

Commandes Shell UNIX

•
Cours de première année de BTS SIO
Ecole IPSSI, 14 Avenue de l'Europe, Montévrain, France

Flavie Tonon
flavie.tonon.pro@gmail.com

11 novembre 2019

Table des matières

1	Système de Gestion de Fichiers	2
1.1	Où suis-je : pwd	2
1.2	Liste des fichiers : ls	2
1.3	Changer de répertoire : cd	2
1.4	Créer un nouveau répertoire : mkdir	3
1.5	Créer un nouveau fichier : touch	3
1.6	Copier un fichier : cp	4
1.7	Déplacer ou renommer un fichier : mv	4
1.8	Supprimer un fichier : rm	4
1.9	Supprimer un répertoire vide : rmdir	5
1.10	Chemin relatif et absolu	5
2	Quelques commandes utiles	6
2.1	Manuel : man	6
2.2	Historique	6
2.2.1	history	6
2.2.2	!	6
2.3	Visualiser un fichier	6
2.3.1	cat	6
2.3.2	more	6
2.3.3	less	6
2.3.4	head	7
2.3.5	tail	7

1 Système de Gestion de Fichiers

1.1 Où suis-je : pwd

Cette commande vous affiche le répertoire de travail, le répertoire courant. Elle signifie "Print Working Directory". Le chemin affiché est un chemin absolu car il débute à la racine /.

```
$ pwd
/Users/Flavie/
```

1.2 Liste des fichiers : ls

La commande ls affiche le contenu de votre répertoire courant.

```
$ ls
Downloads          VirtualBox VMs      AndroidStudioProjects
Dropbox            eclipse             Applications
GvR                eclipse-workspace   AwesomeProject
Library            Blizzard           Movies
Creative Cloud Files Music                Desktop
Pictures           Documents          Public
```

Vous pouvez aussi préciser le répertoire dont vous voulez afficher le contenu. Ici nous allons afficher la liste des fichiers et des répertoires présents à la racine /.

```
$ ls /
Applications      Volumes             home                private
Library           bin                 installer.failurerequests sbin
Network           cores               model               tmp
System            dev                 net                 usr
Users             etc                 opt                 var
```

Pour avoir plus de détails sur vos fichiers et répertoires, utilisez l'option -l

```
$ ls -l
total 1856
drwxr-xr-x  4 Flavie  staff   136 Oct 24 16:00 AndroidStudioProject
drwx-----  9 Flavie  staff   306 Oct  3 11:21 Applications
drwxr-xr-x  6 Flavie  staff   204 Oct 24 16:00 AwesomeProject
drwxr-x----  4 Flavie  staff   136 Aug 29 2017 Blizzard
drwxrwxr-x@  3 Flavie  staff   102 Nov 24 2017 Creative Cloud Files
drwx-----+ 13 Flavie  staff   442 Oct 29 21:22 Desktop
drwx-----+ 33 Flavie  staff  1122 Oct 24 16:12 Documents
drwx-----+ 117 Flavie  staff  3978 Oct 30 21:17 Downloads
drwx-----@ 15 Flavie  staff   510 Aug 21 21:54 Dropbox
drwxr-xr-x  8 Flavie  staff   272 Jan 21 2010 GvR
drwx-----@ 73 Flavie  staff  2482 Aug  6 00:21 Library
drwx-----+  9 Flavie  staff   306 Oct 24 16:23 Movies
drwx-----+ 12 Flavie  staff   408 Feb 28 2018 Music
drwx-----+  6 Flavie  staff   204 Nov 21 2017 Pictures
drwxr-xr-x+  6 Flavie  staff   204 Jan  9 2017 Public
drwxr-xr-x  4 Flavie  staff   136 Oct 27 22:55 VirtualBox VMs
drwxr-xr-x  3 Flavie  staff   102 Aug 22 2017 eclipse
drwxr-xr-x 12 Flavie  staff   408 Aug 22 2017 eclipse-workspace
```

Pour afficher aussi les fichiers cachés (tous ceux qui commencent par un point), utilisez l'option -a. Vous pouvez combiner les options "ls -a -l" ou "ls -al"

1.3 Changer de répertoire : cd

Pour changer de répertoire de travail utilisez la commande cd. Elle signifie "Change Directory".

```
$ pwd
/Users/Flavie/
$ cd Documents
$ pwd
/Users/Flavie/Documents/
```

Grâce à la commande `pwd` je peux voir que j'ai bien changé de répertoire de travail.

"Documents" est un chemin relatif. Il est possible d'accéder directement au répertoire Documents car il se situe dans notre répertoire de travail. Il est également possible d'utiliser la commande `cd` avec un chemin absolu, mais c'est un peu plus long à écrire.

```
$ pwd
/Users/Flavie/
$ cd /Users/Flavie/Documents
$ pwd
/Users/Flavie/Documents/
```

Pour revenir au répertoire parent, utilisez la commande `cd` suivie de `..`. Vous pouvez aussi utiliser le chemin absolu mais c'est plus long à écrire...

```
$ pwd
/Users/Flavie/
$ cd ..
$ pwd
/Users/
```

Pour revenir à votre répertoire personnel (votre homedir), utilisez la commande `cd` sans argument.

```
$ pwd
/Users/Flavie/Documents/Pro/2019–2020/BTSSIO/
$ cd
$ pwd
/Users/Flavie
```

1.4 Créer un nouveau répertoire : `mkdir`

Pour créer un répertoire dans votre répertoire de travail utiliser la commande `mkdir` signifiant Make Directory.

```
$ ls
cours-programmation.pdf
$ mkdir MonDossier
$ ls
cours-programmation.pdf  MonDossier
```

1.5 Créer un nouveau fichier : `touch`

Il n'existe pas de commande dédiée à la création d'un fichier. Cependant, il est possible d'obtenir ce résultat en détournant certaines commandes de leur rôle initial.

```
$ touch monFichier
$ ls
monFichier
```

La commande `touch` permet de modifier l'horodatage d'un fichier, mais utilisé sur un fichier inexistant, la commande `touch` va créer ce fichier. C'est cette fonctionnalité de `touch` que nous utiliserons.

1.6 Copier un fichier : cp

Pour dupliquer un fichier sur votre disque, utilisez la commande cp suivi du chemin vers votre fichier puis du chemin de destination.

```
$ ls
cours-programmation.pdf
$ cp cours-programmation.pdf cours.pdf
$ ls
cours-programmation.pdf  cours.pdf
```

Ce qui est intéressant avec cette commande c'est aussi qu'elle peut copier votre fichier dans un autre répertoire, il suffit de donner le chemin vers le bon répertoire.

```
$ ls
cours-programmation.pdf      MonRepertoire
$ cp cours-programmation.pdf MonRepertoire
$ ls MonRepertoire
cours-programmation.pdf
```

1.7 Déplacer ou renommer un fichier : mv

Pour déplacer un fichier sur votre disque, utilisez la commande mv suivi du chemin vers votre fichier puis du chemin de destination.

```
$ ls
cours.pdf
$ mv cours.pdf S14/cours.pdf
$ ls
$ ls S14
cours.pdf
```

Cette commande peut également renommer un fichier si vous effectuez le déplacement dans le même répertoire que le répertoire du fichier.

```
$ ls
cours.pdf
$ mv cours.pdf cours-unix.pdf
$ ls
cours-unix.pdf
```

1.8 Supprimer un fichier : rm

Pour supprimer définitivement un fichier utiliser la commande rm. Elle signifie Remove. Si vous voulez supprimer un répertoire utilisez l'option -r. Attention à ne pas supprimer le mauvais fichier ou répertoire !

```
$ ls
cours-programmation.pdf  cours.pdf  MonDossier
$ rm cours.pdf
$ ls
cours-programmation.pdf  MonDossier
$ rm -r MonDossier
$ ls
cours-programmation.pdf
```

1.9 Supprimer un répertoire vide : rmdir

Plus rassurant que `rm` si vous souhaitez supprimer un répertoire uniquement si il est vide (pas de perte de données...), utiliser la commande `rmdir`. Elle signifie Remove Directory.

```
$ ls
cours-programmation.pdf  cours.pdf  MonDossier
$ ls MonDossier
$ rmdir MonDossier
$ ls
cours-programmation.pdf  cours.pdf
```

1.10 Chemin relatif et absolu

Nous en avons vu quelques uns dans de ce cours. Le chemin absolu démarre à la racine, son chemin est `"/`". Le chemin relatif dépend de votre répertoire de travail actuel.

2 Quelques commandes utiles

2.1 Manuel : man

Pour obtenir plus d'informations sur une commande, consultez son manuel. Tous les manuels sont accessibles avec la commande man.

```
$ man ls
```

Cette commande ouvre le manuel de la commande ls. Vous pourrez y découvrir les nombreuses options qui permettent de modifier le comportement de la commande ainsi que des explications détaillées sur son fonctionnement. Vous pouvez vous déplacer page par page dans le manuel avec la touche espace. Pour le quitter, tapez "q".

2.2 Historique

2.2.1 history

Pour visualiser les commandes que vous avez effectuées, utilisez la commande history.

```
$ history
```

S'affichera alors la liste de vos commandes numérotées.

2.2.2 !

Pour exécuter une commande de l'historique, il vous suffit d'écrire son numéro, précédé du point d'exclamation "!".

```
$ !201
```

2.3 Visualiser un fichier

Il existe plusieurs commandes pour afficher le contenu d'un fichier dans le terminal. Elles demandent toutes de donner en argument le chemin vers le fichier à afficher. répertoire

2.3.1 cat

La commande cat permet d'afficher tout le fichier d'un bloc sur le terminal.

```
$ cat monfichier
```

2.3.2 more

La commande more permet de visualiser le contenu d'un fichier dans le terminal avec une navigation page par page, plus pratique pour les longs fichiers.

```
$ more monfichier
```

2.3.3 less

La commande less permet de visualiser le contenu d'un fichier dans le terminal avec une navigation page par page plus perfectionnée que more (voir le manuel).

```
$ less monfichier
```

2.3.4 head

La commande `head` permet d'afficher les 10 premières lignes d'un fichier sur le terminal. Avec l'option `-n` il est possible de spécifier le nombre de lignes.

```
$ head -n 4 monFichier
```

Cette commande affiche les 4 premières lignes du fichier `monFichier`.

2.3.5 tail

La commande `tail` permet d'afficher les 10 dernières lignes d'un fichier sur le terminal. Avec l'option `-n` il est possible de spécifier le nombre de lignes.

```
$ tail -n 4 monFichier
```

Cette commande affiche les 4 dernières lignes du fichier `monFichier`.