

Aprendiendo a caminar en la terminal 2/21

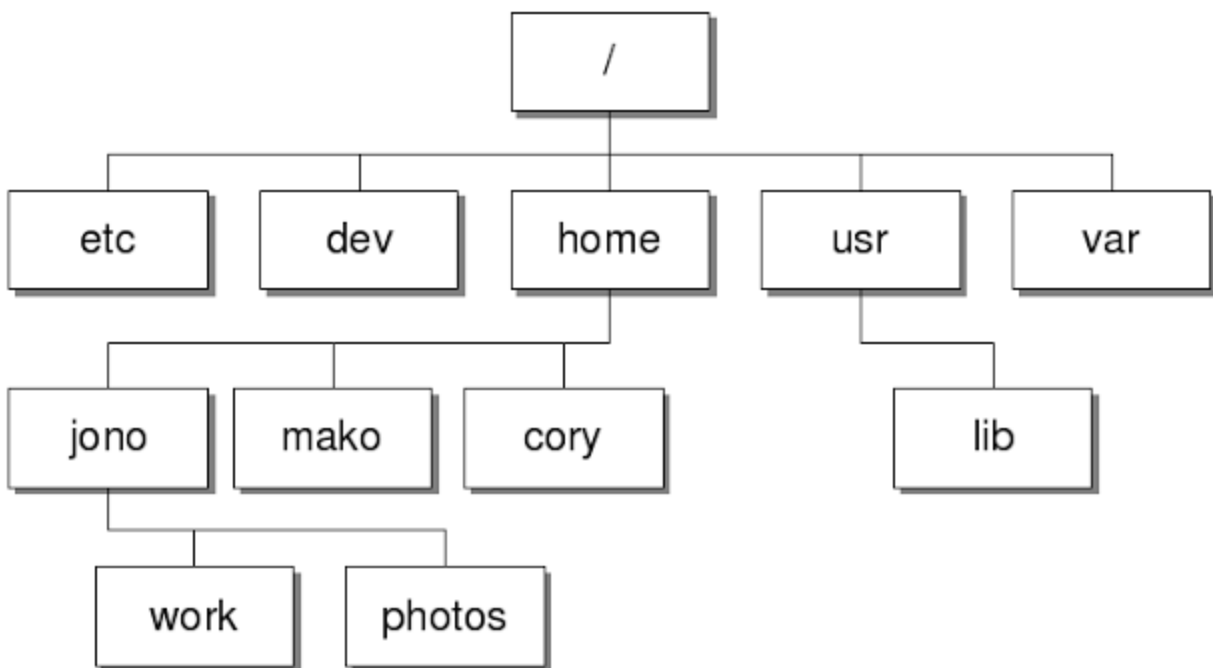
RECURSOS

MARCADORES

En los sistemas operativos organizamos los archivos en una estructura de carpetas en forma de árbol jerárquico. Este árbol cambia dependiendo de los diferentes sistemas operativos (no tanto en Linux y Mac).

Cómo funciona el sistema de carpetas

El sistema operativo con el que trabajaremos es Linux, por lo tanto usaremos su estructura. Observa el siguiente esquema:



La carpeta con el símbolo "/" es la raíz, ahí es donde comienza todo el sistema de ficheros (el equivalente en Windows podría ser el fichero "C:\"). Dentro de esta carpeta hay varios ficheros, el que nos importa en este momento es el "Home".

La carpeta "Home" contiene una carpeta por cada usuario del sistema y ya dentro de cada una de estas carpetas, estarán las carpetas que conocemos de toda la vida como imágenes, documentos, música, etc.

Cómo entender la información al inicio de la terminal

Cuando abrimos la terminal vamos a ver algo como esto:

```
miguelangel@DESKTOP-3R804MK:~$
```

Todo esto parece un mensaje encriptado, pero es más sencillo de lo que parece, vamos por partes:

- **miguelangel** es el nombre del usuario que está activo. En tu caso aparecerá el nombre del usuario que esté activo en tu computadora.
- **DESKTOP-3R804MK** es el nombre que el sistema operativo le dio a la computadora. En este caso se usa Windows y ese es el nombre que en la instalación Windows le asignó al dispositivo. Si usas Linux aparecerá el nombre del PC que hayas colocado en la instalación.
- **~** es la carpeta en la que te encuentras y ahora es cuando empiezan las confusiones porque en el esquema no estaba una carpeta con ese símbolo. Más adelante verás como todo tiene sentido.
- Por último, **\$** significa que somos un usuario normal y no un root o un superusuario. Más adelante hablaremos más acerca de esto.

Tu primer comando (pwd)

Ahora sí, vamos con el poderosísimo comando que te ayudará a descubrir algo muy importante: ¿dónde estás?

Vimos que el símbolo **~** indica la carpeta donde te encuentras, ¿cuál es esa carpeta? Para saberlo escribe el siguiente comando:

```
pwd
```

Te debió haber aparecido algo como esto:

```
/home/miguelangel
```

El comando **pwd**, significa **Print Working Directory** y te muestra el directorio en el que te encuentras. En este caso dice que estoy en el home del usuario que es miguelangel.

Usar el comando Change Directory (cd)

No trabajarás todo desde la misma carpeta, así que necesitas saber como moverte entre carpetas sin salir de la terminal. Para eso usamos el comando **cd** que significa **Change Directory**.

Para usarlo escribimos **cd** seguido del directorio al que queremos movernos, por ejemplo, dentro del home tengo la carpeta Documents, así que para moverme ahí escribo:

```
cd Documents
```

Y ahora vemos que la información que nos muestra la consola cambió y nos indica donde estamos:

```
miguelangel@DESKTOP-3R804MK:~/Documents$
```

Atajos

Virgulilla (~)

De seguro te preguntarás porqué la virgulilla (~) indica la carpeta del usuario en el home, ¿cuál es la utilidad? Supongamos que estamos navegando por diferentes carpetas y nos encontramos en la siguiente dirección:

```
miguelangel@DESKTOP-3R804MK:/mnt/c/users/migue/development$
```

(Si estás usando WSL, eso significa que estás buscando entre carpetas que están en Windows)

Si ahora quieres volver al home habría que escribir el siguiente comando:

```
cd /home/miguelangel
```

Pero puedes mejor escribir:

```
cd ~
```

Y llegas más rápido.

Punto y doble punto (.) (...)

No siempre quieres ir hacia adelante, a veces necesitas devolvete, para eso utilizas el atajo del doble punto (...) que te envía a la carpeta que está atrás. Si haces esto:

```
miguelangel@DESKTOP-3R804MK:~$ cd ..
```

Como estás en la carpeta del usuario, pasarás a estar en la carpeta de home.

Por otro lado, el punto (.) indica la carpeta actual, así puedes hacer direcciones más precisas como esta:

```
miguelangel@DESKTOP-3R804MK:/home$ cd ./miguelangel/Documents/
```

Slash (/)

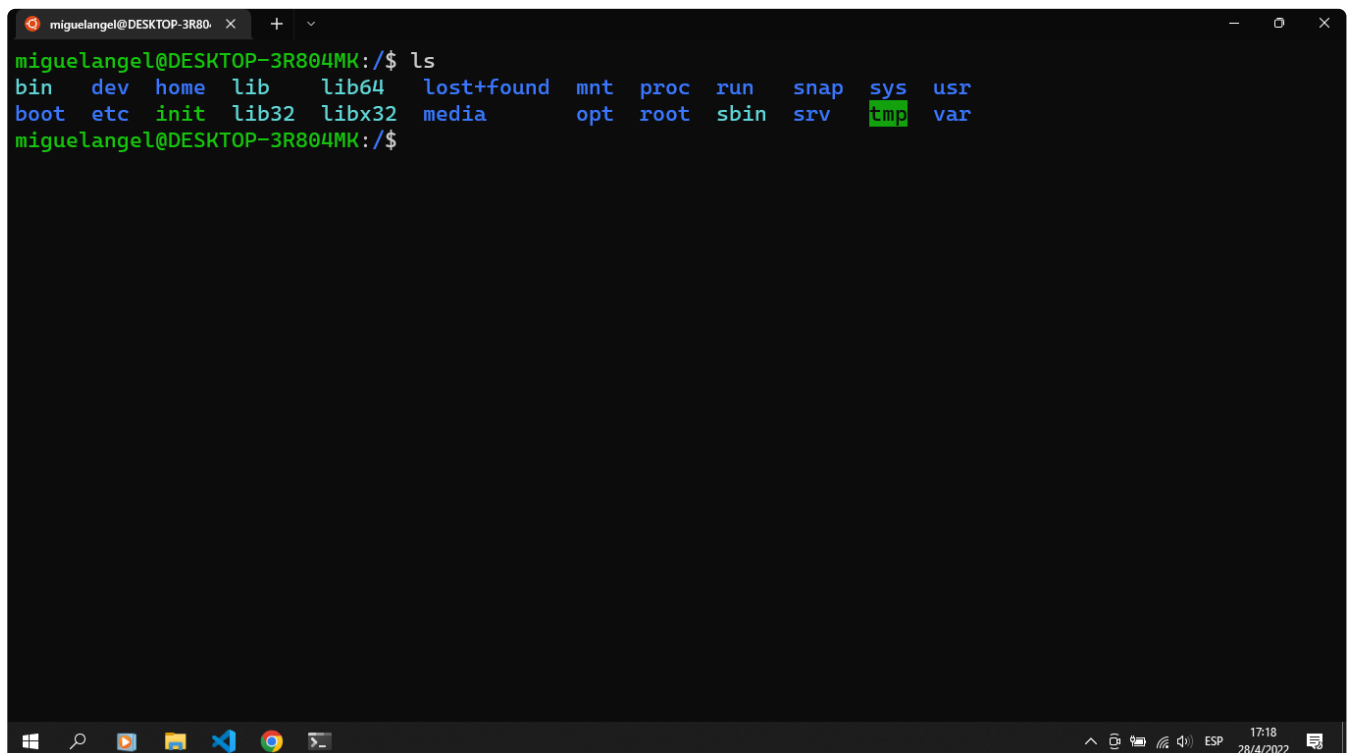
Por último, el atajo slash te lleva a la raíz donde están todas las carpetas del sistema operativo.

```
miguelangel@DESKTOP-3R804MK:/mnt/c/users/migue$ cd /  
miguelangel@DESKTOP-3R804MK:/$
```

¿Cómo saber lo que hay adentro de las carpetas? (ls)

No siempre vas a saber lo que hay adentro de las carpetas, por lo que necesitas listar lo que hay dentro, para eso está el comando `ls`, por lo general le dan el significado de **List**.

Si ejecutamos el comando `ls` veremos las carpetas y archivos que hay dentro:



```
miguelangel@DESKTOP-3R804MK:/$ ls  
bin  dev  home  lib   lib64  lost+found  mnt  proc  run   snap  sys  usr  
boot  etc  init  lib32  libx32  media      opt  root  sbin  srv   tmp  var  
miguelangel@DESKTOP-3R804MK:/$
```

Como puedes ver, cada nombre tiene un color diferente. Para este caso el azul oscuro es para las carpetas y el claro es para los archivos, esto dependerá de la paleta de colores que tenga tu terminal.

Pero, ¿qué tal si queremos conocer más información? Para saber información adicional como la fecha de creación, el peso, los permisos, etc., utilizamos la opción `-l` que significa **Long**:

```
miguelangel@DESKTOP-3R804MK:/$ ls -l
total 684
lrwxrwxrwx 1 root root 7 Mar 24 17:40 bin -> usr/bin
drwxr-xr-x 2 root root 4096 Mar 24 17:47 boot
drwxr-xr-x 8 root root 2720 Apr 28 17:12 dev
drwxr-xr-x 97 root root 4096 Apr 28 17:12 etc
drwxr-xr-x 3 root root 4096 Apr 28 15:19 home
-rwxr-xr-x 2 root root 632096 Apr 23 19:19 init
lrwxrwxrwx 1 root root 7 Mar 24 17:40 lib -> usr/lib
lrwxrwxrwx 1 root root 9 Mar 24 17:40 lib32 -> usr/lib32
lrwxrwxrwx 1 root root 9 Mar 24 17:40 lib64 -> usr/lib64
lrwxrwxrwx 1 root root 10 Mar 24 17:40 libx32 -> usr/libx32
drwx----- 2 root root 16384 Apr 10 2019 lost+found
drwxr-xr-x 2 root root 4096 Mar 24 17:40 media
drwxr-xr-x 4 root root 4096 Apr 28 15:18 mnt
drwxr-xr-x 2 root root 4096 Mar 24 17:40 opt
dr-xr-xr-x 125 root root 0 Apr 28 17:12 proc
drwx----- 2 root root 4096 Mar 24 17:42 root
drwxr-xr-x 9 root root 180 Apr 28 17:12 run
lrwxrwxrwx 1 root root 8 Mar 24 17:40 sbin -> usr/sbin
drwxr-xr-x 6 root root 4096 Mar 24 17:42 snap
drwxr-xr-x 2 root root 4096 Mar 24 17:40 srv
dr-xr-xr-x 11 root root 0 Apr 28 17:12 sys
drwxrwxrwt 2 root root 4096 Apr 28 17:15 tmp
drwxr-xr-x 14 root root 4096 Mar 24 17:41 usr
```

Ahora nos muestra una tabla que tiene mucha información que estaremos viendo más adelante, pero céntrate en la quinta columna, esos números son el peso, pero el peso en bytes, lo cual lo hace un poco complicado de leer. Podemos añadir la opción `-h` que significa **Human Readable**:

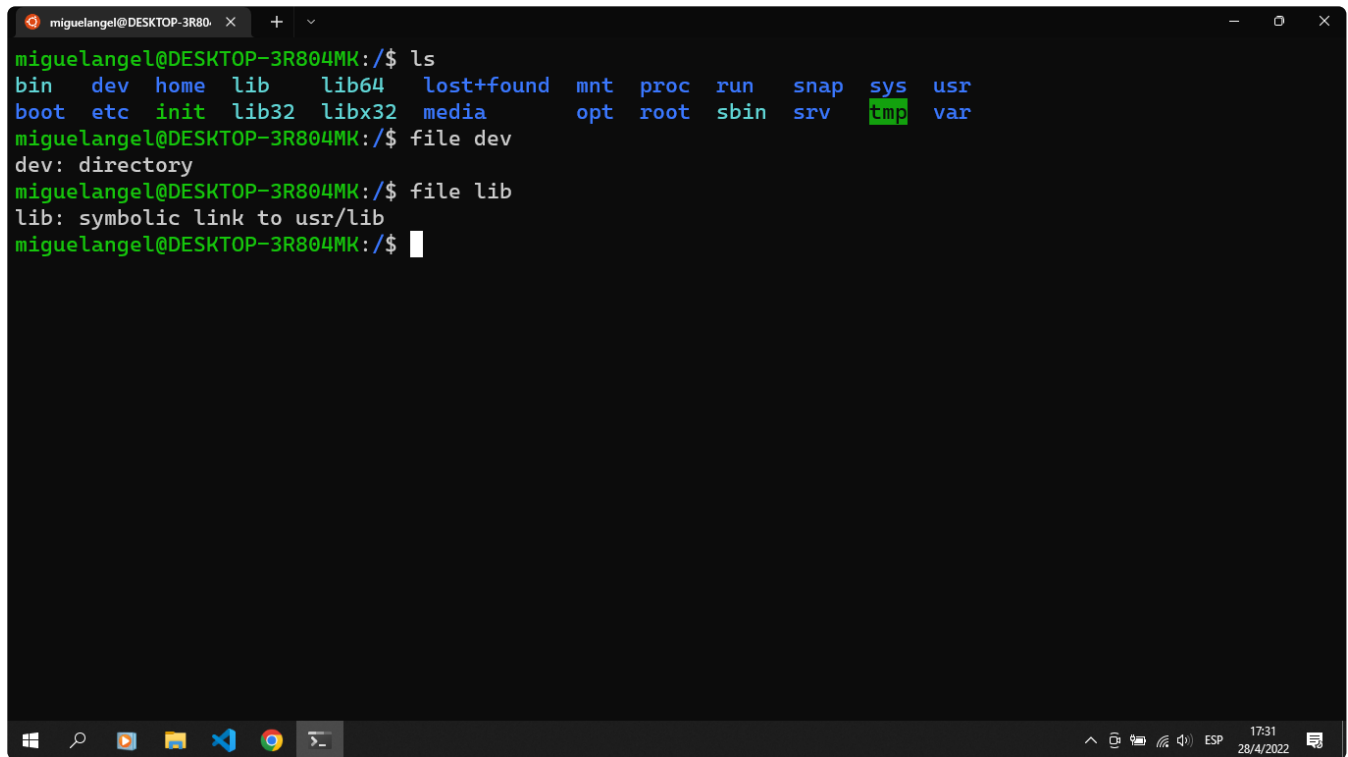
```
miguelangel@DESKTOP-3R804MK:/$ ls -lh
total 684K
lrwxrwxrwx 1 root root 7 Mar 24 17:40 bin -> usr/bin
drwxr-xr-x 2 root root 4.0K Mar 24 17:47 boot
drwxr-xr-x 8 root root 2.7K Apr 28 17:12 dev
drwxr-xr-x 97 root root 4.0K Apr 28 17:12 etc
drwxr-xr-x 3 root root 4.0K Apr 28 15:19 home
-rwxr-xr-x 2 root root 618K Apr 23 19:19 init
lrwxrwxrwx 1 root root 7 Mar 24 17:40 lib -> usr/lib
lrwxrwxrwx 1 root root 9 Mar 24 17:40 lib32 -> usr/lib32
lrwxrwxrwx 1 root root 9 Mar 24 17:40 lib64 -> usr/lib64
lrwxrwxrwx 1 root root 10 Mar 24 17:40 libx32 -> usr/libx32
drwx----- 2 root root 16K Apr 10 2019 lost+found
drwxr-xr-x 2 root root 4.0K Mar 24 17:40 media
drwxr-xr-x 4 root root 4.0K Apr 28 15:18 mnt
drwxr-xr-x 2 root root 4.0K Mar 24 17:40 opt
dr-xr-xr-x 125 root root 0 Apr 28 17:12 proc
drwx----- 2 root root 4.0K Mar 24 17:42 root
drwxr-xr-x 9 root root 180 Apr 28 17:12 run
lrwxrwxrwx 1 root root 8 Mar 24 17:40 sbin -> usr/sbin
drwxr-xr-x 6 root root 4.0K Mar 24 17:42 snap
drwxr-xr-x 2 root root 4.0K Mar 24 17:40 srv
dr-xr-xr-x 11 root root 0 Apr 28 17:12 sys
drwxrwxrwt 2 root root 4.0K Apr 28 17:15 tmp
drwxr-xr-x 14 root root 4.0K Mar 24 17:41 usr
```

Nota: para añadir más de una opción no hacemos esto `ls -l -h` sino esto `ls -lh`

Uso del comando file

Con el comando `ls` podemos listar los elementos dentro de una carpeta, pero a veces no sabemos si es un archivo o si es una carpeta o lo que sea.

Para eso tenemos el comando `file` que te da información acerca del tipo de elemento que hayas seleccionado:

A terminal window titled 'miguelangel@DESKTOP-3R80...' displays the following commands and outputs:

```
miguelangel@DESKTOP-3R804MK:/$ ls
bin  dev  home  lib   lib64  lost+found  mnt  proc  run  snap  sys  usr
boot etc  init  lib32 libx32 media      opt  root  sbin  srv   tmp  var

miguelangel@DESKTOP-3R804MK:/$ file dev
dev: directory

miguelangel@DESKTOP-3R804MK:/$ file lib
lib: symbolic link to usr/lib

miguelangel@DESKTOP-3R804MK:/$
```

The terminal window includes a taskbar at the bottom with icons for Windows, search, and various applications, along with system status icons and a clock showing 17:31 on 28/4/2022.

En este caso vemos que el comando `file` nos dice que “dev” es un directorio y “lib” es un enlace simbólico.

Comando	Función
<code>pwd</code>	Print Working Directory Muestra en la carpeta en la que estas
<code>cd</code>	Change Directory Te mueve a la carpeta que deseas
<code>ls</code>	List Lista los archivos y carpetas dentro del directorio que selecciones
<code>file</code>	Muestra la información del archivo que selecciones