



Ficha de ayuda: Conociendo la consola

Práctica

- `man date` : abre el manual que proporciona más información sobre el comando `date` .
- `clear` : despeja la consola.
- `exit` : cierra la consola.
- `pwd` : me muestra el directorio en el que estoy.
- `ls` : muestra los archivos en el directorio actual.
- `ls -a` : muestra archivos y directorios, incluidos los ocultos.
- `cd first-project` : lleva al directorio `first-project` .
- `cd first-project/qa` : lleva al directorio `qa` en el directorio `first-project` .
- `cd ..` : sube un nivel al directorio padre.
- `cd ../../` : sube dos niveles.
- `cd /` : lleva al directorio raíz.
- `cd ~` : lleva al directorio de inicio.
- `mkdir second-project` : en el directorio actual, crea un directorio llamado `second-project` .
- `rm about.html` : elimina el archivo `about.html` .
- `rmdir images` : elimina el directorio `images` .
- `rm -r second-project` : elimina el directorio `second-project` y todo lo que contiene.
- `touch index.html` : crea un archivo `index.html` en el directorio actual.

- `touch index.html style.css script.js` : si quieres crear varios archivos, puedes introducir sus nombres con un espacio.
- `nano logs/2020/1/apache_2020-01-01.txt` : abre el archivo de texto `apache_2020-01-01.txt` .
- `echo "¿Quién es Morty?"` : muestra el string "¿Quién es Morty?" en la ventana de la terminal.
- `echo "text" > filename` : escribe el string "text" en el archivo `filename` .
- `echo "¿Quién es Morty?" > ~/logs/2020/1/secrets.txt` : escribe el string "¿Quién es Morty?" en el archivo `secrets.txt` .
- `cat ~/logs/2020/1/apache_2020-01-01.txt` : muestra el contenido en la ventana de la terminal.
- `cat a.txt > b.txt` : sobrescribe el contenido del archivo `a.txt` en `b.txt` .
- `cat a.txt >> b.txt` : copia el contenido del archivo `a.txt` al final de `b.txt` .
- `cp brothers.html sisters.html` : copia el archivo `brothers.html` y nombra el nuevo archivo `sisters.html` .
- `cp ../docs/brothers.html ../Documents/` : copia el archivo `brothers.html` en la carpeta `Documents` .
- `mv card.txt ~/` : mueve `card.txt` del directorio actual al directorio de inicio.
- `mv my_app.ssh you_app.ssh` : cambia el nombre del archivo `my_app.ssh` a `you_app.ssh` .
- `grep DELETE apache_2020-01-01.txt` : muestra todos los strings del archivo `apache_2020-01-01.txt` que contienen DELETE.
- `grep -R DELETE ~/logs/2020/1` : muestra todos los strings dentro del directorio que contienen DELETE.
- `grep -n DELETE apache_2020-01-01.txt` : muestra todos los strings y sus números de línea del archivo `apache_2020-01-01.txt` que contienen DELETE.

Para mostrar filas adyacentes, utiliza las claves `B` , `A` y `C` .

Clave	Valor
-B, --before-context	El número de líneas mostradas antes de la línea necesaria
-A, --after-context	El número de líneas mostradas después de la línea necesaria
-C, --context	El número de líneas mostradas antes y después de la línea necesaria

A veces, la palabra DELETE se escribe en un caso diferente o se abrevia como DEL en los logs. En este caso, puedes utilizar lo siguiente:

- La clave `i`, que ignora mayúsculas y minúsculas.
- El carácter de asterisco `*`, que reemplazará cualquier número de caracteres, por ejemplo:

```
grep -i ERR* /test1/test2/test_Logs/Log1.txt .
```

- `[]`: busca coincidencias de patrones. Por ejemplo, este comando `grep -i N[ua]m1 /test1/test2/test_Logs/Log1.txt` significa que el segundo carácter puede ser u o a (Num1, Nam1, num1, nam1).
- `.`: punto, reemplaza cualquier carácter. `grep "204 3.96" ~/logs/2020/1/apache_2020-01-01.txt .`
- `^`: indica el comienzo de un string. El comando mostrará solo aquellos strings que cumplan con el patrón especificado al inicio `grep "^one" text.txt .`
- `$`: indica el final de un string. El comando mostrará solo los strings que tengan el patrón especificado al final. `grep "one$" text.txt .`

Teoría

Un sistema operativo (SO) es un conjunto básico de utilidades que proporciona control sobre el hardware de la computadora.

El **hardware** son los componentes de una computadora; por ejemplo, el procesador, la memoria RAM, el disco duro y la tarjeta de red. Un programa separado, un controlador,

es responsable de cada componente. Cada sistema operativo requiere un recurso de hardware para tener sus propios controladores.

Un sistema operativo puede tener una interfaz gráfica de usuario (GUI) y una interfaz de línea de comandos (CLI).

En términos de uso, los **SO** se dividen en personal y servidor.

Un SO en una computadora personal tiene una interfaz gráfica y aplicaciones para el usuario, como navegadores.

Un SO en el servidor no siempre tiene una interfaz gráfica, pero existen herramientas que ayudan a los usuarios remotos.

Una línea de comandos es una interfaz basada en texto que ayuda a la computadora y al usuario o usuaria a comunicarse a través de comandos de texto. Por ejemplo, puedes navegar por un directorio, crear archivos e instalar programas.