

Découverte de l'interpréteur de commandes

Un terminal pour nos premiers pas

Identifiez-vous auprès du système avec les login et mot de passe qui vous ont été fournis.

Vous accédez à un environnement graphique. Les manipulations proposées ici seront réalisées via l'interpréteur de commandes, dans un terminal.

Un des terminaux installés sur les machines du fil s'appelle tilix, vous le trouverez dans la barre d'icônes ou dans le menu des applications.

Ouvrez un terminal.

Exécutions pas à pas

Les commandes qui suivent sont donc à saisir dans le terminal.

Validez ensuite en appuyant sur Entrée et observez l'action effectuée.

Lorsque vous entrez des commandes dans un terminal, celles-ci s'exécutent dans le répertoire courant, par défaut la racine de votre répertoire personnel.

1. Le contenu d'un répertoire peut être obtenu au moyen de la commande `ls`. Saisissez cette commande, le terminal affiche les fichiers présents dans le répertoire courant.
2. Nous allons maintenant créer un nouveau répertoire : utilisez la commande `mkdir` (make directory) suivie du nom du répertoire. Ici `mkdir bloc3`. Observez le résultat par une nouvelle exécution de `ls`.
3. Déplaçons nous dans le répertoire bloc3 nouvellement créé : `cd bloc3` (change directory). Constatez que l'invite de l'interpréteur a été modifiée, indiquant que nous sommes maintenant dans le répertoire `bloc3`.
4. Utilisez la commande `pwd` pour afficher le répertoire courant.
5. Pour les besoins du TP,
 - créons un nouveau répertoire : `mkdir tim`
 - plaçons nous dans ce répertoire : `cd tim`
 - affichons le répertoire courant : `pwd`
6. Nous allons créer un fichier dans le répertoire `tim` :
 - La commande `touch oleon` crée un fichier (vide) appelé `oleon`,
 - Vérifions à l'aide de la commande `ls` que le fichier a bien été créé.
7. Remontons dans l'arborescence :
 - `cd ..`
 - Vérifions que le répertoire courant a bien été modifié : `pwd`
 - Créons un deuxième répertoire : `mkdir raymond`
8. Pour copier un fichier :
 - on utilise la commande `cp` : `cp tim/oleon raymond/`
 - Cette commande a pour effet de copier le fichier `oleon` dans le répertoire `raymond/`

- Constatez son effet : `ls raymond/` qui affiche le contenu du répertoire `raymond`
9. Nous allons maintenant tenter de supprimer le répertoire `tim` :
- `rmdir tim` (remove directory) vous constatez que la commande produit une erreur puisque le répertoire n'est pas vide.
 - Il est possible de supprimer un répertoire non vide, mais pour l'instant allons effacer le fichier `oleon` qui n'est de toute façon pas d'une grande utilité : `cd tim` puis `rm oleon` (remove).
10. Tentons à nouveau la suppression du répertoire :
- `cd ..`
 - `rmdir tim`

Vous constatez que la commande ne signale plus d'erreur et que le répertoire a été supprimé.

11. Nous pouvons également déplacer des fichiers :
- `mv raymond/oleon .`

Cette commande déplace (move) le fichier `oleon` de `raymond` dans le répertoire courant `./` (ici `bloc3`).

12. La même commande `mv` permet aussi de renommer un fichier. Essayez `mv oleon arthur`. Observez.
13. Le répertoire `raymond` ne contient plus rien :
- `ls raymond/`
 - Nous pouvons le supprimer : `rmdir raymond`
14. Saisissons les commandes suivantes :
- `cp arthur Arthur`
 - `ls`

Observez que le répertoire contient maintenant deux fichiers *différents* dont les noms sont `arthur` et `Arthur` : Le système de fichier est donc *sensible à la casse*.

15. Saisissons :
- `cp arthur .arthur`
 - `ls`

Observez que le contenu du répertoire ne semble pas avoir été modifié. Pourtant le fichier `.arthur` a bien été créé :

- `ls .arthur`

Il s'agit d'un *fichier caché*. La convention adoptée est que tous les fichiers dont le nom commence par un point `.` sont traités comme des fichiers cachés.

16. Pour visualiser *tous* les fichiers d'un répertoire (y compris les cachés) :
- `ls -a`
 - ou `ls -al`

Observez la différence entre les deux commandes. Testez `ls` avec l'option `-l` seule. En déduire l'effet de l'option `-l`.

17. Observons que notre répertoire contient une entrée nommée `.` qui représente le répertoire courant et une entrée nommée `..` qui correspond au répertoire supérieur. On retrouve ces deux entrées dans tous les répertoires.
18. Nettoyons maintenant notre répertoire :
 - `rm arthur Arthur .arthur` (la commande `rm` accepte plusieurs arguments)
 - `cd ..`
 - `rmdir bloc3`

Tous les fichiers sont identifiables par leur nom et leur emplacement dans la hiérarchie. On doit pour cela fournir la liste des répertoires qu'il faut traverser avant de parvenir dans le répertoire contenant le fichier. On appelle cette information le chemin (path en anglais) du fichier. Chaque répertoire d'un chemin est séparé du répertoire suivant par le caractère «/». Il existe deux manières de donner un chemin :

- en spécifiant la liste la plus courte des répertoires à traverser *depuis la racine de la hiérarchie*, on dit alors que c'est un **chemin absolu** ;
- en spécifiant une liste des répertoires à traverser à partir d'un répertoire particulier de la hiérarchie, on dit dans ce cas que c'est un **chemin relatif** à ce répertoire de départ.