

NSI – 1ere	COURS	LFV
Linux : commandes utiles pour utiliser le système de fichiers		

Les systèmes d'exploitation Linux existent en versions avec interface graphiques (par exemple les machines virtuelles que nous utiliserons au lycée) ou en versions serveur (par exemple la machine virtuelle hébergeant le serveur <https://jupyter.ovh>). Lorsqu'on gère un serveur – souvent hébergé à distance – on n'a pas accès à une interface graphique mais à un terminal : toutes les opérations de manipulations des fichiers se font donc en ligne de commande. Nous allons ici étudier un certain nombre de ces lignes de commande.

Instructions de base : Exemples

Lorsqu'on navigue dans les fichiers et répertoires d'un ordinateur on effectue deux sortes d'opérations usuelles :

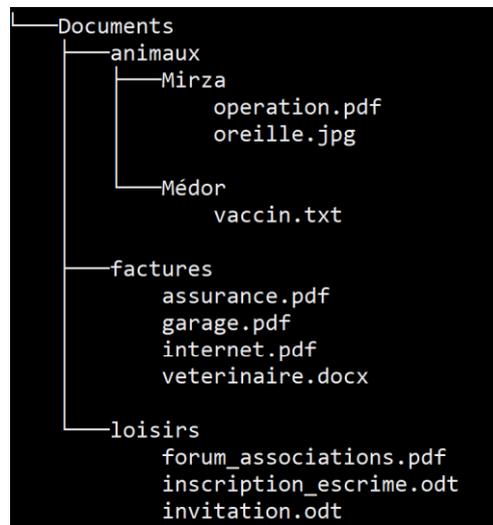
- **se déplacer dans les répertoires et afficher leur contenu** (ouvrir un répertoire, remonter dans le répertoire parent etc.),
- **manipuler les fichiers ou répertoires** (copier-coller ou supprimer un fichier, créer un nouveau répertoire etc.)

Ces opérations dépendent du répertoire dans lequel on se trouve que l'on appelle répertoire courant. Avec ce vocabulaire, se "déplacer" dans les répertoires c'est tout simplement modifier le répertoire courant.

Déplacement et affichage : cd, ls et pwd

On suppose qu'au début /Documents est le répertoire courant

```
$ cd factures # on se déplace dans factures qui devient le répertoire courant
$ ls # liste le contenu du répertoire courant
assurance.pdf garage.pdf internet.pdf veterinaire.docx
$ cd .. # on remonte d'un répertoire : le répertoire courant devient Documents
$ cd animaux/Mirza # on se déplace dans Mirza (en passant par animaux)
$ ls # liste le contenu du répertoire courant
operation.pdf oreille.jpg
$ pwd # affiche le chemin d'accès du répertoire courant depuis la racine
/Documents/animaux/Mirza/
```



Création, déplacement, copie, suppression de fichiers ou répertoires : mkdir, mv, cp, rm

On suppose qu'au début /Documents est le répertoire courant

```
$ cd factures # /documents/factures est désormais le répertoire courant
$ mkdir anciennes_factures # crée un répertoire anciennes_factures dans le répertoire courant
$ mv assurance.pdf anciennes_factures/ # déplace assurance.pdf dans le répertoire anciennes_factures
$ cp garage.pdf anciennes_factures/ # crée une copie de garage.pdf dans le répertoire anciennes_factures
$ rm internet.pdf # supprime le fichier internet.pdf
```

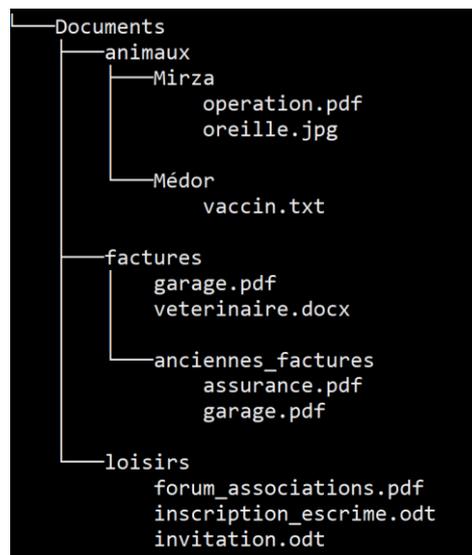
Une fois ces instructions effectuées, voici ce qu'est devenu le répertoire /Documents et son contenu.

On a un répertoire en plus (anciennes_factures) dans lequel il y a une copie de garage.pdf et dans lequel assurance.pdf a été déplacé. Le fichier internet.pdf a quant à lui été effacé.

Remarque : chemin relatif vs chemin absolu

Lorsqu'un chemin d'accès commence par / il est donné depuis la racine de l'arborescence (on parle de chemin absolu).

Lorsqu'un chemin d'accès commence par ./ ou par un nom de répertoire ou de fichier, il est donné depuis le répertoire courant (on parle de chemin relatif).



Instructions de base : Bilan

Se déplacer ou afficher des éléments du système de fichiers

<code>cd (change directory)</code>	changer de répertoire courant
<code>\$ cd chemin</code> <code>\$ cd ../../</code> <code>\$ cd /</code> <code>\$ cd ~</code>	<code># changer le rép. courant (chemin relatif ou absolu)</code> <code># remonter d'un niveau dans l'arborescence</code> <code># le répertoire racine devient le répertoire courant</code> <code># le répertoire de l'utilisateur devient le répertoire courant</code>
Remarque : ~ désigne le répertoire personnel alors que / désigne le répertoire racine (le "haut" de l'arborescence des fichiers)	
<code>Ls (list)</code>	lister le contenu d'un répertoire
<code>\$ ls</code> <code>\$ ls -l</code> <code>\$ ls -l chemin</code>	<code># liste les fichiers et répertoires du répertoire courant</code> <code># idem que ls en affichant les informations des fichiers et rép.</code> <code># idem que ls -l pour un répertoire autre que le répertoire courant</code>
<code>pwd (print working directory)</code>	Afficher le répertoire courant
<code>\$ pwd</code>	<code># affiche le chemin absolu du répertoire courant</code>
<code>cat, less, nano, more</code>	Afficher ou créer fichier texte
<code>\$ cat chemin</code> <code>\$ nano chemin</code>	<code># pour afficher le contenu d'un fichier texte</code> <code># pour éditer le contenu d'un fichier texte</code>

Créer, copier, supprimer ou déplacer des éléments du système de fichiers

<code>mkdir (make directory)</code>	créer un ou plusieurs répertoires
<code>\$ mkdir chemin/rep1</code>	<code># créer un répertoire rep1 dans le répertoire indiqué par le chemin</code>
<code>mv (move)</code>	déplacer ou renommer un fichier ou rép.
<code>\$ mv chemin_origine chemin_destination</code>	<code># déplace la ressource désignée par chemin_origine vers</code> <code># chemin_destination. Permet de renommer si</code> <code># chemin_destination ne correspond pas à un répertoire</code> <code># existant</code>
<code>cp (copy)</code>	copier un fichier ou répertoire
<code>\$ cp chemin_origine chemin_destination</code>	<code># copie la ressource désignée par chemin_origine vers</code> <code># chemin_destination. La copie de la ressource est renommée</code> <code># si chemin_destination ne correspond pas à un répertoire</code> <code># existant.</code>
Remarque importante : Ajouter l'option <code>-r</code> pour copier un répertoire et son contenu.	
<code>rm et rmdir (remove et remove directory)</code>	Supprimer un fichier ou répertoire
<code>\$ rm chemin</code>	<code># supprime la ressource désignée par chemin.</code>
Remarque importante : Ajouter l'option <code>-r</code> pour supprimer un répertoire et son contenu.	
<code>\$ rm -ri chemin_repertoire</code> <code>\$ rm -rf chemin_repertoire</code>	<code># idem en demandant une confirmation avant chaque effacement</code> <code># idem en ne demandant aucune confirmation (très très dangereux)</code>
Attention à la commande <code>rm -rf</code> ! Une erreur classique est de faire :	
<code>\$ rm -rf / home/julien_devillele/documents/archives/2013</code> Par erreur on a introduit un espace entre / et home/... . En conséquence on exécute : <code>\$ rm -rf /</code> C'est-à-dire qu'on efface sans confirmation tout le dossier racine (c-à-d tout le système de fichiers).	