

```
C:\Windows\system32\cmd.exe

echo .
echo Hola Mundo
Hola Mundo
_
```

en Tutoriales

Lenguaje batch tutorial y comandos básicos .bat

Este es un tutorial sencillo para principiantes en el que se intenta explicar de la forma mas simple los principales comandos.

El lenguaje batch no es muy difícil pero para personas que se estén iniciando puede resultar un poco complicado así que intentare explicarlo de la forma mas sencilla posible.

¿Qué es batch?

Un archivo batch es un archivo de texto sin formato guardado con la extensión bat y que contiene un conjunto de ordenes llamadas comandos de DOS.

en resumen un archivo batch es un archivo de texto en el que se a escrito un conjunto de códigos que van a ser ejecutados de forma lineal (ya lo irán entendiendo con ejemplos mas adelante) y ese archivo de texto debe ser guardado con la extensión .bat, es decir un nombre cualquiera.bat ej:
nombredelarchivo.bat

Es muy útil para automatizar tareas, por ejemplo un programa que nos abra todas las herramientas que necesitamos para trabajar en un determinado proyecto, de esta forma cada vez que queramos trabajar solo tendremos que pulsar un botón y no hará falta abrir una a una todas las aplicaciones necesarias, pero su utilidad no se queda en esto, todo hacker debe conocerlo, especialmente su equivalente en linux: los scripts de bash.

La terminación bat viene del propio nombre del lenguaje (batch) que debido a que las extensiones de archivos (.bat, .exe, .txt etc) solo pueden tener 3 letras quedo reducido a eso.

PARTE 1

Lo primero que debemos hacer es abrir un archivo de texto y guardarlo con un nombre seguido de .bat tal como expliqué anteriormente en ese archivo vamos a poner nuestros comandos.

Empecemos quitando el eco, para eso escribimos: **@echo off**

```
1 | @echo off
2 | pause
3 | exit
```

Si no sabes lo que es el eco prueba a quitar **@echo off** y verás lo que pasa.

Pause sirve para hacer una pausa en la secuencia (pulsas una tecla para continuar)
y **exit** sirve para finalizar

Bien ahora vamos a continuar con otro código: el **echo** y el **echo.** (*nótese el punto*)

El **echo** sirve para escribir algo en pantalla por ejemplo echo hola escribiría hola en la pantalla
y **echo.** sirve para dejar un espacio

```
1 @echo off
2 echo.
3 echo Hola TodoHacker
4 echo.
5 echo Como estas?
6 pause
7 exit
```

Prueba a ponerlo y comprueba lo que pasa

Es muy importante que vayas entendiendo todo esto paso a paso sino se irá complicando bastante.

Ahora vamos a ver una variación del **pause**: **pause>NUL**, sirve para hacer una pausa sin poner nada en la pantalla podéis probarlo sustituyendo **pause** por **pause>NUL** en los códigos anteriores

Esto lo vamos a utilizar para poner nosotros el mensaje que queramos en vez del «pulsa una tecla para continuar»

y lo vamos a hacer poniendo algo seguido de **pause>NUL** veréis el efecto

```
1 @echo off echo.
2 echo Pulsa una tecla para ir a la siguiente parte del tutorial de
3 todohacker.com
4 pause>nul
5 echo.
6 echo Bienvenido a la segunda parte del tutorial echo. echo Pulsa una te-
7 cla para salir
  pause>nul
  exit
```

PARTE 2

El comando **cls** sirve para borrar la pantalla.

```
1 @echo off
2 echo.
3 echo Pulsa una tecla para ir a la siguiente parte del tutorial
4 pause>nul
5 cls
6 echo.
7 echo Bienvenido a la segunda parte del tutorial de todohacker
8 echo.
```

```
9 | echo Pulsa una tecla para salir
10 | pause>nul
11 | exit
```

Este es muy sencillo, ejecútalo y te darás cuenta de su utilidad.

Ahora vamos a pasar a las **variables**, algo que tienen casi todos los lenguajes de programación y que, como verás, te da infinitas posibilidades.

Una variable es un nombre que designa un valor y como su nombre indica puede variar. Gracias a esto podemos guardar valores y darles nombre.

Para poner una variable debemos escribir se nombrevARIABLE= valor. ej: **set** nombre= pepito

y para referirnos a esa variable debemos escribir el nombre de la variable entre %% ejemplo: hola %nombre%

```
1 | @echo off
2 | set nombre= Pepito
3 | echo.
4 | echo Hola %nombre%
5 | echo.
6 | pause
7 | exit
```

Es muy fácil pero es importante que lo entiendas, por que en esto se basa todo lo que viene a continuación, si te estas preguntando por que es realmente importante, ahora vas a verlo.

Ahora vamos a hacer que podamos poner nosotros mismos la variable, es decir que la persona que usa el programa puede asignar un valor. para eso en vez de poner **set** debemos poner **set/p**

ej: **set/p** nombrevARIABLE= valor

```
1 | @echo off
2 | echo.
```

```
3  set/p nombre= ¿Como te llamas?
4  cls
5  echo.
6  set/p años= ¿Cuantos años tienes?
7  cls
8  echo.
9  echo Hola %nombre% tienes %años% años.
10 echo.
11 echo Pulsa una tecla para salir.
12 pause>nul
13 exit
```

Una vez entendido esto vamos a empezar a utilizar las variables por ejemplo para hacer operaciones con ellas.

Para eso utilizaremos **set/a**. es como el **set/p** pero no se muestra en pantalla y realiza operaciones, porcentajes, sumas, restas etc.

Estos son los signos que podemos utilizar:

+ para sumar

– para restar

* para multiplicar

/ para dividir

Y lo vamos a expresar así:

```
1  @echo off
2  echo.
3  set/p numero1= Dime un numero
4  cls
5  echo.
6  set/p numero2= Dime otro para sumar al anterior
7  cls
8  set/a suma= %numero1% + %numero2%
9  echo.
10 echo %numero1% + %numero2% = %suma%
11 echo.
12 echo Pulsa una tecla para salir.
13 pause>nul
14 exit
```

PARTE 3

En esta parte del tutorial vamos a aprender a trabajar con **if**, **if not**, **else** y **goto**.

If sirve para expresar una condición. Es decir si pasa esto entonces haz esto.

Para ello vamos a utilizar la siguiente estructura: if condición acción

ejemplo: if %variable%==4 echo bien!! esto quiere decir que si el valor de la variable es igual a cuatro entonces escribe «bien!!»

Las distintas condiciones que podemos comprobar con un if en batch son las siguientes: (operadores)

if variable==valor (para expresar si la variable es igual a valor haz esto)

if variable LSS valor (para expresar si la variable es menor a valor haz esto)

if variable GTR valor (para expresar si la variable es mayor a valor haz esto)

if variable LEQ valor (para expresar si la variable es menor o igual haz esto)

if variable GEQ valor (para expresar si la variable es mayor o igual haz esto)

if not se utiliza igual que if pero expresa todo lo contrario ej:

if not variable==valor (para expresar si la variable NO es igual a «valor») y lo mismo con el resto de condiciones vistas arriba.

```
1  @echo off
2  echo.
3  set/p variable= ¿Cuanto son 2 + 2?
4  cls
5  echo.
6  if %variable%==4 echo bien!!
7  echo.
8  pause
9  exit
```

Aquí también se podría utilizar el goto que sirve para enviar la lectura de comandos a otro sitio del bat por ejemplo if variable==valor goto correcto, después mas abajo pondríamos una «etiqueta» con ese nombre después de dos puntos ej: :correcto así quedaría: si variable es igual a valor entonces ir

al menú «correcto» y en el ya escribiríamos lo que queramos con un ejemplo lo vas a entender mucho mejor.

```
1  @echo off
2  echo.
3  set/p variable= ¿Cuanto son 2 + 2?
4  cls
5  echo.
6  if %variable%==4 goto correcto
7  echo.
8  :correcto
9  echo bien!!
10 pause
11 exit
```

En este caso no tendría mucho sentido pero cuando tenemos más variables y muchas opciones a elegir es esencial útil ya lo iréis viendo

Este ejemplo indicaría que si la respuesta a la pregunta «¿cuanto son 2+2?» es 4 entonces escribe bien!!

pero ¿qué pasa si la respuesta no es esa? pues para eso sirve **else**, se utiliza así:

if variable==valor goto correcto else goto incorrecto esto significaría que si la respuesta es 4 entonces escribe bien pero si la respuesta es cualquier otra cosa escribe mal!!!

```
1  @echo off
2  :inicio
3  echo.
4  set/p pregunta= ¿cuanto son 2 + 2?
5  cls
6  echo.
7  if %pregunta%==4 goto correcto else goto incorrecto
8  :incorrecto
9  echo.
10 echo Mal!!
11 echo.
12 echo Vuelve a intentarlo
13 pause
14 cls
15 goto inicio
16 :correcto
17 echo.
18 echo Bien!!
19 echo Pulsa una tecla para salir de este ejemplo de www.todohacker.com
20 pause>nul
21 exit
```

He añadido lo de inicio por que así si es incorrecto vuelve a empezar.

Una vez que hemos llegado hasta aquí si lo habéis entendido podréis hacer cosas como esta:

```
1  @echo off
2  :inicio
3  echo.
4  set/p variable= ¿cuanto son 2 + 2?
5  cls
6  echo.
7  if %variable%==4 goto correcto else goto incorrecto
8  :incorrecto
9  echo.
10 echo Mal!!
11 echo Intentalo de nuevo
12 pause
13 cls
14 set/p variable1= ¿Cuanto son 5 - 3?
15 cls
16 if %variable1%==2 goto correcto else goto incorrecto1
17 :incorrecto1
18 echo.
19 echo Mal!!
20 echo Intentalo de nuevo
21 pause
22 cls
23 set/p variable2= ¿Cuanto son 4 * 4?
24 cls
25 echo.
26 if %variable2%==16 goto correcto else goto incorrecto2
27 :incorrecto2
28 echo.
29 echo mal!!
30 echo.
31 echo Intentalo de nuevo
32 pause
33 cls
34 goto inicio
35 :correcto
36 echo.
37 echo Bien!!
38 echo Presione una tecla para salir de este ejemplo de
39 www.todohacker.com
40 pause>nul
    exit
```

Te animo a que lo pruebes y te diviertas un rato modificandolo, échale imaginación, no hay limite a lo que puedas hacer 😊

PARTE 4

En esta parte vamos a ver algo muy sencillo, se trata del código de colores de las letras y el fondo de la consola simplemente se expresa con una combinación de números y letras de esta forma:

Primero ponemos **color**, para indicar que lo que le vamos a decir es el color del la letra y el fondo, y seguidamente debemos poner dos números sin espacio, el primero es el color del fondo y el segundo el color de letra.

Código de colores:

A = Verde claro

B = Aguamarina claro

C = Rojo claro

D = Púrpura claro

E = Amarillo claro

F = Blanco brillante

0 = Negro

1 = Azul

2 = Verde

3 = Aguamarina

4 = Rojo

5 = Púrpura

6 = Amarillo

7 = Blanco

8 = Gris

9 = Azul claro

Sabiendo esto, si queremos que el fondo sea blanco y las letras negras pondremos **color 70** pruebalo tu mismo. Abajo hay unas fotos explicativas.

Ahora otra cosa muy sencilla.

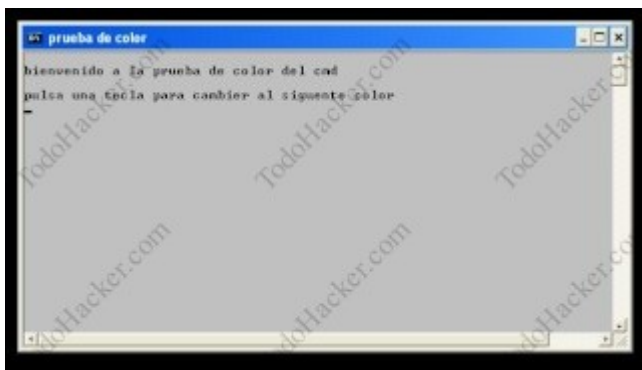
No se si os habéis fijado en la barra de titulo de la terminal, pone por defecto la ubicación en la que nos encontramos, normalmente system32.

Pues esto se puede cambiar poniendo simplemente «title» y el nombre que le queramos poner.

Pondré un ejemplo conjunto para los códigos de color y titulo.

```
@echo off
title Prueba de color www.todohacker.com
color 70
echo.
echo Bienvenido a la prueba de color del cmd de todohacker.com
echo.
echo Pulsa una tecla para cambiaer al siguiente color
pause>nul
color 4f
pause
exit
```

Prueba a cambiar los colores para entenderlo mejor.



Ahora vamos a ver redireccionamientos, son muy útiles ya lo verás.

Se utiliza > para escribir algo y >> para añadir algo a un documento que ya tenia algo escrito por ejemplo.

Se utilizan así: echo hola>archivo.txt esto escribiría hola en un archivo de texto situado en el directorio en el que nos encontramos, es decir por defecto en el que tenemos guardado el archivo de bat que ejecuta esta orden. Si el dicho archivo de texto con nombre «archivo» no existe lo crea.

```
1 | @echo off
2 | echo pulsa una tecla para crear el archivo
3 | pause>nul
4 | echo hola, esto es un ejemplo de todohacker.com > archivo.txt
5 | echo archivo creado
6 | pause
```

7 | `exit`

Si os fijáis nos crearía un documento de texto en el escritorio (en caso de que el bat este en escritorio) con el nombre de «archivo», y escribiría dentro hola se pueden crear otro tipo de archivos, modificando la extensión, .txt, .bat etc.

Pero que pasaría si en el archivo queremos escribir mas lineas? en ese momento es cuando utilizamos el >>. si tenemos un documento de texto que pone hola y le decimos que escriba otra cosa así: echo que tal? > archivo.txt, lo remplazaría, es decir, pasaría a poner que tal? en la primera linea.

Pero si ponemos: echo que tal? >> archivo.txt se añadiría al hola una linea mas abajo. Pruébalo tu mismo, aquí pongo dos códigos de ejemplo para que veas la diferencia.

```
1  @echo off
2  echo hola > archivo.txt
3  echo.
4  echo Abre al archivo creado antes de continuar y mira lo que pone
5  echo Cierra el archivo y continua
6  echo.
7  pause
8  cls
9  echo que tal? > archivo.txt
10 echo.
11 echo Fijate en el archivo creado, pondra que tal? "hola" se ha borrado.
12 echo.
13 pause
14 exit
```

```
1  @echo off
2  echo hola > archivo.txt
3  echo.
4  echo Abre al archivo creado antes de continuar y mira lo que pone
5  echo Cierra el archivo y continua
6  echo.
7  pause
8  cls
9  echo que tal? >> archivo.txt
10 echo.
11 echo Fijate en el archivo creado, "pondra que tal?" despues de hola, no
12 ha borrado nada.
13 echo.
14 pause
    exit
```

Es posible que te hayas dado cuenta de algo, y es que esto es lo mismo que usamos para el `pause>NUL`. Efectivamente lo que estamos haciendo es redireccionar el texto por defecto a un archivo llamado «NUL», en teoría windows no debería crear este archivo, pero se que a veces lo crea, en todo caso no es un problema demasiado grave.

Espero que se haya entendido todo, la parte de las redirecciones es muy importante por que nos va a servir varias pruebas que hagamos posteriormente en creación de virus y scripts automáticos.

PARTE 5

En la anterior parte hemos aprendido a crear archivos, en esta vamos a aprender como ejecutarlos, borrarlos, copiarlos, cambiar sus atributos etc.

Empecemos por ejecutarlos.

Retomemos el ejemplo anterior en el que creamos un archivo de texto con dos lineas, bueno pues ahora vamos a ejecutarlo y a borrarlo.

```
1  @echo off
2  echo hola, esto es un ejemplo de todohacker.com > archivo.txt
3  echo.
4  echo abre al archivo creado antes de continuar y mira lo que pone
5  echo cierra el archivo y continua
6  echo.
7  pause
8  cls
9  echo que tal? >> archivo.txt
10 echo.
11 echo fijate en el archivo creado, "pondra que tal?" despues de hola, no
12 ha borrado nada.
13 echo.
14 pause
    exit
```

Para ejecutar un archivo utilizaremos el comando **start** y para borrarlo el comando **del** acordaos de especificar siempre la extensión.

```
1  @echo off
2  echo hola, esto es un ejemplo de todohacker.com > archivo.txt
3  pause
4  echo que tal? >> archivo.txt
5  echo.
6  echo El archivo se a creado
7  echo.
8  echo Ahora vamos a ejecutarlo (abrirlo)
9  pause
10 start archivo.txt
11 cls
12 echo.
13 echo bien puedes cerrar el archivo por que vamos a borrarlo
14 pause
15 del archivo.txt
16 cls
17 echo.
18 echo Archivo borrado con esto se puede borrar cualquier archivo, solo
19 hay que saber el nombre &#x1f600;
20 pause
    exit
```

Al comando **del** se pueden añadir distintas letras para que funcione de distinta forma:

del/f Modo force. Borra incluso los archivos de solo lectura

del/s Borra también los subdirectorios.

del/p Pide confirmación

del/q Modo silencioso. No pide confirmación en ningún caso

Se pueden utilizar varios en el mismo del

por ejemplo: del/f/p archivo.txt

borraría el archivo aunque fuese de solo lectura y pediría confirmación

Por supuesto para el comando del también se puede especificar la ruta del archivo si este no estuviera en el mismo directorio

ej: del c:/ruta/correspondiente/archivo.txt

Ahora vamos con el comando **cd** que utilizaremos para cambiar el directorio general de la terminal, es decir, si el archivo bat esta en el escritorio la terminal estará en ese directorio por defecto y para cambiarlo utilizaremos el mencionado **cd**.

Por ejemplo si queremos crear el archivo txt del ejemplo en mis documentos en vez de en el escritorio podríamos escribir la ruta cada vez que nos re-

firamos al archivo pero es mas sencillo si cambiamos el directorio actual de la terminal para que siempre se dirija allí.

Para ello escribiremos: `cd C:\Documents and Settings\Propietario\Mis documentos` (los nombres de los directorios pueden cambiar según el ordenador)

y todo lo que creemos o eliminemos sin especificar la carpeta lo hará en mis documentos

También podemos escribir:

cd..(con los puntos) e irá al directorio superior es decir, si estamos en mis documentos se trasladará a propietario, en el caso del ejemplo o **cd ** y se trasladara a la raíz del directorio. en este caso C:

Llegados a este punto tienes los conocimientos necesarios para continuar aprendiendo por tu cuenta a partir de experimentación y experiencia, voy a nombrar algunos comandos más:

rd borra un directorio de raíz solo tenemos que especificar la ruta del directorio y borra todo lo que tenga dentro, carpetas, mas archivos, etc.

format formatea una unidad por ejemplo `format k:` formatearía un pendrive externo o le que tengamos en esa unidad, o `format c:` (ya podéis imaginaros lo que hace)

taskkill termina un proceso como por ejemplo el skype muy útil para bromas ej: `taskkill Skype.exe` cerraría el skype. Puedes ver los procesos que tienes activos pulsando `ctrl+shift+esc`. (shift es la tecla que esta sobre el ctrl de mayusculas.)

shutdown también es muy útil para bromas y según lo que lleve detrás significa una cosa u otra:

-p Apaga el equipo local sin tiempo de espera ni advertencia.

-h Hiberna el equipo local.

-l Cierra la sesion.

-g Cierra la sesion y reinicia el equipo

-s Apaga el equipo.

-r Reinicia el equipo.

-e Documenta la razón del cierre inesperado de un equipo.

- m \\equipo Especifica el equipo de destino. (explicado mas abajo)
- t xxx Establece el período de tiempo de espera antes del cierre en xxx segundos.
- c «comentario» Comentario acerca de la razón del reinicio o apagado.
- f Fuerza el cierre de las aplicaciones sin advertir previamente a los usuarios.
- a Cancela el apagado.

(nota: Estas opciones solo están disponibles a partir de windows 7, en versiones anteriores solo tenemos -s, -r, -t, -a, -c)

Esto es básicamente lo que utilizo en el tutorial de crear un minivirus pero de forma mas sencilla

un ejemplo

shutdown -s -t 60 (tiempo en segundos) -c virus intrusion

Cuando se ejecute pondrá una cuenta atrás de 1 min, pondrá el mensaje de: «virus intrusion» y se apagará

para cancelar la cuenta atrás debemos poner shutdown -a, se puede poner directamente en inicio, ejecutar (simbolo de windows + R) y se cancelara automáticamente.

msg * es otro comando muy útil para bromas hace que salga un mensaje de windows con el texto que le pongan

Por ejemplo: msg * atención su sistema contiene software ilegal estamos enviando la información al FBI

Ahora aprenderemos a utilizar el comando **copy** que como su nombre indica sirve para copiar archivos.

Se utiliza de este modo copy ruta del archivo que queremos copiar «ruta del directorio donde lo queremos copiar incluyendo el nuevo nombre del archivo.

Ejemplo: si queremos copiar un archivo que se llama cosa.txt de el directorio del escritorio a mis documentos, pondremos: copy C:\Documents and Settings\Propietario\Escritorio\cosa.txt «C:\Documents and Settings\Propietario\Mis documentos\cosa2.txt

y por último para cambiar los atributos de un archivo utilizaremos **attrib**

attrib+ nombre del archivo o ruta, mostraría los atributos del archivo.

Los atributos son solo lectura, oculto, sistema

Para cambiarlos utilizaremos:

r => Lectura. Solo lectura con +r, normal con -r

h => oculto. Se activa con +h y desactiva con -h

s => Sistema. Se activa con +s y desactiva con -s

ej:attrib +h se pondría oculto como atributo

Aquí concluye este tutorial, estos son los comandos mas básicos que me ha parecido importante mencionar, hay que decir que he obviado cosas que podrían considerarse importantes, como por ejemplo los bucles, pero este tutorial tenia la misión de ser una introducción, y no me ha parecido conveniente extenderlo mas, ahora es tu misión continuar aprendiendo, lo más importante ya lo sabes y ahora el limite es tu imaginación.

Solo me queda para concluir mencionar un modo de nombrar a los archivos en general

para ello utilizaremos el * que nombra a cualquier archivo es decir si decimos:

del archivo.* borraría todos los archivos que se llamen «archivo» sean de la extensión que sean

del *.txt borraría todos los archivos de texto se llamen como se llamen

del a*.txt borraría todos los documentos de texto que empiecen por a

del nombre.d* borraría todos los archivos que se llamen «nombre» y que su extensión empiece por d como doc o dat

También podemos utilizar ? se usa igual que * pero sustituye a una única letra

ej: **del pan?l.txt** borraría todos los archivos de texto que empezasen por pan siguiese una letra cualquiera y terminase en l por ejemplo panel, panal etc.

Espero que hayan entendido todo. Cualquier duda que tengan por favor pongan un comentario y se responderá con mucho gusto.

Si quieres puedes poner en una ventana de terminal **help** y saldrán todos

los comandos, por supuesto TODOS los comandos mencionados aquí se puede escribir directamente en la ventana de terminal sin necesidad de crear un archivo bat.

Este tutorial tiene copyleft de TodoHacker.com si lo usas, pon las fuentes.



Todohacker by 4m1g0 is licensed under a [Creative Commons](#).

 Escribe un comentario

 426 COMENTARIOS

← Comentarios más antiguos

Zemilo45

12 mayo, 2020

Hello, para lograr ejecutar cada vez que encienda mi pc que código le insertaría
@echo
echo Herramienta de pantalla
echo.
start desk.cpl

[Responder a Zemilo45](#)

jc

6 junio, 2020

Bastara que lo coloques como acceso directo en la carpeta de INICIO, cada vez que inicie windows se ejecutará; es la carpeta que sale cuando oprimes la tecla windows, todos los programas, inicio... suerte.!

[Responder a jc](#)

cath

12 mayo, 2020

Hola , me gustaria si es posible una ayuda , necesito poner a trabajar en windows xp un sistema que trabaja en windows 98, como lo puedo poner a trabajar en *.bat , cuales serian los comandos necesarios para ponerlo igualmente a trabajar sin dificultad estoy necesitando una ayuda por ello me e puesto a estudiar e investigar del tema.

[Responder a cath](#)

Somfug

11 junio, 2020

Ok, muy educativo este tutorial, mucha andragogia, sería muy bueno continuar con otros tutoriales cada vez más avanzados, gracias por el tiempo y la dedicación.

[Responder a Somfug](#)

Elkin Montañez

11 junio, 2020

Hola me gustaria que me ayudaras, tengo este codigoe n un .bat pero cuando lo ejecuto me solicita la confirmacion de si=s o no=n y me gsutaria dejarlo en una tarea programada que seimpre sea si

set rta=s

net session /delete

set %rta%

EXIT

[Responder a Elkin](#)

gilberto

9 julio, 2020

esta genial este tutorial con las bases del batch ojala se pudiera la segunda parte me gusta la informática y con esto empezare a practicar mas. saludos
exito crack

[Responder a gilberto](#)

Oscar Alejandro

9 julio, 2020

Hola, estoy queriendo programar con archivos bat batch, ya se agregarle color a los script y algunas otras cosas mas, pero quisiera ver si hay algun codigo para que este archivo bat se abra en ventana centrada, ya que se abre muy movido. De antemano gracias

[Responder a Oscar](#)

Diego

6 febrero, 2021

proba con el comando mode /? y vas a ver muchas formas de encuadrar la pantalla del cmd en tu pantalla yo uso el comando «mode 180 » antes de cualquier otro comndo para que se vea el cmd a pantalla completa 😊

[Responder a Diego](#)

gilberto

25 agosto, 2020

buenas tardes esta genial el tutorial pero alguien me podria apoyar con lo siguiente estoy haciendo un menu el cual necesito saber que comando necesito para borrar los archivos temporales (temp) a un no consigo hacer que se ejecute esto es lo que tengo

```
dir C:\Windows\Temp
del /p C:\Windows\Temp\
```

[Responder a gilberto](#)**Ricardo**

26 septiembre, 2020

```
@echo off
cd «C:\Windows\Temp»
del/f/s/p tree
exit
```

[Responder a Ricardo](#)**Danse**

28 octubre, 2020

```
C:
del C:\Windows\Temp\*
.* /P
Intenta con esto
```

[Responder a Danse](#)**fernando**

18 septiembre, 2020

Buenas tardes, me podrías ayudar, al momento de ejecutar un código de msg para chat con otro usuario en red me acepta palabras cortas como «hola» pero al mo-

mento de escribir una palabra mas larga como «hola papa como estas» se produce un error y se sale emitiendo un mensaje que dura 1segundo «no se esperaba (papa)»

Responder a fernando

Ricardo

26 septiembre, 2020

Ejem:

«%msg%»

Si lo pones entre comillas si te dejara porque lo interpretara como texto, así no habrá problemas con oraciones o palabras.

Responder a Ricardo

belgo

30 septiembre, 2020

hola soy nueva, tengo pocos conocimientos de .bat
tengo un script que copia las líneas de un fichero Muestras.txt que contengan la palabra Azul y las guarda en Colores.txt, pero mi fichero Muestras.txt va creciendo cada varios minutos (se le añaden líneas nuevas) y necesito que mi script se ejecute en bucle y cada vez que lea Muestras.txt lo haga a partir de la última línea que leyó para que agregue solamente las nuevas líneas a Colores.txt.
mi script ahora mismo es este

@echo off

setlocal enableextensions enabledelayedexpansion

for /f «tokens=*» %%i in («Muestras.txt») do (

goto Buscar

)

:Buscar

type Muestras.txt | find /i «Azul» >> Colores.txt

endlocal

espero que alguien me pueda ayudar llevo varias semanas tratando de lograrlo.

[Responder a bel9o](#)

Aprendiz

17 octubre, 2020

interesante ...

Para mí que también soy nuevo

[Responder a Aprendiz](#)

nlf

6 octubre, 2020

hola

me podrian ayudar?

nose como ejecutar el scandisk estoy usando notepad

espero me puedan ayudar

[Responder a nlf](#)

Christian Carabali

27 noviembre, 2020

@echo off

SET /P HDD=Por favor digita la letra de la unidad a comprobar:

CHKDSK %HDD%: /X /F /R /B

pause

exit

Recuerda que puedes cambiar los parámetros del CHKSDK y ya no se llama SCANDISK.

Espero te sea de ayuda

[Responder a Christian](#)

SKYNETTAPIA

25 noviembre, 2020

Podrá alguien ayudarme con código para oprimir la tecla «shif» y «f» al mismo tiempo...

[Responder a SKYNETTAPIA](#)**erhy garcia**

25 febrero, 2021

buena tarde

como podria borrar una carpeta sin que tenga encuenta todo el nombre tal como se hace con los archivos pero con las carpetas no funciona del pan?l.txt boraria todos los archivos de texto que empezasen por pan siguiese una letra cualquiera y terminase en l por ejemplo panel, panal etc

tal cual como tu ejemplo necesito hacerlo pero con directorios o carpetas

gracias

[Responder a erhy](#)**Giovanni**

2 marzo, 2021

¡Buenas Tardes!

Necesito una linea de comando para copiar y pegar texto automáticamente al ejecutar un archivo en formato .bath pues no logro hacer que funcione.

Les agradeceria mucho si me brindan su ayuda con esto «y los sigo hasta el cielo.»

[Responder a Giovanni](#)

Jhon Sebastián calvo Villegas

25 marzo, 2021

Amigo buenas noches, muchas gracias por su tutorial, no soy programador pero me gusta mucho lo que hace el bath en Windows. Lo que quiero que me ayudes es que desearía que mi equipo al prender inicie automáticamente la conexión VPN que tengo configurada. Eso es muy fácil si quiero que lo intente una sola vez con el bath; el problema es cuando quiero que reintente y reintente en caso dado que no pueda conectarse bien. El código que utilizo es rasdial «NombreVPn» «user» «password» (sin las comas). Y pues meto el bath en la carpeta de inicio. Lo que desearía es saber como decirle al bath, que si la condición da un valor malo, vuelva e intente ejecutar el mismo comando como si no haya un final.

[Responder a Jhon](#)**ZanCudO**

25 marzo, 2021

Hola 4m1g0 me podrías decir como puedo hacer un bucle con todos los ficheros (.bat) utilizando el comando FOR??

[Responder a ZanCudO](#)**Julio Anguita**

22 abril, 2021

¿Es posible hacer una cuenta atrás que se muestre por pantalla y que al finalizar pase al siguiente comando?

[Responder a Julio](#)**Alejandro**

30 abril, 2021

Les paso el código de una agenda desarrollada en batch, comenten que les parece.

```
@echo off
title ***** AGENDA *****
color 17
MODE CON[:] [COLS=70] [LINES=50]

:agenda
cls
echo.
set dia=%date%
echo ***** %dia% *****
echo.
echo ***** AGENDA *****

echo.
echo 1- Crear contacto
echo.
echo 2- Ver contacto
echo.
echo 3- Buscar
echo.
echo 4- Modificar
echo.
echo 9- Imprimir
echo.
echo 0- Salir
echo.
echo.
set /p opcion=» Elija la opcion -> »
if %opcion%==1 (goto opcion1)
if %opcion%==2 (goto opcion2)
if %opcion%==3 (goto opcion3)
if %opcion%==4 (goto opcion4)
if %opcion%==9 (goto opcion9)
if %opcion%==0 (goto opcion0)

:opcion1
cls
echo.
set /p nom=»Apellido y Nombre: »
set /p dir=»Direccion: »
```

```
set /p tel=»Telefono: »
set /p ocu=»Ocupacion: »
set /p emp=»Empresa: »
echo %nom% %dir% %tel% %ocu% %emp% >> agenda.txt
rem attrib +h agenda.txt
set nom=*
set dir=*
set tel=*
set ocu=*
set emp=*
pause>nul
goto agenda

:opcion2
cls
echo.
for /f «tokens=* delims=, » %%i in (agenda.txt) do (echo %%i)
pause>nul
goto agenda

:opcion3
cls
echo.
echo.
echo Buscar un registro
echo.
echo.
set /p palabra=»Registro a buscar... »
cls
echo.
find /i «%palabra%» «agenda.txt»
pause>nul
goto agenda

:opcion4
cls
echo.
echo.
echo Modificar palabra
echo.
echo.
```

```
set /p pala1-o=» Cambiar palabra ->»
set /p pala1-n=» por esta otra ->»
set pal1-old=%pala1-o%
set pal1-new=%pala1-n%
set file=agenda.txt
echo.
echo.
echo texto Original
echo -----
type %file%
echo -----
for /f «tokens=*» %%a in (%file%) do call :wri %%a
type «%file%.bak» > «%file%»
del /f /q /a «%file%.bak»
echo.
echo.
echo texto modificado
echo -----
type %file%
echo -----
pause
goto :agenda

:wri
set lin=%*
call set lin=%%lin:%pal1-old%=%pal1-new%%%
echo %lin%>>»%file%.bak»
goto :eof

:opcion9
Notepad.exe «agenda.txt»
rem Notepad.exe /p «agenda.txt» agregar /p
rem si quieren que vaya directamente a la impresora
goto :agenda

:opciono
echo.
echo.
echo.
echo.
echo _____
```

```
timeout /t 1 >nul
echo [ ]
timeout /t 1 >nul
echo [oo CERRANDO LA ]
timeout /t 1 >nul
echo [oo APLICACION ]
timeout /t 1 >nul
echo [o ]
timeout /t 1 >nul
echo [o ]
timeout /t 1 >nul
echo [ alasistemas ]
timeout /t 1 >nul
echo [=_____]
timeout /t 1 >nul
echo /\
timeout /t 1 >nul
echo _____
timeout /t 1 >nul
echo \ooooooooooooooooooo\
timeout /t 1 >nul
echo \ooooooooooooooooooo\
timeout /t 1 >nul
echo \ooooooooooooooooooo\
timeout /t 1 >nul
echo _____=====_____
timeout /t 1 >nul
::ipconfig /flushdns
echo.
echo.
echo.
echo.
timeout /t 3 >nul
```

Responder a Alejandro

Maravilloso tutorial, gracias a esto ya se programar en batch!!!

[Responder a Rifmario](#)

Springsito4k

9 julio, 2021

Hola!, tu tutorial estuvo muy bueno la verdad, hice vaqrios comandos divertidos pero...sabes que comando tengo que usar para que todo lo que puse en un archivo .bat se repita en bucle?

[Responder a Springsito4k](#)

Springsito4k

9 julio, 2021

Hola de nuevo

Tengo una duda: como hago para que cuando el archivo .bat ejecute un comando, tarde ciertos segundos y se cierre?

[Responder a Springsito4k](#)

Jose noe martinez

4 agosto, 2021

hola buenas tardes soy nuevo y me pusieron una prueba
necesito un scrip que me diga el contenido de una carpeta
mas o menos tengo la idea de como empezarlo pero no logro que me diga el contenido. me gustaria que me dijera cuantos archivos hay y de que extencion son
haci como un organizador me pudieran ayudar por favor
les pongo los comandos que llevo pero es nomas para abrir un directorio no en si el contenido de la carpeta.

```
Set objShell = CreateObject(«Shell.Application»)
```

```
Set objFolder = objShell.BrowseForFolder _  
(WINDOW_HANDLE, «Selecciona una carpeta:», NO_OPTIONS)
```

```
Set objFolderItem = objFolder.Self
```

```
strPath = objFolderItem.Path
```

```
objShell.Explore strPath
```

Responder a Jose

Jondr

21 enero, 2022

Simple bro, no te compliques.

```
@echo off
```

```
cd «direccion de la carpeta donde estan los archivos que quieres listar»
```

```
dir >lista.txt
```

```
pause
```

```
exit
```

creara una lista donde se pone el numero, nombre, extension, fecha de modificacion y mas.

la lista se creara donde pusiste la direccion, puedes moverlo donde quieras con el comando move.

Responder a Jondr

EL POPOCHAS

3 septiembre, 2021

Genial buena informacion.. una hora batallando y que sencillo es crear batchs para abrir mis juegos retro favoritos de PC-Engine

```
cd C:\Games\Ootake
```

```
«Ootake.exe» «C:\ARCADE.82\Ootake\roms\Soldier Blade (Japan).zip»
```

Responder a EL

Julio

10 septiembre, 2021

Hola

Muy bueno todo.

Deseo saber si es posible leer un número de un archivo, luego presentarlo en pantalla, actualizarlo cada hora en pantalla y que tenga una opción para cerrarlo.

[Responder a Julio](#)**daniel molinos**

3 octubre, 2021

hola queria saber como colocar un tiempo de 30 minutos en un bat que copia carpetas con xcopy pero si yo lo ejecuto ,queria que se haga cada 30 m solo desde ya agradecido

[Responder a daniel](#)**Arturo**

18 diciembre, 2021

Con el «TaskScheduler» de windows puedes definir la partida de un bat o de otros programas.

En castellano se llama «Programador de tareas»

[Responder a Arturo](#)**JesusR**

17 noviembre, 2021

Hola a todos, sabéis como asignar el texto en la casilla Title de un reporte....

Notepad++

```
%% @Title = «Título» %%% @DocName = «Reporte_001» %%
```

en Excel

```
CONCAT(«echo %%%%@Title= «Título»%%%% @DocName =  
«»Reporte_001"» %%%%Location: MD2>»».\Reports\» & RTU!AN7  
&».trep»»&»;
```

[Responder a JesusR](#)

Coco93

27 diciembre, 2021

Hola soy nuevo en el lenguaje de batch y quisiera que alguien me ayudara a decir que es lo que hace exactamente este codigo

```
@echo off
```

```
SET batPath=%~dpo
```

```
pushd %batPath%
```

```
set scriptsDir=%batPath%
```

```
set packagesDir=»%batPath%..\packages»
```

```
SET FUSEOPTION=
```

```
call «%scriptsDir%\bin\rgdeploy.exe» –verbose %FUSEOPTION%  
dir=%packagesDir% %*
```

```
popd
```

quiero saber si es una broma o que
muchas gracias por su ayuda

[Responder a Coco93](#)

← Comentarios más antiguos

 Escribe un comentario

WEBMENCIONES

Elimina los *.bak y *.log que crea el Autocad con Batch en Windows - Jonathan Macalupu ²⁷
diciembre, 2021

[...] Nota: Si quieren saber más sobre ello, existe un tutorial básico en todohacker.com y bueno, que puede ayudarnos a entender mejor la herramienta poderosa que Microsoft pone a nuestra disposición: <https://todohacker.com/tutoriales/lenguaje-batch> [...]

CONTENIDO RELACIONADO POR ETIQUETA **BAT** **BATCH** **PROGRAMACIÓN** **TUTORIAL**

Independent Publisher potencializado por WordPress