

## Atelier avancé n°1 : Découverte du terminal UNIX

Lorsque l'on développe en Python dans le monde pro on utilise constamment le terminal.

### Définition du terminal :

Sur les ordinateurs modernes, le mot "terminal" désigne généralement un programme de terminal, ou un émulateur, qui fournit **une interface textuelle pour la saisie de commandes**. Ce type de programme est souvent abrégé "TTY" et peut également être appelé interface de ligne de commande. Les programmes de terminal sont disponibles pour toutes les principales plates-formes informatiques et sont généralement inclus avec le système d'exploitation. Par exemple, Windows inclut le programme "cmd.exe", interface de ligne de commande permettant d'exécuter des commandes DOS et de se connecter à d'autres serveurs. Mac OS X inclut un programme appelé "Terminal", qui peut être utilisé pour exécuter des commandes Unix sous Mac OS ou pour accéder à d'autres ordinateurs. Unix inclut généralement un programme appelé "xterm", qui peut exécuter BASH ou d'autres shells Unix.

Pour en savoir plus : <https://techlib.fr/definition/terminal.html>

### Les commandes UNIX :

Les 2 terminaux les plus utilisés au monde sont respectivement : Le terminal de Linux et le terminal de MAC OS X qui utilise les **commandes Unix**.

En tant que développeur, il est impératif de connaître les commandes Unix les plus utilisées :

- pwd
- cd
- ls
- cat
- cp
- mv
- mkdir
- rm
- clear
- sudo

Pour découvrir ces commandes consultez ce lien :

<https://www.hostinger.fr/tutoriels/commandes-linux>

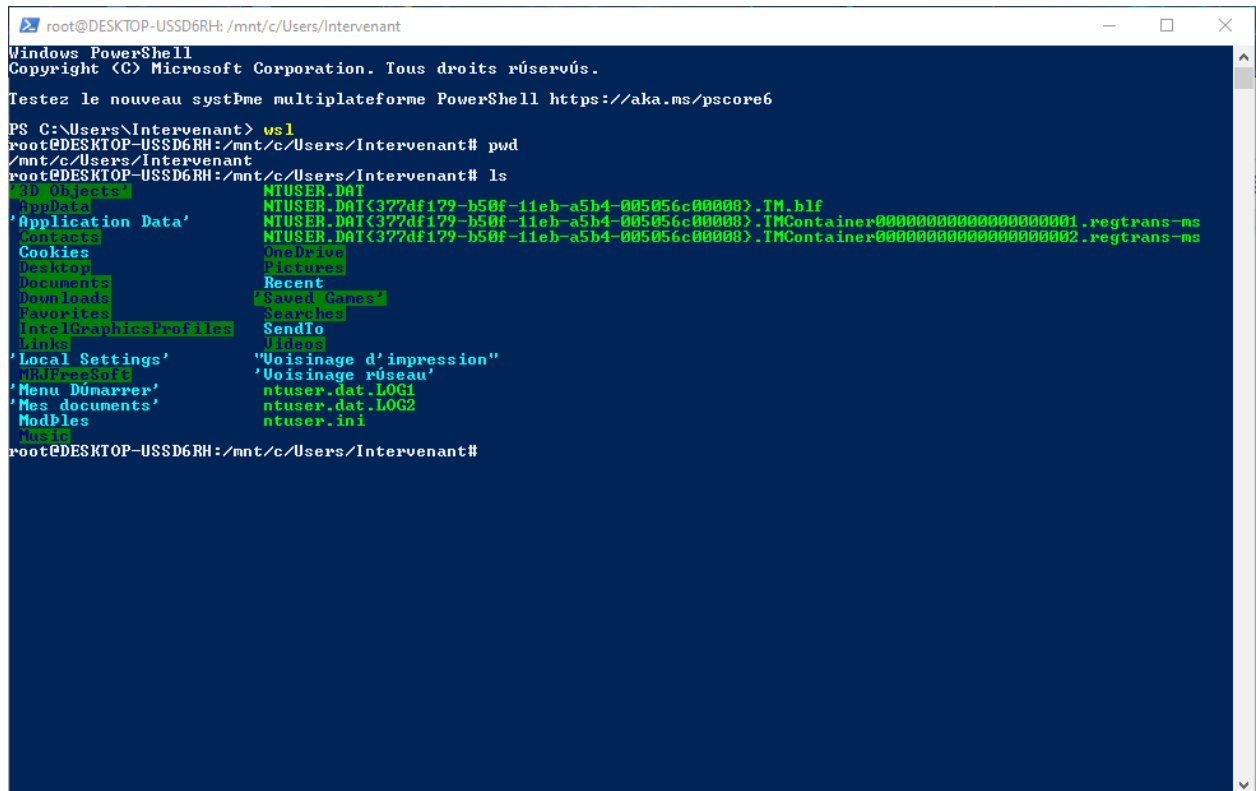
Pour découvrir comment utiliser ces commandes dans un terminal :

[https://doc.fedora-fr.org/wiki/Debuter\\_avec\\_le\\_terminal](https://doc.fedora-fr.org/wiki/Debuter_avec_le_terminal)

## Travail a réaliser :

Sur Windows 10 il n'existe pas de terminal UNIX préinstallé.

- 1) Commencer par installer un sous-système ubuntu sur votre pc ce qui vous permettra d'utiliser les commandes UNIX, en consultant ce lien :  
<https://blog.ineat-group.com/2020/02/utiliser-le-terminal-bash-natif-dans-windows-10/>
- 2) Démarrer le terminal de Windows (Windows PowerShell) utiliser la commande **ws** puis la commande **pwd** pour visualiser ou vous êtes :



```
root@DESKTOP-USSD6RH: /mnt/c/Users/Intervenant
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\Intervenant> ws
root@DESKTOP-USSD6RH:/mnt/c/Users/Intervenant# pwd
/mnt/c/Users/Intervenant
root@DESKTOP-USSD6RH:/mnt/c/Users/Intervenant# ls
Application Data  NTUSER.DAT  Recent
Cookies           NTUSER.DAT  Recent
Desktop           NTUSER.DAT  Recent
Documents         NTUSER.DAT  Recent
Downloads         NTUSER.DAT  Recent
Favorites         NTUSER.DAT  Recent
IntelGraphicsPro  NTUSER.DAT  Recent
Images           NTUSER.DAT  Recent
Local Settings   NTUSER.DAT  Recent
Menu Démarrer    NTUSER.DAT  Recent
Mes documents     NTUSER.DAT  Recent
Modèles          NTUSER.DAT  Recent
root@DESKTOP-USSD6RH:/mnt/c/Users/Intervenant#
```

- 3) Utiliser la commande **apt-get update** pour mettre à jour ubuntu
- 4) Utiliser la commande **cd** pour vous déplacer vers le Bureau (Desktop)
- 5) Utiliser la commande **ls** pour afficher les éléments présent sur votre Bureau dans le terminal.
- 6) Utiliser la commande **clear** pour rafraîchir l'affichage dans le terminal.
- 7) Utiliser la commande **mkdir** pour créer un dossier **tdpython1** sur votre Bureau puis déplacez vous dans ce dossier (essayer de saisir **cd tdp** puis cliquer sur **tab**).
- 8) Une fois dans ce dossier, utiliser la commande **pwd** pour vérifier qu'il est vide.
- 9) Utiliser la commande suivante pour installer l'environnement virtuel de python3 : **apt install python3.8-venv**.
- 10) Créer un environnement virtuel Python en utilisant la commande **python3 -m venv le\_nom\_de\_votre\_env** (pour en savoir plus :  
<https://docs.python.org/fr/3/tutorial/venv.html>)

- 11) Activer l'environnement virtuel Python en utilisant la commande suivante : **source nom\_de\_votre\_env/bin/activate**
- 12) Par la suite créer un fichier python test.py en utilisant la commande **nano** (pour quitter nano utiliser Ctrl + clic)
- 13) Créer un programme permettant d'afficher le message "Hello World"
- 14) Utiliser la commande **ls** pour vérifier que le fichier a bien été créé
- 15) Exécuter le fichier Python en utilisant la commande suivante **py test.py**
- 16) Créer un nouveau Projet sur Pycharm/Visual studio code et lier votre espace de travail avec le dossier tdpython1 et l'environnement virtuel de python présent dans ce dossier.

Exercice à réaliser dans le dossier pythontd1 :

- 1) L'utilisateur donne un entier  $n$  entre 2 et 12, le programme donne le nombre de façons de faire  $n$  en lançant deux dés.
- 2) Même problème que le précédent, mais avec  $n$  entre 3 et 18 et trois dés.
- 3) Généralisation des deux questions précédentes. L'utilisateur saisit deux entrées, d'une part le nombre de dés,  $nbd$  (que l'on limitera pratiquement à 10) et, d'autre part la somme,  $S$ , comprise entre  $nbd$  et  $6.nbd$ . Le programme calcule et affiche le nombre de façons de faire  $S$  avec les  $nbd$  dés.