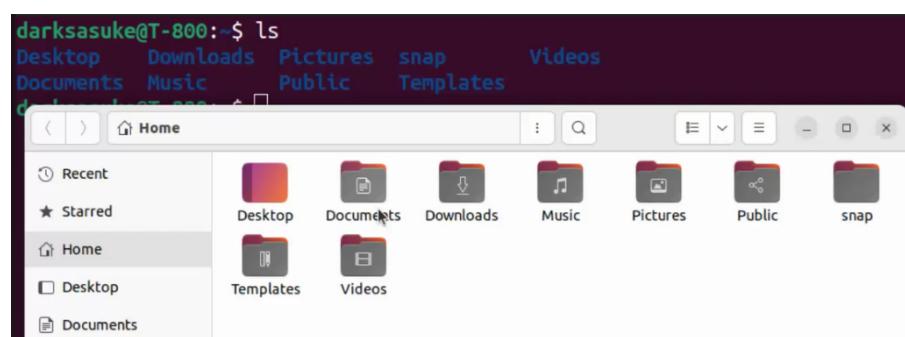
Cette partie dans le shell représente (de gauche à droite) : l'utilisateur avec lequel on est connecté, le nom de la machine et notre position dans la machine



La commande "ls" affiche les dossiers :

```
darksasuke@T-800:~$ ls
Desktop Downloads Pictures snap      Videos
Documents Music     Public    Templates
```

Équivalent à la fenêtre ci-dessous :



Se déplacer dans le dossier "Document" : commande `cd`

```
darksasuke@T-800:~$ cd Documents  
darksasuke@T-800:~/Documents$
```

Si on tape la commande `cd` sans argument, on retourne au dossier home :

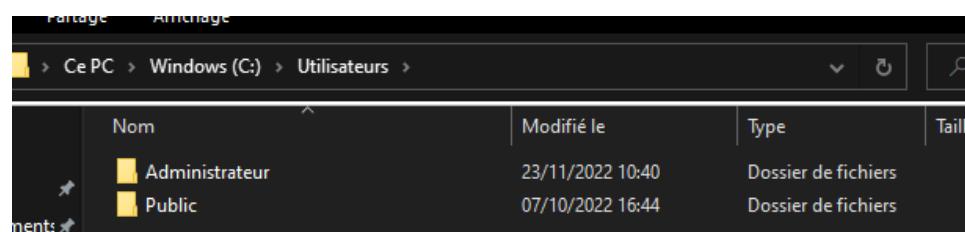
```
darksasuke@T-800:~/Documents$ cd  
darksasuke@T-800:~$ pwd  
/home/darksasuke  
darksasuke@T-800:~$
```

La commande "`pwd`" permet de savoir où l'on se situe dans notre arborescence

Sur Windows, nos dossiers systèmes se situent ici :



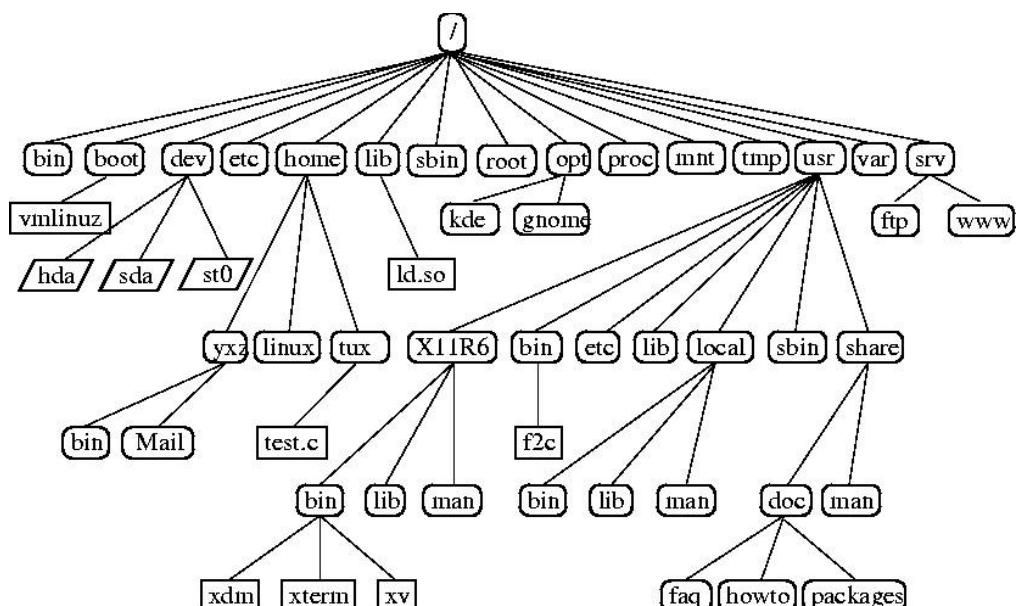
Pour les utilisateurs, ce sera ici :



La racine sous Windows est C:

La racine sous Linux est /

La racine / sous linux mène aux dossiers ETC/BIN/HOME/MNT



Pour revenir à la racine sous Linux :

```
darksasuke@T-800:~$ cd /  
darksasuke@T-800:/$ ls  
bin dev lib libx32 mnt root snap sys var  
boot etc lib32 lost+found opt run srv tmp  
cdrom home lib64 media proc sbin swapfile usr
```

```
darksasuke@T-800:/$ pwd  
/  
darksasuke@T-800:/$
```

```
darksasuke@T-800:/$ cd home/  
darksasuke@T-800:/home$ pwd  
/home  
darksasuke@T-800:/home$ ls  
darksasuke  
darksasuke@T-800:/home$ █
```

Il y a deux type de chemin sous Linux :

**Chemin Relatif** : Dépend de la position où l'utilisateur se situe

**Chemin Absolue** : Commence depuis la racine, à privilégier pour les scripts

```
darksasuke@T-800:~$ cd /  
darksasuke@T-800:~$ ls  
bin dev lib libx32 mnt root snap sys var  
boot etc lib32 lost+found opt run srv tmp  
cdrom home lib64 media proc sbin swapfile usr  
darksasuke@T-800:~$ cd home/darksasuke█
```

```
vel  
● vel@vel-Virtual-Machine:/home$ pwd  
/home  
● vel@vel-Virtual-Machine:/home$ cd /  
● vel@vel-Virtual-Machine:/$ cd home/vel/Documents/  
○ vel@vel-Virtual-Machine:~/Documents$ █
```

Pour obliger le système à commencer par la racine :

```
darksasuke@T-800:~$ cd /home/darksasuke  
darksasuke@T-800:~$ █
```

Pour créer un dossier : **mkdir**

Pour créer un fichier : **touch**

```
darksasuke@T-800:~$ ls  
Desktop dossier Music Public Templates  
Documents Downloads Pictures snap Videos  
darksasuke@T-800:~$ █
```

Exemple en chemin absolue :

```
darksasuke@T-800:~$ mkdir /home/darksasuke/Documents/dossier2
darksasuke@T-800:~$ ls Documents
dossier2 test
```

Exemple de commande pour créer un fichier :

```
darksasuke@T-800:~$ touch Documents/fichier
darksasuke@T-800:~$ touch /home/darksasuke/Documents/fichier2
```

Pour supprimer un programme : **rm** (faire très attention en utilisant cette commande)

```
darksasuke@T-800:~$ rm Documents/fichier
darksasuke@T-800:~$ ls Documents/
dossier2 fichier2 test
darksasuke@T-800:~$
```

Afficher le manuel des commande : **man**

Le résultat de la commande **man rm** :

```
RM(1) User Commands RM(1)

NAME
    rm - remove files or directories

SYNOPSIS
    rm [OPTION]... [FILE]...

DESCRIPTION
    This manual page documents the GNU version of rm. rm removes each specified file. By default, it does not remove directories.

    If the -I or --interactive=once option is given, and there are more than three files or the -r, -R, or --recursive are given, then rm prompts the user for whether to proceed with the entire operation. If the response is not affirmative, the entire command is aborted.

    Otherwise, if a file is unwritable, standard input is a
Manual page rm(1) line 1 (press h for help or q to quit)
```

Affiche un manuel (man) pour savoir comment utiliser la commande rm

Afficher dossier caché : **ls -a**

```
● vel@vel-Virtual-Machine:~$ ls -a
. .bash_history .bashrc .cache Documents .lesshst Modèles .profile .python_history .ssh Téléchargements .vscode-server
.. .bash_logout Bureau .config Images .local Musique Public snap .sudo_as_admin_successful Vidéos .wget-hsts
○ vel@vel-Virtual-Machine:~$
```

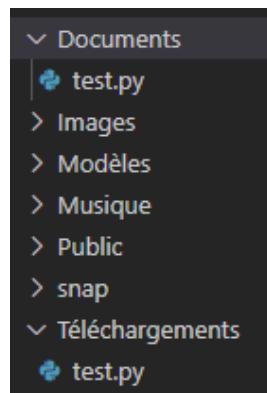
Commande .. : revenir en arrière :

```
● vel@vel-Virtual-Machine:~$ pwd
/home/vel
● vel@vel-Virtual-Machine:~$ cd ..
● vel@vel-Virtual-Machine:/home$ pwd
/home
```

Commande . : répertoire actuel

Pour copié/collé d'un fichier/répertoire : **cp**

```
● vel@vel-Virtual-Machine:~$ cp Documents/test.py /home/vel/Téléchargements/
○ vel@vel-Virtual-Machine:~$
```

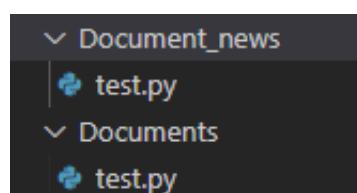


Si on veut copier deux fichiers en même temps, on indique les deux fichiers :

```
darksasuke@T-800:~$ cp fichier fichier.old Downloads/
```

Pour copier un dossier : **cp -r**

```
● vel@vel-Virtual-Machine:~$ cp -r Documents/ Document_news
○ vel@vel-Virtual-Machine:~$
```



Pour supprimer plusieurs éléments d'un coup :

```
● vel@vel-Virtual-Machine:~$ rm Téléchargements/new_fichier_python Téléchargements/test.py
```

Pour couper/coller : **mv**

```
darksasuke@T-800:~$ mv fichier Videos/
darksasuke@T-800:~$ ls
Desktop  Downloads  Pictures  snap      Videos
dossier  Music     Public    Templates
darksasuke@T-800:~$ ls Videos
fichier
```

Pour lui donner un autre nom pendant la commande **mv** :

```
darksasuke@T-800:~$ mv fichier Videos/new_name  
darksasuke@T-800:~$ ls Videos/  
fichier new_name
```

On utilise également la commande mv pour renommer le fichier

```
darksasuke@T-800:~$ mv fichier chocolat  
darksasuke@T-800:~$ ls  
chocolat dossier Music Public Templates  
Desktop Downloads Pictures snap Videos
```

Dans linux : tout est fichier et également très verbeux

Pour lire un fichier : **cat**

Exemple avec la liste de tous les utilisateurs du système : **cat /etc/passwd**

```
vel@vel-Virtual-Machine:~$ cat /etc/passwd  
root:x:0:0:root:/root:/bin/bash  
daemon:x:1:1:daemon:/usr/sbin/nologin  
bin:x:2:2:bin:/bin:/usr/sbin/nologin  
sys:x:3:3:sys:/dev:/usr/sbin/nologin  
sync:x:4:65534:sync:/bin:/bin/sync  
games:x:5:60:games:/usr/games:/usr/sbin/nologin  
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin  
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin  
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin  
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin  
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin  
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin  
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin  
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin  
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin  
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin  
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin  
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin  
systemd-network:x:100:102:system Network Management,,,:/run/systemd:/usr/sbin/nologin  
systemd-resolve:x:101:103:system Resolver,,,:/run/systemd:/usr/sbin/nologin  
messagebus:x:102:105::/nonexistent:/usr/sbin/nologin  
systemd-timesync:x:103:106:system Time Synchronization,,,:/run/systemd:/usr/sbin/nologin  
syslog:x:104:111::/home/syslog:/usr/sbin/nologin  
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin  
tss:x:106:112:TPM software stack,,,:/var/lib/tpm:/bin/false  
uidd:x:107:115::/run/uidd:/usr/sbin/nologin  
systemd-oom:x:108:116:systemd Userspace OOM Killer,,,:/run/systemd:/usr/sbin/nologin  
tcpdump:x:109:117::/nonexistent:/usr/sbin/nologin  
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin  
usbmux:x:111:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin  
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin  
kernoops:x:113:65534:Kernel Ooops Tracking Daemon,,,:/usr/sbin/nologin  
avahi:x:114:121:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin  
cups-pk-helper:x:115:122:user for cups-pk-helper service,,,:/home/cups-pk-helper:/usr/sbin/nologin  
rtkit:x:116:123:RealtimeKit,,,:/proc:/usr/sbin/nologin  
whoopsie:x:117:124::/nonexistent:/bin/false  
sssd:x:118:125:sssd system user,,,:/var/lib/sssd:/usr/sbin/nologin  
speech-dispatcher:x:119:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false  
nm-openvpn:x:120:126:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
```

Ce fichier est un fichier CSV regroupant certaines informations : nom users, x = référence au fichier shadows, nombre 1000 = 1er users, groupe utilisateur, répertoire personnel, bash (/bin/bash)

```
darksasuke@T-800:~$ ls /bin/bash  
/bin/bash
```

Voir les groupes du système :

```
vel@vel-Virtual-Machine:~$ cat /etc/group  
root:x:0:  
daemon:x:1:  
bin:x:2:  
sys:x:3:  
adm:x:4:syslog,vel  
tty:x:5:  
disk:x:6:
```

```
lp:x:7:  
mail:x:8:  
news:x:9:  
uucp:x:10:  
man:x:12:  
proxy:x:13:  
kmem:x:15:  
dialout:x:20:  
fax:x:21:  
voice:x:22:  
cdrom:x:24:vel  
floppy:x:25:  
tape:x:26:  
sudo:x:27:vel  
audio:x:29:pulse  
dip:x:30:vel  
www-data:x:33:  
backup:x:34:  
operator:x:37:  
list:x:38:  
irc:x:39:  
src:x:40:  
gnats:x:41:  
shadow:x:42:  
utmp:x:43:  
video:x:44:  
sasl:x:45:  
plugdev:x:46:vel
```

Nom du groupe – shadows – groupe ID - utilisateurs qui appartiennent à ce groupe

Dans l'idéal : regroupe utilisateur par groupe et donner les autorisation aux groupe pour que les utilisateurs du groupe profitent des droits

Groupe principaux = groupe personnel de l'utilisateur

Groupe secondaire : comme des groupes normaux, mais qui peut regrouper plusieurs utilisateur

Pour invoquer les droits root sur une commande : **root**

```
vel@vel-Virtual-Machine:~$ sudo ls  
[sudo] Mot de passe de vel :  
Bureau Documents Images Modèles Musique Public snap Téléchargements Vidéos  
vel@vel-Virtual-Machine:~$
```

Changer mot de passe : **sudo passwd root**

```
vel@vel-Virtual-Machine:~$ sudo passwd root  
Nouveau mot de passe :  
Retapez le nouveau mot de passe :  
passwd : le mot de passe a été mis à jour avec succès  
vel@vel-Virtual-Machine:~$ su -  
Mot de passe :  
root@vel-Virtual-Machine:~# su vel
```

Ajouter utilisateur : **sudo useradd**

```
vel@vel-Virtual-Machine:~$ sudo passwd root  
Nouveau mot de passe :  
Retapez le nouveau mot de passe :  
passwd : le mot de passe a été mis à jour avec succès  
vel@vel-Virtual-Machine:~$ su -  
Mot de passe :  
root@vel-Virtual-Machine:~# su vel
```

Pour vérifier : **cat /etc/passwd**

```
vel:x:1000:1000:Vel,,,,:/home/vel:/bin/bash  
sshd:x:128:65534::/run/sshd:/usr/sbin/nologin  
naruto:x:1001:1001::/home/naruto:/bin/sh
```

Changer mot de passe utilisateur : **sudo passwd user**

```
vel@vel-Virtual-Machine:~$ sudo passwd naruto  
Nouveau mot de passe :  
MOT DE PASSE INCORRECT : Le mot de passe ne passe pas la vérification dans le dictionnaire - basé sur un mot du dictionnaire  
Retapez le nouveau mot de passe :
```

```
Retapez le nouveau mot de passe :  
passwd : le mot de passe a été mis à jour avec succès
```

Changer utilisateur : **su users**

```
vel@vel-Virtual-Machine:~$ su naruto  
Mot de passe :  
$
```

Pour créer un utilisateur personnalisé :

```
sudo useradd kakashi --uid 2567 --gid 1000 --comment "Entraîneur de l'espace" -m -d /home/terrainDeKakashi --shell /bin/bash
```

```
useradd : l'identifiant d'utilisateur (uid) 2567 n'est pas unique  
vel@vel-Virtual-Machine:~$ sudo useradd kakashi --uid 2568 --gid 1000 --comment "Entraîneur de l'espace" -m -d /home/terrainDeKakashi --shell /bin/bash
```

Pour modifier utilisateur : **usermod**

Pour supprimer utilisateur : **userdel**

Créer groupe : **groupadd**

Supprimer groupe : **groupdel**

Modifier groupe : **groupmod**

Toujours veillez à "exit" les sessions après un su -

Si on ne peut pas supprimer un utilisateur :

```
ps -u "nom de l'utilisateur" pour voir si des bash sont encore utilisés
```

Si des bash sont bien utilisé : utilisé le tag dans la commande **sudo kill -9 "numéro bash"**

```
vel@vel-Virtual-Machine:~$ ps -u kakashi  
    PID TTY      TIME CMD  
 24416 pts/1    00:00:00 bash  
vel@vel-Virtual-Machine:~$ kill -9 24416  
bash: kill: (24416) - Opération non permise  
vel@vel-Virtual-Machine:~$ sudo kill -9 24416  
Processus arrêté  
vel@vel-Virtual-Machine:~$ vel@vel-Virtual-Machine:~$  
exit  
gaara@vel-Virtual-Machine:~$ ls -la  
exit  
  
exit
```

```
○ vel@vel-Virtual-Machine:~$ sudo usermod thrall -s /usr/bin/python3
```

```
○ vel@vel-Virtual-Machine:~$ su thrall  
Mot de passe :  
Python 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> |
```

Pour vérifier la liste des groupes : **cat /etc/group**

Pour rajouter un groupe : **groupadd**

```
darksasuke@T-800:~$ sudo groupadd my_group -g 1234
```

Le nouveau groupe sera présent avec la GID 1234

```
my_group:x:1234:
```

Une fois le groupe créé, on peut rajouter le user dans le groupe primaire

```
darksasuke@T-800:~$ sudo useradd user2 -g 1234
```

Si un groupe est utilisé comme groupe primaire par un user, il sera impossible de le supprimer, il faudra delete tous les utilisateurs du groupe avant de le supprimer

Pour créer groupe secondaire, il faudra utiliser l'argument -G

```
-G, --groups GROUP1[,GROUP2,...[,GROUPN]]  
A list of supplementary groups which the user is also a member of. Each group is separated from the next by a comma, with no intervening whitespace. The groups are subject to the same restrictions as the group given with the -g option. The default is for the user to belong only to the initial group.
```

```
darksasuke@T-800:~$ sudo useradd worker1 -G 1236
```

Notre utilisateur worker1 est bien dans le groupe team\_work

```
team_work:x:1236:worker1  
worker1:x:1002:
```

Commande **history** : affiche la liste de toutes les commandes tapé sur la session par l'utilisateur

```
208 cat /etc/group  
209 cat /etc/passwd  
210 cat /etc/group  
211 cat /etc/passwd  
212 cat /etc/group  
213 sudo groupdel my_group  
214 sudo userdel user2  
215 sudo groupdel my_group  
216 sudo userdel user3  
217 sudo groupdel my_group  
218 sudo userdel user4  
219 sudo groupdel my_group  
220 sudo userdel user4 user3 user2  
221 cat /etc/group  
222 sudo groupadd team_work  
223 cat /etc/group  
224 group  
225 groups  
226 cat /etc/group  
227 sudo useradd worker1 -G 1236  
228 cat /etc/group  
229 sudo useradd worker2 -G 1236  
230 cat /etc/group  
231 history
```

```
darksasuke@T-800:~$ sudo usermod worker1 -G projet  
darksasuke@T-800:~$ sudo usermod worker2 -G projet
```

```
projet:x:1237:worker1,worker2
```

```
drwxr-xr-x  2 darksasuke darksasuke 4096 déc. 12 11:26 Templates
drwxr-xr-x  2 darksasuke darksasuke 4096 déc. 12 14:26 Videos
-rw-----  1 darksasuke darksasuke    30 déc. 12 12:04 .python_history
```

Dans l'ordre : d ou - : information sur la nature du fichier (- pour fichier ou d pour directory)

Puis, dans l'ordre, système de permission pour le users, le groupe et les autres

Permission du users : Permission sur le fichier : rwx = r = droit de lecture | w = droit l'écriture | x = droit d'exécution

Puis Permission sur le groupe

Puis Permission pour les autres

Pour changer le owner d'un fichier : **chown**

```
darksasuke@T-800:~/trash$ sudo chown worker1:worker1 chocolat
```

Ici on change l'utilisateur et le groupe

Méthode en changeant juste l'utilisateur propriétaire sur un fichier :

```
darksasuke@T-800:~/trash$ sudo chown darksasuke chocolat
darksasuke@T-800:~/trash$ ls -la
total 8
drwxrwxr-x  2 darksasuke darksasuke 4096 déc. 13 10:10 .
drwxr-x--- 18 darksasuke darksasuke 4096 déc. 13 10:10 ..
-rw-rw-r--  1 darksasuke worker1      0 déc. 13 10:10 chocolat
darksasuke@T-800:~/trash$
```

```
darksasuke@T-800:~/trash$ sudo chown worker1: dos
darksasuke@T-800:~/trash$ ls -la
total 12
drwxrwxr-x  3 darksasuke darksasuke 4096 déc. 13 10:17 .
drwxr-x--- 18 darksasuke darksasuke 4096 déc. 13 10:10 ..
-rw-rw-r--  1 worker1   worker1      0 déc. 13 10:10 chocolat
drwxrwxr-x  2 worker1   worker1      4096 déc. 13 10:17 dos
```

```
darksasuke@T-800:~/trash$ sudo chown worker1: dos -R
```

Pour tout changer dans un dossier en terme de owner : **sudo chown worker1: dos -R**

```
1 directory, 4 files
darksasuke@T-800:~/trash$ sudo chown worker1: dos -R
darksasuke@T-800:~/trash$ tree -pug
[drwxrwxr-x darksasuke darksasuke] .
└── [-rw-rw-r-- worker1   worker1 ] chocolat
    [drwxrwxr-x worker1   worker1 ] dos
        [-rw-rw-r-- worker1   worker1 ] fraise
        [-rw-rw-r-- worker1   worker1 ] noisette
        [-rw-rw-r-- worker1   worker1 ] vanille
```

Pour modifier les permissions sur les fichiers : **chmod**

Pour calculer, on utilise la table des puissance de 2 :

R (lecture) = 4 | W (écriture) = 2 | X (exécution) = 1

<https://chmod-calculator.com/>

Exemple avec permission 777 :

```
darksasuke@T-800:~/trash$ chmod 777 chocolat
```

```
darksasuke@T-800:~/trash$ ls -la
total 12
drwxrwxr-x 3 darksasuke darksasuke 4096 déc. 13 10:17 .
drwxr-x--- 18 darksasuke darksasuke 4096 déc. 13 10:23 ..
-rwxrwxrwx 1 darksasuke darksasuke    0 déc. 13 10:10 chocolat
drwxrwxr-x 2 worker1   worker1     4096 déc. 13 10:20 dos
```

Test avec 000 :

```
darksasuke@T-800:~/trash$ chmod 000 chocolat
darksasuke@T-800:~/trash$ ls -la
total 12
drwxrwxr-x 3 darksasuke darksasuke 4096 déc. 13 10:17 .
drwxr-x--- 18 darksasuke darksasuke 4096 déc. 13 10:23 ..
----- 1 darksasuke darksasuke    0 déc. 13 10:10 chocolat
drwxrwxr-x 2 worker1   worker1     4096 déc. 13 10:20 dos
```

```
darksasuke@T-800:~/trash$ sudo chown darksasuke: dos/ -R
darksasuke@T-800:~/trash$ chmod 750 dos -R
darksasuke@T-800:~/trash$ tree -pug
[drwxrwxr-x darksasuke darksasuke]
└── [-rwxr-x--- darksasuke darksasuke]  chocolat
    ├── [drwxr-x--- darksasuke darksasuke]  dos
    │   ├── [-rwxr-x--- darksasuke darksasuke]  fraise
    │   ├── [-rwxr-x--- darksasuke darksasuke]  noisette
    │   └── [-rwxr-x--- darksasuke darksasuke]  vanille

1 directory, 4 files
```

Accéder à un dossier compte comme un droit d'exécution et de lecture

```
darksasuke@T-800:~/trash$ sudo su worker1
$ ls -la
total 12
drwxrwxr-x 3 darksasuke darksasuke 4096 déc. 13 10:17 .
drwxr-x--- 18 darksasuke darksasuke 4096 déc. 13 10:23 ..
-rwxr-x--- 1 darksasuke darksasuke    0 déc. 13 10:10 chocolat
drwxr-x--- 2 darksasuke darksasuke 4096 déc. 13 10:20 dos
$ ls -la dos
ls: cannot open directory 'dos': Permission denied
```

Exercice :

Créer une arborescence de fichier projet :

Création 4 users

Création deux groupe

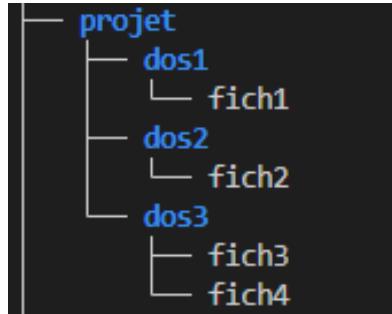
```
Projet
DOS1/ DOS2 / DOS3
Fichier1(DOS1)/fichier2(DOS2)/fichier3 et 4 (DOS 3)
```

1ère étape : je créer mes users

```
159 sudo useradd malfurion --comment "groupe 1" -m --shell /bin/bash
160 sudo useradd varian --comment "groupe 1" -m --shell /bin/bash
161 cat /etc/passwd
162 sudo useradd sylvanas --comment "groupe 2" -m --shell /bin/bash
163 sudo useradd garrosh --comment "groupe 2" -m --shell /bin/bash
```

```
207 sudo passwd sylvanas
208 sudo passwd varian
209 sudo passwd garrosh
210 sudo passwd malfurion
```

2ème étape : je créer mon arborescence



3ème étape : je créer mes groupes

```
213 sudo groupadd grp1 -g 1010
214 sudo groupadd grp2 -g 1020
215
```

4ème étape : je rajoute les groupes en groupes secondaire de mes users

```
● vel@vel-Virtual-Machine:/home$ sudo usermod garrosh -G 1020
● vel@vel-Virtual-Machine:/home$ sudo usermod sylvanas -G 1020
● vel@vel-Virtual-Machine:/home$ sudo usermod varian -G 1010
● vel@vel-Virtual-Machine:/home$ sudo usermod malfurion -G 1010
```

Vérification avec un cat /etc/group

```
grp1:x:1010:varian,malfurion
grp2:x:1020:garrosh,sylvanas
```

5ème étape : on change les propriétaires des dossiers

```
232 sudo chown varian:grp1 dos1
233 sudo chown garrosh:grp2 dos2
234 ls -la
235 history
vel@vel-Virtual-Machine:/home/projet$
```

```
● vel@vel-Virtual-Machine:/home/projet$ ls -la
total 20
drwxr-xr-x  5 root      root 4096 déc. 13 11:25 .
drwxr-xr-x 11 root      root 4096 déc. 13 11:25 ..
drwxr-xr-x  2 varian   grp1 4096 déc. 13 11:26 dos1
drwxr-xr-x  2 garrosh  grp2 4096 déc. 13 11:26 dos2
drwxr-xr-x  2 root      root 4096 déc. 13 11:27 dos3
```

6ème étape : on affine les droit avec chmod

```
● vel@vel-Virtual-Machine:/home/projet$ sudo chmod 770 dos1 -R
● vel@vel-Virtual-Machine:/home/projet$ sudo chmod 770 dos2 -R
● vel@vel-Virtual-Machine:/home/projet$ ls -la
total 20
drwxr-xr-x  5 root      root 4096 déc. 13 11:25 .
drwxr-xr-x 11 root      root 4096 déc. 13 11:25 ..
```

```
drwxrwx--- 2 varian grp1 4096 déc. 13 11:26 dos1  
drwxrwx--- 2 garrosh grp2 4096 déc. 13 11:26 dos2  
drwxr-xr-x 2 root root 4096 déc. 13 11:27 dos3
```

7ème étape : on affine l'accès au dossier en créant un autre groupe "groupe\_projet" pour limiter l'accès aux users

```
grp_project:x:2042:
```

Attention : si on effectue la commande "sudo usermod -G" pour le rajouter dans un autre groupe, cela écrase son groupe secondaire, sous Ubuntu, il faut utiliser la commande "sudo usermod -aG"

<https://linux-note.com/ajouter-un-utilisateur-a-un-ou-plusieurs-groupes/>

Pour supprimer utilisateur d'un groupe :

Avec l'option **--delete**, **gpasswd** supprime un utilisateur d'un groupe.

```
root@debian~#: gpasswd --delete nom-utilisateur nom-groupe
```

<https://linux-note.com/commande-gpasswd-supprime-un-utilisateur-dun-groupe/#:~:text=Avec%20l'option%20%2D%2Ddelete,un%20utilisateur%20d'un%20groupe.&text=Pour%20supprimer%20un%20utilisateur%20d,avec%20l'op%20%2DG%20.>

```
275 sudo gpasswd --delete greymane grp1  
276 sudo gpasswd --delete gallywix grp2
```

```
grp1:x:1010:varian,malfurion  
grp2:x:1020:garrosh,sylvanas  
grp_project:x:2042:malfurion,sylvanas,varian,gallywix,garrosh,greymane
```

8ème étape : finaliser la création des groupes pour l'exercice 2

```
279 sudo groupadd grp3 -g 1030  
280 sudo groupmod greymane -aG 1030  
281 sudo usermod greymane -aG 1030  
282 sudo usermod gallywix -aG 1030
```

```
grp1:x:1010:varian,malfurion  
grp2:x:1020:garrosh,sylvanas  
grp_project:x:2042:malfurion,sylvanas,varian,gallywix,garrosh,greymane  
gallywix:x:2572:  
greymane:x:2573:  
grp3:x:1030:greymane,gallywix
```

8ème étape : On affine à nouveau les droits avec chown et chmod

Je modifie l'accès pour qu'uniquement mon sudoers et mon groupe projet qui embarque tous mes users concerné puisse accéder au dossier

```
285 history  
● vel@vel-Virtual-Machine:/home$ sudo chown vel:grp_project projet  
● vel@vel-Virtual-Machine:/home$ ls -la  
total 52  
drwxr-xr-x 13 root root 4096 déc. 13 12:08 .  
drwxr-xr-x 20 root root 4096 déc. 12 10:28 ..  
drwxr-x--- 2 gallywix gallywix 4096 déc. 13 12:08 gallywix  
drwxr-x--- 2 garrosh garrosh 4096 déc. 13 11:29 garrosh  
drwxr-x--- 2 greymane greymane 4096 déc. 13 12:08 greymane  
drwxr-x--- 2 thrall vel 4096 déc. 12 16:45 maelstromDeThrall  
drwxr-x--- 2 thrall vel 4096 déc. 12 16:32 maisonDeGaara.old  
drwxr-x--- 2 malfurion malfurion 4096 déc. 13 11:18 malfurion  
drwxr-xr-x 5 vel grp_project 4096 déc. 13 11:25 projet
```

Correction :

Création dossier projet

```

darksasuke@T-800:~$ ls -la /home/
total 20
drwxr-xr-x  5 root      root      4096 déc. 13 11:09 .
drwxr-xr-x 20 root      root      4096 déc. 12 11:03 ..
drwxr-x--- 18 darksasuke darksasuke 4096 déc. 13 10:23 darksasuke
drwxr-x---  2      2567 darksasuke 4096 déc. 12 16:20 my_user
drwxr-x---  2 user42    user42    4096 déc. 13 11:09 user42
darksasuke@T-800:~$ sudo mkdir /home/projet
[sudo] password for darksasuke:
darksasuke@T-800:~$ cd /home/projet/
darksasuke@T-800:/home/projet$ ls -la
total 8
drwxr-xr-x  2 root      root      4096 déc. 13 13:54 .
drwxr-xr-x  6 root      root      4096 déc. 13 13:54 ..
darksasuke@T-800:/home/projet$
```

Création dossier dans projet

```
darksasuke@T-800:/home/projet$ sudo mkdir dos1 dos2 dos3
```

Création fichier :

```
darksasuke@T-800:/home/projet$ sudo touch dos1/fic1 dos2/fic2 dos3/fic3 dos3/fic4
```

```

darksasuke@T-800:/home/projet$ sudo tree -pug
[drwxr-xr-x root      root      ] .
└── [drwxr-xr-x root      root      ] dos1
    └── [-rw-r--r-- root      root      ] fic1
└── [drwxr-xr-x root      root      ] dos2
    └── [-rw-r--r-- root      root      ] fic2
└── [drwxr-xr-x root      root      ] dos3
    ├── [-rw-r--r-- root      root      ] fic3
    └── [-rw-r--r-- root      root      ] fic4

3 directories, 4 files
darksasuke@T-800:/home/projet$ █
```

Rajout des groupes :

```

grp1:x:1238:
grp2:x:1239:
grp3:x:1240:
```

```

sudo useradd user1 -G grp1,grp3
sudo useradd user2 -G grp1,grp3
sudo useradd user3 -G grp2,grp3
sudo useradd user4 -G grp2,grp3
```

```

grp1:x:1238:user1,user2
grp2:x:1239:user3
grp3:x:1240:user1,user2,user3
user1:x:1005:
user2:x:1006:
```

```
user3:x:1007:
```

```
3 directories, 4 files
sudo chown :grp3 .
sudo chown :grp1 dos1
sudo chown :grp2 dos2
sudo chown :grp3 dos3
tree -pug
[drwxr-xr-x root      grp3    ] .
└── [drwxr-xr-x root      grp1    ] dos1
    └── [-rw-r--r-- root      root    ] fic1
└── [drwxr-xr-x root      grp2    ] dos2
    └── [-rw-r--r-- root      root    ] fic2
└── [drwxr-xr-x root      grp3    ] dos3
    ├── [-rw-r--r-- root      root    ] fic3
    └── [-rw-r--r-- root      root    ] fic4
```

```
3 directories, 4 files
sudo chown user1 dos1/fic1
sudo chown user2 dos2/fic2
sudo chown user3 dos3/fic3
sudo chown user1:grp2 dos3/fic34
```

```
sudo chmod 070 -R .
tree -pug
. [error opening dir]

0 directories, 0 files
sudo tree -pug
[d---rwx--- root      grp3    ] .
└── [d---rwx--- root      grp1    ] dos1
    └── [----rwx--- user1    root    ] fic1
└── [d---rwx--- root      grp2    ] dos2
    └── [----rwx--- user2    root    ] fic2
└── [d---rwx--- root      grp3    ] dos3
    ├── [----rwx--- user3    root    ] fic3
    └── [----rwx--- user1    grp2    ] fic4

3 directories, 4 files
```

Après modification des dossiers :

```
sudo tree -pug
[d---rwx--- root      grp3    ] .
└── [d---rwx--- root      grp1    ] dos1
    └── [-rwx----- user1    root    ] fic1
└── [d---rwx--- root      grp2    ] dos2
    └── [-rwx----- user2    root    ] fic2
└── [d---rwx--- root      grp3    ] dos3
    ├── [-rwx----- user3    root    ] fic3
    └── [-rwxrwx--- user1    grp2    ] fic4
```

Correction exercice 2 :

```
3 directories, 4 files
sudo touch projet2/dos1/fic1.5
sudo tree -pug projet2
[d---r-x--- root      root    ] projet2
└── [drwxr-xr-x root      root    ] dos1
```

```
[d---r-x--- root      root      ] dos1
  └── [-rwx----- root      root      ] fic1
      └── [-rw-r--r-- root      root      ] fic1.5
[d---r-x--- root      root      ] dos2
  └── [-rwx----- root      root      ] fic2
[d---r-x--- root      root      ] dos3
  └── [-rwx----- root      root      ] fic3
      └── [-rwxr-x--- root      root      ] fic4
3 directories, 5 files
```

Création nouveau groupe

```
grp1:x:1238:user1,user2
grp2:x:1239:user3,user4
user1:x:1005:
user2:x:1006:
user3:x:1007:
user4:x:1008:
group_tot:x:1240:user1,user2,user3,user4
```

```
group_tot:x:1240:user1,user2,user3,user4
sudo groupadd grp3
sudo useradd user5 -G grp3,group_tot
sudo useradd user6 -G grp3,group_tot
```

```
grp1:x:1238:user1,user2
grp2:x:1239:user3,user4
user1:x:1005:
user2:x:1006:
user3:x:1007:
user4:x:1008:
group_tot:x:1240:user1,user2,user3,user4,user5,user6
grp3:x:1241:user5,user6
user5:x:1009:
user6:x:1010:
```

```
sudo groupadd grp1grp2
sudo usermod -aG grp1grp2 user1
sudo usermod -aG grp1grp2 user2
sudo usermod -aG grp1grp2 user
sudo usermod -aG grp1grp2 user3
sudo usermod -aG grp1grp2 user4
```

Affinage des droits :

```
sudo chown :group_tot .
sudo chown :group_tot projet2
sudo chown :grp1grp2 projet2/dos1
sudo chown :grp3 projet2/dos2
sudo chown :group_tot projet2/dos3
sudo tree -pug projet2
[d---r-x--- root      group_tot] projet2
  └── [d---r-x--- root      grp1grp2] dos1
    └── [-rwx----- root      root      ] fic1
        └── [-rw-r--r-- root      root      ] fic1.5
```

```
[d---r-x--- root      grp3      ] dos2
└ [---rwx----- root      root      ] fic2
[d---r-x--- root      group_tot] dos3
└ [---rwx----- root      root      ] fic3
  [---rwxr-x--- root      root      ] fic4
3 directories, 5 files
```

```
3 directories, 5 files
sudo chown :grp1 projet2/dos1/fic1
sudo chown :grp1grp2 projet2/dos1/fic1.5
sudo chown :grp3 projet2/dos2/fic2
sudo chown :grp3 projet2/dos3/fic3
sudo chown :grp1grp2 projet2/dos3/fic4
```

```
3 directories, 5 files
sudo chmod 070 projet2 -R
sudo tree -pug projet2
[d---rwx--- root      group_tot] projet2
└ [d---rwx--- root      grp1grp2] dos1
  [----rwx--- root      grp1      ] fic1
  [----rwx--- root      grp1grp2] fic1.5
└ [d---rwx--- root      grp3      ] dos2
  [----rwx--- root      grp3      ] fic2
  [d---rwx--- root      group_tot] dos3
  [----rwx--- root      grp3      ] fic3
  [----rwx--- root      grp1grp2] fic4
```

Installer différent paquet :

Sur Ubuntu : apt

Sur Arch : pacman

En fonction des distributions, le programme est différents

Apt install : installe un paquet

Apt remove : désinstaller un paquet

Pour installer vlc par exemple :

```
o vel@vel-Virtual-Machine:~$ sudo apt install vlc -y
```

Le -y sert dans la commande apt pour réponse oui automatiquement aux demande pendant l'installation

Apt remove = désinstaller un fichier

Apt update : mettre à jour la liste des paquets disponible

Apt upgrade : mettre à jour l'image de tous les paquets

#### Utilisation de VIM :

```
Leçon 1.1 : DÉPLACEMENT DU CURSEUR

** Pour déplacer le curseur, appuyez les touches h,j,k,l comme indiqué. **
^
k      Astuce : La touche h est à gauche et déplace à gauche.
<h      l>      La touche l est à droite et déplace à droite.
```

```
< n >           La touche i est à droite et déplace à droite.  
j                La touche j ressemble à une flèche vers le bas.  
v
```

- Déplacez le curseur sur l'écran jusqu'à vous sentir à l'aise.
- Maintenez la touche Bas (j) enfoncée jusqu'à ce qu'elle se répète.  
Maintenant vous êtes capable de vous déplacer jusqu'à la leçon suivante.
- En utilisant la touche Bas, allez à la Leçon 1.2.

NOTE : Si jamais vous doutez de ce que vous venez de taper, appuyez <Échap> pour revenir en mode Normal. Puis retapez la commande que vous vouliez.

NOTE : Les touches fléchées devraient également fonctionner. Mais en utilisant hjkl vous pourrez vous déplacer beaucoup plus rapidement, une fois que vous aurez pris l'habitude.

## Leçon 1.2 : SORTIR DE VIM

!! NOTE : Avant d'effectuer les étapes ci-dessous, lisez toute cette leçon !!

- Appuyez la touche <Échap> (pour être sûr d'être en mode Normal).
  - Tapez : :q! <Entrée>  
Ceci quitte l'éditeur SANS enregistrer les changements que vous avez faits.
  - Revenez ici en tapant la commande qui vous a mené à ce tutoriel.  
Cela pourrait être : vimtutor <Entrée>
  - Si vous avez mémorisé ces étapes et êtes confiant, effectuez les étapes 1 à 3 pour sortir puis rentrer dans l'éditeur.
- NOTE : :q! <Entrée> annule tous les changements que vous avez faits. Dans quelques leçons, vous apprendrez à enregistrer les changements.

- Déplacez le curseur à la Leçon 1.3.

## Leçon 1.3 : ÉDITION DE TEXTE - EFFACEMENT

\*\* Appuyez x pour effacer le caractère sous le curseur. \*\*

- Déplacez le curseur sur la ligne marquée ---> ci-dessous.
  - Pour corriger les erreurs, déplacez le curseur jusqu'à ce qu'il soit sur un caractère à effacer.
  - Appuyez la touche x pour effacer le caractère redondant.
  - Répétez les étapes 2 à 4 jusqu'à ce que la phrase soit correcte.
- > La vvache a sauté au-ddessus dde la luune.
- Maintenant que la ligne est correcte, passez à la Leçon 1.4.

NOTE : En avançant dans ce cours, n'essayez pas de mémoriser, apprenez par la pratique.

## Leçon 1.4 : ÉDITION DE TEXTE - INSERTION

\*\* Appuyez i pour insérer du texte. \*\*

- Déplacez le curseur sur la première ligne marquée ---> ci-dessous.
  - Pour rendre la première ligne identique à la seconde, mettez le curseur sur le premier caractère APRÈS l'endroit où insérer le texte.
  - Appuyez i et tapez les caractères qui manquent.
  - Une fois qu'une erreur est corrigée, appuyez <Échap> pour revenir en mode Normal. Répétez les étapes 2 à 4 pour corriger la phrase.
- > Il mnqe caractères cette .  
---> Il manque des caractères dans cette ligne.
- Une fois que vous êtes à l'aise avec l'insertion de texte, allez à la Leçon 1.5.

## Leçon 1.5 : ÉDITION DE TEXTE - AJOUTER

\*\* Appuyez A pour ajouter du texte. \*\*

1. Déplacez le curseur sur la première ligne ci-dessous marquée --->. Peu importe sur quel caractère se trouve le curseur sur cette ligne.
2. Appuyez A et tapez les ajouts nécessaires.
3. Quand le texte a été ajouté, appuyez <Échap> pour revenir en mode Normal.
4. Déplacez le curseur sur la seconde ligne marquée ---> et répétez les étapes 2 et 3 pour corriger la phrase.  
---> Il manque du texte à partir de cet  
          Il manque du texte à partir de cette ligne.  
---> Il manque aussi du te  
          Il manque aussi du texte ici.
5. Quand vous vous sentez suffisamment à l'aise pour ajouter du texte, allez à la Leçon 1.6.

## Leçon 1.6 : ÉDITER UN FICHIER

\*\* Utilisez :wq pour enregistrer un fichier et sortir. \*\*

!! NOTE : Lisez toute la leçon avant d'exécuter les instructions ci-dessous !!

1. Sortez de ce tutoriel comme vous l'avez fait dans la Leçon 1.2 : :q!  
Ou, si vous avez accès à un autre terminal, exécutez-y les actions qui suivent.
2. À l'invite du shell, tapez cette commande : vim tutor <Entrée>  
'vim' est la commande pour démarrer l'éditeur Vim, 'tutor' est le nom du fichier que vous souhaitez éditer. Utilisez un fichier qui peut être modifié.
3. Insérez et effacez du texte comme vous l'avez appris dans les leçons précédentes.
4. Enregistrez le fichier avec les changements et sortez de Vim avec :  
:wq <Entrée>
5. Si vous avez quitté vimtutor à l'étape 1, recommencez vimtutor et déplacez-vous en bas vers le résumé suivant.

## RÉSUMÉ DE LA LEÇON 1

1. Le curseur se déplace avec les touches fléchées ou les touches hjkl.  
h (gauche)      j (bas)      k (haut)      l (droite)
2. Pour démarrer Vim à l'invite du shell tapez : vim FICHIER <Entrée>
3. Pour quitter Vim tapez : <Échap> :q! <Entrée> pour perdre tous les changements.  
OU tapez : <Échap> :wq <Entrée> pour enregistrer les changements.
4. Pour effacer un caractère sous le curseur tapez : x
5. Pour insérer ou ajouter du texte tapez :  
i      tapez le texte à insérer avant le curseur    <Échap>  
A      tapez le texte à ajouter en fin de ligne    <Échap>

NOTE : Appuyer <Échap> vous place en mode Normal ou annule une commande partiellement tapée dont vous ne voulez plus.

Passez maintenant à la leçon 2.

## Leçon 2.1 : COMMANDES D'EFFACEMENT

\*\* Tapez dw pour effacer un mot. \*\*

1. Appuyez <Échap> pour être sûr d'être en mode Normal.
2. Déplacez le curseur sur la ligne marquée ---> ci-dessous.
3. Placez le curseur sur le début d'un mot qui a besoin d'être effacé.
4. Tapez dw pour faire disparaître ce mot.

NOTE : La lettre d apparaîtra sur la dernière ligne de l'écran lors de votre frappe. Vim attend que vous tapez w . Si vous voyez un autre caractère que d vous avez tapé autre chose ; appuyez <Échap> et recommencez.

---> Il y a quelques mots qui n'ont rien à faire cette ligne.

5. Répétez les étapes 3 et 4 jusqu'à ce que la phrase soit correcte et allez à la Leçon 2.2.

---

#### Leçon 2.2 : PLUS DE COMMANDES D'EFFACEMENTS

\*\* Tapez d\$ pour effacer jusqu'à la fin de la ligne. \*\*

1. Appuyez <Échap> pour être sûr d'être en mode Normal.
  2. Déplacez le curseur sur la ligne marquée ---> ci-dessous.
  3. Déplacez le curseur jusqu'à la fin de la ligne correcte (APRÈS le premier . ).
  4. Tapez d\$ pour effacer jusqu'à la fin de la ligne.
- > Quelqu'un a tapé la fin de cette ligne deux fois.
5. Allez à la Leçon 2.3 pour comprendre ce qui se passe.

---

#### Leçon 2.3 : À PROPOS DES OPÉRATEURS ET DES MOUVEMENTS

Plusieurs commandes qui changent le texte sont constituées d'un opérateur et d'un mouvement. Le format pour une commande d'effacement avec l'opérateur d d'effacement est le suivant :

d mouvement

Où :

d - est l'opérateur d'effacement  
mouvement - est le mouvement sur lequel agit l'opérateur (listés ci-dessous)

Une courte liste de mouvements :

w - jusqu'au début du prochain mot, en EXCLUANT son premier caractère.  
e - jusqu'à la fin du mot courant, en EXCLUANT son dernier caractère.  
\$ - jusqu'à la fin de la ligne, en INCLUANT son dernier caractère.

Ainsi, taper de va effacer depuis le curseur jusqu'à la fin du mot.

NOTE : Le seul appui d'un mouvement en mode Normal, sans commande, déplace le curseur comme indiqué.

---

#### Leçon 2.4 : UTILISER UN QUANTIFICATEUR AVEC UN MOUVEMENT

\*\* Taper un nombre avant un mouvement le répète autant de fois. \*\*

1. Déplacez le curseur au début de la ligne marquée ---> ci-dessous.
  2. Tapez 2w pour déplacer le curseur de 2 mots vers l'avant.
  3. Tapez 3e pour déplacer le curseur à la fin du troisième mot vers l'avant.
  4. Tapez 0 (zéro) pour déplacer au début de la ligne.
5. Répétez les étapes 2 et 3 avec des quantificateurs différents.

---> Ceci est juste une ligne avec des mots où vous pouvez vous déplacer.

## 6. Déplacez-vous à la Leçon 2.5.

### Leçon 2.5 : UTILISER UN QUANTIFICATEUR POUR EFFACER PLUS

\*\* Taper un nombre avec un opérateur le répète autant de fois. \*\*

Outre la combinaison de l'opérateur d'effacement avec un déplacement mentionné ci-dessus, vous pouvez insérer un nombre (quantificateur) pour effacer encore plus :

d nombre déplacement

1. Déplacez le curseur vers le premier mot en MAJUSCULES dans la ligne marquée --->.
2. Tapez d2w pour effacer les deux.
3. Répétez les étapes 1 et 2 avec des quantificateurs différents pour effacer les mots suivants en MAJUSCULES à l'aide d'une commande.

---> Cette ligne de mots est .

### Leçon 2.6 : OPÉREZ SUR DES LIGNES

\*\* Tapez dd pour effacer une ligne complète. \*\*

Vu le nombre de fois où l'on efface des lignes complètes, les concepteurs de Vi ont décidé qu'il serait plus facile de taper simplement deux d pour effacer une ligne.

1. Placez le curseur sur la seconde ligne de la phrase ci-dessous.
2. Tapez dd pour effacer la ligne.
3. Maintenant allez à la quatrième ligne.
4. Tapez 2dd pour effacer deux lignes.

---> 1) Les roses sont rouges,  
---> 2) La boue c'est drôle,  
---> 3) Les violettes sont bleues,  
---> 6) Le sucre est doux  
---> 7) Tout comme vous.

### Leçon 2.7 : L'ANNULATION

\*\* Tapez u pour annuler les dernières commandes. \*\*  
\*\* Tapez U pour récupérer toute une ligne. \*\*

1. Déplacez le curseur sur la ligne marquée ---> ci-dessous et placez-le sur la première erreur.
2. Tapez x pour effacer le premier caractère redondant.
3. Puis tapez u pour annuler la dernière commande exécutée.
4. Cette fois, corrigez toutes les erreurs de la ligne avec la commande x .
5. Puis tapez un U majuscule pour remettre la ligne dans son état initial.
6. Puis tapez u deux-trois fois pour annuler le U et les commandes précédentes.
7. Maintenant tapez CTRL-R (maintenez la touche CTRL enfonce pendant que vous appuyez R) deux-trois fois pour refaire les commandes (annuler les annulations).

---> Corrigez les erreurs sur cette ligne et remettez-les avec 'annuler'.

8. Ce sont des commandes très utiles. Maintenant, allez au résumé de la Leçon 2.

### RÉSUMÉ DE LA LEÇON 2

1. Pour effacer du curseur jusqu'au mot suivant tapez : dw
2. Pour effacer du curseur jusqu'à la fin d'une ligne tapez : d\$
3. Pour effacer toute une ligne tapez : dd
4. Pour répéter un déplacement ajoutez un quantificateur : 2w
5. Le format d'une commande de changement est :

```

opérateur [nombre] déplacement

Où :
opérateur - est ce qu'il faut faire, comme d pour effacer.
[nombre] - un quantificateur optionnel pour répéter le déplacement.
déplacement - déplace le long du texte à opérer, tel que w (mot),
$ (jusqu'à la fin de ligne), etc.

6. Pour se déplacer au début de ligne, utilisez un zéro : 0

5. Pour annuler des actions précédentes, tapez : u (u minuscule)
Pour annuler tous les changements sur une ligne tapez : U (U majuscule)
Pour annuler l'annulation tapez : CTRL-R

```

Leçon 3.1 : LE COLLAGE

\*\* Tapez p pour placer après le curseur ce qui vient d'être effacé. \*\*

1. Placez le curseur sur la première ligne ci-dessous marquée --->.
2. Tapez dd pour effacer la ligne et la placer dans un registre de Vim.
3. Déplacez le curseur sur la ligne c) au-dessus où vous voulez remettre la ligne effacée.
4. En mode Normal, tapez p pour remettre la ligne en dessous du curseur.
5. Répétez les étapes 2 à 4 pour mettre toutes les lignes dans le bon ordre.

---> b) Les violettes sont bleues,  
---> c) L'intelligence s'apprend,  
---> d) Et vous, qu'apprenez-vous ?  
---> a) Les roses sont rouges,

---

Leçon 3.2 : LA COMMANDE DE REMPLACEMENT

\*\* Tapez rx pour remplacer un caractère sous le curseur par x . \*\*

1. Déplacez le curseur sur la première ligne marquée ---> ci-dessous.
2. Placez le curseur de manière à ce qu'il surplombe la première erreur.
3. Tapez r suivi du caractère qui doit corriger l'erreur.
4. Répétez les étapes 2 et 3 jusqu'à ce que la première ligne soit égale à la seconde.

---> Quand cette ligne a été saisie, quelqu'un a fait des fautes de frappe !  
---> Quand cette ligne a été saisie, quelqu'un a fait des fautes de frappe !

5. Maintenant, allez à la Leçon 3.3.

NOTE : N'oubliez pas que vous devriez apprendre par la pratique, pas par mémorisation.

Leçon 3.3 : L'OPÉRATEUR DE CHANGEMENT

\*\* Pour changer jusqu'à la fin d'un mot, tapez ce .\*\*

1. Déplacez le curseur sur la première ligne marquée ---> ci-dessous.
2. Placez le curseur sur le u de luhko.
3. Tapez ce et corrigez le mot (dans notre cas, tapez 'igne').
4. Appuyez <Échap> et placez-vous sur le prochain caractère qui doit être changé.
5. Répétez les étapes 3 et 4 jusqu'à ce que la première phrase soit identique à la seconde.

---> Cette ligne contient quelques myqa qui ont besoin d'être changés.  
---> Cette ligne contient quelques mots qui ont besoin d'être changés.

Notez que ce efface le mot et vous place ensuite en mode Insertion.

#### Leçon 3.4 : PLUS DE CHANGEMENTS AVEC c

\*\* L'opérateur de changement fonctionne avec les mêmes déplacements que l'effacement. \*\*

1. L'opérateur de changement fonctionne de la même manière que l'effacement. Le format est :  
`c [nombre] déplacement`
2. Les déplacements sont identiques : `w` (mot) et `$` (fin de ligne).
3. Déplacez-vous sur la première ligne marquée ---> ci-dessous.
4. Placez le curseur sur la première erreur.
5. Tapez `c$` et tapez le reste de la ligne afin qu'elle soit identique à la seconde ligne, puis tapez `<Échap>`.

---> La fin de cette ligne doit être corrigée avec la commande `c$`.  
---> La fin de cette ligne doit être corrigée avec la commande `c$`.

NOTE : Vous pouvez utiliser la touche Retour Arrière pour corriger les erreurs lorsque vous tapez.

#### RÉSUMÉ DE LA LEÇON 3

1. Pour remettre le texte qui a déjà été effacé, tapez `p`. Cela Place le texte effacé APRÈS le curseur (si une ligne complète a été effacée, elle sera placée sous la ligne du curseur).
2. Pour remplacer le caractère sous le curseur, tapez `r` suivi du caractère qui remplacera l'original.
3. L'opérateur de changement vous permet de changer depuis la position du curseur jusqu'où le déplacement vous amène. Par exemple, tapez `ce` pour changer du curseur jusqu'à la fin du mot, `c$` pour changer jusqu'à la fin d'une ligne.
4. Le format pour le changement est :

`c [nombre] déplacement`

Passez maintenant à la leçon suivante.

#### Leçon 4.1 : POSITION DU CURSEUR ET ÉTAT DU FICHIER

\*\* Tapez CTRL-G pour afficher votre position dans le fichier et son état.  
Tapez `G` pour vous rendre à une ligne donnée du fichier. \*\*

NOTE : Lisez toute cette leçon avant d'effectuer l'une des étapes !!

1. Maintenez enfoncée la touche CTRL et appuyez sur `g`. On appelle cela CTRL-G. Une ligne d'état va apparaître en bas de l'écran avec le nom du fichier et le numéro de la ligne où vous êtes. Notez ce numéro, il servira lors de l'étape 3.

NOTE : Vous pouvez peut-être voir le curseur en bas à droite de l'écran.  
Ceci arrive quand l'option 'ruler' est activée (voir `:help 'ruler'`)

2. Tapez `G` pour vous déplacer à la fin du fichier.  
Tapez `gg` pour vous déplacer au début du fichier.
3. Tapez le numéro de la ligne où vous étiez suivi de `G`. Cela vous ramènera à la ligne où vous étiez au départ quand vous aviez appuyé CTRL-G.
4. Si vous vous sentez prêt à faire ceci, effectuez les étapes 1 à 3.

## Leçon 4.2 : LA RECHERCHE

\*\* Tapez / suivi d'un texte pour rechercher ce texte. \*\*

1. Tapez le caractère / en mode Normal. Notez que celui-ci et le curseur apparaissent en bas de l'écran, comme lorsque l'on utilise : .
2. Puis tapez 'errreuur' <Entrée>. C'est le mot que vous voulez rechercher.
3. Pour rechercher à nouveau le même texte, tapez simplement n . Pour rechercher le même texte dans la direction opposée, tapez N .
4. Pour rechercher une phrase dans la direction opposée, utilisez ? au lieu de / .

--> erreur ne s'écrit pas "errreuur" ; errreuur est une erreur.

NOTE : Quand la recherche atteint la fin du fichier, elle reprend au début sauf si l'option 'wrapscan' est désactivée.

---

## Leçon 4.3 : RECHERCHE DES PARENTHÈSES CORRESPONDANTES

\*\* Tapez % pour trouver des ), ] ou } correspondants. \*\*

1. Placez le curseur sur l'un des (, [ ou { de la ligne marquée --> ci-dessous.
2. Puis tapez le caractère % .
3. Le curseur se déplacera sur la parenthèse ou crochet correspondant.
4. Tapez % pour remplacer le curseur sur la parenthèse ou crochet correspondant.
5. Déplacez le curseur sur un autre (,),[],{ ou } et regardez ce que fait % .

--> Voici ( une ligne de test contenant des (, des [ ] et des { } )).

NOTE : Cette fonctionnalité est très utile lors du débogage d'un programme qui contient des parenthèses déséquilibrées !

## Leçon 4.4 : LA COMMANDE DE SUBSTITUTION

\*\* Tapez :s/ancien/nouveau/g pour remplacer 'ancien' par 'nouveau'. \*\*

1. Déplacez le curseur sur la ligne marquée --> ci-dessous.
2. Tapez :s/lee/le <Entrée> . Notez que cette commande change seulement la première occurrence de "lee" dans la ligne.
3. Puis tapez :s/lee/le/g . L'ajout du drapeau g ordonne de faire une substitution globale sur la ligne, et change toutes les occurrences de "lee" sur la ligne.
4. Pour changer toutes les occurrences d'un texte, entre deux lignes, tapez :#, #s/ancien/nouveau/g où #,# sont les numéros de lignes de la plage où la substitution doit être faite.  
Tapez :%s/ancien/nouveau/g pour changer toutes les occurrences dans tout le fichier.  
Tapez :%s/ancien/nouveau/gc pour trouver toutes les occurrences dans tout le fichier avec une invite pour confirmer ou infirmer chaque substitution.

## RÉSUMÉ DE LA LEÇON 4

1. CTRL-G affiche la position dans le fichier et l'état de celui-ci.  
G déplace à la fin du fichier.  
nombre G déplace au numéro de ligne.  
gg déplace à la première ligne.
2. Taper / suivi d'un texte recherche ce texte vers l'AVANT.  
Taper ? suivi d'un texte recherche ce texte vers l'ARRIÈRE.  
Après une recherche tapez n pour trouver l'occurrence suivante dans la même direction ou Maj+U pour rechercher dans la direction opposée.

```
mais direction ou Maj-N pour rechercher dans la direction opposée.
```

3. Taper % lorsque le curseur est sur (, ), [ , ] , { ou } déplace celui-ci sur le caractère correspondant.
4. Pour remplacer le premier aa par bb sur une ligne tapez :s/aa/bb  
Pour remplacer tous les aa par bb sur une ligne tapez :s/aa/bb/g  
Pour remplacer du texte entre deux numéros de ligne tapez :#, #s/aa/bb/g  
Pour remplacer toutes les occurrences dans le fichier tapez :%s/aa/bb/g  
Pour demander une confirmation à chaque fois ajoutez 'c' :%s/aa/bb/gc

Commande intéressante à connaître :

**echo** : affiche un message dans le terminal (équivalent du print)

**cat** : renvoie le contenu d'un fichier sur la sortie standard

**head** : affiche les premières ligne d'un fichier

**tail** : affiche les dernières ligne d'un fichier

Pour **head** et **tail** : argument -n permet d'afficher le nombre de ligne

```
[darksasuke@T-800 ~]$ tail -n4 /etc/passwd
user3:x:1007:1007::/home/user3:/bin/sh
user4:x:1008:1008::/home/user4:/bin/sh
user5:x:1009:1009::/home/user5:/bin/sh
user6:x:1010:1010::/home/user6:/bin/sh
```

**less/more** : visionneur

**cut** : couper les ligne

Argument possible dans **cut** : **cut -d":": -f**

```
saned
colord
geoclue
pulse
gnome-initial-setup
hplip
gdm
darksasuke
fwupd-refresh
user
worker1
worker2
user42
user1
user2
user3
user4
user5
user6
[darksasuke@T-800 ~]$ cut -d":": -f1 /etc/passwd
```

```
[darksasuke@T-800 ~]$ cut -d":": -f1,3,7 /etc/passwd
```

**-f1-3** est aussi possible pour afficher les colonnes de 1 à 3

**date** : affiche la date, pratique pour les timers

**sed** : s'utilise avec le mode substitut : permet de remplacer les éléments d'un fichier par autre chose sans altérer le contenu du fichier

```
[darksasuke@T-800 ~]$ sed "s/bonjour/bonsoir/g"
```

g = pour faire le remplacement de manière globale

```
mon texte
chocolat
caffé
noisette
bonjour
bonjour
```

```
[darksasuke@T-800 ~]$ sed "s/bonjour/bonsoir/g" fichier
mon texte
chocolat
caffé
noisette
bonsoir
bonsoir
```

Afficher un texte en enlevant les espaces en trop :

```
mon texte
chocolat
caffé
noisette
bonjour
bonjour
une phrase avec un      probleme      d'espace
```

Sed -E "s/ +/ /g" fichier : ajuste l'affichage pour enlever les espaces en trop

```
[darksasuke@T-800 ~]$ sed -E "s/ +/ /g" fichier
mon texte
chocolat
caffé
noisette
bonjour
bonjour
une phrase avec un probleme d'espace
```

Pour inscrire les modifications, il faut utiliser l'argument : -i

```
[darksasuke@T-800 ~]$ sed -E "s/ +/ /g" -i fichier
```

```
[darksasuke@T-800 ~]$ sed -E "s/ +/ /g" -i fichier
[darksasuke@T-800 ~]$ cat fichier
```

```
[darksasuke@T-800 ~]$ cat fichier
mon texte
chocolat
caffé
noisette
bonjour
bonjour
une phrase avec un problème d'espace
```

**grep** : chercher des patterns dans un fichier

```
[darksasuke@T-800 ~]$ grep darksasuke /etc/passwd
```

```
● vel@vel-Virtual-Machine:~$ grep vel /etc/passwd
vel:x:1000:1000:Vel,,,,:/home/vel:/bin/bash
```

```
[darksasuke@T-800 ~]$ grep "a" fichier
chocolat
caffé
une phrase avec un problème d'espace
[darksasuke@T-800 ~]$ grep "a" fichier -v
mon texte
noisette
bonjour
bonjour
```

Argument **-v** dans grep : affiche toutes les lignes où il n'y a pas un a

Argument **-o** dans grep : affiche la chaîne de caractère sélectionné, à utiliser avec une regex pour être sûr d'avoir le résultat voulu

```
[darksasuke@T-800 ~]$ grep "a" fichier -o
a
a
a
a
a
[darksasuke@T-800 ~]$
```

**-A2** : affiche les deux résultats avant le grep

```
[darksasuke@T-800 ~]$ grep "cho" fichier -A2
chocolat
caffé
noisette
```

**-B2** : affiche les deux résultats après le grep

```
[darksasuke@T-800 ~]$ grep "cho" fichier -B2
mon texte
chocolat
[darksasuke@T-800 ~]$ grep "cho" fichier -B2 -A1
mon texte
chocolat
caffé
```

Commande **sort** : permet de trier les lignes

Argument **-K2** avec sort : permet d'afficher dans l'ordre alphabétique

```
[darksasuke@T-800 ~]$ sort -k2 fichier
bonjour bonsoir
bonjour monde
```

```
chocolat noir
caffé noisette
une phrase
noisette pistache
mon texte
```

Sort fonctionne uniquement avec des chiffre, pas des nombres (10+), pour lui demander de faire le tri avec des nombres, il faut utiliser les arguments -n ou -

```
[darksasuke@T-800 ~]$ sort -k1 fichier -n

2G une phrase
4K noisette pistache
7T caffé noisette
8M bonjour monde
9K bonjour bonsoir
9M chocolat noir
11G mon texte
```

Argument -h : trie par ordre alphabétique :

```
[darksasuke@T-800 ~]$ sort -k1 fichier -h

4K noisette pistache
9K bonjour bonsoir
8M bonjour monde
9M chocolat noir
2G une phrase
11G mon texte
7T caffé noisette
```

Commande du : affiche les dossiers et leurs tailles, il faut l'utiliser avec l'argument -h pour **human readable**

```
vel@vel-Virtual-Machine:~$ du -h
8,0K ./Téléchargements
8,0K ./ssh
64K ./snap/snap-store/common/.cache/fontconfig
4,0K ./snap/snap-store/common/.cache/appstream
72K ./snap/snap-store/common/.cache/immodules
28K ./snap/snap-store/common/.cache/gnome-software/fwupd
1,4M ./snap/snap-store/common/.cache/gnome-software/odrs
5,8M ./snap/snap-store/common/.cache/gnome-software/appstream
324K ./snap/snap-store/common/.cache/gnome-software/cssresource
3,3M ./snap/snap-store/common/.cache/gnome-software/icons
5,4M ./snap/snap-store/common/.cache/gnome-software/screenshots/624x351
296K ./snap/snap-store/common/.cache/gnome-software/screenshots/112x63
5,7M ./snap/snap-store/common/.cache/gnome-software/screenshots
17M ./snap/snap-store/common/.cache/gnome-software
24K ./snap/snap-store/common/.cache/gio-modules
17M ./snap/snap-store/common/.cache
17M ./snap/snap-store/common
8,0K ./snap/snap-store/582/.config/fontconfig
4,0K ./snap/snap-store/582/.config/gtk-2.0
4,0K ./snap/snap-store/582/.config/ibus
4,0K ./snap/snap-store/582/.config/dconf
4,0K ./snap/snap-store/582/.config/gtk-3.0
```

```
27M ./cache
4,0K ./Public
260M .
vel@vel-Virtual-Machine:~$ du -h -a
```

Le chevron > = redirige la sortie standard vers un fichier (l'écrase)

```
[darksasuke@T-800 ~]$ sudo du -h -s ~ > result
```

Cette commande renvoie le résultat dans un fichier et le créer automatiquement s'il n'existe pas

Pour incrémenter à la fin d'un fichier, il faut utiliser un double chevron : >>

On peut rediriger vers **/dev/null** -> poubelle des flux

**cd~** = ramène au répertoire personnel

Affiche les sorties en erreur :

```
[darksasuke@T-800 ~]$ du -h -s / 2> result
```

Pour les sorties standard, on utilise l'argument 1 à la place de deux

On utilisera l'argument & pour afficher les deux sorties

```
[darksasuke@T-800 ~]$ du -h -s / 2> /dev/null  
18G /  
[darksasuke@T-800 ~]$ du -hs / 2> /dev/null  
18G /
```

La redirection > est donc une nouvelle méthode de création de fichier

Autre type de redirection : le pipe | sortie vers un autre cmd

Exemple d'utilisation :

```
[darksasuke@T-800 ~]$ cat /etc/passwd | grep darksasuke
```

```
zoom .  
● vel@vel-Virtual-Machine:~$ cat /etc/passwd | grep vel  
vel:x:1000:1000:Vel,,,:/home/vel:/bin/bash  
● vel@vel-Virtual-Machine:~$
```

Pour récupérer l'UID :

```
● vel@vel-Virtual-Machine:~$ cat /etc/passwd | grep vel | cut -d: -f3  
1000
```

Autre exemple :

```
user1:1005  
user2:1006  
user3:1007  
user4:1008  
user5:1009  
user6:1010  
[darksasuke@T-800 ~]$ cat /etc/passwd | cut -d: -f1,3
```

```
user3 1007  
user4 1008  
user5 1009  
user6 1010  
[darksasuke@T-800 ~]$ cat /etc/passwd | cut -d: -f1,3 | sed "s/:/ /g"
```

```
[darksasuke@T-800 ~]$ cat /etc/passwd | cut -d: -f1,3 | sed "s/:/ /g" | sort
```

```
[root@darkasuke ~]# cat /etc/passwd | grep $(cat /etc/group | grep -n -k2 | tail -n1)
nobody 65534
```

Commande : \$(cmd)

\$() -> exécuter une commande dans une commande

```
[darksasuke@T-800 ~]$ cat /etc/group | grep $(cat /etc/passwd | grep darksasuke | cut -d: -f3) | cut -d: -f1
```

```
[darksasuke@T-800 ~]$ cat /etc/group | grep $(cat /etc/passwd | grep user4: | cut -d: -f3) | cut -d: -f1
```

```
[darksasuke@T-800 ~]$ cat $(echo "fichier")
11G mon texte
9M chocolat noir
7t caffé noisette
4m noisette pistache
9K bonjour bonsoir
8M bonjour monde
2G une phrase
```

#### Exercice

1) afficher les lignes 5 a 9 du fichier /etc/passwd

```
● vel@vel-Virtual-Machine:~$ sed -n "5,9 p" /etc/passwd
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
```

correction :

```
[darksasuke@T-800 ~]$ head -n9 /etc/passwd | tail -n4
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
```

2) enregister dans un fichier la liste des utilisateurs du système et leur UID et GID

```
vel@vel-Virtual-Machine:~$ cat /etc/passwd | cut -d":" -f1,3,4 > liste_users_1412
```

```
● vel@vel-Virtual-Machine:~$ cat /etc/passwd | cut -d":" -f1,3,4 > liste_users_1412
● vel@vel-Virtual-Machine:~$ cat liste_users_1412
root:0:0
daemon:1:1
bin:2:2
sys:3:3
sync:4:65534
games:5:60
man:6:12
```

```
lp:7:  
mail:8:8  
news:9:9  
uucp:10:10  
proxy:13:13  
www-data:33:33  
backup:34:34  
list:38:38  
irc:39:39  
gnats:41:41  
nobody:65534:65534  
systemd-network:100:102  
systemd-resolve:101:103  
messagebus:102:105  
systemd-timesync:103:106  
syslog:104:111  
_apt:105:65534  
tss:106:112  
uuid:107:115  
systemd-oom:108:116  
tcpdump:109:117  
avahi-autoipd:110:119  
usbmux:111:46  
dnsmasq:112:65534  
kernoops:113:65534  
avahi:114:121  
cups-pk-helper:115:122  
rtkit:116:123  
whoopsie:117:124  
sssd:118:125  
speech-dispatcher:119:29  
nm-openvpn:120:126  
saned:121:128  
colord:122:129  
geoclue:123:130  
pulse:124:131  
gnome-initial-setup:125:65534  
hplip:126:7  
gdm:127:133
```

correction :

```
[darksasuke@T-800 ~]$ cut -d: -f1,3,4 /etc/passwd > users
```

3) en fonction d'un nom d'utilisateur retrouver son nom de groupe (2 commandes)

```
● vel@vel-Virtual-Machine:~$ cat /etc/group | grep $(cat /etc/passwd | grep vel | cut -d":" -f4) | cut -d":" -f1,3  
vel:1000
```

correction :

```
[darksasuke@T-800 ~]$ grep $(grep darksasuke /etc/passwd | cut -d: -f4) /etc/group  
darksasuke:x:1000:
```

4) afficher les utilisateurs qui ont une uid qui commence par 1

```
● vel@vel-Virtual-Machine:~$ cut -d: -f1,3 /etc/passwd | grep ":1"  
daemon:1  
uucp:10  
proxy:13  
systemd-network:100  
systemd-resolve:101
```

Correction :

```
[darksasuke@T-800 ~]$ grep x:1 /etc/passwd | cut -d: -f1,3
```

5) En fonction d'un nom d'utilisateur afficher l'espace disque utilisé par son home directory  
vous aurez besoin de la commande 'du' , je vous laisse regarder sur le manuel

```
● vel@vel-Virtual-Machine:~$ du -h -s $(grep vel /etc/passwd | cut -d":" -f6)
 260M      /home/vel
```

Correction :

```
[darksasuke@T-800 ~]$ du -hs $(grep darksasuke /etc/passwd | cut -d: -f6)
449M      /home/darksasuke
```

6) A partir d'un nom d'user, afficher tout les users appartenant a son groupe primaire (2 commandes)

```
● vel@vel-Virtual-Machine:~$ cat /etc/group | grep $(cat /etc/passwd | grep greymane | cut -d":" -f4) | cut -d:" -f1,4
greymane:
```

Correction :

```
[darksasuke@T-800 ~]$ grep $(grep darksasuke /etc/passwd | cut -d: -f4) /etc/passwd
darksasuke:x:1000:1000:darksasuke,,,:/home/darksasuke:/bin/bash
user42:x:1004:1000::/home/user42:/bin/sh
```

7) afficher rangés dans l'ordre croissant les noms de groupes qui n'ont pas de chiffre '1'  
dans leur GID

```
● vel@vel-Virtual-Machine:~$ cat /etc/group | grep -v -e ":1" | cut -d:" -f1 | sort -g
adm
audio
backup
bin
cdrom
dialout
dip
disk
fax
floppy
gallywix
games
garrosh
gnats
greymane
grp_project
irc
```

Correction :

```
[darksasuke@T-800 ~]$ cat /etc/group | grep "1" -v | cut -d: -f1,3 | sort -k2 -t: -n
```

8) afficher l'UID de l'utilisateur OWNER du dossier /home/<utilisateur>

```
● vel@vel-Virtual-Machine:/home$ cat /etc/passwd | grep $(ls -la | grep 17 | cut -d" " -f3) | cut -d:" -f1,3
vel:1000
```

Correction :

```
[drwxr-x--- 19 darksasuke darksasuke 4096 dec. 14 16:48 /home/darksasuke/
[darksasuke@T-800 ~]$ ls -ld /home/darksasuke/ | sed "s/ +/ /g" -E | cut -d" " -f3
darksasuke
```

```
[darksasuke@T-800 ~]$ grep $(ls -ld /home/darksasuke/ | sed "s/ +/ /g" -E | cut -d" " -f3) /etc/passwd | cut -d: -f3
1000
```

```
grep $(ls -ld /etc | sed "s/ +/ /g" -E | cut -d" " -f3):x /etc/passwd | cut -d: -f3
```

9) - creer un nouvel utilisateur sans l'option '-m'

```
vel@vel-Virtual-Machine:~$ sudo useradd deathwing --shell /bin/bash
```

- creer son home directory a partir de /etc/passwd

Correction

```
[darksasuke@T-800 ~]$ grep user1:x /etc/passwd | cut -d: -f6  
/home/user1  
[darksasuke@T-800 ~]$ $(grep user1:x /etc/passwd | cut -d: -f6)  
bash: /home/user1: No such file or directory  
[darksasuke@T-800 ~]$ sudo mkdir $(grep user1:x /etc/passwd | cut -d: -f6)  
[sudo] password for darksasuke:  
[darksasuke@T-800 ~]$ ls /home/  
darksasuke my_user projet projet2 user1 user42
```

```
$>sudo mkdir $(grep user566:x /etc/passwd | cut -d: -f6) -p
```

10) afficher l'utilisateur qui a son home directory (/home/\*) le plus lourd

```
10) afficher l'utilisateur qui a son home directory (/home/*) le plus lourd  
$> sudo du /home/* -sh 2> /dev/null | sort -h | tail -1
```

11) afficher tous les fichier dont l'utilisateur demandé est le propriétaire

```
[darksasuke@T-800 ~]$ sudo find / -type f -user darksasuke 2> /dev/null
```

12) ecrire dans un fichier le nom de tout les fichiers qui ont été modifiés aujourd'hui

Pour avoir la date en format anglo-saxon

```
[darksasuke@T-800 ~]$ date "+%Y-%m-%d:%S"  
2022-12-15:28
```

```
12) ecrire dans un fichier le nom de tout les fichiers qui ont été modifiés aujourd'hui  
$>find . -newermt $(date "+%Y-%m-%d")
```

13) - Trouver tout les fichiers et leur taille dont le chemin contient 'log' (commande 'du')

```
13) - Trouver tout les fichiers et leur taille dont le chemin contient 'log' (commande 'du')  
$> sudo du / -ah | grep log
```

Exercice 13 BiS

Compresser dossier :

```
[darksasuke@T-800 ~]$ tar -zcvf archive.tar.gz temp/
```

```
13.1) copier les trois fichiers les plus lourds qui contiennent log dans un dossier temporaire  
$> sudo du / -ah 2> /dev/null | grep log | sort -h | tail -3 | cut -f2 > fichier
```

```
$> cp $(cat /etc/cent) temp -i

13.2) faire une archive compressée de ce dossier temporaire (tar)

$>tar -zcvf archive.tar.gz temp/
# decompression

$>tar -zxvf archive.tar.gz
```

Expression régulière

Une expression régulière est une suite de caractères typographiques (qu'on appelle plus simplement « motif » – « pattern » en anglais) décrivant un ensemble de chaînes de caractères. Par exemple l'ensemble de mots « ex-equ, ex-equo, ex-aquo et ex-aquo » peut être condensé en un seul motif « ex-(a? e|æ|é)quo ».

#### Opérateurs [ modifier | modifier le code ]

Opérateurs	Description	Exemples		
		Expression régulière	Chaines décrites	Chaines non décrites
<code>expr<sub>1</sub>expr<sub>2</sub></code>	Opérateur de concaténation de deux expressions (implicite).	ab	« ab », « b », chaîne vide	
.	Un caractère et un seul	.	« a », « b », etc.	chaîne vide, « ab »
<code>expr?</code>	Ce quantificateur correspond à ce qui le précède, présent zéro ou une fois. Si de multiples correspondances existent dans un texte, il trouve d'abord ceux placés en tête du texte et retourne alors la plus grande longueur possible à partir de cette position initiale.	a?	chaîne vide, « a »	« aa », « b »
<code>expr+</code>	Ce quantificateur correspond à ce qui le précède, répété une ou plusieurs fois. Si de multiples correspondances existent dans un texte, il trouve d'abord ceux placés en tête du texte et retourne alors la plus grande longueur possible à partir de cette position initiale.	a+	« a », « aa », « aaaa », etc.	chaîne vide, « b », « aaab »
<code>expr*</code>	Ce quantificateur correspond à ce qui le précède, répété zéro ou plusieurs fois. Si de multiples correspondances existent dans un texte, il trouve d'abord ceux placés en tête du texte et retourne alors la plus grande longueur possible à partir de cette position initiale.	a*	chaîne vide, « a », « aaa », etc.	« b », « aaab »
<code>expr<sub>1</sub> expr<sub>2</sub></code>	C'est l'opérateur de choix entre plusieurs alternatives, c'est-à-dire l'union ensembliste. Il peut être combiné autant de fois que nécessaire pour chacune des alternatives possibles. Il fait correspondre l'une des expressions placées avant ou après l'opérateur. Ces expressions peuvent éventuellement être vides, et donc (x) équivaut à x?.	a b	« a », « b »	chaîne vide, « ab », « c »
[liste]	Un des caractères entre crochets (« classe de caractères »)	[aeiou]	« a », « e », « i », etc.	chaîne vide, « b », « ae »
[^liste]	Un caractère n'étant pas entre crochets (« classe de caractères »)	[^aeiou]	« b », etc.	chaîne vide, « a », « bc »
(expr)	Groupement de l'expression entre parenthèses	(déTECTé)	« détECTé »	« détECT », « détECTA », « détECTÉS »
<code>expr{n}</code>	Exactement <i>n</i> occurrences de l'expression précédant les accolades	a{3}	« aaa »	« aa », « aaaa »
<code>expr{n,m}</code>	Entre <i>n</i> et <i>m</i> occurrences de l'expression précédant les accolades	a{2,4}	« aa », « aaa », « aaaa »	« a », « aaaa »
<code>expr{n,}</code>	Au moins <i>n</i> occurrences de l'expression précédant les accolades	a{3,}	« aaa », « aaaa », « aaaaa », etc.	« aa »
<code>^</code>	Ce prédictat ne correspond à aucun caractère mais fixe une condition nécessaire permettant de trouver un accord sur ce qui le suit en indiquant que ce doit être au début d'une ligne (donc être au début du texte d'entrée ou après un saut de ligne). Il ne peut être considéré ainsi qu'au début de l'expression régulière, ailleurs il est considéré littéralement. Il s'applique comme condition à la totalité du reste de l'expression régulière (et concerne donc toutes les alternatives représentées).		^a trouve « a » en début de ligne mais pas dans « ba ».	
<code>\$</code>	Ce prédictat ne correspond à aucun caractère mais fixe une condition nécessaire permettant de trouver un accord sur ce qui le précède en indiquant que ce doit être à la fin d'une ligne (donc être à la fin du texte d'entrée ou juste avant un saut de ligne). Il ne peut être considéré ainsi qu'à la fin de l'expression régulière, ailleurs il est considéré littéralement. Il s'applique comme condition à la totalité du reste de l'expression régulière (et concerne donc toutes les alternatives représentées).		a\$ trouve « a » en fin de ligne mais pas dans « ab ».	

Site pour tester les regex :

<https://regex101.com/>

Exemple d'utilisation du site :

The screenshot shows a 'REGULAR EXPRESSION' tool interface. The 'REGEX' input field contains the pattern `.\com$`. The 'TEST STRING' input field contains the following lines:  
l%l  
l@l  
.com  
.org  
loup.com

Caractère : "a" "c"

N'importe quel char : .

Opérateur "OU" : a|b

Classe de char : [ab0-9f-w]

Début ligne : ^

Fin ligne : \$

Priorité : 0

Quantifieur :

expr?	Ce quantificateur correspond à ce qui le précède, présent <b>zéro ou une fois</b> . Si de multiples correspondances existent dans un texte, il trouve d'abord ceux placés en tête du texte et retourne alors la plus grande longueur possible à partir de cette position initiale.
expr+	Ce quantificateur correspond à ce qui le précède, répété <b>une ou plusieurs fois</b> . Si de multiples correspondances existent dans un texte, il trouve d'abord ceux placés en tête du texte et retourne alors la plus grande longueur possible à partir de cette position initiale.
expr*	Ce quantificateur correspond à ce qui le précède, répété <b>zéro ou plusieurs fois</b> . Si de multiples correspondances existent dans un texte, il trouve d'abord ceux placés en tête du texte et retourne alors la plus grande longueur possible à partir de cette position initiale.

Exemple de quantifieur :

The screenshot shows a 'REGULAR EXPRESSION' tool interface. The 'REGEX' input field contains the pattern `^le?`. The 'TEST STRING' input field contains the following lines:  
la  
A  
Z  
C  
F  
G

le ↵  
lu ↵

Regex Numéro de téléphone :

The screenshot shows a regular expression tester. The regular expression field contains: `^ [0-9]+[ \. \-] [0-9]+[ \. \-] [0-9]+[ \. \-] [0-9]+[ \. \-] [0-9]+`. The test string field contains several phone number patterns: `04 12 43 56 76`, `02 42 52 42 42`, `02.42.52.42.42`, and `02-42-52-42-42`. The results pane shows the matches highlighted in blue.

Regex faire matcher des chaines de caractères :

The screenshot shows a regular expression tester. The regular expression field contains: `a{3,5}`. The test string field contains the same phone number patterns as the previous screenshot. The results pane shows matches for the first four phone numbers, with the last one failing to match because it is longer than 5 characters.

Quantifieur ? : 0 ou 1  
 Quantifieur + : 1 ou +  
 Quantifieur \* : 0, 1 ou +  
 Quantifieur {n} -> n fois

Exercice pratique Regex :

### https://regex101.com/

##### regex

## url

# match

```
http://wikipedia.de
http://wikipedia.fr
http://www.framasoft.org
https://mail.ville.de
wikipedia.xyz
facebook.com
www.lyon-ville9.fr
```

```
# pas match
http://ww.ww.w..www..Ww..
http://google.fr
https://www.data.drive.fr
http://fr
facebook
```

REGULAR EXPRESSION

```
:^(http(s)?:(\/\/)?[a-z]*\.[a-zA-Z0-9\-\_]*[a-z]*$)
```

TEST STRING

```
## url
## match
http://wikipedia.de
http://wikipedia.fr
http://www.framasoft.org
https://mail.ville.de
wikipedia.xyz
facebook.com
www.lyon-ville9.fr
## pas match
http://ww.ww.w..www..Ww..
http://google.fr
https://www.data.drive.fr
http://fr
facebook
```

Correction :

REGULAR EXPRESSION

```
^(https?:\/\/)?[A-Z0-9]+\.?[A-Z0-9\-\_]+\.[A-Z0-9]+$
```

```
# mail

# match

loupfournment@gmail.com
loup.fourment@site.fr
loup.fou.ment@yahoo.de
loup-fourment_13@site.xyz

loup-fourment_13@site.xyz
```

```
# pas match

@gmail.com
loup@@gmail.com
loup@gmail.gmail.com
loup@gmail
loup$!/*%"#@gmail.com
loupgmail.com
```

TEST STRING

```

# mail
# match
loupfourment@gmail.com
loup.fourment@site.fr
loup.fou.ment@yahoo.de
loup-fourment_13@site.xyz
loup-fourment_13@site.xyz
```

Correction :

REGULAR EXPRESSION

```

#[A-z0-9\.\-\_]+@[A-z0-9\-\-]+\.\?[A-z0-9]+\$
```

```

## MAC hexadecimal :00:FF:
0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
```

# match

```

01:2A:3B:4C:5D:6E
AA:BB:23:54:ef:9b
```

# pas match

```

11:22:33:44:55:66:
:11:22:33:AA:BB:CC
11:AA:22:BB:33
44:55:EE:FF:22:33:22
11:RR:GG:44:RR:23
```

REGULAR EXPRESSION

```

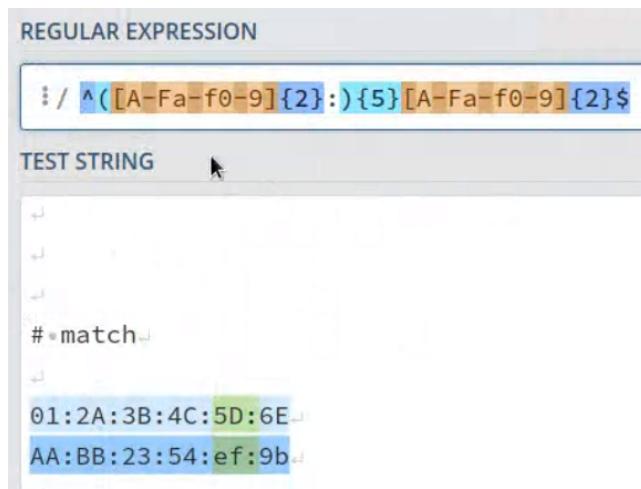
#[A0-1]*:[A-B0-2]*:[B-B2-3]*:[C-C4-5]*:[D-f5]*:[E-b6-9]*$
```

TEST STRING

```

## MAC hexadecimal :00:FF:
0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
# match
01:2A:3B:4C:5D:6E
AA:BB:23:54:ef:9b
# pas match
11:22:33:44:55:66:
:11:22:33:AA:BB:CC
```

```
11:AA:22:BB:33:  
44:55:EE:FF:22:33:22:  
11:RR:GG:44:RR:23:  
4
```



```
## /etc/passwd

# match avec toutes les lignes dans /etc/passwd

git:x:976:976:git daemon user:/usr/bin/git-shell
mongodb:x:973:973:/var/lib/mongodb:/usr/bin/nologin
ntp:x:87:87:Network Time Protocol:/var/lib/ntp:/bin/false/
avahi:x:975:975:Avahi mDNS/DNS-SD daemon:/usr/bin/nolo
colord:x:974:974:Color management daemon:/var/lib/colord:/usr/bin/nologin
polkitd:x:102:102:PolicyKit daemon:/usr/bin/nologin
mpd:x:45:45:/var/lib/mpd:/usr/bin/nologin
rtkit:x:133:133:RealtimeKit:/proc:/usr/bin/nologin
mongodb:x:973:973:/var/lib/mongodb:/usr/bin/nologin
```

# pas match

```
git:x:976:976:git daemon user:/usr/bin/git-shell
ntp:x:87:87:Network Time Protocol:/var/lib/ntp:56
avahi:x:avaji:975:Avahi mDNS/DNS-SD daemon:/usr/bin/nologin
colord:x:974:974:/var/lib/colord:/usr/bin/nologin
:polkitd:x:102:102:PolicyKit daemon:/usr/bin/nologin
mpd:x:45:45:/var/lib/mpd:/usr/bin/nologin:
```

Correction :



```
## grammes
```

```
# match
```

```
1 mdg
```

```
1mg
```

```
2 cg
```

```
13 dg
```

```
123.43 g
```

```
23.01 dag
```

```
0.01 hg
```

```
400kg
```

```
# pas match
```

```
1 ae
```

```
23,2mg
```

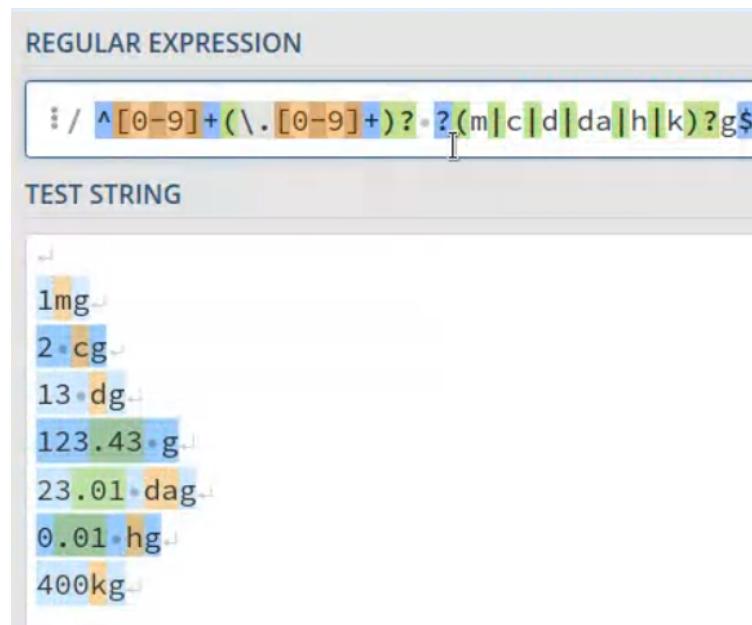
```
13 gg
```

```
12.12.32 kg
```

```
12.32 kg kg
```

```
12 12 kg
```

Correction :



adresse

```
# match ( avec route+nom_de_route ; quartier+nom_de_quartier ; arrondissement+numero_arrondissement(1-99))
```

```
13300 Paris
```

```
12 route arc en ciel
```

```
14200 Poitiers
```

```
9 quartier Mazaugues
```

11111 Marseille  
99 arrondissement 99

11111 Marseille  
99 arrondissement 40

# pas match

11111 Marseille  
00 arrondissement 44

Paris  
9 route douce

13300 li4lo  
22 quartier fleur

13333 Lyon  
AR arrondissement 99

11111 Marseille  
99 arrondissement 00

11111 Marseille  
99 arrondissement 123

13333 Lyon  
12 espace 99

Correction :

REGULAR EXPRESSION TEST STRING 12 •route•arc•en•ciel 42@O.PoitiersJ 9•quartier.Mazauge 99 •arrondissement •99 99 • Arrondissement'44 4 matches (660 steps, 1.0ms) — +\n 199it@k9f'.aroutelquartier' IA9z+Iarrondissement\* / gm

##### en plus

# Chemins  
# match

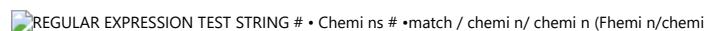
/  
/chemin/

```
/chemin/chemin/  
/chemin/chemin
```

```
# pas match
```

```
///  
/chemin//chemin  
chemin/chemin
```

Correction :



REGULAR EXPRESSION TEST STRING # • Chemi ns # •match / chemi n/ chemi n (Fhemis n/chemi

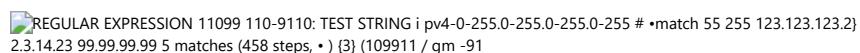
```
## ipv4 0-255.0-255.0-255.0-255
```

```
# match
```

```
0.0.0.0  
255.255.255.255  
123.123.123.123  
12.3.14.231  
99.99.99.99
```

```
# pas match
```

```
555.555.555.555  
1.2.3  
1.32.42.12.43  
13.42.33.23.  
001.2.002.019.2  
564.23.299.1
```



REGULAR EXPRESSION 11099 110-9110: TEST STRING i pv4-0-255.0-255.0-255.0-255 # •match 55 255 123.123.123.2}  
2.3.14.23 99.99.99.99 5 matches (458 steps, • ) {3} (109911 / gm -91

 Fonctions avancées modifier I modifier le code La plupart des standards et moteurs d'expressions régulières proposent des fonctions avancées. Notamment : • Quantificateurs non gloutons : par défaut, les quantificateurs « + » et \* recherchent la plus grande séquence correspondant au motif recherché. Ces nouveaux quantificateurs, souvent notés « +? » et « \*? » permettent à l'inverse de rechercher la plus petite séquence correspondante. par exemple, l'expression régulière ab. ? appliquée à la chaîne abbbb entre en correspondance avec la sous-chaine ab n plutôt que « abbbb ». • Capture des groupements : La capture des groupements permet de réutiliser un groupement entré en correspondance pour un traitement ultérieur, par exemple une substitution. Dans la plupart des syntaxes, il est possible d'accéder au groupement capturé par la syntaxe `\nouparfois`, où `n` est un entier. • Groupements non capturants : Lorsqu'elle est implémentée, la capture des groupements est souvent le comportement par défaut. Comme elle a un coût algorithmique important, il est parfois possible de désactiver la capture de certains groupements. On peut par exemple citer la syntaxe (?: groupement) `n`. • Captures nommées : Les longues expressions régulières avec de nombreux groupements peuvent être complexes à lire et à maintenir. Pour faciliter cette tâche, certains moteurs permettent de nommer les groupements, par exemple avec la syntaxe « `{name}` » en Python. • Références arrières : Les références arrières permettent de faire référence à un même groupement capturé, par exemple « `b.` » entrera en correspondance avec `bébé` et « `bobo` mais pas `baby` ». Cette fonctionnalité, proposée par la plupart des moteurs, permet de reconnaître des langages non rationnels tels que `anban` pour tout `n` entier positif. • Modificateurs de mode de correspondance : ces modificateurs permettent de faire varier localement le comportement de l'expression régulière. Par exemple, alors qu'elles sont normalement sensibles à la casse, l'expression `perl` « (: nom) -Prénom » entrera en correspondance avec `NOM-Prénom` et `Nom-prénom` mais pas `Nom-prénom n`. Parmi les autres modificateurs, on peut citer le mode multi-lignes, le mode non-gourmand ou le free-spacing mode `n`. • Conditions : certains moteurs permettent de construire des structures « `if . then . el se . au` » dans le sein même des expressions régulières. Par exemple, en Perl, l'expression `"\defl 1 \ghi)" » selit « sila chaine` commence par la lettre `a`, boucle, chercher à `\A`, ir suite la lettre `d`, `e` ou `f`, sinon chercher la `g`, `h` ou `i`. • Commentaires : Dans un sotin de lisibilité, des moteurs permettent de commenter les expressions régulières au sein même de celles-ci. • Code embarqué : Lorsqu'une expression régulière est utilisée au sein d'un programme, cette fonctionnalité permet de déclencher des actions lorsqu'une partie de la chaîne est entrée en correspondance.

La commande curl et le code HTML :

Installation curl : `sudo apt install curl -y`

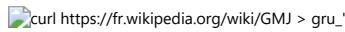
 sudo apt install curl [sudo] Mot de passe de root. Lecture des listes de paquets... Fait Construction de 1 arbre des dépendances.. Fait Lecture des informations d'état.. Fait Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires : i965-virtual intel-media-video-driver libvaS2-a.7.4 libvaacsa libvaom3 libvaibb24-a libass9 libavcodec58 libavformat58 libavutil156 libbdplusa libb1uray2 libcddb2 libdvdrad8 libebml15 libfaad2 libflashrom libftdi1-2 libgmea libgsml libigdgmm12 libixmlla libkate1 lib1uaS2-a libmada libmatroska7 libmd4ca libpgm-5.3-a libp1aceb0192 libpostprocSS libprotobuf-1.16.2 libproxy-tools libqtScoreSa libqtSdbusS libqtSguis libqtSnetwork5 libqtSsvgS libqtSwidgetsS libqt libsnappy1v5 libsnappy1v5 libsnappy1v5 libsnappy1v5 libspatialaudioa libsr1.4-gnu1s libssh-gcrypt-4 libssh2-1 libswresamp1e3 libswsc1eS libudfreada libupnp13 libvadrm2 libvawaylan libxcb-composite libxcb-xinerama libxcb-xinput libxvidcore4 libzmq5 libzvbi-common libzvbia mesa-virtual drivers mesa-vdpau-drivers qts-gtk-platformtheme qttra vlc-plugin-base vlc-plugin-notify vlc-plugin-qt vlc-plugin-samba vlc-p1ugin-skins2 vlc-plugin-video-output vlc-plugin-video-splitter vlc-plugin-visualization Veuillez utiliser « sudo apt autoremove » pour les supprimer. Les paquets suivants seront installés : curl a mis à jour, 1 nouveau(s) installé(s), 0 à enlever et 4 non mis à jour. Il est nécessaire de prendre 194 Ko dans les archives. Après cette opération, 453 Ko d'espace disque supplémentaires seront utilisés. Réception de : 1 http://fr.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.1-1ubuntu1.6 [194 Ko]

Exercice GNU / HTML :

Avec 'curl', recuperer le code HTML de la page: <https://fr.wikipedia.org/wiki/GNU> et le mettre dans le fichier 'gnu'

```
curl https://fr.wikipedia.org/wiki/GNU > gnu
```

Récupération de la page wikipédia GNU dans un fichier avec la commande curl : `curl https://fr.wikipedia.org/wiki/GNU > gnu`

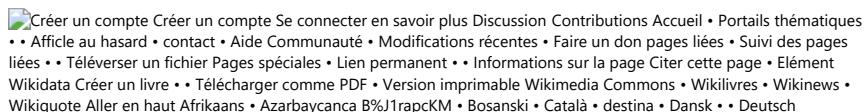


```
[EXPLORATEUR VEL [SSH: VEL-VIRTUAL-MACHINE].cache > .config >.gnupg >.local >.ssh .vscode-server Bureau Documents exerciceRegex o gnu >
Images > Modèles Musique > Public > snap Téléchargements jeNeSuisPasUnFichierPython.py > Vidéos .bash_history o gnu 9. Documents > gnu > 4 8 14 <
!DOCTYPE html <html class="client-nojs" <head ><meta >Wikipédia</title> <script >document.documentElement.className=" client -j s";<
wgBreakFrames": false, "wgSeparatorTransformTable": [", " >"Portail:Informatique/Articles liés", "Portail:Technologies/Articles liés",
"Portail:Sciences/Articles liés", "Portail:L01 "wgCentralAuthMobi1eDomain": false, "wgULSposition": " interlanguage ", "wgULSisCompactLinksEnabled"
: true, "wgWikibaseItemId": "ext . visual Editor. targetLoader" " ext . eventLogging" " ext . wikimediaEvents ", " ext . navigationTiming", " ext . cx . eventlogging,
<script>(RLQ=window. RLQI I C) .push(function(){mu.loader.implement("user.options@12s5i",function($,jQuery, require, module <link rel="stylesheet"
cite="styless%7Cext.U1s . interlanguage%7Cext.visualEditor0 <script async="" src=" "w' load_ ; module ; $kin -vector -2822" / <script <meta <link <meta
<meta <meta <meta <meta <meta <meta name= name= name= name= name= name= name= name= "ResourceLoaderDynamicStyles" content="" rel=""
stylesheet" " generator" "referrer" "referrer" h ref" "/w/ o ad . ; modules = $ite styl es&amp;only=styl es&amp; ; $kin n -vector -2822" / >
content="MediaWiki content="origin-when-crossorigin" /> "robots" content=" max -image -preview : standard" / > "format-detection" image"
content="https://upload.wikimedia.org/wikipedia/commons/d/d/HJURD Live_CD. image width" content=" 12ae"
```

En HTML les liens sont notés dans des balises "`<a href=> </a>`"

Récuperer toutes les balises de liens dans la page gnu et les stocker dans un fichier 'liens'

# Si on fait 'firefox liens' on ne voit que des liens"



Correction :

 En HTML les liens sont notés Récuperer toutes les balises un fichier liens on fait 'firefox liens' dans des de liens balises "<a href=" dans la page gnu et les stocker dans on ne voit que des liens" grep # on sed —Po "<a . \*?</a>" gnu > liens enleve les —E "s@<img balises < img> —i liens

-----  
Soigner l'affichage pour afficher une jolie liste en html (sed).

Il faut rajouter devant chaque ligne contenant <a>:

<li>

Lazy mode : grep -P

a la fin de chaque ligne contenant </a>:

</li>

Pour réussir cet exercice, j'ai modifié les délimiteur de base "/" par des "@"

Il faut rajouter au tout début du fichier 'liens':

<html><body><ul>

 4 href= " i ndex. php ? titl : href= " 'w/ index. php ? titl : href= " index. php : Connexion& returnto= href= "/wiki/Aide: Introduction" aria-labe1="En savoir plus sur href=" :Mes\_discussions" page de d href=" tit1e="Accue1 href=" /wiki/Aide:DZC3ZA9buter" sur Wikipédia </s href=" d href= " 'w/ index. php : Livre& bookcmd=book href="https://commons.wikimedia.org/wiki/GNU" hreflang="en"> < CONSOL DE DÉBOGAGE TERMINAL 'ROBLÈMES SORTIE 'elUe1-Virtua1-Machine:m\$ sed - i \*

 Ajouter une ligne en début de fichier Pour insérer une ligne au début d'un fichier, il faudra utiliser la commande sed. Voici un exemple d'utilisation S cat test.txt ligne2 S sed -i '1l\_igne' test.txt S cat test.txt ligne1 ligne2 On notera l'utilisation du code li au début de la chaîne à insérer. Le chiffre 1 indique à sed d'insérer cette phrase sur la première ligne.

Et tout en bas du fichier 'liens':

```
</ul><body><html>
```

 325 326 PROBLÈMES SORTIE CONSOLE DE DÉBOGAGE TERMINAL ve1Ue1-Virtua1-Machine:m\$ echo 'PORTS'

# si on fait 'firefox liens' on voit une jolie liste de liens HTML

---

++++

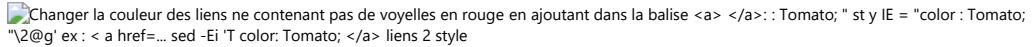
Changer la couleur des liens ne contenant pas de voyelles en rouge

en ajoutant dans la balise <a> </a>:

```
style="color:Tomato;"
```

ex:

```
<a href=... style="color:Tomato;"> </a>
```

 Changer la couleur des liens ne contenant pas de voyelles en rouge en ajoutant dans la balise <a> </a>: : Tomato; " st y IE = "color : Tomato; "\2@g' ex : < a href=... sed -Ei 'T color: Tomato; </a> liens 2 style

Quelques conseils de recherche :

- écrire nos questions en anglais
- Spécifier langage | langage utilisé
- StackOverFlow
- La documentation officielle (doc ubuntu et arch peut-être utilise pour toutes les distributions)

Exercice de recherche & introduction aux scripts bash :

work.sh

quand l'user lance:

```
$>bash work.sh
```

on met un filtre jaune a l'écran

on baisse de volume de sortie

on ouvre code|libreoffice a gauche de l'écran

on ouvre firefox a droite de l'écran avec deux onglets:

un onglet youtube/musique, un onglet doc python

on lance quake

cinema.sh

quand l'user lance

```
$> bash cinema.sh
```

on baisse la luminosité de l'écran

Sous VM impossible car il ne détecte pas le display

```
↳ xrandr | grep "connected" | cut -f1 -d " " | sed -t open display
```

on monte le volume de sortie

```
↳ vel@vel-Virtual-Machine : - / Document tsS vel@vel-Virtual-Machine : -/Document tsS pactl eactl set-stnk-volume set-stnk-volume e 150%
```

on ouvre vlc en plein écran

