

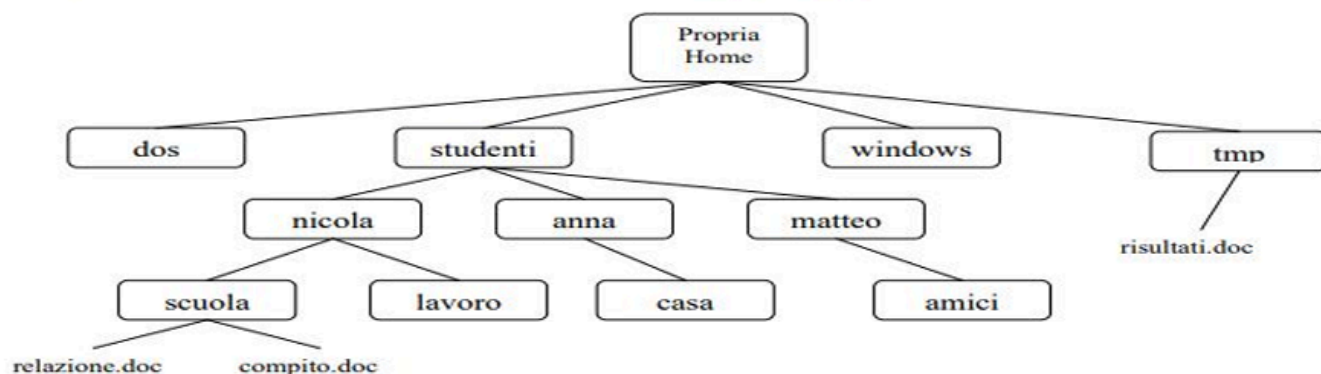
cd	Cambia Directory
Mkdir	Make Directory (fa una directory)
rmdir	rimuove una directory (se vuota)
mv	sposta un file - directory
cp	Copia un file (directory)
rm	Cancella un file
ls	visualizza il contenuto della cartella
pwd	print working directory (stampa il percorso assoluto dove mi trovo)
man argomento	Visualizza il manuale di un comando

Collegarsi al sistema con utenza e password

Esercizio 1

Come prima cosa creare le seguenti cartelle e sottocartelle (usando i comandi "terminale" mkdir cd rmdir ... a partire dalla propria HOME e visualizzarle a video:

(Per "Propria home" si intende il posto dove vi posiziona quando aprite il terminale!)



Ti trovi nella directory **lavoro** (sotto nicola), scrivere il comando per passare alla directory **casa** (sotto anna) con percorso relativo e percorso assoluto.

- Copia il file compito.doc (dalla directory scuola) nella directory corrente (casa).
- Sposta il file relazione.doc nella directory corrente (casa).
- Cancella la cartella **\tmp**
- Creare il file pippo.txt nella cartella lavoro
- Cambiare gli attributi del file pippo.txt e renderlo scrivibile e leggibile solo per il proprietario, mentre per tutti gli altri solo leggibile...
- Nascondere il contenuto della cartella anna
- Spostarsi nella cartella lavoro e visualizzare il contenuto del file pippo.txt
- Rimuovere la cartella amici
- Rimuovere tutte le cartelle precedentemente create

- 1) iniziamo con il creare le cartelle richieste dall'esercizio con il comando **mkdir**, **cd** per spostarsi dalle directory e **touch** per creare file. Si torna indietro con **cd ..**

```
File Actions Edit View Help
(kali㉿kali)-[~]
$ mkdir dos studenti windows tmp

(kali㉿kali)-[~]
$ cd studenti

(kali㉿kali)-[~/studenti]
$ mkdir nicola anna matteo
File System
(kali㉿kali)-[~/studenti]
$ cd nicola

(kali㉿kali)-[~/studenti/nicola]
$ mkdir scuola lavoro

(kali㉿kali)-[~/studenti/nicola]
$ cd scuola

(kali㉿kali)-[~/studenti/nicola/scuola]
$ touch relazione.doc compito.doc

(kali㉿kali)-[~/studenti/nicola/scuola]
$ cd ..

(kali㉿kali)-[~/studenti/nicola]
$ cd ..

(kali㉿kali)-[~/studenti]
$ cd anna

(kali㉿kali)-[~/studenti/anna]
$ mkdir casa

(kali㉿kali)-[~/studenti/anna]
$ cd ..

(kali㉿kali)-[~/studenti]
$ cd matteo

(kali㉿kali)-[~/studenti/matteo]
$ mkdir amcici

(kali㉿kali)-[~/studenti/matteo]
$ cd ..

(kali㉿kali)-[~/studenti]
$ cd ..

(kali㉿kali)-[~]
$ cd tmp

(kali㉿kali)-[~/tmp]
$ touch risultati.doc

(kali㉿kali)-[~/tmp]
$ cd ..

(kali㉿kali)-[~]
$
```

- 2) Copia il file **compito.doc** nella directory **anna/casa** usando il comando **CP**

```
(kali㉿kali)-[~/studenti/anna/casa]
$ cp /home/kali/studenti/nicola/scuola/compito.doc /home/kali/studenti/anna/casa
```

- 3) Sposta il file **relazione.doc** nella directory **casa** usando il comando **CD**

```
(kali㉿kali)-[~/studenti/anna/casa]
$ cp /home/kali/studenti/nicola/scuola/relazione.doc /home/kali/studenti/anna/casa
```

- 4) la cartella deve essere vuota prima di essere eliminata, andiamo ad utilizzare il **comando rm** (server per eliminare i file) e successivamente il **comando rmdir** (server per eliminare le directory)

```
(kali㉿kali)-[~]
$ rm tmp/risultati.doc

(kali㉿kali)-[~]
$ rmdir tmp
```

- 5) creare file **pippo.txt** tramite il comando **touch** e verificare con il comando **ls**

```
(kali㉿kali)-[~]
$ touch studenti/nicola/lavoro/pippo.txt

(kali㉿kali)-[~]
$ cd studenti/nicola/lavoro

(kali㉿kali)-[~/studenti/nicola/lavoro]
$ ls
pippo.txt
```

- 6) assegnare i permessi per ogni categoria di utente con il comando **chmod** e verificare con il comando **ls -l**

```
(kali㉿kali)-[~/studenti/nicola/lavoro]
$ chmod g-w pippo.txt

(kali㉿kali)-[~/studenti/nicola/lavoro]
$ ls -l
total 0
-rw-r--r-- 1 kali kali 0 Jan 10 09:44 pippo.txt

(kali㉿kali)-[~/studenti/nicola/lavoro]
$
```

- 7) Nascondere il contenuto della cartella casa, bisogna aggiungere il . prima del nome e verra inserito dentro al file di testo .hidden

```
(kali㉿kali)-[~/studenti/anna]  
$ mv casa .casa
```

- 8) modificare file **pippo.txt**

```
GNU nano 7.2 pippo.txt  
sto modificando il file pippo.txt
```

- 9) elimina la cartella amici con il comando rmdir (con cartella vuota)

```
(kali㉿kali)-[~/studenti/matteo]  
$ rmdir amcici
```

- 10) elimina tutto con il comando rm -r

```
(kali㉿kali)-[~/studenti/matteo]  
$ rm -r /home/kali/dos  
  
(kali㉿kali)-[~/studenti/matteo]  
$ rm -r /home/kali/studenti  
  
(kali㉿kali)-[~/studenti/matteo]  
$ rm -r /home/kali/windows
```

who	lista utenti collegati
who am i	chi sono io
jobs	elenco lavori sul terminale
&	apre processo in background
fg	metti in foreground
bg	metti in background
ps	elenco processi
kill	termina processo

Provare i comandi:

w
who
who am i

Esercizi - processi:

1. Aprire un terminale
2. leggere il manuale del comando job, ps e kill
3. lanciare il comando vi pippo
4. aprire un nuovo terminale e visualizzare tutti i propri processi...
5. cercare di terminare (killare) il processo vi per sbloccare il terminale precedente
6. lanciare il comando firefox in background
7. portarlo in background
8. cercare di terminare il processo firefox
9. verificare quanto spazio si sta occupando su disco

- 1) Per leggere un manuale nella shell linux bisogna precedere un comando con **man**

```
(kali㉿kali)-[~]
$ man ps
```

- 4) per visualizzare i processi possiamo usare il comando **ps -a** oppure **ps-e** (per visualizzarli tutti)

```
(kali㉿kali)-[~]
$ ps -a
  PID TTY          TIME CMD
 31454 pts/0    00:00:00 vi
 31854 pts/2    00:00:00 ps
```

5) per terminare un programma bisogna utilizzare il comando **kill** seguito dal **PID** del programma da arrestare

```
(kali㉿kali)-[~]
$ kill 31454

(kali㉿kali)-[~]
$ ps -a
zsh PID TTY      Stopped   Vi  TIME CMD
    32692 pts/2                00:00:00 ps
```

6) Avvio in beckground firefox con il comando `firefox&`

```
(kali㉿kali)-[~]
$ firefox&
[1] 34529

(kali㉿kali)-[~]
$ bg firefox
bg: job already in background

(kali㉿kali)-[~]
$
```

7) Killo il processo firefox

```
(kali㉿kali)-[~]
$ ps -a
PID TTY      Stopped   Vi  TIME CMD
    34529 pts/2                00:00:03 firefox-esr
    34604 pts/2                00:00:00 Socket Process
    34630 pts/2                00:00:00 Privileged Cor
    34707 pts/2                00:00:00 file:// Conten
    34744 pts/2                00:00:00 WebExtensions
    34815 pts/2                00:00:00 Web Content
    34819 pts/2                00:00:00 Web Content
    34823 pts/2                00:00:00 Web Content
    35793 pts/2                00:00:00 ps

(kali㉿kali)-[~]
$ kill 34529

(kali㉿kali)-[~]
$
Exiting due to channel error.
Exiting due to channel error.
Exiting due to channel error.
Exiting due to channel error.
Exiting due to channel error.

[1] + terminated  firefox

(kali㉿kali)-[~]
$ ps -a
PID TTY      Stopped   Vi  TIME CMD
    36089 pts/2                00:00:00 ps
```

8) per verificare quanto spazio sul disco abbiamo stiamo utilizzando useremo il comando df meglio aggiungiamo l'opzione -h cioè human che lo rende leggibile

```
(kali㉿kali)-[~]
$ df
Filesystem      1K-blocks      Used Available Use% Mounted on
udev            966784         0    966784  0% /dev
tmpfs           201428         972    200456  1% /run
/dev/sda1      82083148 15217776 62649824 20% /
tmpfs          1007136         0    1007136  0% /dev/shm
tmpfs           5120          0         5120  0% /run/lock
tmpfs           201424        112    201312  1% /run/user/1000

(kali㉿kali)-[~]
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            945M     0  945M   0% /dev
tmpfs           197M  972K  196M   1% /run
/dev/sda1       79G   15G   60G  20% /
tmpfs           984M     0  984M   0% /dev/shm
tmpfs           5.0M     0   5.0M   0% /run/lock
tmpfs           197M  112K  197M   1% /run/user/1000

(kali㉿kali)-[~]
$
```