

## R1.08 – TP 2 : La commande FIND

### 1. Création d'un fichier vide ~/toto

Commande utilisée :

```
touch ~/toto
```

Cette commande permet de créer un fichier vide nommé "toto" dans le répertoire personnel.

### 2. Trouver tous les fichiers dans /usr dont le nom commence par "std" et se termine par ".h"

Commande utilisée :

```
find /usr -type f -name 'std*.h' -exec basename {} \;
```

-type f : cherche uniquement des fichiers.

name 'std\*.h' : cherche les fichiers commençant par "std" et finissant par ".h".

basename {} : affiche uniquement le nom du fichier (sans le chemin complet).

### 3. Afficher les informations détaillées sur les fichiers trouvés à la question précédente

Commande utilisée :

```
find /usr -type f -name 'std*.h' -exec ls -l {} \;
```

On utilise ici ls -l pour obtenir les droits, propriétaires, tailles et dates de modification des fichiers.

### 4. Copier les fichiers trouvés dans /tmp/tp\_intro\_sys

Commandes utilisées :

```
mkdir -p /tmp/tp_intro_sys
```

```
find /usr -type f -name 'std*.h' -exec cp {} /tmp/tp_intro_sys/ \;
```

```
ls -l /tmp/tp_intro_sys
```

mkdir -p : crée le dossier destination (et ses parents si besoin).

cp : copie chaque fichier trouvé dans le dossier.

On vérifie avec ls -l que les fichiers ont bien été copiés.

### 5. Afficher les infos détaillées sur tous les répertoires dans l'arborescence du home

Commande utilisée :

```
find $HOME -type d -exec ls -ld {} \;
```

- type d : cherche les répertoires.

ls -ld : affiche les détails du dossier sans lister son contenu.

## 6. Trouver tous les fichiers de taille 0 octet créés aujourd'hui

Commande utilisée :

```
find $HOME -type f -size 0 -daystart -ctime 0
```

Explication :

-size 0` : taille 0 octet.

-ctime 0` : créés ou modifiés aujourd'hui.

-daystart` : le calcul commence à minuit (pas 24h en arrière).

## 7. Créer les fichiers ~/titi et ~/tata

Commandes utilisées :

```
touch ~/titi
```

```
touch ~/tata
```

On crée deux fichiers vides dans le répertoire personnel, utiles pour tester les commandes de comparaison de dates.

## 8. Trouver les fichiers dans \$HOME qui sont antérieurs à toto

Commande utilisée :

```
find $HOME -type f ! -newer ~/toto
```

! -newer : sélectionne les fichiers \*\*plus anciens\*\* que `toto` (pas modifiés après lui).

## 9. Trouver les fichiers modifiés hier et de taille > 1 Kio

Commande utilisée :

```
find $HOME -type f -size +1024c -mtime 1
```

-size +1024c : fichiers de plus de 1024 octets (1 Kio).

-mtime 1 : fichiers modifiés entre 24h et 48h avant maintenant, donc "hier".

## 10. Différence entre :

```
ls -l ~/*.txt
```

et

```
find ~ -name "*.txt" -exec ls -l {} \;
```

ls -l ~/\*.txt : liste les fichiers `.txt` \*\*dans le répertoire personnel uniquement\*\*, pas dans les sous-dossiers.

find ~ -name "\*.txt" -exec ls -l {} \; : cherche récursivement \*\*dans tous les sous-dossiers de\*\* les fichiers `.txt` et affiche leurs détails. Plus complet.

## 11. Chercher tous les fichiers simples exécutables par l'utilisateur

Commande utilisée :

```
find $HOME -type f -perm -u=x -exec ls -l {} \;
```

-perm -u=x : cherche les fichiers ayant le droit d'exécution pour l'utilisateur.

-type f : uniquement les fichiers simples.

ls -l : pour afficher les infos détaillées.