

## I. Afficher le contenu d'un répertoire : **ls**

### La commande : **ls [options]**

#### **ls**

Affiche la liste des fichiers et sous-répertoire du répertoire courant

#### **ls Rep1/toto**

Affiche la liste des fichiers et sous-répertoires du répertoire **Rep1/toto**

#### **ls -l**

Affiche le contenu du répertoire courant sous forme d'une liste détaillée (droits, propriétaire, taille, etc...).

#### **ls -a**

Affiche également les fichiers cachés.

#### **ls -t**

Affiche par ordre de date de dernière modification

#### **ls -lat**

**==> ls -l & ls -a & ls -t**

#### **ls -R**

Afficher récursivement le contenu des sous-répertoires.

#### **ls -r**

Affiche la liste des fichiers et sous-répertoire du répertoire courant , en inversant le tri du contenu.

.....

Voir la commande : **man ls**

.....

## II. Créer un répertoire: **mkdir Répertoire**

### **mkdir Rep1**

crée le répertoire Rep1 comme sous répertoire du répertoire courant.

### **mkdir -p Rep2/Rep3**

crée le répertoire **Rep2** comme sous répertoire du répertoire courant puis un sous répertoire **Rep3** dans **Rep2**.

## III. Parcourir l'arborescence des répertoires :

### 1. Où suis-je?

**pwd**

### 2. Changer de répertoire :

**cd Rep1**

**cd ..**

**cd Rep2/Rep3**

...

## IV. Manipulation de fichiers

### 1. Création de fichiers :

#### **touch f1**

Cette fonction permet de créer un fichier **f1** vide

#### **echo "Bonjour" > f2**

Créer le fichier **f2** et insérer la chaîne "**Bonjour**" dans **f2**

#### **echo "Coucou" > f2**

Écraser l'ancien fichier **f2** puis créer de nouveau **f2** et ...

#### **echo "comment ça va ?" >> f2**

Insérer la chaîne "comment ça va ?" à la fin du fichier **f2**

#### **cat > f3**

....

**texte**

....

**^C**

Cette fonction permet de créer le fichier **f3** et insérer un texte dans ce fichier, arrêter le saisi du texte en tapant sur les deux touches Contrôle et c

### 2. Afficher le contenu d'un fichier :

1. **cat nom\_du\_fichier**
2. **more nom\_du\_fichier**
3. ...

### 3. Effacer un fichier ou un répertoire : **rm**

#### **rm f1**

Supprimer le fichier **f1**

#### **rm Rep1/\***

Supprime tout les fichiers dans Rep1 (voir sens de \* ci-dessous)

#### **rm -i Rep1/\***

Même chose, avec demande de confirmation.

#### **rm -rf Rep1**

Supprime récursivement **Rep1** et ses sous répertoires.

#### **rmdir Rep1**

Supprimer le répertoire **Rep1** à condition que le répertoire **Rep1** soit vide.



**Attention : ces dernières opérations ne sont pas réversibles.**

**4. Déplacer, renommer un fichier, un répertoire : mv****mv fich1 fich2**Renommer **fich1** en **fich2**.**mv fich1 Rep/**Déplacer le fichier **fich1** dans le répertoire Rep en conservant le même nom.**mv fich1 ../fich2**Déplacer en le renommant le fichier **fich1** vers le répertoire parent.**5. Copier un fichier, un répertoire : cp****cp fich1 fich2**Copier le fichier **fich1** dans le fichier **fich2** du répertoire courant.**cp fich1 Rep1/fich2**Copier le fichier **fich1** du répertoire courant vers un sous-répertoire**Rep1** sous le nom **fich2**.**cp -r Rep1 Rep2/.**Copier toute la arborescence de **Rep1** dans **Rep2**.**V. Compression de fichier****1. gzip (gunzip) fich1**Compresser / Décompresser **fich1**, et le renomme **fich1.gz****2. bzip2 (bunzip2) fich1**Compresser / Décompresser **fich1**, et le renomme **fich1.bz2**

Un fichier compressé doit être décompressé avant de pouvoir être utilisé.



Le gain de taille peut varier de 0% à 90% de la taille initiale:

- Types de fichiers déjà compressés: .pdf, .jpeg, .mpeg. ...

- Types de fichiers non compressés: .ps, .tiff, texte, ...



Il est préférable de compresser avant d'envoyer en pièce attachée ou de transférer par le réseau!

**VI. Créer une archive**

Il est utile, en vue de le compresser et de le transférer, de transformer toute l'arborescence d'un répertoire en un seul fichier.

**tar [options] ..... (voir : man tar)****tar c(z)vf nom\_archive.tar(.z) répertoire\_à\_archiver****(tar cvf Rep1.tar Rep1)**Permet de créer une archive "**Rep1.tar**" à partir du répertoire source "**Rep1**"**tar xvf nom\_archive.tar**

Reconstituer le répertoire d'origine

**tar tvf nom\_archive.tar**

Lister le contenu du répertoire d'origine

.....

## VII. Le contrôle des processus

**Un processus est un programme en cours d'exécution.**

Il lui est attribué un PID (process ID), un nombre qui le caractérise de manière unique.

### 1. Afficher les informations sur les processus en cours d'exécution :

**ps -u**

Affiche la liste de tout les processus dont l'utilisateur est propriétaire, et des informations comme le PID, occupation en mémoire, etc ...

**top**

Liste en temps réel les processus sur l'ordinateur, et ressources utilisées.

### 2. Processus & signaux

Il est possible de modifier un processus en cours d'exécution en lui envoyant un signal

**Crtl-C**

Interrompre un processus qui est en mode interactif.

**kill -9 763287**

Tuer le processus de PID **763287**

**nice 19 -p 763287**

Fixe à 19 la priorité du processus de PID **763287** (19 = priorité la plus faible).

- [?] À chaque processus Unix est associée une valeur du paramètre système **nice**, entre -20 et 19. Par défaut, cette valeur est égale à 0. Plus cette valeur est faible, plus le processus est prioritaire pour accéder à la ressource cpu de la machine.

**at [-csm] heure [date] commande**

L'exécution à une date et une heure précise

Exécute la commande à une heure donnée, éventuellement en précisant la date. Si la date n'est pas précisée, elle est par défaut égale à la date du jour.

**at 1030am May 03 mon\_programme**

Lance le script "**mon\_programme**" à **10h30 le 3 Mai**

.....

### 3. Exécution en arrière plan : &

**commande &**

Lance le job "**commande**" en arrière plan et vous redonne la main tout de suite.

**firefox &**

Lancer l'application **firefox** en tâche de fond et vous redonne la main.

**fg**

Remettre le processus en avant plan

**Contrôle-Z**

Stopper le processus en cours d'exécution

**bg**

Remettre le processus en arrière plan

## VIII. Opérateurs de redirection

Rediriger la sortie standard (sortie écran) dans un fichier : >

**ls -l > liste**

Redirige le résultat d'exécution de **ls -l** vers le fichier **liste**.

**prog.out > resultats**

Redirige le résultat d'exécution de prog.out dans le fichier resultats, au lieu de l'afficher à l'écran.

**ls -l >> liste**

Insérer les résultats d'exécution de **ls -l** à la fin du fichier **liste**.

Idem pour :

**prog.out >> resultats**

.....

## | Redirection de la sortie standard vers l'entrée standard

**ls | grep fich1**

La sortie standard de la fonction **ls** devient l'entrée standard de la fonction **grep**