

But

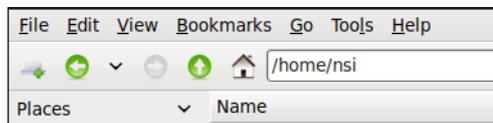
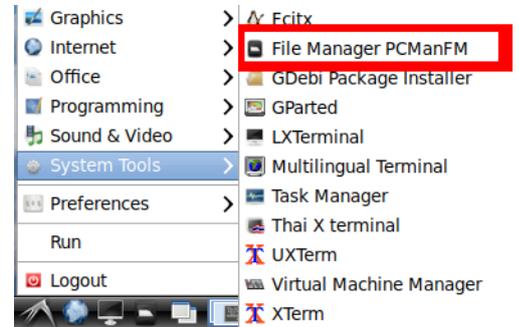
Ce TP-projet devrait vous permettre de vous familiariser avec la distribution GNU/Linux Debian 10 buster [<https://www.debian.org/>], mieux comprendre l'arborescence des fichiers ainsi que découvrir ce qu'est un terminal.

I Arborescence

Ouvrir le manager de fichiers.



ou



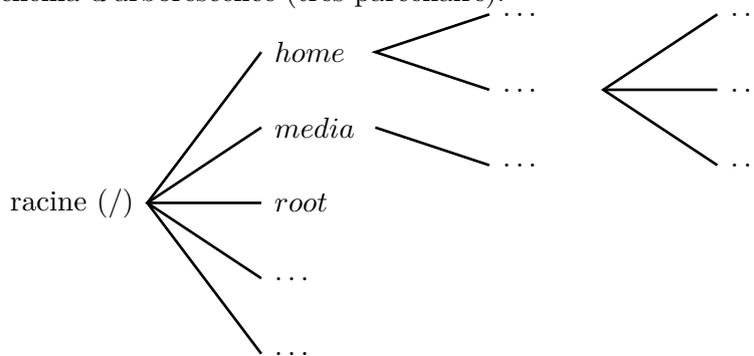
1. Par défaut, dans quel répertoire êtes-vous ?

2. Lister tous les répertoires à l'intérieur (ces répertoires sont les 'enfants' du précédent).

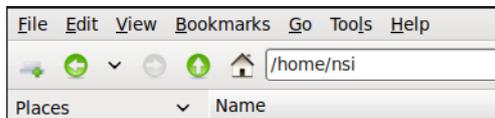
3. Existe-t-il un répertoire 'parent' ? Quel est-il ?

4. Comment est noté le seul répertoire qui est son propre parent (il s'appelle le répertoire 'racine') ?

On peut dessiner sous forme d'arbre (vertical ou horizontal) l'ensemble des relations entre les répertoires et fichiers. Compléter ce schéma d'arborescence (très parcellaire).



Dans la barre d'adresse du file manager



, on peut lire l'adresse

absolue d'un répertoire. L'adresse absolue démarre à la racine (e.g. /home/user/ ou /home/user/Desktop/).

5. Dans cette barre d'adresse,

a) taper '/'. Où vous trouvez-vous ?

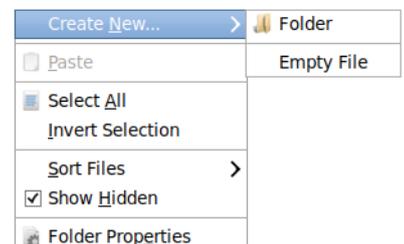
b) Taper '~' [Alt-Gr 2], où vous trouvez-vous ? Ajouter '/../', où vous trouvez-vous ?

c) Retaper '~' puis ajouter '/../', où vous trouvez-vous ?

6.

Sous le répertoire ~/ , selon votre NOM et votre prénom, créer un répertoire NOMp (e.g. Eiichiro ODA créera le répertoire ODAe).

Puis dans ce répertoire créer un fichier onepiece.txt. Préciser de deux manières l'adresse de ce fichier.



clic-droit puis choisir l'option voulue :

Récapitulons,

- ◇ pour chaque utilisateur (user), ~/ équivaut à l'adresse absolue /home/user/. L'adresse ~/Documents/ est donc une adresse absolue.
- ◇ à partir d'un répertoire, '../' fait référence au répertoire parent. Tandis que './' fait référence au répertoire courant.
- ◇ À partir du répertoire ~/ , on peut définir l'adresse relative du fichier onepiece.txt par ./ODAE/onepiece.txt (ou simplement ODAE/onepiece.txt). Ces deux adresses relatives à ~/ font référence à l'adresse absolue /home/user/ODAE/onepiece.txt.

Exo 1 : Préciser les adresses absolues des fichiers suivants :

~/Documents/file1 ; ~/../file2 ; ~/../../file3 ; ~/ODAE../file4 ; ~/../../file5

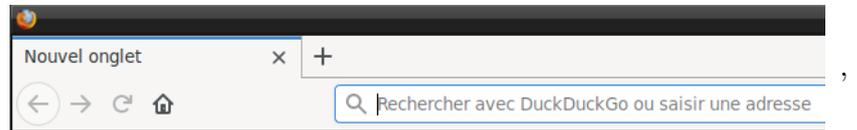
Exo 2 : Préciser les adresses relatives au répertoire /home/ des fichiers suivants :

~/Documents/file1 ; ~/../file2 ; ../../file3 ; /media/file4 ; ~/../../file5

Ouvrir un navigateur internet et aller sur le site [<http://forhan.maths.free.fr/>] section NSI.

7. Télécharger le fichier de ce TP-projet ; NSI-projet00.pdf. Sous quel répertoire a-t-il été téléchargé par défaut ? Le copier dans votre répertoire ~/NOMp/.

8. Dans la barre url de votre navigateur



taper file:/// . Où vous trouvez-vous ?

9. Que faudrait-il taper pour pouvoir lire le contenu du répertoire ~/NOMp/ ? Le vérifier avec votre répertoire.

Exo 3 : Que faut-il taper dans l'url du navigateur firefox pour voir le contenu du répertoire :

~/Documents/? de ~/NOMp/.../? de ~/../Documents/~/NOMp/..?

II Shell et lignes de commandes

Toutes les opérations que vous venez de faire peuvent être faites en ligne de commandes, notamment dans un terminal qui vous donne accès au shell (le shell est le programme qui interprète vos commandes).

Parmi les commandes de bases, il y a :

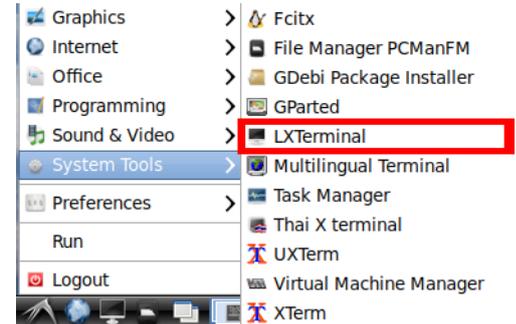
commande	acronyme de	signification	exemple
pwd	path working directory	affiche l'adresse du répertoire courant	pwd
cd arg	change directory	change de répertoire	cd ../user/Documents/
ls arg	list	liste les fichiers/répertoires contenus à l'adresse courante ou spécifiée	ls ls ../Documents/
ls -al arg	list all long	affiche tous les fichiers avec les détails des droits selon un ordre chronologique	ls -al ./
chmod arg	change moderation	change les droits des fichiers	chmod 755 ./file
mkdir arg	make directory	créé le répertoire spécifié à l'adresse fournie	mkdir ../NOMp/
rmdir arg	remove directory	efface le répertoire spécifié s'il est vide	rmdir ../NOMp/
rm arg	remove	efface le fichier spécifié	rm file
rm -r arg	remove recursively	efface tous les fichiers/répertoires contenus dans le répertoire et ses enfants	rm -r ../Documents/
cp arg	copy	copie le fichier1 dans le fichier2	cp file1 file2
mv arg	move	déplace le fichier1 dans le fichier2	mv file1 file2
more arg	more	affiche le contenu du fichier (similaire à less ou cat ; Q pour quitter)	more file1 (⇔ cat file1)
touch arg	touch	créé un fichier vide	touch file1
echo arg	echo	affiche la chaîne de caractères fournie	echo "Bonjour"
man arg	manual	affiche le manuel de la commande (Q pour quitter)	man ls (⇔ ls --help)

Rq : De nombreuses options existent pour chacune de ces commandes, e.g. `ls -ltp` permet de lister avec détail (l : long) par ordre chronologique (t : time) en ajoutant un slash (/) pour différencier les répertoires (p : print directory indicator). Se renseigner sur chaque commande à l'aide du `man`.

Ouvrir un terminal



ou



Remarquer qu'après le `:` il y a le sigle `~` suivi d'un dollar `$`. Vous êtes dans le répertoire `~/`.

1. afficher l'adresse du répertoire courant et vérifier qu'il corresponde au répertoire `~/`.
2. lister les fichiers et répertoire contenus dans ce répertoire.
3. se déplacer à la racine et lister le contenu.
4. se déplacer dans votre répertoire `~/NOMp/` et vérifier qu'un fichier `onepiece.txt` existe.
5. renommer ce fichier en `mangas.txt` et afficher le contenu du fichier (il doit être vide).
6. créer un répertoire `TP0` et déplacer le fichier `mangas.txt` dedans. Se déplacer dans le répertoire `TP0` puis lister le contenu.

Exo 4 : On vient d'ouvrir un terminal, comment faire pour

- a) créer un répertoire `~/NOMp/Test/` ?
- b) puis créer trois fichiers `eleve1`, `eleve2`, `eleve3` ?
- c) enfin effacer tout le répertoire ainsi créé ?

À ce niveau, vous êtes capables de créer, renommer, naviguer dans, des répertoires et des fichiers.

Sigles et caractères spéciaux : `*`, `?`, `[]`, `>`, `<`, `|`

- ◇ `*` remplace autant de caractères inconnus que possible (il se peut que `*` représente 0 caractère)
- ◇ `?` remplace un caractère inconnu
- ◇ `[abc]` remplace un seul caractère parmi a, b ou c.
- ◇ `>` ou `<` redirige la sortie d'une commande vers un fichier ou un flot
- ◇ `|` redirige la sortie d'une commande vers l'entrée d'une autre commande

Une fois l'exercice 4 fait, il ne devrait plus y avoir de fichiers dans votre répertoire `~/NOMp/`.

7. Créer cinq fichiers `file`, `file2`, `file3`, `file4`, `file21`.
8. Taper `ls file*`, qu'est-ce que cela affiche ?
9. Taper `ls file?`, qu'est-ce que cela affiche ?
10. Que va afficher `ls file[123]` ? Vérifier.
11. Que va afficher `ls file[23]*?` ? Vérifier.
12. Taper `ls > filelist` et afficher le contenu du fichier `filelist`.
13. Taper `ls | more` (ou `ls | less`), qu'est-ce que cela affiche ? Quelle différence avec la commande `ls` ?
En fait, la commande `ls | less` correspond aux commandes chaînées : `ls > filetmp ; less < filetmp ; rm filetmp`

 `ls | echo` ne fonctionne pas car `echo` n'a pas d'entrée, il faudrait écrire `ls | xargs -L 1 echo`.

14. taper `echo "onepiece" >> mangas.txt` puis afficher le contenu du fichier `mangas.txt`.
15. taper `echo "dragon ball" >> mangas.txt` et réafficher le contenu du fichier `mangas.txt`. Que s'est-il passé? Que permettent les sigles `>>`?
16. ajouter quelques mangas que vous connaissez dans le fichier.
17. faire une copie du fichier `mangas.txt` en tant que `livres.txt`. Ajouter quelques livres que vous connaissez dans ce fichier.

Exo 5 : Proposer une solution pour lister les fichiers du répertoire dont le nom commencent par `eleve`; dont le nom contient le mot `eleve`; dont le nom terminent par `eleve`.

À ce niveau, vous êtes capables d'afficher le contenu d'un répertoire, créer un fichier à la volée et ajouter du texte en fin de fichier.

D'autres commandes existent, *e.g.* `grep`, `diff`, `find`, `top`, `kill`, `jobs`, `bg`, `fg`, `ps`, `chown`, `which`, ... [https://doc.ubuntu-fr.org/tutoriel/console_commandes_de_base]. Nous ne verrons que les 3 premières.

commande	nom	signification	exemple
<code>grep arg</code>	global search regular expression & print	recherche un mot dans un fichier	<code>grep mot file1</code>
<code>diff arg</code>	difference	compare deux fichiers entre eux	<code>diff file1 file2</code>
<code>find arg</code>	find	trouve le nom du fichier dans le répertoire et sous-répertoires fourni	<code>find ./ -name file</code>

Rq : pour ne pas avoir d'affiché les messages d'erreurs d'une commande, on peut les rediriger (`2>`) dans une sorte de trou noir (`/dev/null`), *e.g.* `find ./ -name file -print 2>/dev/null`.

18. Taper `grep "onepiece" mangas.txt`, le manga onepiece est-il bien contenu dans le fichier `mangas.txt`?
19. Qu'affiche `grep "one" mangas.txt`?
20. Taper `diff mangas.txt livres.txt`, quelles sont les différences entre les deux fichiers? Que faut-il ajouter au fichier `mangas.txt` pour obtenir `livres.txt`?
21. Se placer dans le répertoire `~/` et taper `find ./ -name "mangas.txt"`, que s'est-il affiché? Quelle commande doit-on taper pour avoir le chemin de tous les fichiers se finissant par `.txt`?

Enfin pour lancer des programmes à partir du terminal, on écrit le nom du programme suivi d'arguments (et possiblement de `&`), le shell lancera alors le programme voulu; *e.g.* `python file.py`.

22. taper `gedit livres.txt &`; à quoi correspond la commande `gedit`? Ajouter d'autres titres de livres puis sauvegarder.

Exo 6 : Comment faire pour comparer le contenu du répertoire `~/` avec le contenu du répertoire `~/NOMp/`?

À ce niveau, vous êtes capables de faire une recherche dans des répertoires/fichiers, comparer deux fichiers entre eux et ouvrir des applications depuis un terminal.