

Actividad 09. Comandos básicos CMD de Windows

¿CÓMO FUNCIONA UNA LÍNEA DE COMANDOS?

Una línea de comandos (a menudo también conocida como consola o terminal) es una **interfaz basada en texto dentro de un sistema operativo** a través de la que los usuarios envían comandos al sistema operativo. De esta manera pueden, por ejemplo, organizarse archivos, iniciarse programas o ejecutarse otros comandos que afectan al sistema operativo, al ordenador o a la red.

En los sistemas operativos más antiguos como, por ejemplo MS-DOS, no podían utilizarse interfaces gráficas de usuario ni se navegaba con ratón. En lugar de ello, tenían que **introducirse todos los comandos**, mostrándose así en la pantalla las estructuras de directorio como texto sin formato. Tras el paso a los sistemas operativos gráficos, la línea de comandos siguió basándose en texto.

Generalmente, la conexión entre el sistema operativo y el usuario, ya sea gráfica (GUI, Graphical User Interface) o basada en texto (CLI, Command Line Interpreter), se engloba en el término **shell**. El CLI estándar se llama en Windows CMD.EXE, pero también existe, por ejemplo, **PowerShell**, que ofrece más posibilidades que CMD. Con PowerShell, por ejemplo, es más fácil crear automatizaciones, para lo cual los usuarios pueden crear programas de scripts con el PowerShell Scripting Language.

¿CÓMO SE ABRE LA TERMINAL DE COMANDOS?

Al hablar de la línea de comandos se hace referencia estrictamente al **programa CMD.EXE**, archivo ejecutable que se encuentra en el directorio de sistema de la carpeta de Windows (si haces clic con el botón derecho del ratón en

el archivo, Windows también te ofrece la posibilidad de iniciar el programa como administrador si dispones de los datos de acceso para ello). Para acceder más rápido al programa, puedes utilizar el explorador o el menú Ejecutar (Run), que en Windows 7,8 y 10 se abre con la combinación de teclas Win+ R. Tras ello basta con introducir “cmd” en el campo de entrada y hacer clic en el botón “OK”.

Desde hace algún tiempo, además, la terminal tiene un lugar fijo en el **menú de inicio** de Windows. En Windows 7 y 10 el programa se encuentra en la carpeta “Accesorios”. En Windows 8 Microsoft ha cambiado el menú de inicio por una pantalla de inicio pero también aquí se encuentra a la consola a través de las apps. Como alternativa, tanto Windows 8 como Windows 10 ofrecen un menú de usuario avanzado (power user) que puede abrirse con la combinación de teclas Win+X. De esta manera, la consola se encuentra tanto en la ejecución normal como en la variante de administrador.

¿CÓMO SE INTRODUCEN LOS COMANDOS EN LA TERMINAL?

CMD funciona con comandos definidos que deben introducirse correctamente, pues de lo contrario la línea de comandos no puede ejecutar la tarea. Lo mismo se aplica a los **parámetros** vinculados a los comandos. Algunos comandos pueden ampliarse con opciones (también llamadas argumentos). Generalmente, estas solo deben contener una consonante y un signo menos (-), un signo más (+) o una barra (/). Asimismo, CMD también permite espaciadores. Para las denominadas wildcards se ha de utilizar el símbolo del asterisco (*) o el de la interrogación (?). Mientras que el último reemplaza a un solo carácter, ya sea consonante, número o carácter, el asterisco representa a muchos caracteres.

Algunos comandos CMD pueden combinarse entre sí. A través de una denominada **pipe**, la salida del primer comando se interpreta como entrada para el siguiente. En la terminal, estos se vinculan entre sí mediante una raya vertical (|). Muchos denominan a esta raya como *pipe*, aunque el nombre oficial es barra vertical (vertical bar). Las rayas verticales conducen los datos de un comando al siguiente:

- comandoA | comandoB

Asimismo, también existe la posibilidad de la **redirection**, en la que la salida de un comando se envía directamente a un archivo (>) o el contenido de un archivo de texto se entiende como entrada para el comando (<):

- Comando > archivo (salida al archivo)
- Comando < archivo (entrada desde archivo)

Por último, los comandos CMD de Windows pueden vincularse de modo que estos **se ejecuten directamente uno detrás de otro**. Esto puede regularse de una manera relativamente sencilla (&), pero también mediante dos variantes más complejas. Por un lado, es posible ejecutar el segundo comando únicamente si el primero se ejecutó con éxito, pero también puede tener lugar lo contrario, es decir, el segundo comando solo se ejecutará si el primero no funciona (||). Ambas opciones pueden combinarse para que una esté disponible:

- comandoA & comandoB (el segundo comando se ejecuta directamente tras el primero)
- comandoA && comandoB (el segundo comando se ejecuta si el primero se ejecutó con éxito)
- comandoA || comandoB (el segundo comando solo se ejecuta si el primero fracasó)

- comandoA && comandoB || comandoC (el segundo comando solo se ejecuta si el primero se ejecutó con éxito. De lo contrario, se ejecutará el tercer comando)

En este sentido, es especialmente útil el hecho de que la línea de comandos recuerde los últimos comandos CMD introducidos. Con las teclas de dirección puedes volver a **llamar a las entradas anteriores**, lo que también significa que estas teclas no sirven para desplazarse hacia arriba y hacia abajo, sino que para ello debe utilizarse el ratón. Si, por el contrario, prefieres el teclado y estás preparado para renunciar a la función de recuperar comandos, puedes realizar los ajustes correspondientes: para ello, haz clic con el botón derecho del ratón en la línea de comandos y activa la opción “Desplazar” (Scroll).

Los comandos, los nombres de archivo o las rutas de acceso se separan entre sí con espacios. No obstante, en la mayoría de los casos es irrelevante el uso de las mayúsculas o de las minúsculas. Que los parámetros se escriban con mayúsculas o minúsculas solo es decisivo en unos pocos casos (generalmente solo cuando se abrevian dos parámetros diferentes con la misma consonante, por lo que se diferencian entre sí en las mayúsculas y las minúsculas).

COMANDOS DE LA CMD PARA LA CMD

Antes de entrar en materia con comandos básicos y comandos avanzados de CMD, conviene conocer algunos de los comandos más útiles para interactuar con la propia consola de Windows.

- **help:** mostrar todos los comandos compatibles con la CMD mediante una lista que detalla todas sus funciones.
- **exit:** cerrar la ventana actual de la CMD.
- **cls:** limpiar las líneas de comando introducidas, así como las respuestas por parte de la máquina.
- **color:** cambia el color de la consola, ya sea el fondo de pantalla o las letras. Puede ser usado con otros comandos para ampliar sus funciones. Por ejemplo, color 1F para cambiar el fondo a azul y las letras a blanco.
- **ver:** comprobar la versión de Windows 10 que tenemos instalada.
- **time:** ver la hora actual del ordenador.
- **date:** mostrar o establecer una fecha nueva.
- **systeminfo:** mostrar la información relativa al hardware de nuestro ordenador, tal como la cantidad de memoria RAM instalada, el modelo de procesador o la cantidad de almacenamiento disponible.
- **logoff:** cerrar la sesión de usuario actual completamente.
- **shutdown:** apagar el ordenador completamente. Junto a otros comandos complementarios, podemos configurar aspectos como el tiempo o la hora exacta de apagado del equipo. Por ejemplo, shutdown -r para reiniciar el ordenador o shutdown -s -t 60 para apagar el ordenador en 60 segundos.

COMANDOS BÁSICOS DE CMD DE WINDOWS

Si somos usuarios neófitos y queremos iniciarnos en el mundo de la CMD con comandos básicos y fáciles de recordar, el programa integra una serie de comandos que nos permiten **crear archivos, cambiar la extensión de un determinado elemento o copiar carpetas** enteras en discos duros y unidades extraíbles, entre otras tantas funciones.

- **dir**: listar el contenido de un determinado directorio o carpeta para visualizar todos los archivos y carpetas alojados en la ruta indicada.
- **cd**: acceder o retroceder a un determinado directorio o carpeta del sistema. Por ejemplo, `cd /fotos` o `cd:` para volver atrás en el directorio o carpeta actual.
- **md**: crear una carpeta con el nombre que se indica junto al comando. Por ejemplo, `md fotos` o `md recuerdos`.
- **type**: crear un archivo con el nombre y la extensión que se indica junto al comando. Por ejemplo, `type cuentas.txt` o `canción.rtf`. También puede ser usado para escribir texto en archivos compatibles. Por ejemplo, `robocopy C:\foo C:\bar /E` o `robocopy C:\foo C:\bar /COPYALL /E /R:0`
- **copy**: copiar uno o varios archivos en la ruta indicada. Por ejemplo, `copy cancion /Musica` o `copy cancion.mp4 /Musica`.

- **robocopy:** copiar uno o varios archivos en la ruta indicada. La diferencia con respecto a copy se basa en la posibilidad de copiar archivos pesados y árboles de directorios, así como detener la copia e incluso cancelarla si así lo deseamos.
- **move:** mover uno o varios archivos en la ruta indicada. Por ejemplo, move cancion /Musica o move cancion.mp4 /Musica.
- **rename:** cambiar el nombre o la extensión de un determinado archivo. Por ejemplo, rename cancion.mp3 cancion.mp4 o rename texto1 texto2.
- **del:** borrar el archivo o la carpeta indicada. Por ejemplo, del cancion o del cancion.mp4.

COMANDOS DE LA CMD PARA DISCOS DUROS Y UNIDADES EXTRAIBLES

Más allá de CHKDSK y DISKPART, existen multitud de comandos que nos permiten interactuar con discos duros internos y unidades de almacenamiento internas, como *pendrives*, tarjetas SD o discos duros externos. Funciones que van desde **reparar una determinada unidad a formatearla** en caso de que esta no funcione correctamente.

- **diskpart**: mostrar y configurar las propiedades de la partición de un determinado disco. Una vez se ha introducido el comando, ha de ser usado junto con otros comandos complementarios. Por ejemplo, clean C para limpiar el disco C, select disk C para seleccionarlo o list disk para ver una lista con todos los discos insertados.
- **diskcomp**: comparar el contenido de dos discos o unidades de almacenamiento. Por ejemplo, diskcomp [C: [E:]].
- **format**: formatear y borrar los datos de una determinada unidad en un formato compatible con Windows. Por ejemplo, format C o format E:.
- **convert**: convertir discos en formato FAT a discos en formato NTFS. Por ejemplo, convert F o convert E:.
- **chkdsk**: analizar una determinada unidad de almacenamiento. Junto a otros comandos suplementarios, es capaz de llevar acciones a cabo como formatear un disco o comprobar los sectores del mismo para repararlos posteriormente. Por ejemplo, chkdsk /F para solucionar los errores de una unidad o chkdsk /R para reparar los sectores defectuosos y recuperar la información.

- **defrag**: desfragmentar un disco duro para ordenar la información y mejorar su rendimiento en caso de tratarse de un disco duro mecánico. Por ejemplo, defrag C o defrag E:.
- **cleanmgr**: analizar y limpiar la unidad principal de Windows para liberar espacio de forma similar a la herramienta Liberador de espacio de Windows. Por ejemplo, cleanmgr C o cleanmgr E:.

COMANDOS DE LA CMD PARA REDES

Otra de las funciones principales que admite CMD tiene que ver con el **uso y el diagnóstico de las redes** con comandos como el conocido IPCONFIG.

- **ipconfig:** mostrar la información de red de nuestro ordenador, tal como la IP de nuestro PC, la de nuestro router o la máscara de subred.
- **netstat:** mostrar la información de red relativa a los protocolos de conexiones TCP/IP y los puertos de conexión.
- **getmac:** mostrar la dirección MAC de nuestro ordenador para conocer la identidad del mismo en red.
- **tracert:** mostrar la ruta de nuestro ordenador a la hora de enviar un determinado paquete a la dirección host que le indiquemos para verificar el estado y la seguridad de la conexión. Puede ser usado junto con otros comandos para expandir sus funciones. Por ejemplo, tracert 11.1.0.1, tracert -h 4 11.1.0.1 o tracert google.com.

COMANDOS AVANZADOS DE CMD PARA WINDOWS

Además de los comandos que acabamos de ver, existe una gran cantidad de instrucciones que nos permiten acceder a funciones concretas de la CMD. También podemos hacer uso de estas mismas instrucciones para **crear pequeños programas pegando el código en un archivo BAT** para realizar acciones de manera automática, comúnmente conocidos como archivos o programas por lotes.

- **call [X]**: llamar a un programa por lotes desde otro programa
- **chcp**: mostrar o establecer el número de página de códigos.
- **chdir [X]**: mostrar o cambiar el nombre del directorio actual.
- **chkntfs [X]**: mostrar o cambiar la comprobación de disco al arrancar el sistema.
- **cmd**: inicia una nueva ventana de CMD de Windows.
- **compact [X]**: mostrar o cambiar el estado de compresión de archivos en particiones con formato NTFS.
- **doskey**: edita, recuperar y crear líneas de comando para la CMD de Windows.
- **echo «X»**: lanzar mensajes y cadenas de texto en la CMD.
- **erase [X]**: borrar uno o varios archivos.
- **fc [X]**: compara dos o más archivos y mostrar las diferencias entre ellos. Es menos completo que compare.
- **find [X]**: busca un mensaje en uno o varios archivos a la vez.
- **findstr [X]**: busca una cadena de texto en uno o varios archivos a la vez.

- **for**: parámetro empleado para la ejecución de bucles en archivos ejecutables, el cual ejecuta una serie de líneas tantas veces como se le indique siempre y cuando se cumple la condición impuesta.
- **if**: parámetro empleado para la ejecución de bucles en archivos ejecutables, el cual ejecuta una serie de líneas si se cumple una determinada condición.
- **label [X]**: crear, borrar o cambiar la etiqueta de volumen de un determinado disco.
- **mode [X]**: configurar un dispositivo de sistema.
- **openfiles**: mostrar todos los archivos abiertos que han sido compartidos con otros usuarios
- **pause**: suspender la ejecución de un archivo por lotes.
- **print [X]**: imprimir un determinado archivo de texto por pantalla.
- **rd [X]**: borrar un determinado directorio.
- **ren [X]**: cambiar el nombre de uno o más archivos a la vez.
- **replace [X]**: reemplazar archivos.
- **rmdir [X]**: elimina una determinada carpeta o directorio.
- **sc**: mostrar los servicios del sistema que corren en segundo plano.
- **title [X]**: establecer el título de la ventana de CMD.
- **tree [X]**: mostrar la estructura de directorios de una determinada unidad o ruta.
- **type [X]**: mostrar el contenido de uno o varios archivos de texto con formato TXT, RTF, BAT...
- **verify [X]**: comprobar que los archivos se escriben de forma correcta en un determinado disco.

- **vol [X]**: mostrar la etiqueta de un determinado volumen y el número de serie del disco elegido.
- **xcopy**: copiar archivos y árboles de directorios.

Se pide:

1. Estudiar y utilizar los comandos cmd en Windows
2. Realizar los ejercicios del documento **Ejercicios de uso de la terminal CMD Windows**