

# La consola git CMD:

La consola de git CMD consiste en una pantalla en negro En la que escribimos comandos que realizan distintas acciones.

## Navegación con lector de pantalla:

Es aconsejable utilizar NVDA, de preferencia con el complemento Speech History.

Con el complemento speech history podemos navegar por el historial de todo lo que NVDA va leyendo, con el comando shift+F11, leemos hacia atrás en el historial, con el comando shift+F12 leemos hacia delante en el historial, y con F12 copiamos el mensaje seleccionado.

Lo anterior nos ayuda al trabajar con la consola, debido a que en ocasiones se complica leer el resultado de los comandos que ejecutamos.

De forma adicional podemos utilizar el cursor de revisión de NVDA o el cursor de jaws para leer los resultados de la consola.

Los comandos para utilizar cada uno cambian según el tipo de teclado, por lo que deberemos buscar los comandos para el lector de pantalla que utilizamos según nuestro teclado ya sea portátil o de escritorio.

En la consola podemos leer el comando carácter por carácter que escribimos navegando con flechas izquierda y derecha.

También podemos ver los comandos ejecutados anteriormente con las flechas arriba y abajo.

Si pulsamos enter sobre algún comando lo ejecutaremos nuevamente.

Con la tecla tab, podremos autocompletar nombres de carpetas o archivos, escribiendo sólo las primeras letras y luego tab nos pondrá el nombre completo.

## Personalizar git CMD

Si ya la tenemos abierta, veamos cómo obtener una interfaz a nuestro gusto mediante sus opciones de personalización.

Para acceder a las opciones, solo tendremos que pulsar el comando alt+espacio, y elegir "Propiedades". Una ventana se abrirá y tendremos varias pestañas para modificar diferentes opciones de la consola.

- Opciones: sirve para modificar el funcionamiento básico de la consola, buffer de comandos, interacciones, selección de texto y tamaño del cursor.
- Fuente: podremos elegir tamaño de fuente y tipo de fuente.
- Diseño: podremos modificar el tamaño de la ventana por defecto.
- Colores: desde aquí podremos modificar el color del fondo de ventana, color del texto y hasta la transparencia del fondo de la ventana.

## Introducir comandos y opciones en CMD (Siempre utilizar la ayuda)

Seguramente todos nos imaginamos cómo utilizar comandos en CMD, pero si es la primera vez que abres la ventana merece la pena tener unas cuantas cosas presentes.

Lo primero, para ejecutar un comando lo único que tendremos que hacer es escribirlo detrás del Prompt, la línea que muestra el directorio donde estamos en cada momento. Tras escribirlo, pulsaremos Enter para que se ejecute, si se muestra un error, significa que lo hemos escrito mal, que no tenemos permiso, o que el comando no existe.

Probemos a ejecutar el comando “help” que nos mostrará todos los comandos disponibles en CMD.

Te aparecerá toda la lista de comandos que puedes ejecutar de forma básica en CMD.

## Comandos útiles para trabajar con la consola de git CMD:

Al escribir el comando help y luego enter, vemos uno muy interesante llamado “CD” que sirve para navegar por los directorios de nuestro sistema. Vamos a ver cómo se utiliza, y aparte echaremos mano nuevamente de help o en este caso /? Debemos de escribirlo detrás del comando.

cd /?

La expresión “/?” se puede utilizar absolutamente con todos los comandos para mostrar su ayuda correspondiente. Es básico en CMD saber esto.

Vemos que aparece una ayuda de cómo utilizar el comando, en ella se muestra que si colocamos “cd..” estaremos saliendo de un directorio, y si escribimos “cd <ruta>” estaremos accediendo a un nuevo fichero.

En este punto, nos va a interesar conocer el comando “dir” para listar los archivos que hay en un directorio, de esta forma podremos saber el nombre de la carpeta donde queremos entrar. Si escribimos dir, se nos muestran todos los ficheros y directorio que hay en la carpeta actual en donde estamos, es decir en “C:\Users\tuUsuario.

Vamos a acceder a nuestra carpeta de documentos y a listar lo que hay. Cuando comencemos a escribir el nombre del archivo o directorio, podremos pulsar la tecla “Tab” para que el nombre se complete automáticamente.

cd Documents

dir

Ahora prueba a ingresar y salir de las diferentes carpetas que tienes en tu carpeta de documentos.

## Copiar y pegar ruta y otros textos

Seguramente si queremos acceder a una ruta compleja de forma directa y no sabemos exactamente cómo se escriben sus directorios, lo más fácil será acceder de forma gráfica y pegar la ruta en CMD. Veamos cómo.

Accedemos al directorio que nos interese en modo gráfico y pulsamos sobre la barra de direcciones para seleccionar la ruta. En este instante pulsamos “Ctrl + C” para copiar.

Ahora nos vamos a CMD nuevamente y solamente tendremos que escribir “cd”, luego pegamos con ctrl+V, Automáticamente se pegará el texto que antes copiamos, si no te funciona pegar con este comando, debes habilitarlo desde las opciones de la consola.

De esta forma también podemos pegar cualquier texto; podemos pegar por ejemplo comandos complejos o identificadores extensos.

Limpiar CMD

Si te das cuenta, tu ventana de consola está llena de los comandos y resultados que has ejecutado.

Con el comando “cls”, limpiaremos la ventana de la consola.

La pantalla se quedará limpia, aunque siempre podremos volver a ver lo anterior si navegamos con las teclas avance y retroceso de página, o con ayuda del ratón.

Mirar el historial de comandos usados en CMD

Cuando llevemos un tiempo utilizando diferentes comandos, quizás queramos repetir con alguno de ellos. CMD cuenta con un historial de comandos utilizados para acceder a ellos sin necesidad de volver a escribirlos. Para hacer esto tendremos dos opciones:

Pulsando las flechas “Arriba” y “Abajo” del teclado podremos seleccionar los últimos comandos utilizados.

Tenemos también un comando para listar este historial.

`doskey /history`

Guardar el resultado en un archivo de texto:

Los resultados que obtenemos al ejecutar comandos en la consola en ocasiones son complejos de leer con lector de pantalla, en estos casos es mejor enviar el resultado a un archivo de texto y leerlo luego con detenimiento.

Para enviar el resultado de algún comando a un archivo de texto, pondremos el comando, luego pondremos el símbolo de mayor que >, y luego el nombre del archivo con la extensión .txt.

`git status > estado.txt`

Con el comando anterior consultamos el estado del repositorio actual, y guardamos el resultado en un archivo llamado estado con extensión .txt.

Para salir de la consola, podemos escribir `exit` o pulsar control+d, Control+d imprime el carácter especial de final de archivo.

Si pulsamos control+c cancelaremos la ejecución de un comando. Útil si, por ejemplo, tarda mucho o queremos interrumpir una operación.

## Comandos para iniciar con git:

`git clone url_repositorio`

Clona el repositorio en nuestro equipo en la carpeta que estamos ubicados.

`git init`

Inicializa un repositorio, en otras palabras, convierte una carpeta cualquiera en un repositorio que podemos seguir con git.

`git add archivo.extensión`

Agrega el archivo especificado a la cola de cambios.

`git add --all`

Agrega todos los archivos modificados a la cola de cambios.

`git add .`

Lo mismo que el comando anterior.

`git commit -m "texto del commit"`

Crea un nuevo commit con los cambios agregados y la descripción que le pusimos.

`git push origin rama_predeterminada.`

Sube los cambios recientes a nuestro repositorio remoto.

`git pull`

Baja los cambios de nuestro repositorio remoto a nuestro repositorio local.