Commandes pour Linux :

1. Commandes de base :

- `ping [adresse IP/hostname]`

- `traceroute [adresse IP/hostname]`

- `ifconfig`

- `arp -a`

- `nslookup [adresse IP/hostname]`

2. Commandes TCP/IP :

- `netstat -a`

- `telnet [adresse IP/hostname] [port]`

- `nmap [adresse IP]`

3. Configuration réseau

- `ifconfig [interface] up/down`

- `ip addr show`

- `dhclient [interface]` (pour obtenir une adresse IP via DHCP)

4. Gestion des fichiers :

- `ls`

- `cp [source] [destination]`

- `mv [source] [destination]`

5. Gestion des processus :

- `ps`

- `kill [PID]`

- `top` (affiche les processus en cours d'exécution en temps réel)

6. Gestion des utilisateurs :

- `useradd` et `userdel`

- `passwd [utilisateur]`

- `sudo` (pour exécuter des commandes avec des privilèges administratifs)

7. Gestion des permissions :

- `chmod`

- `chown` (change le propriétaire d'un fichier)

Commandes pour Windows :

1. Commandes de base :

- `ping [adresse IP/hostname]`

- `tracert [adresse IP/hostname]`

- `ipconfig`

- `arp -a`

- `nslookup [adresse IP/hostname]`

2. Commandes TCP/IP :

- `netstat -a`

- `telnet [adresse IP/hostname] [port]`

- `nmap` (peut être installé, mais moins courant)

3. Configuration réseau :

- `ipconfig /release` et `ipconfig /renew`

- `netsh interface show interface`

- `netsh interface ipv4 set address [interface] static [IP] [Subnet] [Gateway]`

4. Gestion des fichiers :

- `dir`

- `copy [source] [destination]`

- `move [source] [destination]`

5. Gestion des processus :

- `tasklist` (liste les processus)

- `taskkill /F /PID [PID]` (force la fin d'un processus)

6. Gestion des utilisateurs :

- `net user [username] [password] /add`

- `net user [username] /delete`

- `runas` (exécute une commande avec des privilèges élevés)

7. Gestion des permissions :

- `icacls` (pour gérer les contrôles d'accès)

Commandes pour Git Bash (Linux/Windows) :

1. Configuration Git :

- `git config --global user.name "Votre Nom"`

- `git config --global user.email "votre@email.com"`

- `git config --list` (affiche la configuration Git)

2. Initialisation et clonage de dépôts :

- `git init` (initialise un nouveau dépôt Git)

- `git clone [URL du dépôt]` (clone un dépôt existant)

3. Commandes de base :

- `git status` (affiche l'état des fichiers dans le dépôt)

- `git add [fichier]` (ajoute un fichier à l'index)

- `git commit -m "Message du commit"` (enregistre les modifications avec un message)

- `git pull` (récupère les modifications depuis le dépôt distant)

- `git push` (envoie les modifications locales vers le dépôt distant)

4. Gestion des branches :

- `git branch` (affiche les branches)

- `git checkout [nom de la branche]` (change de branche)

- `git merge [nom de la branche]` (fusionne une branche dans la branche active)

- `git branch -d [nom de la branche]` (supprime une branche)

5. Gestion des conflits :

- `git diff` (affiche les différences entre les modifications)

- `git mergetool` (outil de fusion graphique pour résoudre les conflits)

- `git reset --hard` (annule toutes les modifications locales non validées)

6. Historique et informations :

- `git log` (affiche l'historique des commits)

- `git show [commit]` (affiche les détails d'un commit spécifique)

- `git blame [fichier]` (montre qui a modifié chaque ligne d'un fichier)

7. Remotes :

- `git remote add origin [URL du dépôt distant]` (ajoute un dépôt distant)

- `git remote -v` (affiche les dépôts distants associés)

BDD :

1. Création de tables :

CREATE TABLE nom\_table (

colonne1 type1,

colonne2 type2,

...

);

2. Insertion de données :

INSERT INTO nom\_table (colonne1, colonne2, ...) VALUES (valeur1, valeur2, ...);

3. Sélection de données :

SELECT colonne1, colonne2, ... FROM nom\_table WHERE condition;

4. Mise à jour de données :

UPDATE nom\_table SET colonne1 = valeur1 WHERE condition;

5. Suppression de données :

DELETE FROM nom\_table WHERE condition;

6. Filtrage de données distinctes :

SELECT DISTINCT colonne FROM nom\_table;

7. Filtrage de résultats avec des conditions :

SELECT \* FROM nom\_table WHERE colonne > valeur;

8. Regroupement de données :

SELECT colonne, COUNT(\*) FROM nom\_table GROUP BY colonne;

9. Jointure de tables :

SELECT \* FROM table1 INNER JOIN table2 ON table1.colonne = table2.colonne;

10. Tri des résultats :

SELECT \* FROM nom\_table ORDER BY colonne ASC|DESC;

11. Création d'index :

CREATE INDEX index\_nom ON nom\_table (colonne);

12. Suppression de tables :

DROP TABLE nom\_table;

test