



Rapport de Travaux Pratiques

**LST : Ingénierie de données et développement
logiciel(IDDL)**

TP 02: Réseaux et Système

Réalisé par :

-Nom et prénom : EL MAHRAOUI AMAL

Encadré par : Pr. EL MAMOUN ABDELLAH

Année Universitaire : 2023/2024

Table des matières

I. Introduction :	3
II. Objectifs :	3
III. Déroulement du TP	3
IV. Manipulation: Manipulation des commandes de gestion de l'arborescence des fichiers	3
Exercice 1 :	4

TP N° 2

Introduction à l'arborescence des fichiers

I. Introduction :

Dans le cadre du TP N° 2, nous avons abordé le concept fondamental de l'arborescence des fichiers. L'arborescence des fichiers est une structure hiérarchique utilisée pour organiser et gérer les fichiers et dossiers sur un système informatique.

II. Objectifs :

L'objectif principal de ce TP était de comprendre les concepts de base liés à l'arborescence des fichiers, notamment :

- La signification de l'arborescence des fichiers.
- La hiérarchie des dossiers et fichiers.
- Comment naviguer dans l'arborescence des fichiers.
- Comment créer, copier, déplacer et supprimer des fichiers et des dossiers.

III. Déroulement du TP

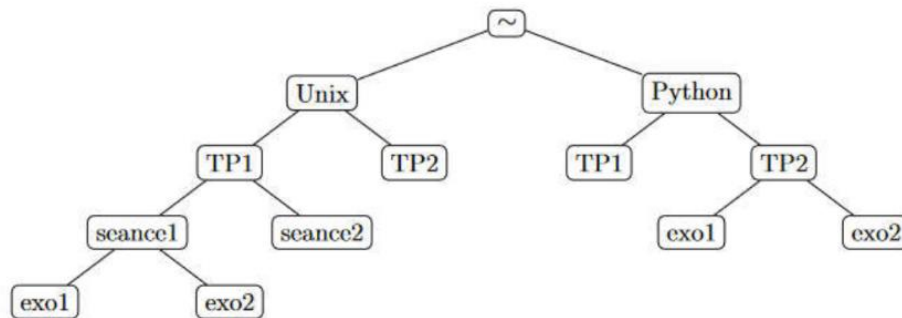
Au cours de la séance de TP, nous avons effectué les activités suivantes :

- Introduction théorique sur l'arborescence des fichiers et son importance dans la gestion des données.
- Démonstration de la structure de l'arborescence des fichiers sur un système d'exploitation (Linux).
- Exercices pratiques de navigation dans l'arborescence des fichiers.
- Création de dossiers et fichiers, copie et déplacement de fichiers, et suppression de fichiers et dossiers.
- Utilisation de commandes spécifiques selon le système d'exploitation (commandes Terminal pour Linux).

IV. Manipulation: Manipulation des commandes de gestion de l'arborescence des fichiers.

Exercice 1 :

1. Créer dans votre répertoire personnel les fichiers selon l'arborescence suivante.

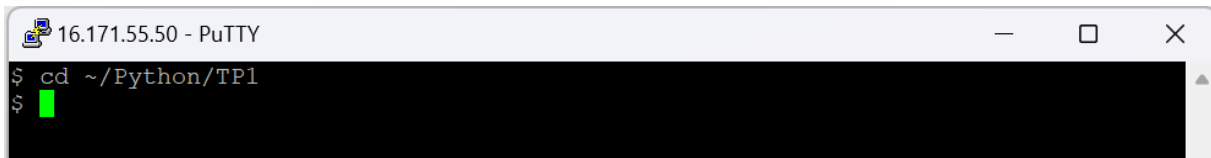


```
16.171.55.50 - PuTTY
$ mkdir ~/unix
$ mkdir ~/Python
mkdir: cannot create directory '/home/elmahraoui/Python': File exists
$ mkdir ~/Unix/TP1
mkdir: cannot create directory '/home/elmahraoui/Unix/TP1': File exists
$ mkdir ~/Unix/TP2
mkdir: cannot create directory '/home/elmahraoui/Unix/TP2': File exists
$ cp -r ~/Unix/TP1 ~/Python
$ mki ~/Unix/TP1/seance1
-sh: 13: mki: not found
$ mkdir ~/Unix/TP1/seance1
$ mkdir ~/Unix/TP2/seance2
$ touch ~/Unix/TP1/seance1/exo1
$ touch ~/Unix/TP1/seance1/exo2
$ cp ~/Unix/TP1/seance1/exo1 ~/Python/TP2
$ cp ~/Unix/TP1/seance1/exo2 ~/Python/TP2
$
```

2. Placez-vous dans le répertoire Python/TP1. Vérifiez que vous y êtes, puis, en une seule commande, placez-vous dans le répertoire Unix/TP2 en utilisant le chemin relatif du répertoire.

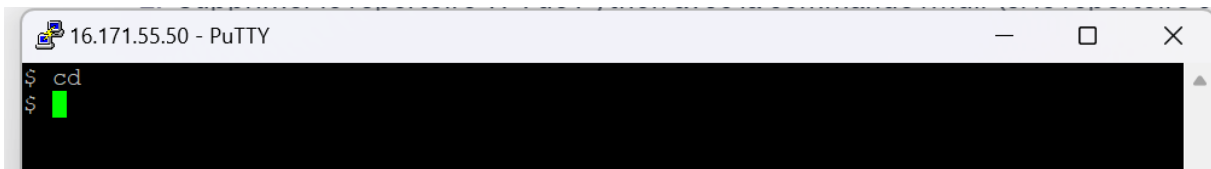
```
16.171.55.50 - PuTTY
$ cd ~/Python/TP1
$ cd ../../Unix/TP2
$
```

Pour revenir dans Python/TP1 en utilisant le chemin absolu :



```
16.171.55.50 - PuTTY
$ cd ~/Python/TP1
$
```

Pour revenir à la racine de votre répertoire personnel :



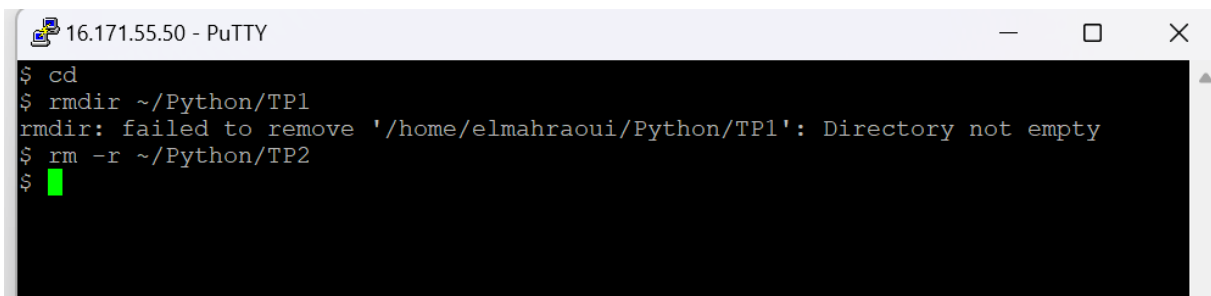
```
16.171.55.50 - PuTTY
$ cd
$
```

3. Supprimer le répertoire TP1 de Python avec la commande `rmdir`.

(elle échoue car le répertoire n'est pas vide)

4. Supprimer le répertoire TP2 de Python avec la commande `rmdir`, pourquoi la commande a échoué. Supprimer en une commande le répertoire TP2 de Python.

elle échoue car le répertoire n'est pas vide). Pour supprimer le répertoire TP2 de Python en une commande (y compris son contenu), en utilise `rm -r` :



```
16.171.55.50 - PuTTY
$ cd
$ rmdir ~/Python/TP1
rmdir: failed to remove '/home/elmahraoui/Python/TP1': Directory not empty
$ rm -r ~/Python/TP2
$
```

5. Déplacer le fichier `exo1` dans le répertoire TP1 et supprimer les répertoires `seance1` et `seance2`.

N.B : J'ai déjà déplacé le fichier `exo1` dans le répertoire TP1 et supprimer le répertoire séance 1, mais j'ai pas pu a prendre des captures d'écrans, c'est pour cela j'ai écrit les commandes à nouveau.

```
16.171.55.50 - PuTTY
$ mv ~/Unix/TP1/seance1/exo1 ~/Unix/TP1
mv: cannot stat '/home/elmahraoui/Unix/TP1/seance1/exo1': No such file or directory
$ rm -r ~/Unix/TP1/seance1
rm: cannot remove '/home/elmahraoui/Unix/TP1/seance1': No such file or directory
$ rm -r ~/Unix/TP1/seance2
$
```

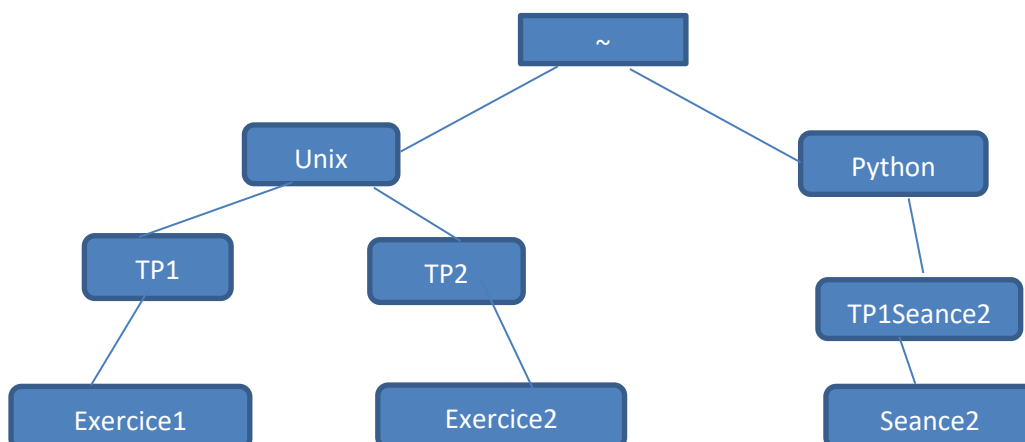
6. Renommer les fichiers `exo1` et `exo2` en respectivement `exercice1` et `exercice2`.

N.B : J'ai déjà renommé le fichier `exo1` et `exo2` en respectivement `exercice1` et `exercice2`, mais j'ai pas pu à prendre des captures d'écrans, c'est pour cela j'ai écrit les commandes à nouveau.

```
16.171.55.50 - PuTTY
$ mv ~/Unix/TP1/exo1 ~/Unix/TP1/exercice1
mv: cannot stat '/home/elmahraoui/Unix/TP1/exo1': No such file or directory
$ mv ~/Unix/TP1/exo2 ~/Unix/TP1/exercice2
mv: cannot stat '/home/elmahraoui/Unix/TP1/exo2': No such file or directory
$
```

7. Dessiner l'arborescence des fichiers contenus dans le répertoire personnel de manière théorique.

Vérifier le à l'aide d'une commande.



Pour dessiner l'arborescence des fichiers contenus dans le répertoire personnel de manière théorique, on utilise la commande "**tree ~/**" aussi :

```
$ tree ~/
/home/elmahraoui/
├── Python
│   └── TP1
│       └── seance2
├── Unix
│   ├── TP1
│   │   └── exercicel
│   ├── TP2
│   │   └── seance2
└── unix
```

8 directories, 1 file

On peut vérifier ça avec la commande : "**ls -R ~/**"

```
16.171.55.50 - PuTTY
$ ls -R ~/
/home/elmahraoui/:
Python  Unix  unix

/home/elmahraoui/Python:
TP1

/home/elmahraoui/Python/TP1:
seance2

/home/elmahraoui/Python/TP1/seance2:

/home/elmahraoui/Unix:
TP1  TP2

/home/elmahraoui/Unix/TP1:
exercicel

/home/elmahraoui/Unix/TP2:
seance2

/home/elmahraoui/Unix/TP2/seance2:

/home/elmahraoui/unix:
$
```