**IF**

Conditionally perform a command. Evaluar una condición y dependiendo del resultado, ejecutar un comando.

**File syntax** **Sintaxis para trabajo con archivos**

IF [NOT] EXIST *filename command* IF [NOT] *EXIST* archivo *comando*

IF [NOT] EXIST *filename* ( *command* ) ELSE ( *command* ) IF [NOT] *EXIST* archivo *(comando)* else *(comando)*

**String syntax** **Sintaxis para trabajo con cadenas**

IF IF [/I] [NOT] **item1** == **item2** *command*[/ I] [NOT] **item1 == item2** *comando*

IF [/I] **item1** *compare-op* **item2** *command*IF [/ I] **item1** *compara-op* **item2** comando

IFIF [/I] **item1** *compare-op* **item2** ( *command* ) ELSE ( *command* ) [/ I] **item1** *compara-op* **item2** *(comando)* else *(comando)*

**Error Check Syntax** **Sintaxis para testear errores**

IF [NOT] DEFINED *variable command* IF [NOT] DEFINED *variable comando*

IF [NOT] [ERRORLEVEL](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/errorlevel.html&usg=ALkJrhihQ0Rq95nHVNYaX7N5mN9mt8p60w) *number command* IF [NOT] ERRORLEVEL *número comando*

IF CMDEXTVERSION *number command*IF CMDEXTVERSION *número comando*

**key** **claves**

item May be a text string or an environment variable **ítem**  puede ser una cadena de texto o una variable de entorno.

a variable may be modified using either Una variable puede ser modificada utilizando [Substring syntax](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-substring.html&usg=ALkJrhhWH1WUMXw493jtxLCJPsy8-msepA) or [Search syntax](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-replace.html&usg=ALkJrhg5JPUqOPJEawFXEJks6c8saVJhEQ)LaLaaala expansión de cadenas.

*command* The command to perform **comando** El co*mando*, o comandos a ejecutar.

NOT perform the command if the condition is false. **NOT** Ejecuta el comando si la condición es falsa.

**==** perform the command if the two strings are equal. **==** Ejecutar el comando si las dos cadenas son iguales.

/I Do a case Insensitive string comparison. **/I** Comparación de cadenas que no distinguen entre mayúsculas y minúsculas.

*compare-op* May be one of **comparar-OP** puede ser una de

EQU : Equal EQU: Igual

NEQ : Not equal NEQ: No es igual

LSS : Less than < LSS: Menos que

LEQ : Less than or Equal <= LEQ: Menor o igual

GTR : Greater than > GTR: Mayor que

GEQ : Greater than or equal >= GEQ: Mayor o igual

This 3 digit syntax is necessary because the > aEsta sintaxis de 3 dígitos es necesaria porque el **>** y **<** symbols are recognised as redirection operators son reconocidos como operadores de redirección

IF ERRORLEVEL *n* statements should be read as IF *Errorlevel* >= *number* IF ERRORLEVEL *n* declaración debe leerse como *IF* ERRORLEVEL >= *número* ie es decir,   
IF ERRORLEVEL 0 will return TRUE when the errorlevel is 64 IF ERRORLEVEL 0 devolverá TRUE cuando el errorlevel es de 64

An alternative and often better method of checking Errorlevels is to use the string syntax along with the %ERRORLEVEL% variable: Un método alternativo de comparación de Errorlevels es usar la sintaxis de la cadena junto con la variable **% ERRORLEVEL%**:

IF %ERRORLEVEL% GTR 0 Echo An error was found IF% ERRORLEVEL% GTR 0 echo ha detectado un error   
IF %ERRORLEVEL% LSS 0 Echo An error was found IF% ERRORLEVEL% LSS 0 echo ha detectado un error   
  
IF %ERRORLEVEL% EQU 0 Echo No error found IF% ERRORLEVEL% EQU 0 echo No hay error encontrado   
IF %ERRORLEVEL% EQU 0 (Echo No error found) ELSE (Echo An error was found) IF% ERRORLEVEL% EQU 0 (Echo sin encontrar ningún error) else (Echo ha detectado un error)   
IF %ERRORLEVEL% EQU 0 Echo No error found [||](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-conditional.html&usg=ALkJrhh-oX1AQHlGOGBAO-kPjwuWKZjk0A) Echo An error was found IF% ERRORLEVEL% EQU 0 echo No hay error encontrado [| |](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-conditional.html&usg=ALkJrhh-oX1AQHlGOGBAO-kPjwuWKZjk0A) Eco ha detectado un error

Note some errors are negative numbers. Ten en cuenta que algunos errores son números negativos.   
When working with errorlevels in a batch file it's a good idea to also use [SETLOCAL](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/setlocal.html&usg=ALkJrhiQdrYX0BZK-6DtxzpHaCsfF3oiew) so that the %ERRORLEVEL% variable is reset each time the batch file runs. Cuando se trabaja con errorlevel en un archivo por lotes es una buena idea utilizar también [setlocal](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/setlocal.html&usg=ALkJrhiQdrYX0BZK-6DtxzpHaCsfF3oiew) para que la variable %ERRORLEVEL% se pone a cero cada vez que el archivo por lotes se ejecuta.

IF EXIST *filename* will return true if the file exists (this is not case sensitive). IF *EXIST* archivo devolverá true si el archivo existe (no se trata de mayúsculas y minúsculas).

**Examples:** **Ejemplos:**

IF EXIST C:\install.log (echo complete) ELSE (echo failed) IF EXIST C: \ install.log (eco completar) ELSE (echo no)

IF IF DEFINED \_department ECHO Got the department variable DEFINED departamento echo Tienes la variable departamento

IF DEFINED \_commission SET /A \_salary=%\_salary% + %\_commission% IF IF DEFINED \_department ECHO Got the department variable DEFINED commision SET / A sueldo=%sueldo%+%commision%

IF IF CMDEXTVERSION 1 GOTO start\_processCMDEXTVERSION 1 GOTO start\_process

IF %ERRORLEVEL% EQU 2 goto sub\_problem2 IF %ERRORLEVEL% EQU 2 GOTO subProblema2

**Does %1 exist?** **Existe %1 ?**   
  
To test for the existence of a [command line parameter](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-args.html&usg=ALkJrhgrLvaE_gB1Y_8JOqh1xKGAml3odw) - use empty brackets like this Para probar la existencia de un parámetro de línea de comandos - usar comillas vacíos como en este ejemplo:   
  
IF [%1]==[] ECHO Value Missing IF “%1” == “” ECHO Falta parámetro  
or o   
IF [%1] EQU [] ECHO Value Missing IF ”%1” EQU “” ECHO Falta parámetro  
  
In the case of a variable that may be NULL - a null variable will remove the variable definition altogether, so testing for NULLs becomes easy: En el caso de una variable que puede ser NULL - una variable nula eliminará la definición de la variable por completo, lo prueba para conocer si es nula se hace fácil:   
  
IF NOT DEFINED *\_example* ECHO Value Missing IF NOT DEFINED *ejemplo* ECHO Falta valor   
  
IF DEFINED will return true if the variable contains any value (even if the value is just a space) IF DEFINED devolverá true si la variable contiene un valor (incluso si el valor es sólo un espacio)

**Test the existence of files and folders** **Prueba de la existencia de los archivos y carpetas**

IF EXIST *name* - will detect the existence of a file or a folder - the script [empty.cmd](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/empty.cmd.txt&usg=ALkJrhhUTeHUqxVlAQKpYX0nT-6LfOxOVA) will show if the folder is empty or not. IF EXIST *nombre* Detecta la existencia de un archivo o una carpeta

**BracketsParéntesis**   
  
You can improve the readability of a batch script by writing a complex IF...ELSE command over several lines using [brackets](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-brackets.html&usg=ALkJrhiYbYTX1VWXAfv4tsHsI4xa1y0log) Puedes mejorar la legibilidad de un archivo bat escribiendo un comando IF ... ELSE complejo en varias líneas utilizando los paréntesis

eg por ejemplo,

IF EXIST filename ( IF EXIST archivo (

Del filename Del nombre del archivo

) ELSE ( ) ELSE (

Echo The file was not found. Echo El archivo no fue encontrado.

) )

When using brackets the CMD shell will expand [read] all the [variables](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-variables.html&usg=ALkJrhjN-BgKXL534hnOrFfPTPb7DlntpQ) at the beginning of the code block and use those values even if the variables value has just been changed. Cuando se utilizan paréntesis el Shell CMD expande a su valor [leer] todas las **variables**  al principio del bloque de código y utiliza esos valores, incluso si el valor de las variables ha sido cambiado. Turning on [DelayedExpansion](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/delayedexpansion.html&usg=ALkJrhjT3_U_MfY29eu0IkHerkkCTlEQvQ) will force the shell to read variables at the start of every line. Veremos como activando [**DelayedExpansion**](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/delayedexpansion.html&usg=ALkJrhjT3_U_MfY29eu0IkHerkkCTlEQvQ)seobligará al Shell a leer las variables al comienzo de cada línea.

**Delimiters** **Delimitadores**

If the string being compared by an IF command includes [delimiters](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-esc.html&usg=ALkJrhj1IBjTHfe0qVDewyTPXgp6V6VW6w) such as [Space] or [Comma], then either the delimiters must be escaped with a caret ^ or the whole string must be "quoted". Si la cadena a comparan por un comando IF incluye **delimitadores** tales como [Espacio] o [coma], cualquiera de los delimitadores deben ser precedidos por un acento circunflejo **^** o toda la cadena debe ser envuelta en comillas.   
This is so that the IF statement will treat the string as a single item and not as several separate strings. Esto es para que la instrucción IF trate la cadena como un único elemento y no como varias palabras separadas.

**Testing Numeric values** **Prueba de los valores numéricos**

Do not use brackets or quotes when comparing numeric values No utilices paréntesis o comillas cuando se comparen valores numéricos, eg por ejemplo:   
IF (2) GEQ (15) echo "bigger"

IF (2) GEQ (15) echo "más grande"   
or o   
IF "2" GEQ "15" echo "bigger" IF "2" GEQ "15" echo "más grande"

These will perform a character comparison and will always echo "bigger" Estas instrucciones llevarán a cabo una comparación de carácter y siempre se hará eco de "más grande"   
however the command sin embargo, el comando   
 IF 2 GEQ 15 echo "bigger"IF 2 GEQ 15 echo "más grande"   
Will perform a numeric comparison and works as expected - notice that this behaviour is exactly opposite to the [SET](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/set.html&usg=ALkJrhj0S8c5P_d-8qGUbMSne-D4EX-k1g) /a command where quotes are required. Se realiza una comparación numérica y funciona como se espera .

Ten en cuenta que este comportamiento es exactamente opuesto a la **SET /a comando**, donde las comillas son obligatorias.

The examples here all use GEQ, but this applies equally to all the compare-op operators: EQU, NEQ, LSS, LEQ, GTR, GEQ En los ejemplos uso GEQ, pero esto se aplica igualmente a todos los operadores de comparación: EQU, NEQ, LSS, LEQ, GTR, GEQ when comparing numbers as a string "026" > "26" cuando se comparan los números como una cadena "026"> "26"

**Wildcards** **Los comodines**

Wildcards are not supported by IF, so %COMPUTERNAME%==SS6\* will not match SS64 Los comodines no son soportados por IF, por lo que %COMPUTERNAME% == ES6\* no coincidirá con ES64   
  
A workaround is to retrieve the substring and compare just those characters: Una solución consiste en recuperar la subcadena y comparar sólo los caracteres:

SET \_prefix=%COMPUTERNAME:~0,3% SET prefix =%COMPUTERNAME:~0,3%   
IF %\_prefix%==SS6 GOTO they\_matched IF %prefix% == ES6 GOTO encontrado

**Pipes** **Tuberías**

When [piping](http://translate.googleusercontent.com/translate_c?act=url&hl=es&ie=UTF8&prev=_t&rurl=translate.google.es&sl=en&tl=es&u=http://ss64.com/nt/syntax-redirection.html&usg=ALkJrhi2HYJGdC12872EBTvajV3r7wqUxQ) commands, the expression is evaluated from left to right, so Cuando se unen comandos con tuberías, la expresión se evalúa de izquierda a derecha, por lo que

IF... IFI ... | ... is equivalent to (IF ... ) | ... | ... equivalente a (IFI ...) | ...

you can also use the explicit syntax IF (... | ...) también puedes utilizar la sintaxis explícita IF (... | ...)