

TP 1.2 - Nicolas PIPLARD - 07/09/2020

I .MANIPULATION DES FICHIERS ET REPERTOIRES

Exercice 1:

La racine du système contient tous les répertoires principaux.

Nous avons :

Dossier	Description
bin	Exécutables essentiels au système, utilisables par tous les utilisateurs (ls pwd cp)
boot	Ensemble des fichiers permettant à Linux de démarrer
dev	Point d'entrée de tous les périphériques (disque dur, écran, partition, consoles TTY)
etc	Ce dossier contient les commandes et fichiers nécessaires à l'administrateur système (XXX.conf, passwd, inittab, runlevels)
home	Répertoire personnel des utilisateurs
lib	Ce dossier contient les bibliothèques partagées essentielles au système lors du démarrage
lib64	idem /lib mais pour les 64bits (parfois, on trouvera lib et lib32. Dans ce cas, lib = 64bits et lib32 = 32bits)
mnt /media	Ce dossier contient les point de montage des partitions temporaires (clés USB, partitions de données)
opt	Répertoire générique pour l'installation de programmes compilés par
proc	Ce dossier n'existe pas physiquement sur un disque, elle est créée par le noyau dans la mémoire. Cette partition permet de donner des informations sur le système.
root	Répertoire personnel de l'administrateur
sbin	Ce dossier contient les programmes système essentiels utilisables par l'administrateur uniquement.
srv	Ce dossier n'est pas présent dans toutes les distributions. C'est un répertoire de données pour divers services (stockage des documents de comptes FTP, ou pages de sites web)
tmp	Répertoire fichiers temporaires
usr	Ce dossier contient des programmes installés (/usr/bin) avec leurs librairies (/usr/lib ou /usr/lib64) tels que firefox, libreoffice, ... quelques programmes réservés à l'administrateur système (/usr/sbin) et les fichiers de code source (/usr/src)
var	Ce dossier contient les données variables (fichiers de log) mais parfois les bases de données (/var/lib/mysql) et les pages de site web (/var/www/html)
sys	Ce dossier permet de connaître les informations sur le système et ses composants (matériels attachés et installés) d'une manière structurée

source : rebootinformatique.org

Exercice 2:

2.1)

L'option -R de la command **ls** permet d'afficher récursivement le contenu des sous-répertoires.

2.2)

Par défaut, les fichiers sont triés par ordre alphabétique en fonction de la première lettre des fichiers. Plus précisément, l'ordre générale de classement de dossiers et fichiers est le suivant : **ponctuations, chiffres, majuscules, minuscules, selon l'ordre défini par la table des caractères ASCII.**

La commande **sort** permet de changer l'ordre d'affichage. Notamment avec l'option **-r** qui permet d'inverser l'ordre de classement. l'option **-t** qui permet de trier en fonction de la date de dernière modification du fichier ou dossier.

Voici toutes les options de la commande sort :

Option	Signification
-b	Saute les colonnes constituées de blancs.
-d	Trie de type dictionnaire.
-n	Trie par ordre numérique.
-f	Aucune différenciation n'est faite entre minuscules et majuscules.
-b	Ignore les espaces placés en début de champ.
-r	Trie inverse.
-M	Trie chronologiquement les mois.
-t:	Trie suivants les champs séparés par les caractères deux points (``:").

2.3)

Voici les commandes à réaliser :

```
cd && ls -t  
ou  
cd && sort -t
```

2.4)

2.5)

La commande **ls** permet de lister le contenu d'un répertoire, à savoir ses fichiers et dossiers.

La commande **ls** suivi de l'argument **-l** permet, en plus du nom, d'afficher le type du fichier, les permissions d'accès, le nombre de liens physiques, le nom du propriétaire et du groupe, la taille en octets, et l'horodatage.

2.6)

Voici la commande à réaliser :

```
ls -l dossier/
```

Exercice 3:

3.1)

```
mkdir d1 && mkdir d2 && mkdir d3
```

3.2)

```
rmdir d*
```

3.3)

Le répertoire **mkdir** est créé

```
nicolas@debian:~$ mkdir mkdir
nicolas@debian:~$ ls
Desktop    Downloads  Music      Public     themes
Documents  mkdir      Pictures   Templates  Videos
nicolas@debian:~$
```

3.4)

Pour trouver les fichiers cachés on fait :

```
ls -la
```

3.5)

Oui, mkdir dispose de plusieurs options :

Option	Description
-m	Créer les répertoires avec le mode d'accès indiqué.
-p	S'assurer que chaque répertoire indiqué existe. Créer les répertoires parents manquants.

3.6)

La commande ne peut que fonctionner avec l'option **-p** :

```
mkdir -p Rapport/annexes
```

3.7)

L'option **-p** permet cela.

3.8)

L'option **-m** permet cela.

3.9)

L'exécution de cette commande va supprimé le répertoire tutu. L'option **-p** permet de supprimé également les répertoires parents s'ils deviennent vides après la suppression des répertoires mentionnés en arguments.

3.10)

Le dossier toto et dossier enfant tutu seront créés.

Exercice 4:

4.1)

Oui, l'option **-R** ou **-r** permet de le faire de façon récursive avec tous les fichiers et sous-dossier.

```
cp -R /dossier /destination
```

4.2)

Une erreur s'affiche, **cp: -r not specified; omitting directory 'mkdir'**

Il nous montre qu'il faut utiliser l'option **-R**

Une fois l'option **-R**, la commande créé une copie de /dossier dans /destination.

4.3)

Le fichier de destination est écrasé et remplacé par le nouveau.

4.4)

Les 3 fichiers **fichier1**, **fichier2** et **fichier3** sont copiés dans **/répertoire**.

4.5)

Pour renommer un répertoire, on utilise la commande **mv**, comme il n'y a pas de commande **"rename"**, on doit déplacer le dossier et changer son nom de destination pour le renommer.

```
mv dossier/ nouveau-nom-du-dossier/
```

Si le nom du dossier existe déjà, le dossier sera alors déplacé dans le dossier de destination, et il ne sera donc pas renommé.

4.6)

Pour déplacer un fichier et changer son nom en même temps il faut lui donner le chemin que l'on veut du dossier et à la suite le nouveau nom du fichier :

```
nicolas@debian:~/toto$ ls
mkdir test tttt tut tutu
nicolas@debian:~/toto$ mv test tttt/test-nouveau-nom
nicolas@debian:~/toto$ cd tttt/
nicolas@debian:~/toto/tttt$ ls
test-nouveau-nom testoss
nicolas@debian:~/toto/tttt$
```

Si le nom du fichier existe déjà, il sera alors écrasé par le nouveau.

4.7)

Pour ce qui est du dossier, le même principe s'applique que pour le déplacement et renommage d'un fichier.

Si, pour le dossier, lors du déplacement et du renommage il s'avère qu'un dossier est déjà présent avec le même nom, alors le dossier sera déplacé dans le dossier portant le même nom et deviendra alors un sous-dossier de celui-ci.

4.8)

L'option **-i** permet de demander confirmation avant écrasement d'un fichier déjà existant.

L'option **-f** permet de forcer l'écrasement d'un fichier sans demander confirmation.

D'après la documentation de la commande **mv**, l'option **-f** prend le dessus sur l'option **-i**, la commande sera donc semblable à **mv -f fichier1 fichier2**.

II. Commandes de base (suite).

Exercice 1:

a) Compter le nombre de fichiers se trouvant dans le répertoire courant :

```
ls | wc -l
```

Pour exemple, si cela donne **"12"** il y a alors 12 fichiers/dossiers dans le dossier courant. Cette commande n'est pas récursive.

b) Afficher (en utilisant le fichier **/etc/passwd**) la liste des utilisateurs pouvant travailler en bash classée par ordre alphabétique :

```
cat /etc/passwd | sort | grep bash$
```

```
nicolas@debian: ~
File Edit View Search Terminal Help
nicolas@debian:~$ cat /etc/passwd | sort | grep bash$
nicolas:x:1000:1000:nicolas,,,:/home/nicolas:/bin/bash
root:x:0:0:root:/root:/bin/bash
nicolas@debian:~$
```

Exercice 2:

Voici le résultat de exécution :

```
nicolas@debian: ~/Documents
File Edit View Search Terminal Help
nicolas@debian:~$ ls
Desktop  Downloads  Music      Public      Templates  toto
Documents  mkdir      Pictures  Rapport    themes     Videos
nicolas@debian:~$ cd Documents/
nicolas@debian:~/Documents$ touch hello
nicolas@debian:~/Documents$ nano hello
nicolas@debian:~/Documents$ gcc -o hello hello.c
gcc: error: hello.c: No such file or directory
gcc: fatal error: no input files
compilation terminated.
nicolas@debian:~/Documents$ mv hello hello.c
nicolas@debian:~/Documents$ gcc -o hello hello.c
hello.c:1:21: warning: extra tokens at end of #include directive
#include <stdio.h>  int main() { printf ("Hello World\n"); return(0); }
                    ^~~
/usr/bin/ld: /usr/lib/gcc/x86_64-linux-gnu/8/../../../../x86_64-linux-gnu/Scrt1.o:
in function `_start':
(.text+0x20): undefined reference to `main'
collect2: error: ld returned 1 exit status
nicolas@debian:~/Documents$ nano hello.c
nicolas@debian:~/Documents$ gcc -o hello hello.c
nicolas@debian:~/Documents$ ./hello
Hello World
nicolas@debian:~/Documents$
```

Exercices simples :

```
nicolas@debian: ~/Documents/TP1
File Edit View Search Terminal Help
nicolas@debian:~/Documents$ mkdir TP1
nicolas@debian:~/Documents$ cd TP1/
nicolas@debian:~/Documents/TP1$ pwd
/home/nicolas/Documents/TP1
nicolas@debian:~/Documents/TP1$ man touch
nicolas@debian:~/Documents/TP1$ touch 1.txt 2.txt 3.txt MonFichierVide.txt 4.txt
nicolas@debian:~/Documents/TP1$
```

Exercice 1:

```
nicolas@debian: ~/Documents/TP1
File Edit View Search Terminal Help
nicolas@debian:~/Documents/TP1$ ls
1.txt 2.txt 3.txt 4.txt MonFichierVide.txt
nicolas@debian:~/Documents/TP1$ ls -l
total 0
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 1.txt
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 2.txt
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 3.txt
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 4.txt
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 MonFichierVide.txt
nicolas@debian:~/Documents/TP1$ ls -al
total 8
drwxr-xr-x 2 nicolas nicolas 4096 Sep 10 21:45 .
drwxr-xr-x 3 nicolas nicolas 4096 Sep 10 21:44 ..
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 1.txt
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 2.txt
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 3.txt
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 4.txt
-rw-r--r-- 1 nicolas nicolas 0 Sep 10 21:45 MonFichierVide.txt
nicolas@debian:~/Documents/TP1$
```

ls permet seulement d'afficher les fichier et dossier.

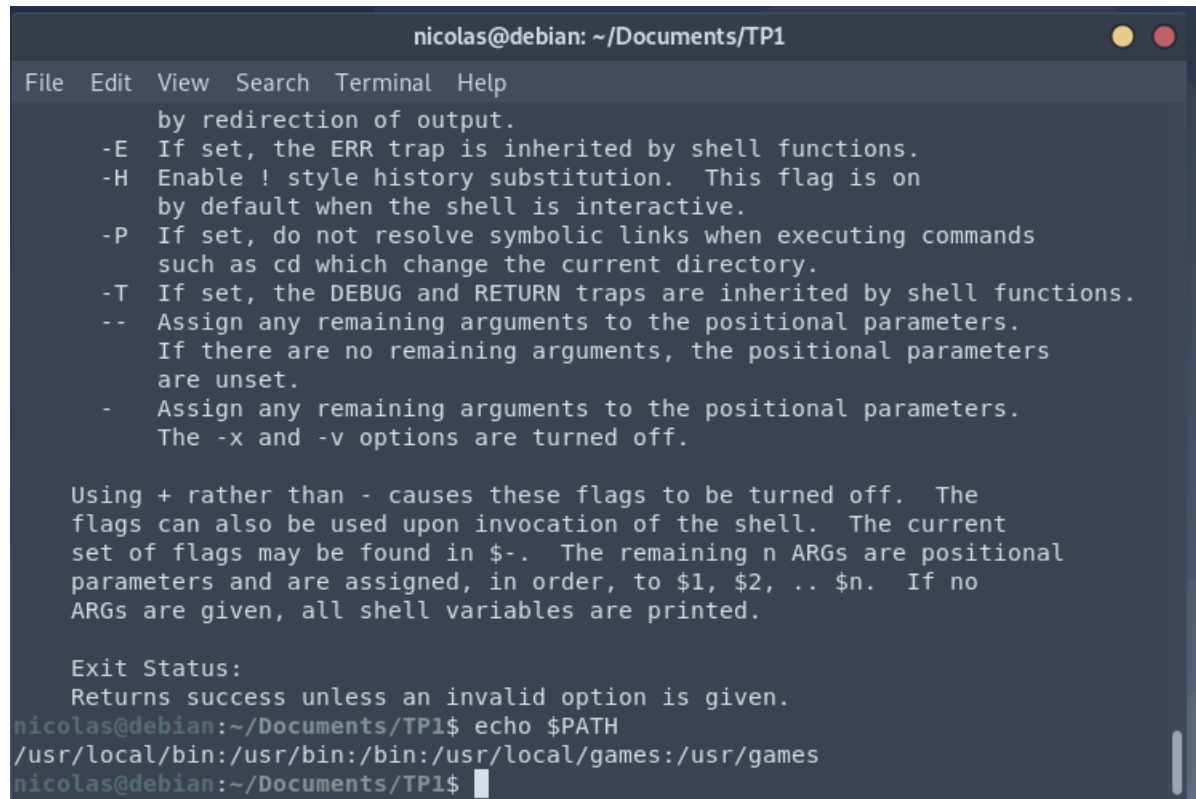
ls -l permet, en plus du nom, d'afficher le type du fichier, les permissions d'accès, le nombre de liens physiques, le nom du propriétaire et du groupe, la taille en octets, et l'horodatage.

ls -l permet d'afficher les même caractéristique que **ls -l** mais, avec tous les fichiers/dossiers cachés et également les fichiers commençant par ".".

Exercise 2:

```
set
man set
set --help
```

La commande **set** permet d'afficher toutes les variables d'environnements.

A terminal window titled 'nicolas@debian: ~/Documents/TP1' showing the output of the 'set' command. The output lists various shell options and their descriptions, followed by the current shell flags and the result of the 'echo \$PATH' command.

```
nicolas@debian: ~/Documents/TP1
File Edit View Search Terminal Help
by redirection of output.
-E If set, the ERR trap is inherited by shell functions.
-H Enable ! style history substitution. This flag is on
  by default when the shell is interactive.
-P If set, do not resolve symbolic links when executing commands
  such as cd which change the current directory.
-T If set, the DEBUG and RETURN traps are inherited by shell functions.
-- Assign any remaining arguments to the positional parameters.
  If there are no remaining arguments, the positional parameters
  are unset.
- Assign any remaining arguments to the positional parameters.
  The -x and -v options are turned off.

Using + rather than - causes these flags to be turned off. The
flags can also be used upon invocation of the shell. The current
set of flags may be found in $-. The remaining n ARGs are positional
parameters and are assigned, in order, to $1, $2, .. $n. If no
ARGs are given, all shell variables are printed.

Exit Status:
Returns success unless an invalid option is given.
nicolas@debian:~/Documents/TP1$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
nicolas@debian:~/Documents/TP1$
```

Exercise 3:

```
alias ls='ls --color=auto'
```

La commande **rm** à été remplacé par **rm -i** grâce à l'alias (on a créé un alias pour la commande **rm -i**). De ce fait, la commande **rm *** nous demande une confirmation avant suppression.

```
nicolas@debian: ~/Documents/TP1
File Edit View Search Terminal Help
nicolas@debian:~/Documents/TP1$ alias
alias ls='ls --color=auto'
nicolas@debian:~/Documents/TP1$ alias rm="rm -i"
nicolas@debian:~/Documents/TP1$ rm *
rm: remove regular empty file '1.txt'? y
rm: remove regular empty file '2.txt'? n
rm: remove regular empty file '3.txt'? n
rm: remove regular empty file '4.txt'? n
rm: remove regular empty file 'MonFichierVide.txt'? n
nicolas@debian:~/Documents/TP1$ ls
2.txt 3.txt 4.txt MonFichierVide.txt
nicolas@debian:~/Documents/TP1$
```

Exercice 4:

ls -al > listefic, cette commande va écrire le résultat de la commande **ls -al** dans le fichier **listefic**.

more listefic va lire le fichier listefic avec des pages.

cat listefic va ouvrir le fichier **listefic** dans l'éditeur.

```
nicolas@debian: ~/Documents/TP1
File Edit View Search Terminal Help
nicolas@debian:~/Documents/TP1$ ls -al > listefic
nicolas@debian:~/Documents/TP1$ more listefic
total 8
drwxr-xr-x 2 nicolas nicolas 4096 Sep 10 21:56 .
drwxr-xr-x 3 nicolas nicolas 4096 Sep 10 21:44 ..
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:45 2.txt
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:45 3.txt
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:45 4.txt
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:56 listefic
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:45 MonFichierVide.txt
nicolas@debian:~/Documents/TP1$ cat listefic
total 8
drwxr-xr-x 2 nicolas nicolas 4096 Sep 10 21:56 .
drwxr-xr-x 3 nicolas nicolas 4096 Sep 10 21:44 ..
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:45 2.txt
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:45 3.txt
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:45 4.txt
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:56 listefic
-rw-r--r-- 1 nicolas nicolas    0 Sep 10 21:45 MonFichierVide.txt
nicolas@debian:~/Documents/TP1$ ls
2.txt 3.txt 4.txt listefic MonFichierVide.txt
nicolas@debian:~/Documents/TP1$
```

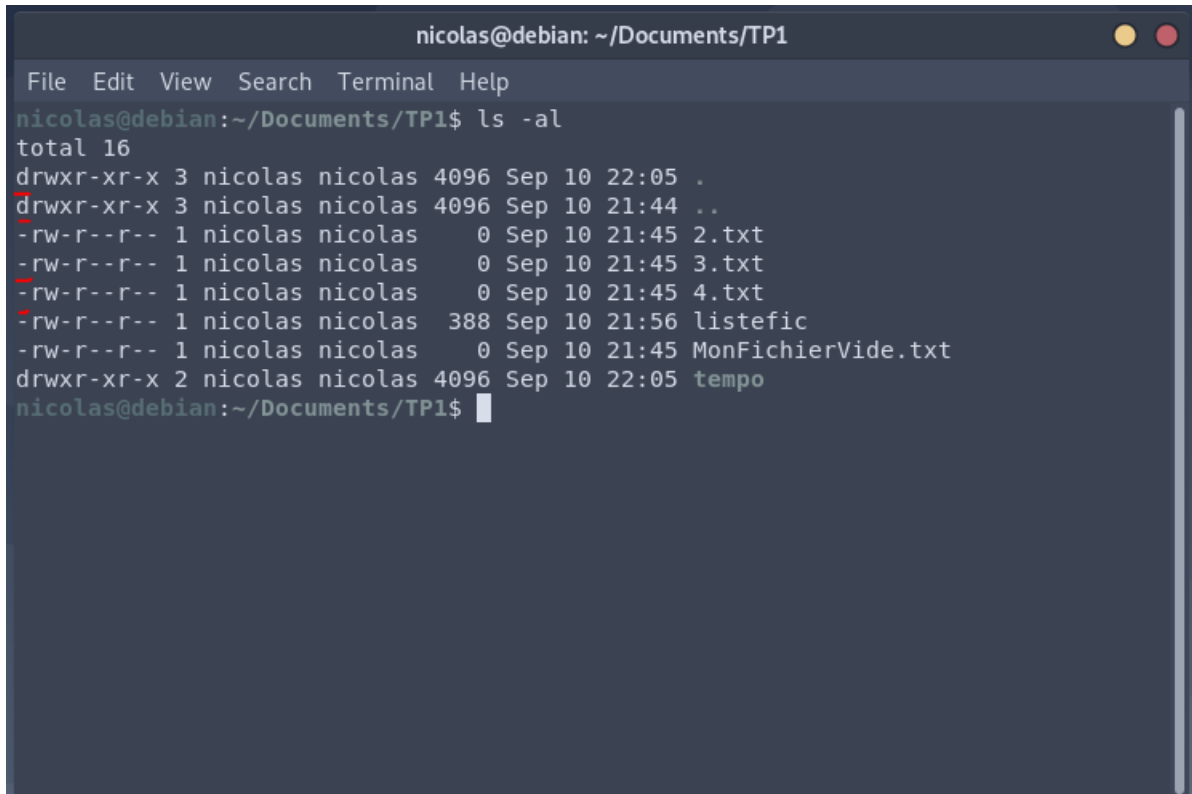
Exercice 5:

Exemple si tmpo et tempo sont dans le répertoire courant.

```
cp -r ~/tmpo/exos ~/tempo
```

Exercice 6:

Pour différencier un fichier d'un répertoire on regarde le premier caractère de la ligne affichée, s'il s'agit d'un **d**, alors c'est un dossier pour directory. Si c'est un tiret, alors c'est un fichier.

A terminal window titled 'nicolas@debian: ~/Documents/TP1' with a menu bar (File, Edit, View, Search, Terminal, Help). The user has entered 'ls -al' and the output shows a directory listing. The first character of each line indicates the file type: 'd' for directory and '-' for file. The 'tempo' directory is highlighted in the original image.

```
nicolas@debian:~/Documents/TP1$ ls -al
total 16
drwxr-xr-x 3 nicolas nicolas 4096 Sep 10 22:05 .
drwxr-xr-x 3 nicolas nicolas 4096 Sep 10 21:44 ..
-rw-r--r-- 1 nicolas nicolas   0 Sep 10 21:45 2.txt
-rw-r--r-- 1 nicolas nicolas   0 Sep 10 21:45 3.txt
-rw-r--r-- 1 nicolas nicolas   0 Sep 10 21:45 4.txt
-rw-r--r-- 1 nicolas nicolas  388 Sep 10 21:56 listefic
-rw-r--r-- 1 nicolas nicolas   0 Sep 10 21:45 MonFichierVide.txt
drwxr-xr-x 2 nicolas nicolas 4096 Sep 10 22:05 tempo
nicolas@debian:~/Documents/TP1$
```

Exercice 7:

```
rm -r ~/tempo
```

```
nicolas@debian: ~/Documents/TP1
File Edit View Search Terminal Help
nicolas@debian:~/Documents/TP1$ rm -rfv tempo/
removed directory 'tempo/'
nicolas@debian:~/Documents/TP1$
```

Exercice 8:

La commande **df** (disk free) permet d'afficher la taille de l'espace disque occupée et la taille de l'espace disque libre.

La commande **du** (disk usage) permet d'afficher la taille d'un répertoire et de tous les sous répertoires qu'il contient (récursivement).

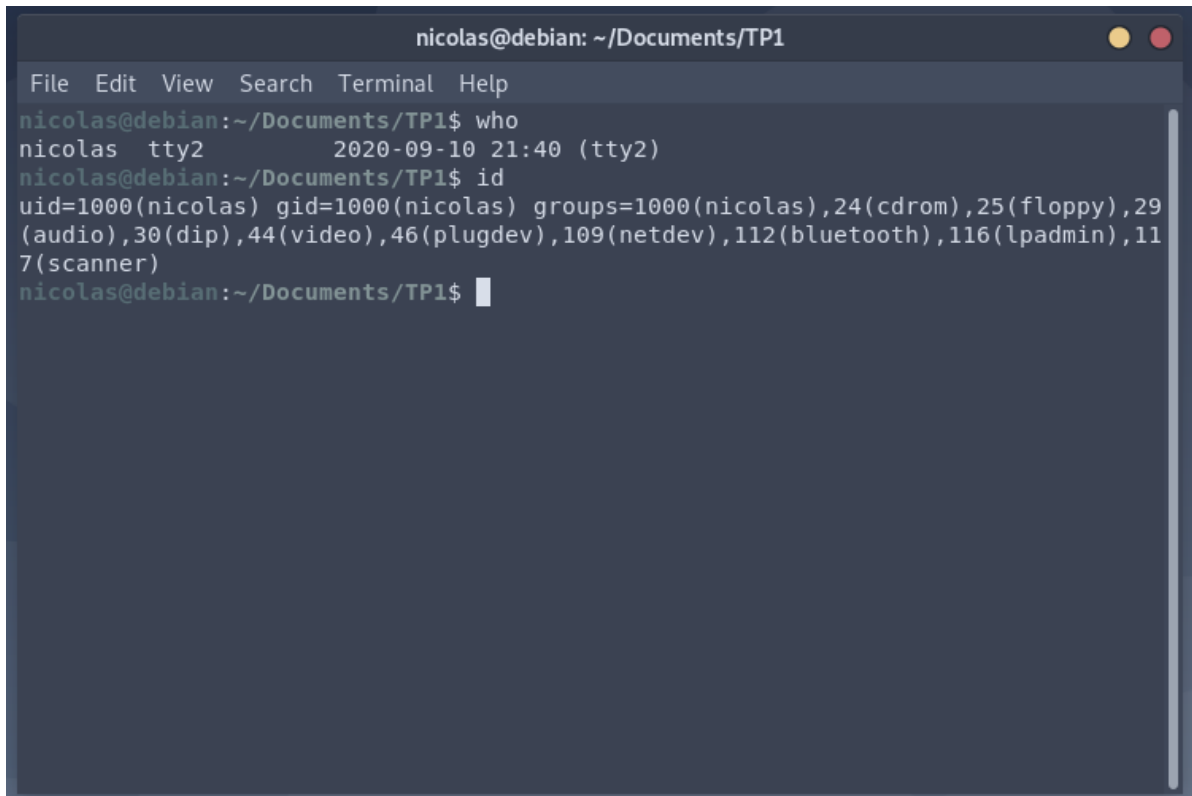
```
nicolas@debian: ~/Documents/TP1
File Edit View Search Terminal Help
nicolas@debian:~/Documents/TP1$ du
8
.
nicolas@debian:~/Documents/TP1$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            1545600         0   1545600  0% /dev
tmpfs           313088      5028   308060  2% /run
/dev/sda1       12263304 4380548   7240096 38% /
tmpfs           1565428         0   1565428  0% /dev/shm
tmpfs           5120         4     5116  1% /run/lock
tmpfs           1565428         0   1565428  0% /sys/fs/cgroup
tmpfs           313084      3512   309572  2% /run/user/1000
/dev/sr0         59206     59206         0 100% /media/cdrom0
nicolas@debian:~/Documents/TP1$
```

Exercice 9:

La commande **who** permet de savoir qui est connecté sur le système et que font-ils.

La commande **id** permet d'afficher tous les groupes et l'uid de l'utilisateur actuel.

(Vous avez marqué la réponse dans la question)

A terminal window titled 'nicolas@debian: ~/Documents/TP1' with standard window controls. The terminal shows the following commands and output:

```
nicolas@debian:~/Documents/TP1$ who
nicolas  tty2          2020-09-10 21:40 (tty2)
nicolas@debian:~/Documents/TP1$ id
uid=1000(nicolas) gid=1000(nicolas) groups=1000(nicolas),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),109(netdev),112(bluetooth),116(lpadmin),117(scanner)
nicolas@debian:~/Documents/TP1$
```