

Terminal y manejo de archivos

Terminal y manejo de archivos	1
¿Qué aprenderás?	2
Introducción	2
Introducción al terminal	3
Inicialización de terminal	3
Utilizando el terminal	3
Explicación de las estructuras de directorio	4
Conocer en qué directorio estamos (pwd)	5
Listar archivos (ls)	5
Comandos de navegación entre directorios (cd)	7
Anatomía de un comando	8
Manejo de archivos y carpetas	9
Creación de archivos	9
Creación de directorios	9
Copia de archivos	9
Copia de directorios	10
Mover archivos y directorios	10
Borrar archivos	10
Ejercicio guiado: Manejando archivos y carpetas	12
Contexto	12
Resumen	13



¡Comencemos!

¿Qué aprenderás?

- Realizar operaciones de navegación de directorios, usando los comandos básicos del terminal, para crear y manipular archivos y directorios.

Introducción

En un desarrollo es importante realizar la trazabilidad sobre los cambios que se realizan en el código fuente, también poder compartirlo y a la vez mantener registros de los cambios por los que va pasando.

Por lo anterior, existen sistemas de versionado de código que se basan en respaldar archivos y carpetas que conforman un repositorio y que permiten operaciones como: volver atrás modificaciones, conocer las diferencias entre las versiones, obtener una copia y publicar los cambios realizados. Estos sistemas nos van a ser de mucha utilidad, ya que nos permitirán gestionar rápidamente los proyectos y visualizar los estados por lo que pasa un proyecto.

A lo largo de esta unidad, conocerás cómo controlar las versiones de tu código y cómo respaldar de forma online, esto te permitirá compartirlo y trabajar colaborativamente.

¡Vamos con todo!



Introducción al terminal

El terminal, es una poderosa herramienta basada en una interfaz de texto que sirve para comunicarse directamente con un computador. Utiliza líneas de comandos para navegar por archivos y directorios, al mismo tiempo, se utiliza para interactuar con programas que no tienen interfaz gráfica.

Inicialización de terminal

Para inicializar el terminal según los diferentes sistemas operativos utilizaremos los siguientes atajos:

- **En Linux:** Presiona `ctrl + alt + t`.
- **En Mac:** Presiona `⌘ + espacio`, busca por spotlight terminal.
- **En Windows:** Presiona `inicio(tecla de windows) + r`, escribe “cmd” en la caja de texto y presiona aceptar.

No obstante, considerando que no todas las terminales utilizan las mismas líneas de comando, para homologar los procesos que aprenderemos a hacer más adelante ocuparemos un programa especial llamado [git Bash](#), el cual se instala por defecto cuando instalas Git en tu sistema computador.

Utilizando el terminal

A continuación, revisaremos los comandos básicos que nos ayudarán a movernos y ubicarnos dentro de los directorios de nuestro computador, como también listar los archivos o carpetas al interior de un directorio.

Es necesario que sepamos movilizarnos y utilizar la terminal, ya que algunas herramientas solo permiten la interacción a través de ella. En nuestro caso la utilizaremos para controlar las versiones de nuestro proyecto.

Explicación de las estructuras de directorio

Las carpetas de nuestro computador tienen una estructura del tipo árbol donde existen directorios y subdirectorios.

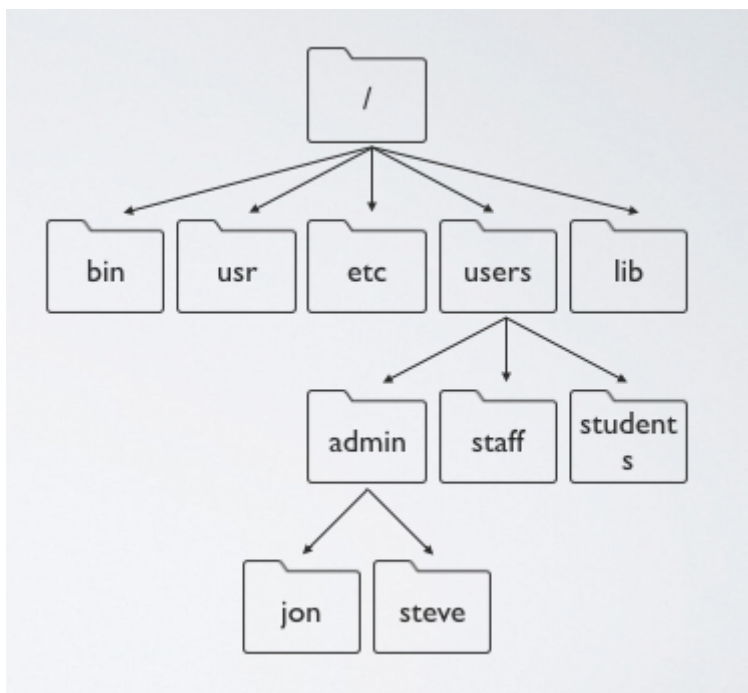


Imagen 3. Estructura de escritorio.
Fuente: Desafío Latam.

Esto quiere decir que el árbol de directorios comienza en la raíz y contiene ramas o directorios, al mismo tiempo que al interior de estos directorios pueden existir archivos u otros directorios.



A veces se utiliza el término técnico **nodo** para referirse a un archivo o un directorio en algún punto del árbol.

El nodo principal de tu computador regularmente se identifica con el símbolo /, también llamado raíz.

Todos los directorios y/o archivos de nuestro computador están dentro de este súper directorio general.



La estructura de directorios puede cambiar dependiendo del sistema operativo con el cual estés trabajando, así que puede que en tu computador no veas los mismos directorios.

Conocer en qué directorio estamos (pwd)

Una de las cosas más importantes que necesitamos saber al trabajar con la terminal, es conocer en qué directorio estamos trabajando. Para ello, existe un comando que nos entregará esta información.

Escribe en tu terminal lo siguiente `pwd` y presiona enter.

```
[MacBook-Pro-de-Cristian:~ adacher$ pwd  
/Users/adacher  
MacBook-Pro-de-Cristian:~ adacher$
```

Imagen 4. Uso de `pwd`.

Fuente: Desafío Latam.

En tu consola se imprimirá la ruta en la cual estás posicionado, esto quiere decir que aparecerá como texto.

Recuerda que siempre que escribamos un comando, necesitaremos presionar la tecla `enter` para que se ejecute.

Listar archivos (ls)

Observa los archivos o directorios que están en tu carpeta raíz o usuario, para ellos utilizaremos el comando `ls` o sea, list, que como su nombre lo indica, hará una lista de los archivos y directorios contenidos en el directorio en el que se está ejecutando el comando. Para usarlo escribe `ls` en tu terminal y presiona enter.

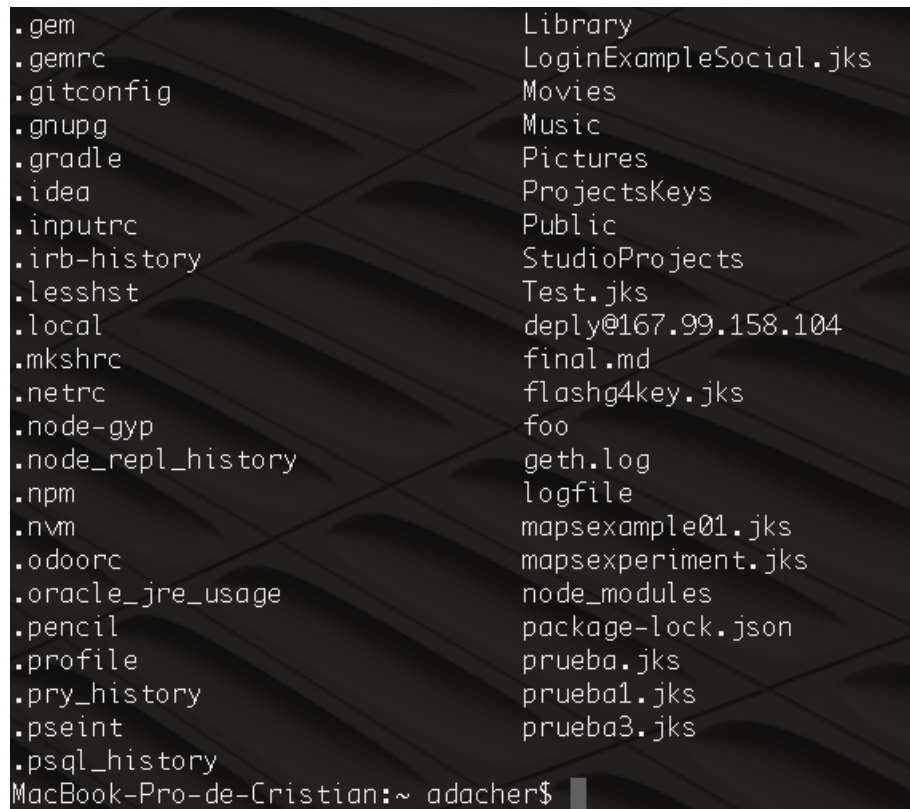
```
Last login: Thu Apr 19 13:10:41 on ttys000  
MacBook-Pro-de-Cristian:~ adacher$ ls  
Adlm                Music                geth.log  
AndroidStudioProjects Pictures              logfile  
Applications         ProjectsKeys          mapsexample01.jks  
Desktop              Public                mapsexperiment.jks  
Documents             StudioProjects        node_modules  
Downloads             Test.jks              package-lock.json  
Dropbox               deploy@167.99.158.104 prueba.jks  
Library               final.md              prueba1.jks  
LoginExampleSocial.jks flashg4key.jks        prueba3.jks  
Movies               foo  
MacBook-Pro-de-Cristian:~ adacher$
```

Imagen 5. Listar archivos.

Fuente: Desafío Latam.

Existe también una opción del comando `ls` que nos permitirá observar los archivos ocultos de un directorio. Los archivos ocultos, son ficheros que tienen un atributo que hace que de forma normal no aparezcan visibles, regularmente son archivos de configuración que no tienen que ser modificados por usuarios normales.

Para poder verlos debemos escribirlo así: `ls -a` el resultado será un listado de directorios y archivos ocultos que comienzan con un punto (.) antes de su nombre:



```
.gem Library
.gemrc LoginExampleSocial.jks
.gitconfig Movies
.gnupg Music
.gradle Pictures
.idea ProjectsKeys
.inputrc Public
.irb-history StudioProjects
.lesshist Test.jks
.local deploy@167.99.158.104
.mkshrc final.md
.netrc flashg4key.jks
.node-gyp foo
.node_repl_history geth.log
.npm logfile
.nvm mapsexample01.jks
.odoorc mapsexperiment.jks
.oracle_jre_usage node_modules
.pencil package-lock.json
.profile prueba.jks
.pry_history prueba1.jks
.pseint prueba3.jks
.psql_history
MacBook-Pro-de-Cristian:~ adacher$
```

Imagen 6. Respuesta `ls -a`.

Fuente: Desafío Latam.

Comandos de navegación entre directorios (cd)

Ahora que sabemos en qué directorio estamos posicionados y qué archivos u otros directorios hay dentro de él, aprenderemos a navegar entre directorios. Para ello, utilizaremos el comando llamado `cd` lo que significa change directory.

Escribe `cd /` y presiona enter, en la terminal. Este comando te lleva a la raíz de tu usuario dentro del computador,, lo que puedes comprobar escribiendo `pwd`.

O sea, si quisiéramos volver al home de nuestro equipo, bastará con escribir `cd /` y clickear enter en la terminal.

Si queremos recorrer e introducirnos en algún otro directorio con `cd` debemos escribir `cd` + la ruta al archivo que queremos llegar. Por ejemplo, accederemos a la carpeta `Desktop` o `Escritorio`, dependiendo de tu sistema operativo, escribiendo:

```
cd Desktop
```

Podemos comprobar que estamos en la carpeta correcta escribiendo `pwd`.

Podemos, nuevamente, escribir `ls` para ver qué archivos o directorios hay dentro de esa carpeta `Desktop` e introducirnos en alguno de ellos con `cd`.

Si ingresamos a un directorio y necesitamos volver atrás, podemos escribir lo siguiente: `cd ..` así seremos dirigidos hacia la carpeta contenedora.

Anatomía de un comando

Todos los comandos tienen un nombre que los distingue, por ejemplo `ls` y `pwd`, serían el nombre del comando.

Algunos como `cd` además tienen uno o más argumentos, por ejemplo `cd carpeta`. En algunos comandos los argumentos son opcionales y en otros obligatorios.

Hay comandos que pueden recibir opciones. Las opciones las especificamos anteponiendo `-` o `--` al igual que con el comando `ls` que ya habíamos realizado, es decir, `ls -a`. En este caso el `-a` da la opción de ver archivos ocultos.



Importante

Linux es sensible a las mayúsculas. Esto implica que es distinto escribir `CD` o `cd`. OSX no lo es, pero tendremos esto en cuenta y para seguir la convención escribiremos todos los comandos en minúsculas.



Si quieres configurar los colores y estilo de tu terminal puedes leer esta [lectura adicional](#).

Manejo de archivos y carpetas

A continuación, veremos algunos comandos útiles al trabajar con archivos y carpetas, que optimizarán nuestro tiempo una vez que adquiramos un poco de práctica.

Creación de archivos

Para crear un archivo nuevo, utilizaremos el comando `touch`. La forma de utilizarlo es `touch nombre_del_archivo.extension`, esto creará un nuevo archivo con el nombre que hayamos ingresado y la extensión que apunta al tipo de archivo.

```
touch archivo.extension
```

Creación de directorios

Para crear un directorio, se utiliza el comando `mkdir`, que significa make directory, de la siguiente manera:

```
mkdir directorio
```

Esto creará un nuevo directorio con el nombre que lo acompañe.

Copia de archivos

Para copiar archivos, se utiliza el comando `cp`, el nombre del comando, luego se añade el archivo que vamos a copiar, y finalmente la ruta donde queremos copiarlo, de la siguiente forma:

```
cp archivo.extension ruta_destino/archivo_nuevo.extension
```

Copia de directorios

Para copiar directorios, también se utiliza el comando `cp`, agregándole la opción `-r`, la carpeta a copiar y la carpeta de destino.

```
cp -r directorio_copiado directorio_destino
```

Mover archivos y directorios

Para mover archivos, se utiliza el comando `mv` que significa move, el cual se utiliza de una forma muy similar que el anterior `cp`.

```
mv archivo.extension directorio_destino/
```

`mv` también nos permite renombrar el archivo que estemos moviendo, indicando el nuevo nombre en la ruta de destino.

Para mover un directorio utilizaremos la misma sintaxis que el comando `cp`:

```
mv directorio_origen directorio_destino
```

Borrar archivos

Otro comando importante que debemos conocer y manejar con mucho cuidado es `rm`, ya que con este comando podremos eliminar un archivo y con una opción en su sintaxis, también directorios completos.

Para eliminar un archivo utilizaremos el siguiente comando:

```
rm archivo.extension
```



¡Importante!

¡Ten mucho cuidado! Ya que los archivos eliminados de esta forma no van a parar en la papelera de reciclaje de tu computador.

Si por algún motivo escribimos el comando y no existe al interior del directorio, obtendremos la siguiente respuesta:

```
No such file or directory
```

Para eliminar un directorio completo, utilizaremos el mismo comando `rm` pero con una opción.

```
rm -r directorio
```

Esto eliminará por completo el directorio, por lo que debemos tener mucho cuidado al utilizar este comando.

Como ves, la terminal es una poderosa herramienta. Existen muchos comandos más que los que revisaremos, pero por el momento con lo aprendido ya puedes moverte libremente por los directorios, verificar en qué carpetas estás y además crear, copiar, mover y eliminar a través de la terminal.



Antes de continuar:

¿Existe algún concepto que no hayas comprendido?

Vuelve a revisar los conceptos que más te hayan costado antes de seguir adelante.

Ejercicio guiado: Manejando archivos y carpetas

Contexto

Como vimos en las páginas anteriores, existen múltiples comandos que nos ayudarán a navegar a través de los archivos y directorios, facilitando nuestro trabajo. Para familiarizarnos con ellos, haremos un recorrido por los comandos que acabamos de revisar:

- **Paso 1:** crea un nuevo directorio con un comando. Para ello, dirígete a la carpeta raíz de tu computador. Si no estás en esta carpeta, ve a ella escribiendo `cd`.

Ahora escribe en la consola:

```
mkdir proyecto1
```

Esto creará un nuevo directorio llamado `proyecto1`.

- **Paso 2:** crea un archivo desde la terminal con el comando `touch`, con el nombre `index.html`:

```
touch index.html
```

Esto crea el archivo `index.html` en la ubicación actual, o sea la carpeta raíz del computador. Si utilizamos `ls` podremos ver el archivo creado.

- **Paso 3:** ahora que ya sabes crear un directorio y un archivo, copia el archivo `index.html` dentro del directorio que habías creado, o sea en la carpeta `proyecto1`, de la siguiente manera:

```
cp index.html /proyecto1/index.html
```

- **Paso 4:** utilizando este mismo comando también puedes cambiarle el nombre a los archivos copiados. Vuelve al directorio anterior con `cd ..`, luego ocuparás `cp` nuevamente pero con un nuevo nombre de archivo.

```
cp index.html /proyecto1/index2.html
```

Si ingresas a `proyecto1` y listamos los archivos con `ls`, debería aparecer el archivo copiado con el nuevo nombre.

- **Paso 5:** si quieres copiar un archivo dentro de la misma carpeta donde se encuentra, solo debes escribir el nombre del archivo seguido del nombre que le quieres poner.

```
cp index.html index3.html
```

Resumen

- El terminal es una herramienta basada en una interfaz de texto que sirve para comunicarse directamente con un computador. Utiliza líneas de comandos para navegar por archivos y directorios, al mismo tiempo se utiliza para interactuar con programas que no tienen interfaz gráfica.
- Los directorios o carpetas de nuestro computador tienen una estructura del tipo árbol, es decir, comienzan en la raíz y contienen ramas o directorios, al mismo tiempo que al interior de estos directorios pueden existir archivos u otros directorios.
- Todos los comandos tienen un nombre que los distingue, por ejemplo `ls` y `pwd`, serían el nombre del comando.