

1 - Le repertoire personnel est le dossier qui est chargé lorsque notre profil se connecte. A savoir le dossier 22113572t (numéro d'étudiant).

Obtenu en tapant la commande "cd ../"

```
22113572t@ST-2019008090:~$ cd ../  
22113572t@ST-2019008090:/home$ ls  
22113572t 22303710t
```

2 - La commande « cd ~ » permet de se placer dans son repertoire personnel.

```
22113572t@ST-2019008090:~$ cd /  
22113572t@ST-2019008090:/$ ls  
bin      dev      lib      media    proc      root      sbin     tmp      vmlinuz  
boot     etc      lib32    mnt      rc.local.log run      srv      usr      vmlinuz.old  
copyright home     lib64    opt      ro        rw       sys      var  
22113572t@ST-2019008090:/$ cd ~  
22113572t@ST-2019008090:~$ ls  
Bureau  Documents  Images  Modèles  Musique  Public  Téléchargements  Vidéos  
22113572t@ST-2019008090:~$
```

3 - Le nom du shell utilisé pour notre profil est bash, La commande utilisée est « echo \$SHELL »

```
22113572t@ST-2019008090:~$ echo $SHELL  
/bin/bash
```

4 - La commande « pwd » permet de savoir le path qui mène à notre fichier courant depuis la racine.

```
22113572t@ST-2019008090:~$ pwd  
/home/22113572t
```

5 - Avec les commandes « mkdir » et « cd » on peut créer et entrer dans un dossier

```
22113572t@ST-2019008090:~$ mkdir LinuxTPS  
22113572t@ST-2019008090:~$ cd LinuxTPS/  
22113572t@ST-2019008090:~/LinuxTPS$
```

6 - Si on fait une redirection de la commande « echo » dans le fichier voulu (même si il n'existe pas) nous créons le fichier avec le contenu voulu.

```
22113572t@ST-2019008090:~/LinuxTPS$ echo Les fiches de synthèse déposées sur Célène donnent des points essentiels sur le langage de commandes de linux. >> tp1.txt  
22113572t@ST-2019008090:~/LinuxTPS$ cat tp1.txt  
Les fiches de synthèse déposées sur Célène donnent des points essentiels sur le langage de commandes de linux.
```

7 - On va créer le fichier en faisant un « echo » de la commande « date » dans un fichier

```
22113572t@ST-2019008090:~$ echo $(date) > datetp1.txt  
22113572t@ST-2019008090:~$ cat datetp1.txt  
lun. 09 sept. 2024 11:01:08 CEST
```

8 - La commande « ls » permet d'afficher les fichiers et dossiers contenus dans le repertoire courant.

```
22113572t@ST-2019008090:~$ ls  
Bureau      Images      Modèles     Public      Vidéos  
Documents   LinuxTPS    Musique     Téléchargements
```

9 – Avec la commande « mkdir » on peut créer les dossiers voulus

```
22113572t@ST-2019008090:~$ cd LinuxTPS/
22113572t@ST-2019008090:~/LinuxTPS$ mkdir REPTP1
22113572t@ST-2019008090:~/LinuxTPS$ mkdir REPTP1/Rep1Q
22113572t@ST-2019008090:~/LinuxTPS$ mkdir REPTP1/RepTP1B/
22113572t@ST-2019008090:~/LinuxTPS$ mv REPTP1/Rep1Q/ REPTP1/Rep1A
22113572t@ST-2019008090:~/LinuxTPS$ ls
REPTP1  tp1.txt
22113572t@ST-2019008090:~/LinuxTPS$ ls REPTP1/
Rep1A  RepTP1B
```

10 – La commande « mv » permet de déplacer les deux fichiers

```
22113572t@ST-2019008090:~/LinuxTPS$ mv tp1.txt REPTP1/Rep1A/
22113572t@ST-2019008090:~/LinuxTPS$ mv ../
.bash_history      .dmrc              LinuxTPS/          Public/
.bash_logout       Documents/         .local/            .sh_history
.bashrc            .dotnet/           Modèles/           Téléchargements/
Bureau/            .emacs.d/          .mozilla/          Vidéos/
.cache/            .gnupg/            Musique/           .vscode/
.config/           .ICEauthority      .pki/              .Xauthority
datetp1.txt        Images/            .profile           .xsession-errors
22113572t@ST-2019008090:~/LinuxTPS$ mv ../datetp1.txt REPTP1/Rep1A/
```

11 - La commande « cat » permet d'afficher le contenu des fichiers

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Rep1A$ cat datetp1.txt
lun. 09 sept. 2024 11:01:08 CEST
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Rep1A$ cat tp1.txt
Les fiches de synthèse déposées sur Célène donnent des points essentiels sur le
langage de commandes de linux.
```

12 – On cree les trois fichiers voulus a l'aide de la commade « touch »

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Rep1A$ touch fic1.txt
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Rep1A$ touch fic2.txt
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Rep1A$ touch fic3.txt
```

13 – On cree le dossier voulu avec « mkdir » puis on copie recursivement avec « cp » tout le contenu de Rep1A

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1$ mkdir SauveRep1A
22113572t@ST-2019008090:~/LinuxTPS/REPTP1$ cp -r Rep1A/* SauveRep1A/
22113572t@ST-2019008090:~/LinuxTPS/REPTP1$ ls SauveRep1A/
datetp1.txt  fic1.txt  fic2.txt  fic3.txt  tp1.txt
```

14 – On peut créer une archive avec la commande « tar » avec l'option c(copier) v et f

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/SauveRep1A$ tar -cvf Tp1A.tar *.txt
datetp1.txt
fic1.txt
fic2.txt
fic3.txt
tp1.txt
```

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/SauveRep1A$ tar --list -f Tp1A.tar
datetp1.txt
fic1.txt
fic2.txt
fic3.txt
tp1.txt
```

15 – En utilisant la commande « rm » avec le bon globbing, on peut supprimer seulement les fichiers voulus

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/SauveRep1A$ ls
datetp1.txt  fic1.txt  fic2.txt  fic3.txt  Tp1A.tar  tp1.txt
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/SauveRep1A$ rm *.txt
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/SauveRep1A$ ls
Tp1A.tar
```

16 – On cree le dossier voulu avec mkdir

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1$ mkdir Verif
22113572t@ST-2019008090:~/LinuxTPS/REPTP1$ ls
Rep1A  RepTP1B  SauveRep1A  Verif
```

17 – On va extraire les fichiers dans l'archive avec la commande « tar » et l'option x (extraire) v et f en se plaçant dans le dossier Verif

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1$ cd Verif/
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Verif$ tar -xvf ../SauveRep1A/Tp1A.tar
datetp1.txt
fic1.txt
fic2.txt
fic3.txt
tp1.txt
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Verif$ ls
datetp1.txt  fic1.txt  fic2.txt  fic3.txt  tp1.txt
```

18 – On liste Verif avec « ls » et on fait une redirection avec un chevron dans le fichier voulu

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Verif$ ls > contenuRep1A.txt
22113572t@ST-2019008090:~/LinuxTPS/REPTP1/Verif$ cat contenuRep1A.txt
contenuRep1A.txt
datetp1.txt
fic1.txt
fic2.txt
fic3.txt
tp1.txt
```

19 – On supprime le dossier avec rm et l'option r pour supprimer tous les fichiers a l'interieur recursivement

```
22113572t@ST-2019008090:~/LinuxTPS/REPTP1$ rm -r Verif/
22113572t@ST-2019008090:~/LinuxTPS/REPTP1$ ls
Rep1A  RepTP1B  SauveRep1A
```

20 – id nous renvoie des informations utilisateurs et l'option u permet d'obtenir seulement l'identifiant effectif et l'option n le numero de cet identifiant
whoami a le meme effet

21 – On se place a la racine avec « cd / » puis on peut lister l'arborescence de niveau 1 (les dossiers fils de la racine avec ls)1

```
22113572t@ST-2019008090:~$ cd /
22113572t@ST-2019008090:/$ ls
bin      dev      lib      media    proc      root      sbin      tmp      vmlinuz
boot     etc      lib32    mnt      rc.local.log  run      srv      usr      vmlinuz.old
copyright home     lib64    opt      ro        rw        sys      var
```

22 – Pour obtenir le nombre d'elements dans notre repertoire personnel, on va mettre un pipe de la commande « ls » a la commande « wc -w » qui va compter le nombre d'elements renvoyes par ls

```
22113572t@ST-2019008090:~$ ls
Bureau      Images      Modèles     Public      Vidéos
Documents   LinuxTPS    Musique     Téléchargements
22113572t@ST-2019008090:~$ ls | wc -w
9
22113572t@ST-2019008090:~$ ls * | wc -w
18
```

23 – On va utiliser « grep » sur la commande appropiee

```
22113572t@ST-2019008090:~$ ls /usr/include/ | grep ^c.*
c++
cabvars.h
cairo
clang
clif.h
complex.h
cpio.h
crypt.h
cspmatrix.h
cspredef.h
ctype.h
cursesapp.h
cursesf.h
curses.h
cursesm.h
cursesw.h
cursslk.h
```

24 – Dans le meme principe, on va utiliser le pattern matching suivant

```
22113572t@ST-2019003264:~$ ls /usr/include/ | grep .*i.*
aio.h
aliases.h
asm-generic
blkid
brotli
bzlib.h
cairo
clif.h
cpio.h
cspmatrix.h
dirent.h
endian.h
eti.h
etip.h
evutil.h
execinfo.h
features-time64.h
finclude
fontconfig
gio-unix-2.0
glib-2.0
gnu-versions.h
gobject-introspection-1.0
```

25 – Meme principe mais on escape le point pour qu'il soit considere en tant que caractere

```
22113572t@ST-2019003264:~$ ls /usr/include/ | grep ..*i.*\.h
aio.h
aliases.h
bzlib.h
clif.h
cpio.h
cspmatrix.h
```

26 – La chaine est contenue dans les cas suivants

```
22113572t@ST-2019003264:~$ ls /usr/include/ | grep .*plu.*
sudo_plugin.h
```

27 – a) renvoie les fichiers qui contiennent une fois ou plus 415 ou 416

b) renvoie les fichiers qui ne contiennent que 415 ou que 416

28 – a) echo \$HOME affiche le contenu de la variable HOME

- b) echo '\$HOME' affiche la chaine de caractere « \$HOME »
- c) echo "\$HOME" affiche une string vide puis le contenu de la variable HOME puis une string vide
- d) echo /\$HOME affiche le dollar car l'antislash lui enleve sa signification syntaxique puis HOME