



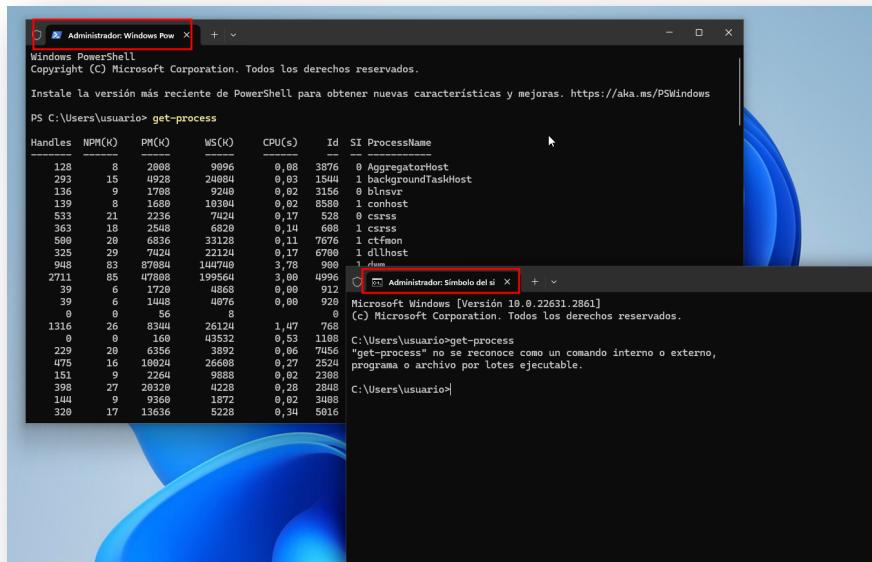
Comandos del Sistema de Ficheros Windows

La gestión de carpetas y ficheros desde el explorador de archivos de Windows es algo que ya debemos dar por supuesto. El objetivo de este documento es saber hacer lo mismo desde una interfaz de comandos. Gracias a ello podremos incluir estas operaciones de archivos y carpetas dentro de scripts o programas con el fin de automatizar una secuencia de acciones.

El entorno CLI (Command Line Instructions) es mucho más duro que una interfaz gráfica en el sentido de que requiere un mayor conocimiento de la estructura de archivos. Windows no es sensible a las mayúsculas por lo que un comando se puede escribir indiferentemente en minúsculas, mayúsculas o en cualquier combinación de estas.

Para empezar abriremos una terminal de comandos. En Windows 11 tenemos dos tipos de interfaz de comandos. La más antigua es la llamada "Símbolo de sistema" y consiste en un emulador del antiguo sistema monousuario y monotarea MSDOS. La versión más moderna se llama "PowerShell" que, además de incluir todas las herramientas del "Símbolo de sistema" incluye otras relacionadas con las nuevas habilidades que proporciona Windows como sistema operativo multiusuario y multitarea.

Por ejemplo, si queremos obtener información de los procesos que se están ejecutando en el sistema usaremos PowerShell porque en los tiempos de MSDOS (monotarea) no existían estos comandos.

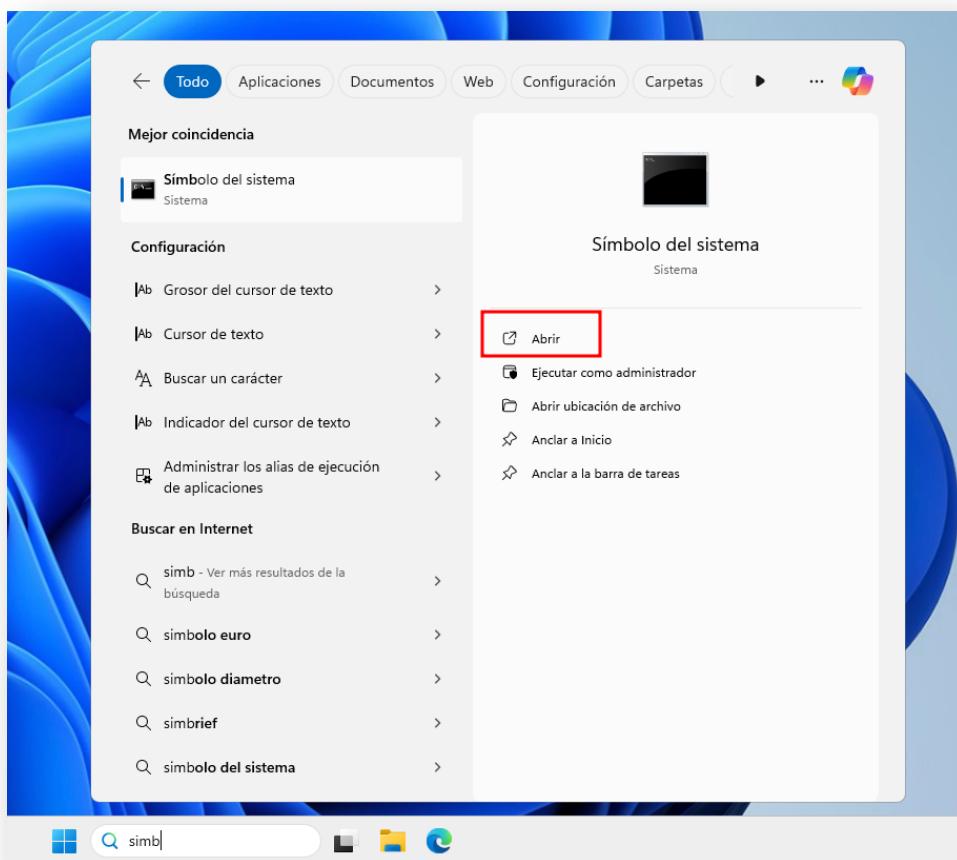




En la captura anterior puedes ver como la ejecución del mismo comando funciona en Powershell y nos ofrece la información de los procesos activos, y en cambio en el símbolo de sistema no se reconoce el comando.

Los comandos que vamos a usar están relacionados con archivos y carpetas que ya existían y se gestionaban con MSDOS por lo que todos los comandos funcionarán en el símbolo de sistema.

Empezaremos abriendo un terminal de símbolo de sistema. Hay múltiples maneras de hacerlo, la más sencilla consiste en usar la barra de búsqueda y escribir en ella “Símbolo de sistema”. A medida que escribas, el buscador irá delimitando los contenidos que te ofrece y rápidamente llegarás al lanzador.

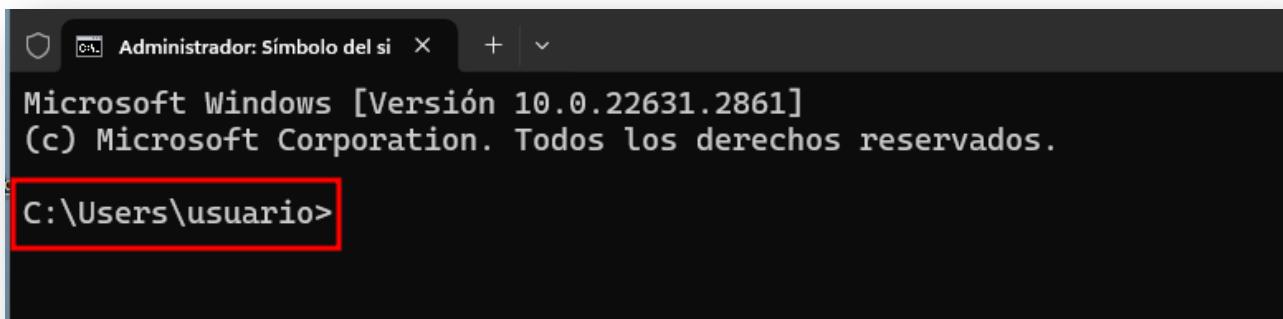


Una vez abierta la terminal encontraremos la información de la versión y una línea de “prompt” que nos indica que el sistema está listo para recibir un nuevo comando.

Al estar usando la terminal de símbolo de sistema debemos recordar que estamos emulando un sistema operativo monotarea por lo que si lanzamos un proceso pesado perderemos el “prompt” y con ello la posibilidad de ejecutar otro comando mientras no termine la ejecución del primero.



Por defecto, Windows especifica en el “prompt” la unidad de disco y la ruta del directorio actual, todo ello acabado con el símbolo “>”. Con ello sabemos que todos los comandos que ejecutemos tendrán efecto en esa unidad y ruta concreta.



```
Administrator: Símbolo del sistema + ▾
Microsoft Windows [Versión 10.0.22631.2861]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\usuario>
```

En esta captura el “prompt” nos está diciendo que estamos en el disco C: y dentro de este disco en la carpeta “Users” y subcarpeta “usuario”.

DIR

Este comando nos lista el contenido de la ruta activa indicada en el “prompt”.



```
C:\Users\usuario>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 02DE-EC24

Directorio de C:\Users\usuario

23/10/2024 12:35    <DIR>          .
25/04/2024 13:22    <DIR>          ..
25/04/2024 13:18    <DIR>          Contacts
25/04/2024 13:18    <DIR>          Desktop
25/04/2024 13:18    <DIR>          Documents
18/10/2024 15:03    <DIR>          Downloads
25/04/2024 13:18    <DIR>          Favorites
25/04/2024 13:18    <DIR>          Links
25/04/2024 13:18    <DIR>          Music
25/04/2024 13:19    <DIR>          OneDrive
25/04/2024 13:18    <DIR>          Pictures
18/10/2024 14:59    <DIR>          prueba
23/10/2024 12:35          12 prueba.txt
25/04/2024 13:18    <DIR>          Saved Games
25/04/2024 13:22    <DIR>          Searches
25/04/2024 13:18    <DIR>          Videos
                           1 archivos           12 bytes
                           15 dirs   77.863.960.576 bytes libres

C:\Users\usuario>
```



Un ejemplo lo podemos ver en esta captura anterior. Aquí vemos la información en forma de tabla donde cada fila es un archivo o subcarpeta y cada columna indica alguna información como fecha y hora de creación, indicación de si es una carpeta, tamaño y nombre.

Siempre tendremos dos líneas especiales nombradas con un punto (.) y con dos puntos seguidos (..). El punto hace referencia a esta carpeta activa y los dos puntos seguidos hacen referencia a su carpeta padre. En el caso de esta captura, el punto equivale a "C:\Users\usuario" y los dos puntos seguidos equivalen a "C:\Users"

Los sistemas operativos siempre usan rutas absolutas donde se indica todo el camino desde el principio de la unidad. Si pensamos en una estructura de árbol, una ruta absoluta nos indicará el camino desde la raíz hasta una rama u hoja indicando explícitamente qué desvío tomar en cada nudo.

En ocasiones, cuando la ruta es larga y queremos simplificarla o bien cuando queremos ofuscar parte de la ruta absoluta porque contiene información que no queremos que se vea, utilizamos las rutas relativas. Con estas rutas la diferencia consiste en que el origen ya no es la raíz sino lo indicado por el o los puntos iniciales.

En general cuando una ruta de Windows empieza por una letra de unidad como por ejemplo "C:" sabemos que lo que sigue es una ruta absoluta que nos indicará todo el camino desde la raíz del disco C:. Por otro lado, cuando no especificamos explícitamente la unidad de disco, estamos usando una ruta relativa desde la carpeta actual en la que nos encontramos, es decir, el punto.

El comando "dir" acepta muchos parámetros. Podemos acceder a la lista completa de opciones usando el parámetro "/"?

```
C:\Users\usuario>dir /?
Muestra la lista de subdirectorios y archivos de un directorio.

DIR [drive:][:path][filename] [/A[:|:lattributes]] [/B] [/D] [/L] [/N]
[/O[[:]sortorder]] [/P] [/Q] [/R] [/S] [/T[[:]timefield]] [/W] [/X] [/4]

[drive:][:path][filename]
  Especifica la unidad, el directorio o los archivos que se mostrarán.

/A      Muestra los archivos con los atributos especificados.
Atributos   D Directarios          R Archivos de solo lectura
            H Archivos ocultos       A Archivos listos para el archivado
            S Archivos de sistema    I Archivos indizados que no son de contenido
            L Puntos de reanálisis     O Archivos sin conexión
            - Prefijo de exclusión

/B      Usa el formato simple (sin encabezados ni resúmenes).
/C      Muestra el separador de miles en los tamaños de los archivos. Es la opción
        predeterminada. Usa /-C para deshabilitar la aparición del separador.
/D      Similar al listado ancho, pero los archivos aparecen en una lista ordenada por columnas.
/L      Usa letras minúsculas.
/N      Nuevo formato de lista larga donde los nombre de archivo aparecen en el lado derecho.
/O      Muestra los archivos según el orden indicado.
        N Por nombre (orden alfabético)      S Por tamaño (orden creciente)
        E Por extensión (orden alfabetico)  D Por fecha y hora (el más antiguo primero)
        G Agrupa primero los directorios - Prefijo para invertir el orden
/P      Hace una pausa después de cada pantalla completa de información.
/Q      Muestra el propietario del archivo.
/R      Muestra las secuencias alternativas de datos del archivo.
/S      Muestra los archivos del directorio especificado y todos sus subdirectorios.
/T      Controla el campo de hora que se mostrará o usará para realizar la ordenación.
        C Creación
        A Último acceso
        W Última modificación
/W      Usa el formato de listado ancho.
/X      Muestra los nombres cortos generados para los nombres
        de archivo sin formato 8dot3. El formato es el mismo que para /N, con el nombre corto especificado
        antes del nombre largo. Si no existe ningún nombre corto, se muestran
        espacios en blanco en su lugar.

Presione una tecla para continuar . . . |
```



Si probamos el parámetro "/a" veremos más archivo y carpetas que antes estaban ocultos.

```

Administrator: Símbolo del sistema + 
El número de serie del volumen es: 02DE-EC24

Directorio de C:\Users\usuario

23/10/2024 12:35   <DIR>       .
25/04/2024 13:22   <DIR>       ..
25/04/2024 13:18   <DIR>       AppData
25/04/2024 13:18   <JUNCTION> Configuración local [C:\Users\usuario\AppData\Local]
25/04/2024 13:18   <DIR>       Contacts
25/04/2024 13:18   <JUNCTION> Cookies [C:\Users\usuario\AppData\Local\Microsoft\Windows\INetCookies]
25/04/2024 13:18   <JUNCTION> Datos de programa [C:\Users\usuario\AppData\Roaming]
25/04/2024 13:18   <DIR>       Desktop
25/04/2024 13:18   <DIR>       Documents
18/10/2024 15:03   <DIR>       Downloads
25/04/2024 13:18   <JUNCTION> Entorno de red [C:\Users\usuario\AppData\Roaming\Microsoft\Windows\Network Shortcuts]
25/04/2024 13:18   <DIR>       Favorites
25/04/2024 13:18   <JUNCTION> Impresoras [C:\Users\usuario\AppData\Roaming\Microsoft\Windows\Printer Shortcuts]
25/04/2024 13:18   <DIR>       Links
25/04/2024 13:18   <JUNCTION> Menú Inicio [C:\Users\usuario\AppData\Roaming\Microsoft\Windows\Start Menu]
25/04/2024 13:18   <JUNCTION> Mis documentos [C:\Users\usuario\Documents]
25/04/2024 13:18   <DIR>       Music
18/10/2024 15:20   1.572.864 NTUSER.DAT
25/04/2024 13:18   221.184 ntuser.dat.LOG1
25/04/2024 13:18   495.616 ntuser.dat.LOG2
25/01/2024 13:18   65.536 NTUSER.DAT{52f8dc78-cdd7-11ec-8680-00224848b4d3}.TM.blf
25/04/2024 13:18   524.288 NTUSER.DAT{52f8dc78-cdd7-11ec-8680-00224848b4d3}.TMContainer000000000000000000000001.regtrans-ms
25/04/2024 13:18   524.288 NTUSER.DAT{52f8dc78-cdd7-11ec-8680-00224848b4d3}.TMContainer000000000000000000000002.regtrans-ms
25/04/2024 13:18   20 ntuser.ini
25/04/2024 13:19   <DIR>       OneDrive
25/04/2024 13:18   <DIR>       Pictures
25/04/2024 13:18   <JUNCTION> Plantillas [C:\Users\usuario\AppData\Roaming\Microsoft\Windows\Templates]
18/10/2024 14:59   <DIR>       prueba
23/10/2024 12:35   12 prueba.txt
25/04/2024 13:18   <JUNCTION> Reciente [C:\Users\usuario\AppData\Roaming\Microsoft\Windows\Recent]
25/04/2024 13:18   <DIR>       Saved Games
25/04/2024 13:22   <DIR>       Searches
25/04/2024 13:18   <JUNCTION> SendTo [C:\Users\usuario\AppData\Roaming\Microsoft\Windows\SendTo]
25/04/2024 13:18   <DIR>       Videos
     8 archivos      3.403.888 bytes
    26 dirs    77.991.194.624 bytes libres

C:\Users\usuario>

```

Es interesante que probemos alguno por nuestra cuenta, no por memorizar y sí por conocer las posibilidades que ofrece el sistema operativo.

Además de estos parámetros también podemos usar filtros. Veremos el uso del asterisco (*) que interpretaremos como cualquier cadena (incluida la vacía).

La siguiente captura muestra el resultado del comando "dir" con 3 filtros distintos.



```
C:\Users\usuario>dir *.txt
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 02DE-EC24

Directorio de C:\Users\usuario

23/10/2024 12:35           12 prueba.txt
              1 archivos          12 bytes
              0 dirs   77.991.194.624 bytes libres

C:\Users\usuario>dir *rue*
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 02DE-EC24

Directorio de C:\Users\usuario

18/10/2024 14:59      <DIR>        prueba
23/10/2024 12:35           12 prueba.txt
              1 archivos          12 bytes
              1 dirs   77.991.194.624 bytes libres

C:\Users\usuario>dir prueb*
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 02DE-EC24

Directorio de C:\Users\usuario

18/10/2024 14:59      <DIR>        prueba
23/10/2024 12:35           12 prueba.txt
              1 archivos          12 bytes
              1 dirs   77.991.194.624 bytes libres

C:\Users\usuario>
```

En el primer caso estamos buscando archivos o carpetas que acaben con la cadena ".txt", en el segundo caso estamos buscando archivos o carpetas que contengan la subcadena "rue" en cualquier posición. Por último, estamos buscando archivos o carpetas que empiecen por la subcadena "prueb".

Es interesante fijarnos cómo en estas ejecuciones del comando "dir" no hemos puesto ninguna ruta, ni absoluta ni relativa. Lo que ha hecho el sistema operativo es usar por defecto el punto para indicar que debemos ejecutar sobre la carpeta actual.

En la siguiente captura podemos ver el mismo resultado del comando "dir" buscando archivo con extensión "txt" pero usando la ruta por defecto, una ruta relativa y una ruta absoluta.



```

Administrator: Símbolo del sistema + ▾
C:\Users\usuario>dir *.txt
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 02DE-EC24

Directorio de C:\Users\usuario

23/10/2024 12:35           12 prueba.txt
               1 archivos          12 bytes
                  0 dirs  77.992.452.096 bytes libres

C:\Users\usuario>dir .\*.txt
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 02DE-EC24

Directorio de C:\Users\usuario

23/10/2024 12:35           12 prueba.txt
               1 archivos          12 bytes
                  0 dirs  77.992.452.096 bytes libres

C:\Users\usuario>dir c:\Users\usuario\*.txt
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 02DE-EC24

Directorio de c:\Users\usuario

23/10/2024 12:35           12 prueba.txt
               1 archivos          12 bytes
                  0 dirs  77.992.452.096 bytes libres

C:\Users\usuario>

```

En el primer caso no se aprecia ni letra de unidad ni puntos por lo que el sistema operativo interpreta que hay que usar un punto y lo hace internamente.

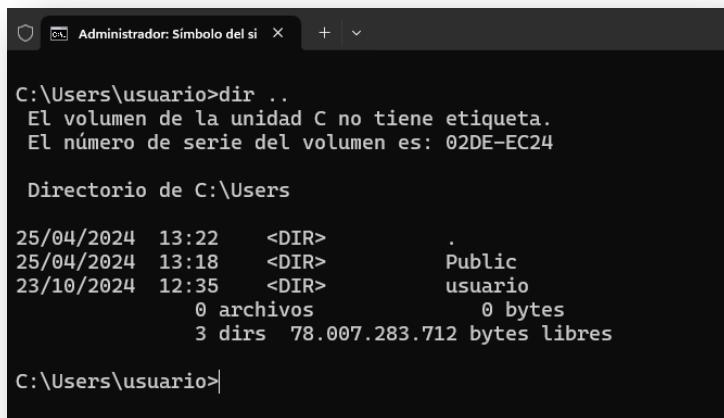
En el segundo caso hay una ruta relativa. Lo sabemos porque la parte de la ruta empieza por punto indicando este mismo directorio y a continuación el filtro.

En el tercer caso tenemos la ruta absoluta de la carpeta. Lo sabemos porque la parte de la ruta empieza por una unidad de disco, c:.

Usar un método u otro para indicar una ruta dependerá de lo que queremos hacer. En unos casos será imprescindible hacer referencia a una ruta absoluta. Por ejemplo cuando nos referimos a algún archivo de configuración del sistema operativo que se encuentra obligatoriamente en una determinada ruta. En otros casos será mucho más interesante usar rutas relativas. Por ejemplo, en una estructura de carpetas de un proyecto de desarrollo es mucho más interesante entrar en las subcarpetas teniendo como origen la carpeta principal del proyecto. Es muy probable que el proyecto pase por distintos usuarios que lo almacenarán en rutas distintas. Si pensamos en la captura anterior, deberíamos entender que el primer y segundo comando funcionarán siempre independientemente de tu nombre de usuario, son rutas relativas que ponen el origen en la carpeta actual, cualquiera que sea esta. En cambio, el tercer comando tiene una ruta absoluta que solo funcionará si en tu equipo hay un usuario llamado "usuario".



Las rutas relativas también pueden empezar por los dos puntos seguidos (..) y en ese caso estarían haciendo referencia a la carpeta padre.



```
C:\Users\usuario>dir ..
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 02DE-EC24

Directorio de C:\Users

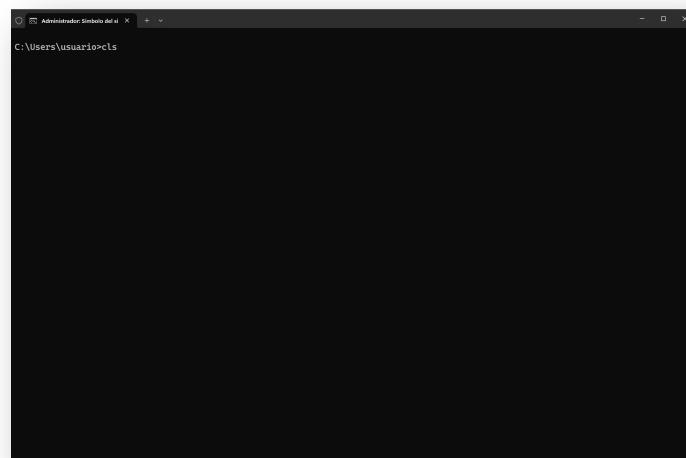
25/04/2024 13:22    <DIR>          .
25/04/2024 13:18    <DIR>          Public
23/10/2024 12:35    <DIR>          usuario
              0 archivos            0 bytes
              3 dirs    78.007.283.712 bytes libres

C:\Users\usuario>
```

En esta captura anterior vemos como desde la carpeta de usuario hemos ejecutado el comando "dir" con una ruta relativa. Lo que queremos ver con ese comando es un listado de archivos y carpetas contenidos en la carpeta padre, que en este caso es "C:\Users"

CLS

A esta altura puede que ya tengamos la terminal con demasiada información y queremos limpiar la pantalla. Este comando limpia la terminal dejándola solamente con una línea de "prompt".

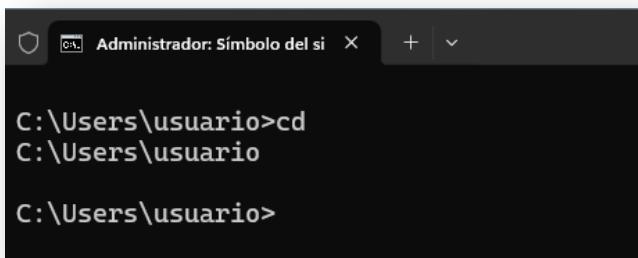


```
C:\Users\usuario>cls
```



CD

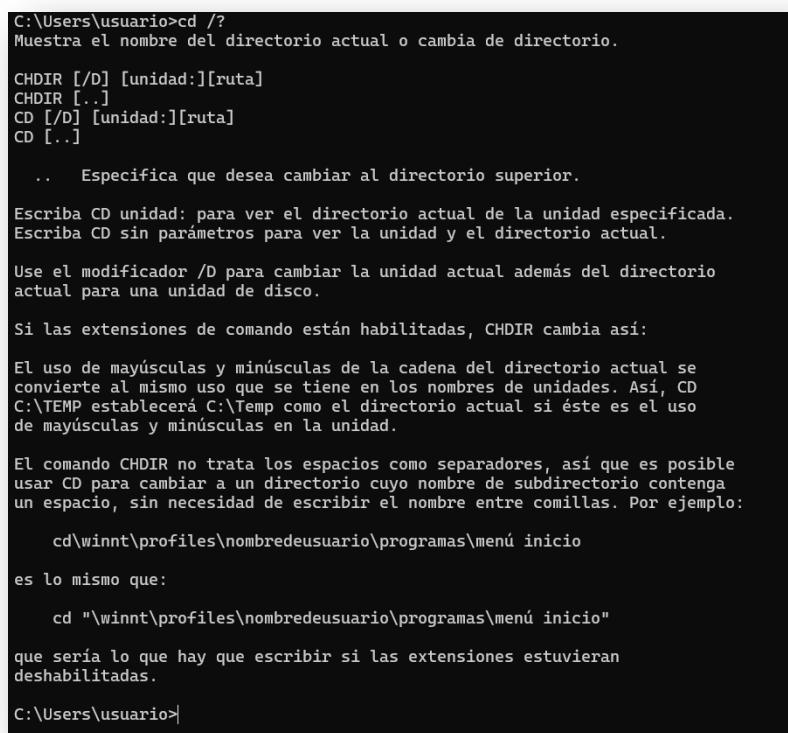
El comando "cd" tiene varios usos. Si lo utilizamos sin ningún parámetro nos devolverá la ruta absoluta actual en la que nos encontramos.



```
C:\Users\usuario>cd
C:\Users\usuario

C:\Users\usuario>
```

El comando "cd" admite algunos parámetros así que consultamos la ayuda usando "/" para obtener más información sobre estos.



```
C:\Users\usuario>cd /?
Muestra el nombre del directorio actual o cambia de directorio.

CHDIR [/D] [unidad:][ruta]
CHDIR [...]
CD [/D] [unidad:][ruta]
CD [...]

... Especifica que desea cambiar al directorio superior.

Escriba CD unidad: para ver el directorio actual de la unidad especificada.
Escriba CD sin parámetros para ver la unidad y el directorio actual.

Use el modificador /D para cambiar la unidad actual además del directorio
actual para una unidad de disco.

Si las extensiones de comando están habilitadas, CHDIR cambia así:

El uso de mayúsculas y minúsculas de la cadena del directorio actual se
convierte al mismo uso que se tiene en los nombres de unidades. Así, CD
C:\TEMP establecerá C:\Temp como el directorio actual si éste es el uso
de mayúsculas y minúsculas en la unidad.

El comando CHDIR no trata los espacios como separadores, así que es posible
usar CD para cambiar a un directorio cuyo nombre de subdirectorios contenga
un espacio, sin necesidad de escribir el nombre entre comillas. Por ejemplo:

    cd\winnt\profiles\nombredeusuario\programas\menú inicio
es lo mismo que:

    cd "\winnt\profiles\nombredeusuario\programas\menú inicio"
que sería lo que hay que escribir si las extensiones estuvieran
deshabilitadas.

C:\Users\usuario>
```

Antes de continuar, es importante prestar atención a la ayuda. Desde que Windows deja usar nombres largos para archivos y carpetas también es posible usar el carácter espacio en blanco como parte del nombre. Esto es un inconveniente para el símbolo de sistema porque aquí se usa el espacio para separar comandos o parámetros. Encontrar un espacio en blanco en un nombre supone un problema. Para lidiar



```
C:\Users\usuario\prueba>cd ..\desktop  
C:\Users\usuario\Desktop>
```

con ello podemos encerrar la ruta entre comillas y así será interpretada en conjunto. Como podemos leer en la ayuda, disponemos de otro comando llamado "chdir" que no interpreta los espacios como un separador de parámetros. Lo más habitual es usar las comillas o escapar el espacio en blanco.

En esencia, usaremos el comando "cd" para cambiar de directorio indicando la ruta absoluta o relativa a la que queremos ir.

En nuestro ejemplo, si quisiera entrar dentro de la subcarpeta "prueba" tendría que escribir el comando "cd" y la ruta para llegar hasta la subcarpeta "prueba".

Como estoy en la carpeta inmediatamente anterior podría poner directamente el nombre de la carpeta. Recordemos que si Windows no detecta ni nombre de unidad, ni puntos, asumirá que es una ruta relativa partiendo de la carpeta activa, es decir, usando el punto.

También podríamos poner la ruta absoluta en la que tendríamos que indicar todo el camino desde la raíz de la unidad de disco hasta la subcarpeta.

Fijémonos como al cambiar de carpeta también cambia el "prompt" para mantenernos informados de cuál es la carpeta activa en cada momento.

Para el resto del documento vamos a cambiar de carpeta y nos vamos a situar dentro de la subcarpeta "Desktop" que todos tenemos dentro de la carpeta de nuestro usuario. Recordemos que Windows, durante su instalación, crea una carpeta llamada "Users" donde almacena las carpetas personales de los usuarios de su sistema. En mi caso, la ruta absoluta a la que quiero ir es "C:\Users\usuario\Desktop"

```
C:\Users\usuario\prueba>cd ..\desktop  
C:\Users\usuario\Desktop>
```

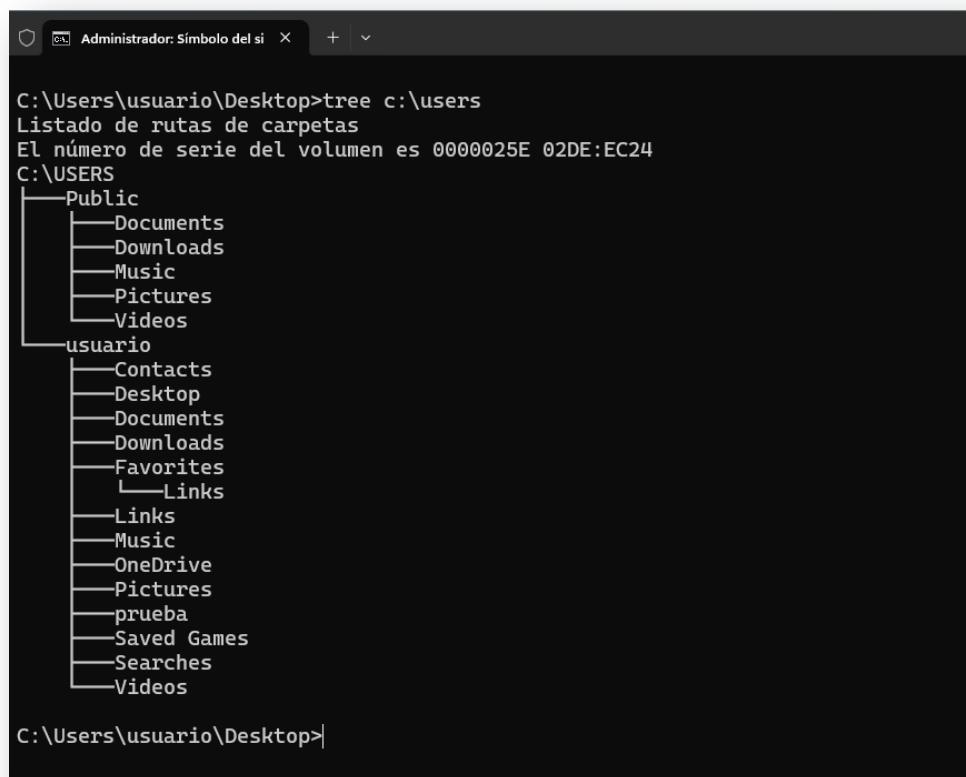
En la captura anterior podemos ver como se ha utilizado una ruta relativa cuyo origen es el padre de la carpeta actual. Es decir, hemos ido a "c:\users\usuario" con los dos puntos seguidos y a continuación a la carpeta "desktop".



Esta carpeta es la que toma Windows para representar el "Escritorio" de cada usuario. Así que desde ahora, todos los comandos que efectuemos en esta carpeta se verán gráficamente representados en el escritorio de Windows.

TREE

No es que sea un comando fundamental en Windows pero hay ocasiones en las que cuesta recordar una ruta o moverte por las carpetas con una organización complicada. El comando "tree" nos representa las subcarpetas una vista en forma de árbol partiendo desde la ruta que le indiquemos como origen.



```
C:\Users\usuario\Desktop>tree c:\users
Listado de rutas de carpetas
El número de serie del volumen es 0000025E 02DE:EC24
C:\USERS
└── Public
    ├── Documents
    ├── Downloads
    ├── Music
    ├── Pictures
    └── Videos
└── usuario
    ├── Contacts
    ├── Desktop
    ├── Documents
    ├── Downloads
    ├── Favorites
    │   └── Links
    ├── Links
    ├── Music
    ├── OneDrive
    ├── Pictures
    ├── prueba
    ├── Saved Games
    ├── Searches
    └── Videos

C:\Users\usuario\Desktop>
```

En la captura anterior le he pedido al comando "tree" que me muestre el árbol de subdirectorios que tienen a "C:\Users" como directorio origen. Con esta ayuda podríamos escribir las rutas relativas con mayor facilidad. ¿Podrías escribir la ruta relativa que nos lleva desde "links" como origen hasta "Documents" de la carpeta "Public"?...

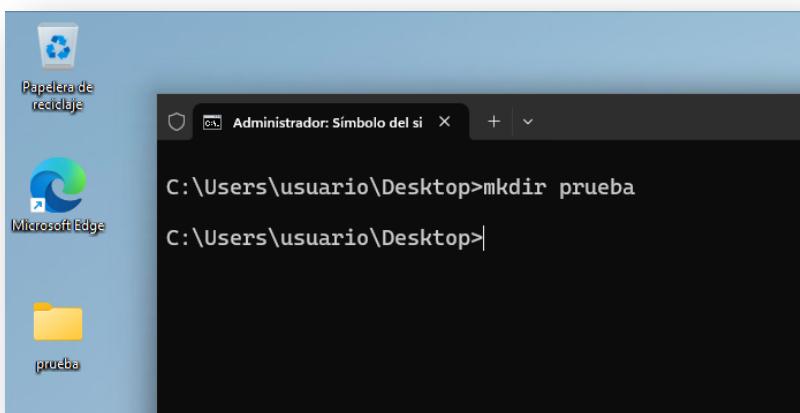
MKDIR

Este comando es muy sencillo. A estas alturas ya deberíamos saber consultar su ayuda llamando al parámetro "?". Este comando crea una nueva carpeta de la que tenemos que indicar su ruta.

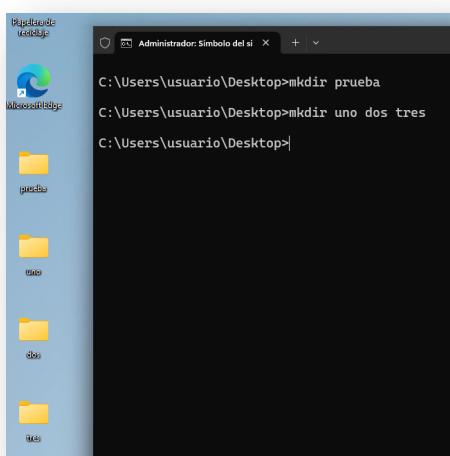


Nuevamente aquí jugamos con las distintas posibilidades para escribir rutas que nos ofrece Windows. Recordemos que estas pueden ser absolutas o relativas y que si no indicamos nada explícitamente tomará el punto como inicio de una ruta relativa.

Con el comando “mkdir prueba” estoy indicando al sistema operativo que cree una nueva carpeta en la ruta indicada. Como la ruta no empieza por unidad de disco ni por puntos tomaremos que implícitamente es una ruta absoluta que empieza por punto, es decir, desde la carpeta activa. El resultado es que nos crea la carpeta “prueba” en el “Escritorio” y tan pronto como se ejecuta el comando y nos devuelve el “prompt”, ya podemos verlo en la interfaz gráfica del escritorio.

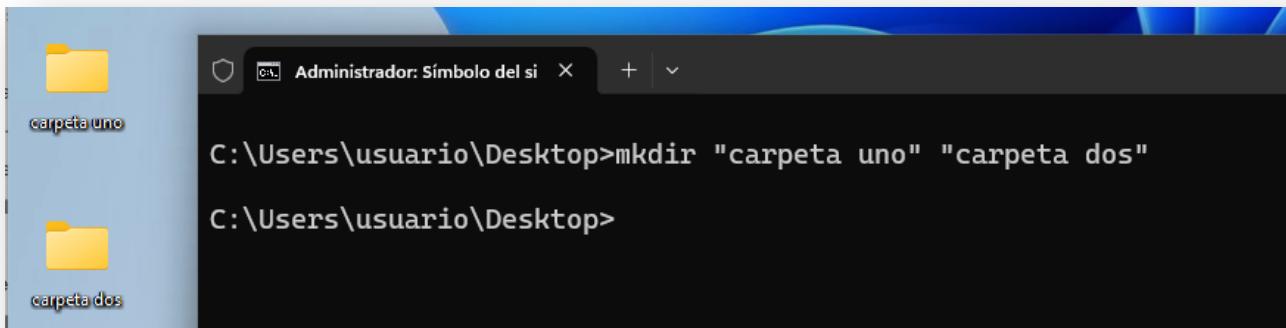


Con el comando mkdir también podemos crear varias carpetas a la vez. Simplemente tenemos que separar con un espacio el nombre de cada carpeta. En la siguiente captura veremos como creamos tres carpetas distintas usando una sola línea de comando.

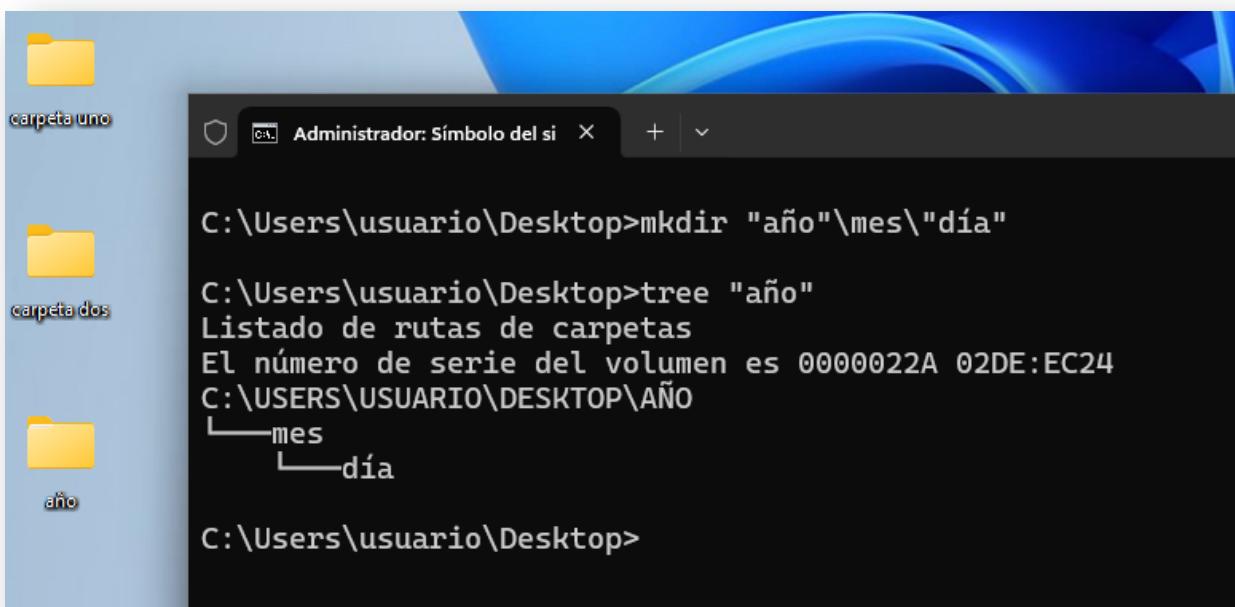




Si queremos crear carpetas con nombres largos en los que se incluyan espacios en blanco debemos recordar el problema asociado y su solución usando comillas para las cadenas de texto. En la siguiente captura crearemos dos carpetas con nombres que incluyen espacios en blanco. Para evitar que el sistema operativo piense que son varias carpetas lo que hacemos es encerrar entre comillas los nombres.



Otra utilidad del comando mkdir es que puede crear las subcarpetas incluidas en la ruta. Por ejemplo, en un solo comando puedo crear una carpeta llamada “año” (debo utilizar comillas por contener la ñ) y dentro una subcarpeta llamada “mes” y dentro de esta otra subcarpeta llamada “día” (también con comillas por el acento de la i)

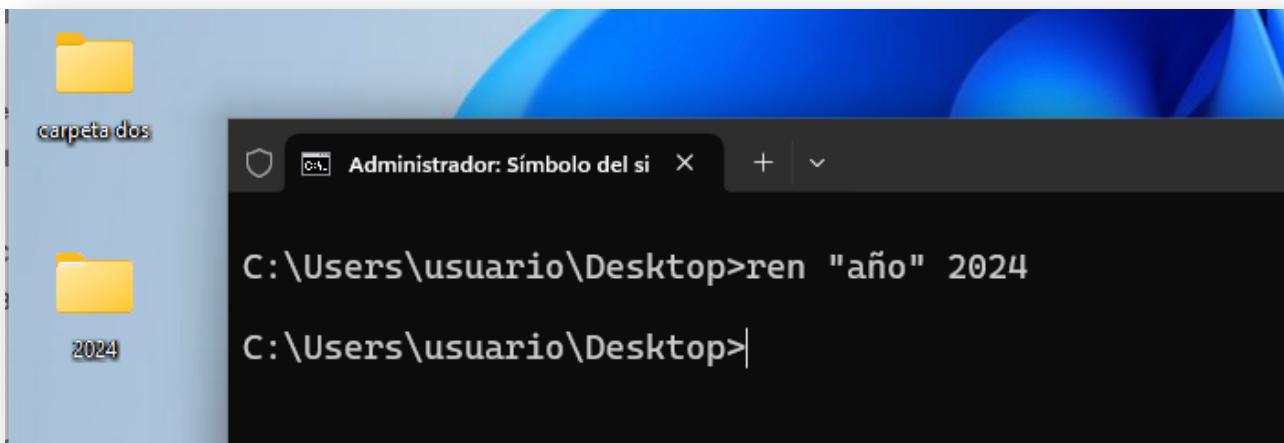




REN

Cambia el nombre de archivos o carpetas. Para ello, a este comando hay que pasarle dos parámetros. El primero es el origen, es la ruta donde se encuentra el archivo o carpeta que queremos cambiar de nombre y como segundo parámetro el nuevo nombre que tendrá.

El primer parámetro es una ruta y el segundo solamente el nuevo nombre. Debemos tener en cuenta que la ruta del primer parámetro sigue el patrón visto anteriormente, es decir que debe ser una ruta absoluta o relativa.



En esta captura podemos ver como hemos cambiado el nombre de la carpeta "año" a 2024. Recordamos que en la ruta que he puesto ni empieza por la unidad de disco ni por puntos así que se asume que es una ruta relativa usando el punto por lo que el origen estaría (en este caso) en la carpeta "C:\Users\usuario\Desktop"

RD

Este comando elimina un directorio. Si consultamos la ayuda con el parámetro "/" veremos que nos ofrece



```

Símbolo del sistema  +  ▾

Microsoft Windows [Versión 10.0.22631.3374]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\jabellan>rd /?
Quita (elimina) un directorio.

RMDIR [/S] [/Q] [unidad:]ruta
RD [/S] [/Q] [unidad:]ruta

/S      Quita todos los directorios y archivos del directorio además
        del mismo directorio. Se usa principalmente cuando se
        desea quitar un árbol.

/Q      Modo silencioso, no pide confirmación para quitar un árbol
        de directorio con /S

C:\Users\jabellan>

```

En la ayuda no se dice explícitamente pero este comando no se ejecuta si hay algo dentro de la carpeta que queremos borrar. En realidad, es una medida de seguridad contra borrados accidentales. Para borrar carpetas con contenido en su interior debemos o bien borrar los archivos en su interior a mano o bien utilizar el parámetro "/s". En este caso nos pedirá una confirmación explícita de que queremos borrar la carpeta con todo su contenido.

```
C:\Users\jabellan\Desktop>rd año /s
año, ¿Está seguro (S/N)?
```

Si queremos evitar la pregunta tendremos que añadir el modificador "/q" y se realizará un borrado sin preguntas ni interacción con el usuario.

Algo para tener en cuenta es que los borrados que hagamos desde terminal no pasan por la papelera de reciclaje, se eliminan inmediatamente.

Es buen momento para probarlo, crea un par de carpetas y añádeles contenido. Borra una de ellas con el explorador de ficheros y otra desde terminal para ver las diferencias. Fundamentalmente el explorador de ficheros te deja borrar sin preguntarte nada porque sabe que lo puedes recuperar desde la papelera de reciclaje. En cambio desde terminal te advertirá y pedirá confirmación de que realmente queremos borrar.



Crear archivos

Desde el símbolo de sistema de Windows no hay un comando específico para crear archivos., se espera que sea el programa de usuario el que lo haga programáticamente. Aun así, tenemos varios trucos que nos permiten crearlos.

El primer truco es usar el comando "copy"

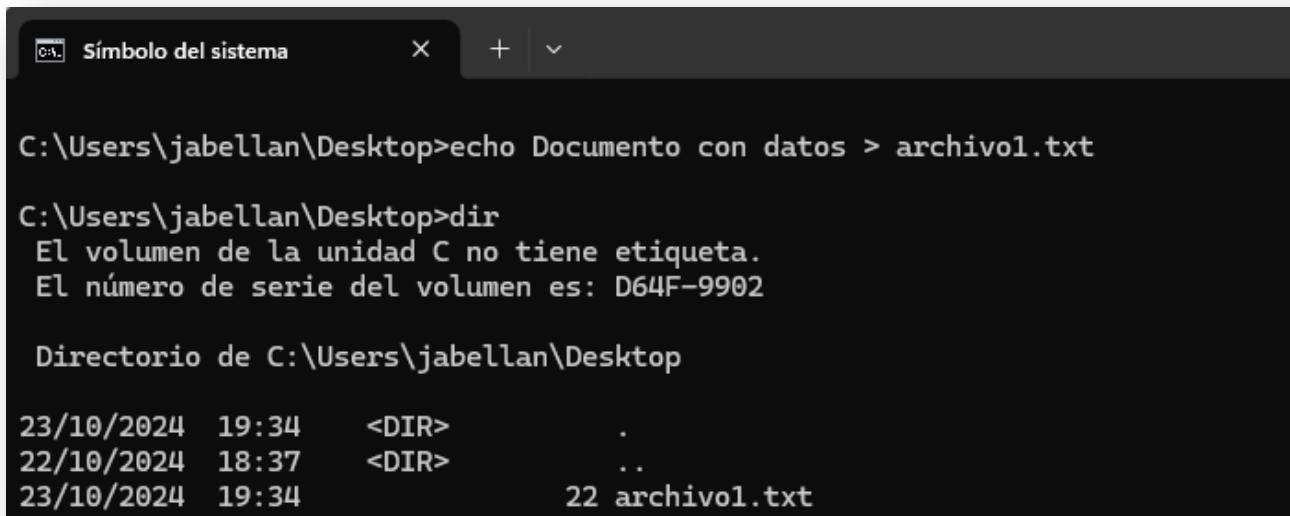
```
copy nul archivo.txt
```

Con este comando lo que hacemos es copiar un nulo a un nuevo archivo llamado "archivo1.txt" en la carpeta actual.

Otro truco consiste en usar el siguiente comando:

```
Echo "Archivo con contenido" > "archivo2.txt"
```

Con este comando hacemos un eco, algo similar a un print en cualquier lenguaje de programación y redirigimos su salida con ">" para que, en lugar de salir por pantalla, salga a un archivo llamado "archivo2.txt" en la carpeta actual.



```
C:\Users\jabellan\Desktop>echo Documento con datos > archivo1.txt

C:\Users\jabellan\Desktop>dir
 El volumen de la unidad C no tiene etiqueta.
 El número de serie del volumen es: D64F-9902

 Directorio de C:\Users\jabellan\Desktop

23/10/2024 19:34    <DIR>      .
22/10/2024 18:37    <DIR>      ..
23/10/2024 19:34           22 archivo1.txt
```

Dejo como reto averiguar la razón por la que este archivo ocupa 22 bytes.

Además del operador ">" que nos redirecciona una salida también disponemos del operador ">>" que nos anexa datos a continuación de los datos que ya contiene el documento.



TYPE

Este comando nos permite ver el contenido de un archivo. Si consultamos la ayuda veremos que este comando no permite ningún parámetro que no sea la ruta del archivo del que queremos ver el contenido.

Si es un archivo de texto veremos el contenido en plano tal y como está guardado en el fichero como en la siguiente captura.

```
C:\Users\usuario\Desktop>copy con prueba.txt  
Esto es una prueba  
de como se puede guardar lo escrito desde la consola directamente a un archivo  
Fin  
    1 archivo(s) copiado(s).  
  
C:\Users\usuario\Desktop> type prueba.txt  
Esto es una prueba  
de como se puede guardar lo escrito desde la consola directamente a un archivo  
Fin  
  
C:\Users\usuario\Desktop>
```

En cambio, si ejecutamos un "type" sobre un archivo de tipo binario tendremos una salida con muchos caracteres raros fruto de interpretar esa secuencia de bits usando la tabla ASCII.

DEL

Usamos el comando "del" para borrar archivos. En primer lugar, consultamos la ayuda con el parámetro "?".



```
Símbolo del sistema x + ▾
C:\Users\jabellan\Desktop>del /?
Elimina uno o más archivos.

DEL [/P] [/F] [/S] [/Q] [/A[[:]attributes]] names
ERASE [/P] [/F] [/S] [/Q] [/A[[:]attributes]] names

names      Especifica una lista de uno o más archivos o directorios.
           Se pueden usar caracteres comodín para eliminar varios archivos. Si se
           especifica un directorio, todos los archivos de este se
           eliminarán.

/P          Pide confirmación antes de eliminar cada archivo.
/F          Fuerza la eliminación de los archivos de solo lectura.
/S          Elimina los archivos especificados de todos los subdirectorios.
/Q          Modo silencioso. No pide confirmación sobre si es correcto eliminar el carácter comodín global
/A          Selecciona los archivos que se van a eliminar basándose en los atributos.
Atributos   R  Archivos de solo lectura          S  Archivos de sistema
           H  Archivos ocultos          A  Archivos preparados para el archivado
           I  Archivos indexados que no son de contenido L  Puntos de reanálisis
           O  Archivos sin conexión      -  Prefijo de exclusión

Si las extensiones de comando están habilitadas, DEL y ERASE cambian
de la siguiente manera:

La semántica que se muestra para el modificador /S está invertida
de tal modo que le muestra solamente los archivos eliminados y no
los que no se encontraron.

C:\Users\jabellan\Desktop>
```

De la ayuda podemos concluir que tenemos 3 alternativas. La primera consiste en poner la ruta a uno o varios archivos que queremos eliminar. La segunda consiste en usar comodín para eliminar un número desconocido de archivos que cumplan la condición del filtro. La tercera opción consiste en indicar la ruta de una carpeta para borrar los archivos que contiene. Recuerda que aquí mantenemos todo lo visto con las rutas, es decir, podemos escribir rutas absolutas que empiecen por una letra y dos puntos indicando la unidad de disco o bien rutas relativas que empiecen por punto o dos puntos seguidos para tener como origen la carpeta actual o su parente. En caso de no indicar ninguna de estas opciones, Windows tomará el punto por defecto indicando que la ruta parte de la carpeta actual.

```
C:\Users\jabellan\Desktop>del prueba
C:\Users\jabellan\Desktop\prueba\*, ¿Está seguro (S/N)?
```

Si nos fijamos, hacer un "del" de una carpeta se transforma en un borrar todos los archivos d esa carpeta gracias al asterisco del comodín.

Crea una carpeta que contenga un par de archivos, repite el proceso con una subcarpeta dentro de la primera y haz pruebas. Recuerda que los archivos borrados desde la interfaz de comandos no pasan por la papelera de reciclaje, se borran directamente.



COPY

Con este comando copiamos archivos desde una ruta origen hasta una ruta destino. Al usar este comando es necesario tener bastante claro el uso de las rutas tanto absolutas como relativas porque recordemos que cuando las rutas no empiezan por unidad de disco o puntos, Windows toma el punto por defecto.

Otro asunto para tener en cuenta es el orden de los parámetros. Primero siempre va el origen y como segundo parámetro tenemos el destino.

La forma más sencilla consiste en especificar la ruta absoluta tanto de origen como de destino.

```
copy c:\users\usuario\carpeta\archivo1.txt c:\users\usuario\carpeta\archivo2.txt
```

Este comando anterior copia el contenido del archivo1.txt cuya ruta es absoluta, a otro archivo que crea con el nombre de archivo2.txt cuya ruta es también absoluta. El archivo origen se creará pero en caso de que ya existiera previamente aparecerá un texto de advertencia preguntando si queremos sobrescribir el archivo o no.

El comando "copy" crea el archivo en destino, pero no crea las carpetas que pudiera haber en la ruta así que antes de copiar es necesario asegurarse que la carpeta destino ya existe.

En ocasiones podemos usar únicamente el parámetro origen. Windows tomará como parámetro destino el punto y usará el mismo nombre de archivo.

```
copy c:\archivo.txt
```

Este comando copia el fichero "archivo.txt" situado en la raíz del disco c: a la carpeta activa en ese momento y usará el mismo nombre ("archivo.txt") para guardarlo.

De la misma manera podemos establecer todas las combinaciones que se nos ocurran como por ejemplo la siguiente:

```
copy c:\archivo.txt ..
```

Este comando copiará el archivo llamado "archivo.txt" desde la raíz del disco c: hasta la carpeta padre de la carpeta actual desde donde estemos lanzando este comando.

El comando copy nos permite copiar varios archivos a una misma carpeta destino. En este caso podemos hacerlo de dos maneras. La primera es con el uso del comodín asterisco.

```
copy c:\archivo*.txt ..
```

Este comando copia todos los archivos cuyo nombre empiece por la subcadena "archivo" y los deja con su mismo nombre en la carpeta actual (se ve mal, pero fijémonos que hay un punto como destino indicando la carpeta actual)



Otra alternativa consiste en separar los nombres de archivo origen con un "+"

```
copy c:\archivo.txt + datos.txt + informes.txt c:\resumen
```

Este archivo copia los archivos llamados "archivo.txt" de la raíz del disco, el archivo llamado "datos.txt" e "informes.txt" de la carpeta actual (fijémonos que no tienen ruta absoluta ni relativa por lo que se usa el punto por defecto) y finalmente, estos tres archivos se guardan en la carpeta resumen (con ruta absoluta) con su mismo nombre de origen.

Con el comando "copy" también podemos copiar y pegar el archivo con distinto nombre. En este caso habría que indicar la ruta al origen y la ruta al destino especificando en este último caso como queremos que sea el nuevo nombre del archivo.

```
copy archivo1.txt archivo2.txt
```

Este comando me hace una copia del fichero "archivo1.txt" que se encuentra en la carpeta actual a un nuevo archivo llamado "archivo2.txt" que también dejaremos en la carpeta actual.

MOVE

El comando "move" funciona de una manera casi idéntica al comando "copy". Si con el comando "copy" hacíamos una copia de un archivo, con el comando "move" solamente lo movemos. Estos dos comandos son el equivalente al copiar/pegar y mover/pegar.

Igual que con "copy" al mover uno o varios archivos a otra carpeta le podemos cambiar el nombre durante el proceso.

XCOPY

El comando "xcopy" es la solución a cuando queremos copiar archivos y también carpetas. Si imaginamos una estructura de directorios compleja que quisiéramos copiar en el escritorio u otro disco, tendríamos que crear las carpetas con "mkdir" y después ir copiar los archivos a cada subcarpeta. "xcopy" nos resuelve este problema haciendo todos los pasos en uno solo. Para conocer en detalle cómo funciona consultaremos la ayuda con "/"?



```
C:\ Símbolo del sistema x + ▾
C:\Users\jabellan\Desktop\prueba\prueba>
C:\Users\jabellan\Desktop\prueba\prueba>xcopy /?
Copia archivos y árboles de directorios.

XCOPY source [destination] [/A | /M] [/D[:date]] [/P] [/S [/E]] [/V] [/W]
    [/C] [/I] [/R] [/Q] [/F] [/L] [/G] [/H] [/R] [/T]
    [/U] [/K] [/N] [/O] [/X] [/Y] [/Z] [/B] [/J]
    [/EXCLUDE:file1[+file2][+file3]... ] [/COMPRESS]

source      Especifica los archivos por copiar.
destination  Especifica la ubicación o el nombre de los nuevos archivos.
/A          Copia solo archivos con el conjunto de atributos de archivo,
            no cambia el atributo.
/M          Copia solo archivos con el conjunto de atributos de archivo,
            desactiva el atributo de archivo.
/D:m-d-y   Copia archivos modificados en la fecha especificada o después de esta.
            Si no se proporciona una fecha, copia solo aquellos archivos cuya
            hora de origen es más reciente que la hora de destino.
/EXCLUDE:file1[+file2][+file3]...
            Especifica una lista de archivos que contienen cadenas. Cada cadena
            debe estar en una línea separada en los archivos. Cuando cualquiera de las
            cadenas coincide con cualquier parte de la ruta absoluta del archivo que se va a
            copiar, ese archivo se excluirá y no se copiará. Por
            ejemplo, especificar una cadena como \obj\ o .obj excluirá
            todos los archivos debajo del directorio obj o todos los archivos con
            la extensión .obj respectivamente.
/P          Pregunta antes de crear cada archivo de destino.
/S          Copia directorios y subdirectorios excepto los vacíos.
/E          Copia directorios y subdirectorios, incluidos los vacíos.
            igual que /S /E. Puede usarse para modificar /I.
/V          Comprueba el tamaño de cada nuevo archivo.
/W          Solicita presionar una tecla antes de copiar.
/C          Continúa copiando incluso si se producen errores.
```

Aquí podemos ver que si usamos el parámetro /E se copiarán toda la estructura de carpetas incluyendo aquellas subcarpetas sin contenido.

```
xcopy /e c:\fotos c:\almacen\2024
```

Este comando nos copiará todo el contenido de la ruta c:\fotos en la ruta c:\almacen\2024. Usar este comando nos dará independencia de la estructura de archivos y carpetas que contenga.