

# Comandos para terminal

## man

Proporciona información sobre los comandos u otros programas del entorno Linux (manual pages). En el caso particular de los comandos de la shell de Linux, dado que estos suelen tener una gran cantidad de modificadores, lo usual es conocer solo algunas funcionalidades de los comandos más conocidos y recurrir a las manual pages del comando cuando se desee más información.

Por más información, ejecutar:

```
man man
```

## ls

Lista el contenido de un directorio. Si lo usamos sin argumentos, lista el contenido del directorio actual o corriente.

Por más información:

```
man ls
```

## more

Filtro que permite desplegar en una terminal el contenido de un archivo de texto de a una pantalla a la vez.

```
<COMANDO> | more
```

El comando more tiene dos usos muy importantes:

Permite ver el contenido de un archivo:

```
more <NOMBRE_ARCHIVO>
```

- 

Le da formato a la salida de un comando para visualizarlo por páginas. Esto es muy útil cuando el resultado en pantalla de un comando tiene más líneas que la terminal.

Por ejemplo:

```
ls | more
```

- 

## cd

Cambia el directorio corriente (Change Directory) en que nos encontramos posicionados. Si se utiliza sin argumentos, cambia al directorio raíz del homedir del usuario.

Ejemplos:

```
cd <NOMBRE_DIRECTORIO> Cambia al directorio <NOMBRE_DIRECTORIO>
cd                      Cambia al directorio homedir del usuario.
cd ..                  Cambia al directorio padre del actual.
cd -                   Cambia al directorio previo.
```

## **mkdir**

Crea entradas de directorios.

```
mkdir <NOMBRE_DIRECTORIO>
```

## **rmdir**

Elimina entradas de directorios vacíos. Para eliminar un directorio y todo su contenido, ver el comando rm más adelante.

```
rmdir <NOMBRE_DIRECTORIO>
```

## **rm**

Elimina archivos o directorios.

```
rm <NOMBRE_DE_LOS_ARCHIVOS>
```

```
rm -r <NOMBRE_DE_LOS_DIRECTORIOS>
```

Ejemplos:

```
rm *.class
rm prueba.txt mi_directorio/prueba2.txt
rm -r mi_directorio
```

Importante:

Los archivos y directorios eliminados con este comando o el anterior no van a parar a la papelera de reciclaje, serán eliminados permanentemente.

## **cp**

Copiar archivos o directorios. Para copiar un directorio se debe agregar el modificador "-r".

```
cp <ARCHIVOS_ORIGEN> <DESTINO>
```

```
cp -r <DIRECTORIOS_ORIGEN> <DESTINO>
```

Ejemplos:

```
cp *.txt mi_dir/mi_dir2
cp prueba.txt mi_directorio/prueba2.txt
cp -r mi_directorio mi_directorio_2
```

## mv

Mueve archivos o directorios. La sintaxis de este comando es similar a la del cp, solo que no es necesario utilizar el modificador "-r".

Este comando es útil a la hora de renombrar archivos o directorios (moviéndolo con el nombre con el que se lo quiere renombrar) dado que la mayoría de las shell de Linux no poseen un comando específico para esto.

Ejemplos:

```
mv arch01.txt arch02.inf
```

## du

Reporta el espacio de disco utilizado por archivos y directorios (du = disk usage). Este comando generalmente se utiliza de esta forma:

```
du -ks <ARCHIVOS_O_DIRECTORIOS>
```

Con el modificador **-k** el comando devuelve el resultado en Kilobytes.

Ejemplos:

```
du -ks .  
du -ks mi_directorio/mi_directorio2  
du -ks *.java
```

## exit

Cierra la shell actual.