

Commandes linux

CD : Change Directory (Commande qui permet de changer de dossier)

```
cd /
```

La commande du dessus me permet de me déplacer à la racine, car '/' représente le dossier racine.

La racine du pc c'est le dossier '/'.

Le premier paramètre de la commande cd c'est le chemin du dossier dans lequel on veut aller

2 types de chemins :

1. Chemin absolue : Démarre toujours de la racine '/' jusqu'à l'endroit où on veut aller
2. Chemin relatif : Démarre de là où on est

Exemple :

Admettons qu'on soit dans '/home/imed/Downloads' et qu'on veuille aller dans '/home/imed/Vidéos'

Via un chemin relatif on ferait :

```
cd ../Vidéos
```

Via un chemin absolue on ferait :

```
cd /home/imed/Vidéos
```

LS : List (Liste tous les éléments d'un dossier)

Ls, sans aucun paramètre, affiche les éléments du dossier dans lequel je me situe.

Par exemple si je suis dans /home/imed et que je fais ls voilà ce qui va se passer :

```
imed@imed-virtual-machine:~/Downloads$ cd ..
imed@imed-virtual-machine:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  Videos
```

Ca a listé tous les éléments du dossier dans lequel je suis.

Pour info => Ls peut prendre un paramètre, et la particularité ici c'est que contrairement à cd, ici le paramètre de ls n'est pas obligatoire.

LS peut prendre en paramètre un chemin (relatif ou absolue), et ça va me permettre d'afficher les éléments du dossier passé en paramètre.

Exemple : Pour lister les éléments de '/home/imed/Downloads'

```
imed@imed-virtual-machine:~/Downloads$ ls /home/imed/Downloads
Mojave-CT-Classic.tar.xz  Mojave-Dark.tar.xz
```

TOUCH : Commande qui crée un fichier (PAS UN DOSSIER)

touch (touche en français) est une commande qui permet de créer un fichier, et pas un dossier.

touch c'est la commande et touch peut prendre une infinité de paramètres, et ces paramètres représentent tout simplement les fichiers qu'on veut créer.

```
imed@imed-virtual-machine:~$ ls
Desktop Documents Downloads Music Pictures Public snap Templates Videos
imed@imed-virtual-machine:~$ touch fichier1 fichier2 fichier3
imed@imed-virtual-machine:~$ ls
Desktop Documents Downloads fichier1 fichier2 fichier3 Music Pictures Public snap Templates Videos
```

Ici je crée 3 fichiers fichier1 fichier2 fichier3.

MV : move => déplacer + renommer (commande 2 en 1)

```
imed@imed-virtual-machine:~$ ls
Desktop Documents Downloads fichier1 fichier2 fichier3 Music Pictures Public snap Templates Videos
imed@imed-virtual-machine:~$ mv fichier1 Documents/
imed@imed-virtual-machine:~$ ls Documents/
fichier1
imed@imed-virtual-machine:~$
```

Mv permet de déplacer un fichier un fichier

1. Le premier paramètre c'est la source (ce qu'on veut déplacer)
2. Le deuxième paramètre c'est la destination (où on veut le déplacer).

Mv permet aussi de renommer un fichier

Si je veux renommer fichier1 en toto :

```
imed@imed-virtual-machine:~/Documents$ ls
fichier1 fichier3 tata
imed@imed-virtual-machine:~/Documents$ mv fichier1 toto
imed@imed-virtual-machine:~/Documents$ ls
fichier3 tata toto
imed@imed-virtual-machine:~/Documents$
```

Dossier courant et dossier parent

Le dossier parent => c'est tout simplement le dossier avant le dossier où je suis

Le dossier courant => c'est tout simplement le dossier où je suis.

Si je suis dans /home/imed/Vidéos

Alors le dossier courant c'est : Vidéos

Et le dossier parent c'est : imed

Exemple : pour reculer d'un cran on fait cd .., car .. représente le dossier parent

```
imed@imed-virtual-machine:~/Videos$ ls
imed@imed-virtual-machine:~/Videos$ cd ..
imed@imed-virtual-machine:~$ ls
Desktop Documents Downloads Music Pictures Public snap Templates Videos
imed@imed-virtual-machine:~$
```

Et pour aller la où je suis déjà (le dossier courant), je fais cd . (ça n'a aucun intérêt) :

```
imed@imed-virtual-machine:~$ cd .
imed@imed-virtual-machine:~$ ls
Desktop Documents Downloads Music Pictures Public snap Templates Videos
imed@imed-virtual-machine:~$
```

Enfaite le '.' représente le chemin de là où je suis.

PWD : Print Working Directory (Afficher le dossier de travail/courant)

La commande pwd permet d'afficher le chemin absolu de là où je suis.

Admettons que je sois dans /home/imed/Vidéos, et que je fais un pwd, voilà ce qui va s'afficher :

```
imed@imed-virtual-machine:~/Downloads$ pwd
/home/imed/Downloads
imed@imed-virtual-machine:~/Downloads$
```