

DOS Commands (DOS Prompt)

Command-line reference A-Z

To find information about a command, on the A-Z button menu at the top of this page, click the letter that the command starts with, and then click the command name.

In addition to the tools installed with Windows XP, there are over 40 support tools included on the Windows XP CD. You can use these tools to diagnose and resolve computer problems. For more information about these support tools, see [Windows Support Tools](#). For information about installing support tools, see [Install Windows Support Tools](#).

For more information about changes to the functionality of MS-DOS commands, new command-line tools, command shell functionality, configuring the command prompt, and automating command-line tasks, see [Command-line reference](#).

Command-line tools must be run at the prompt of the Cmd.exe command interpreter. To open Command Prompt, click **Start**, click **Run**, type **cmd**, and then click **OK**. To view help at the command-line, at the command prompt, type the following:

CommandName /?

A

[Arp](#)

[Assoc](#)

[At](#)

[Atmadm](#)

[Attrib](#)

B

[Batch files](#)

[Bootcfg](#)

[Break](#)

C

[Cacls](#)

[Call](#)

[Certreq](#)

[Certutil](#)

[Change](#)

[Chcp](#)

[Chdir](#)

[Chkdsk](#)

[Chkntfs](#)

[Cipher](#)

[Cls](#)

[Cluadmin](#)

[Cluster](#)

[Cmd](#)

[Cmstp](#)

[Color](#)

[Command shell overview](#)

[Comp](#)

[Compact](#)

[Convert](#)

[Convlog](#)

[Copy](#)

[Cprofile](#)

[CScript overview](#)

[CSVDE](#)

D

[Date](#)

[DCGPOFix](#)

[Defrag](#)

[Del](#)

[Dfscmd](#)

[Dir](#)

[Diskcomp](#)

[Diskcopy](#)

[DiskPart](#)

[Doskey](#)

[Driverquery](#)

[Ds \(directory service\)](#)

[Dsadd](#)

[Dsget](#)

[Dsmod](#)

[Dsmove](#)

[Dsquery](#)

[Dsrm](#)

E

[Echo](#)

[Endlocal](#)

[Eventcreate](#)

[Eventquery](#)

[Eventtriggers](#)

[Evtcmd](#)

[Exit](#)

[Expand](#)

F

[Fc](#)

[Filter_commands](#)

[Find](#)

[Findstr](#)

[Finger](#)

[Flattemp](#)

[For](#)

[Format](#)

[Esutil](#)

[Etp](#)

[Etp_subcommands](#)

[Etype](#)

G

[Getmac](#)

[Goto](#)

[Gpresult](#)

[Gpupdate](#)

[Graftabl](#)

H

[Help](#)

[Helpctr](#)

[Hostname](#)

I

[If](#)

[Iisapp](#)

[Iisback](#)

[Iiscnfg](#)

[Iisftp](#)

[Iisftpr](#)

[Iisreset](#)

[Iisvdir](#)

[Iisweb](#)

[Ipconfig](#)

[Ipxroute](#)

[Irftp](#)

L

[Label](#)

[Ldifde](#)

[Lodctr](#)

[Logman](#)

[Lpq](#)

[Lpr](#)

M

[Macfile](#)

[Mkdir \(md\)](#)

[Mmc](#)

[Mode](#)

[More](#)

[Mountvol](#)

[Move](#)

[Mrinfo](#)

[MS-DOS subsystem configuration commands](#)

[Msiexec](#)

[Msinfo32](#)

N

[Nbtstat](#)

[Net services overview](#)

[Net services commands](#)

[Netsh](#)

[Netsh commands for AAAA](#)

[Netsh commands for DHCP](#)

[Netsh diagnostic \(diag\) commands](#)

[Netsh commands for IP](#)

[Netsh commands for RAS](#)

[Netsh commands for Routing](#)

[Netsh commands for WINS](#)

[Netstat](#)

[Nlb](#)

[Nslookup](#)

[Nslookup_subcommands](#)

[Ntbackup](#)

[Ntcmdprompt](#)

[Ntdsutil](#)

[Ntsd](#)

O

[Openfiles](#)

P

[Pagefileconfig](#)

[Path](#)

[Pathping](#)

[Pause](#)

[Pbadmin](#)

[Pentnt](#)

[Perfmon](#)

[Ping](#)

[Popd](#)

[Print](#)

[Prncnfg](#)

[Prndrvr](#)

[Prnjobs](#)

[Prnmngr](#)

[Prnport](#)

[Prnqctl](#)

[Procon](#)

[Prompt](#)

[Pushd](#)

Q

[Query](#)

R

[Rasdial](#)

[Rcp](#)

[Recover](#)

[Redirection operators](#)

[Reg](#)

[Regsvr32](#)

[ReLog](#)

[Rem](#)

[Rename](#)

[Replace](#)

[Reset session](#)

[Rexec](#)

[Risetup](#)

[Rmdir](#)

[Route](#)

[Rsh](#)

[Rsm](#)

[Rss](#)

[Runas](#)

S

[Sc](#)

[Schtasks](#)

[Secedit](#)

[Set](#)

[Setlocal](#)

[Shift](#)

[Shutdown](#)

[Sort](#)

[Start](#)

[Subst](#)

[Systeminfo](#)

[System File Checker \(sfc\)](#)

T

[Tapicfg](#)

[Taskkill](#)

[Tasklist](#)

[Tcmsetup](#)

[TCP/IP utilities and services](#)

[Telnet commands](#)

[Terminal Services commands](#)

[Iftp](#)

[Time](#)

[Title](#)

[Tracerpt](#)

[Tracert](#)

[Tree](#)

[Tsecimp](#)

[Type](#)

[Typeperf](#)

U

[Unlodctr](#)

V

[Ver](#)

[Verify](#)

[Vol](#)

[Vssadmin](#)

W

[W32tm](#)

[Winnt](#)

[Winnt32](#)

[Wlbs](#)

[WMIC overview](#)

X

[Xcopy](#)

Batch Files

Using batch files

With batch files, which are also called batch programs or scripts , you can simplify routine or repetitive tasks. A batch file is an unformatted text file that contains one or more commands and has a .bat or .cmd file name extension. When you type the file name at the command prompt, Cmd.exe runs the commands sequentially as they appear in the file.

You can include any command in a batch file. Certain commands, such as **for**, **goto**, and **if**, enable you to do conditional processing of the commands in the batch file. For example, the **if** command carries out a command based on the results of a condition. Other commands allow you to control input and output and call other batch files.

The standard error codes that most applications return are 0 if no error occurred and 1 (or higher value) if an error occurred. Please refer to your application help documentation to determine the meaning of specific error codes.

For more information about batch file operations, see the following topics:

- ✎ [Using batch parameters](#)
- ✎ [Using filters](#)
- ✎ [Using command redirection operators](#)

For more information about commands that you can use in batch files, click a command:

- ✎ [Call](#)
- ✎ [Echo](#)
- ✎ [Endlocal](#)
- ✎ [For](#)
- ✎ [Goto](#)
- ✎ [If](#)
- ✎ [Pause](#)
- ✎ [Rem](#)
- ✎ [Setlocal](#)
- ✎ [Shift](#)

Batch Files: Echo

Echo

Turns the command-echoing feature on or off, or displays a message. Used without parameters, **echo** displays the current echo setting.

Syntax

```
echo [{on|off}] [message]
```

Parameters

{on|off}

Specifies whether to turn the command-echoing feature on or off.

message

Specifies text you want to display on the screen.

/?

Displays help at the command prompt.

Remarks

- ⚡ The **echo message** command is useful when echo is turned off. To display a message that is several lines long without displaying other commands, you can include several **echo message** commands after the **echo off** command in your batch program.
- ⚡ If you use **echo off**, the command prompt does not appear on your screen. To display the command prompt, type **echo on**.
- ⚡ To prevent echoing of a line, insert an at sign (@) in front of a command in a batch program.
- ⚡ To echo a blank line on the screen, type:
echo.
- ⚡ To display a pipe (|) or redirection character (< or >) when you are using **echo**, use a caret character immediately before the pipe or redirection character (for example, ^>, ^<, or ^|). If you need to use the caret character (^), type two (^ ^).

Examples

The following example is a batch program that includes a three-line message preceded by and then followed by a blank line:

```
echo off
echo.
echo This batch program
echo formats and checks
echo new disks
echo.
```

If you want to turn echo off and you do not want to echo the **echo** command, type an at sign (@) before the command as follows:

```
@echo off
```

You can use the **if** and **echo** commands on the same command line. For example:

```
if exist *.rpt echo The report has arrived.
```

Formatting legend

Format	Meaning
<i>Italic</i>	Information that the user must supply
Bold	Elements that the user must type exactly as shown
Ellipsis (...)	Parameter that can be repeated several times in a command line
Between brackets ([])	Optional items
Between braces ({}); choices separated by pipe (). Example: {even odd}	Set of choices from which the user must choose only one
Courier font	Code or program output



[Pause](#)

Batch Files: If

If

Performs conditional processing in batch programs.

Syntax

if [**not**] **errorlevel** *number* *command* [**else** *expression*]

if [**not**] *string1* ==*string2* *command* [**else** *expression*]

if [**not**] **exist** *filename* *command* [**else** *expression*]

If command extensions are enabled, use the following syntax:

if [/i] *string1* *CompareOp* *string2* *command* [**else** *expression*]

if **cmdextversion** *number* *command* [**else** *expression*]

if **defined** *variable* *command* [**else** *expression*]

Parameters

not

Specifies that the command should be carried out only if the condition is false.

errorlevel *number*

Specifies a true condition only if the previous program run by Cmd.exe returned an exit code equal to or greater than *number*.

command

Specifies the command that should be carried out if the preceding condition is met.

string1 ==*string2*

Specifies a true condition only if *string1* and *string2* are the same. These values can be literal strings or batch variables (for example, %1). You do not need to use quotation marks around literal strings.

exist *filename*

Specifies a true condition if *filename* exists.

CompareOp

Specifies a three-letter comparison operator. The following table lists valid values for *CompareOp*.

Operator	Description
EQU	equal to
NEQ	not equal to
LSS	less than
LEQ	less than or equal to
GTR	greater than
GEO	greater than or equal to

/i

Forces string comparisons to ignore case. You can use /i on the *string1* ==*string2* form of **if**. These comparisons are generic, in that if both *string1* and *string2* are both comprised of all numeric digits, the strings are converted to numbers and a numeric comparison is performed.

cmdextversion *number*

Specifies a true condition only if the internal version number associated with the Command Extensions feature of Cmd.exe is equal to or greater than *number*. The first version is 1. It is incremented by one when significant enhancements are added to the command extensions. The **cmdextversion** conditional is never true when command extensions are disabled (by default, command extensions are enabled).

defined *variable*

Specifies a true condition if *variable* is defined.

expression

Specifies a command-line command and any parameters to be passed to the command in an *else* clause.

/?

Displays help at the command prompt.

Remarks

- ⚡ If the condition specified in an **if** command is true, the command that follows the condition is carried out. If the condition is false, the command in the **if** clause is ignored, and executes any command in the **else** clause, if one has been specified.
- ⚡ When a program stops, it returns an exit code. You can use exit codes as conditions by using the *errorlevel* parameter.
- ⚡ Using **defined** variable

If you use **defined** variable, the following three variables are added: **%errorlevel%**, **%cmdcmdline%**, and **%cmdextversion%**.

%errorlevel% expands into a string representation of the current value of *errorlevel*, provided that there is not already an environment variable with the name ERRORLEVEL, in which case you get the ERRORLEVEL value instead. The following example illustrates how you can use **errorlevel** after running a batch program:

```
goto answer%errorlevel%
:answer0
echo Program had return code 0
:answer1
echo Program had return code 1
goto end
:end
echo done!
```

You can also use the *CompareOp* comparison operators as follows:

```
if %errorlevel% LEQ 1 goto okay
```

%cmdcmdline% expands into the original command line passed to Cmd.exe prior to any processing by Cmd.exe, provided that there is not already an environment variable with the name **cmdcmdline**, in which case you get the **cmdcmdline** value instead.

%cmdextversion% expands into the a string representation of the current value of **cmdextversion**, provided that there is not already an environment variable with the name CMDEXTVERSION, in which case you get the CMDEXTVERSION value instead.

- ⚡ Using the **else** clause

You must use the **else** clause on the same line as the command after the **if**. For example:

```
IF EXIST filename. (
del filename.
) ELSE (
echo filename. missing.
)
```

The following code does not work because you must terminate the **del** command by a new line:

```
IF EXIST filename. del filename. ELSE echo filename. missing
```

The following code does not work because you must use the **else** clause on the same line as the end of the **if** command:

```
IF EXIST filename. del filename.
ELSE echo filename. missing
```

If you want to format it all on a single line, use the following form of the original statement:

```
IF EXIST filename. (del filename.) ELSE echo filename. missing
```

Examples

If the file Product.dat cannot be found, the following message appears:

```
if not exist product.dat echo Can't find data file
```

If an error occurs during the formatting of the disk in drive A, the following example displays an error message:

```
:begin
@echo off
format a: /s
```

```
if not errorlevel 1 goto end
echo An error occurred during formatting.
:end
echo End of batch program.
```

If no error occurs, the error message does not appear.

You cannot use the **if** command to test directly for a directory, but the null (NUL) device does exist in every directory. As a result, you can test for the null device to determine whether a directory exists. The following example tests for the existence of a directory:

```
if exist c:mydir\nul goto process
```

Formatting legend

	Format	Meaning
	<i>Italic</i>	Information that the user must supply
	Bold	Elements that the user must type exactly as shown
	Ellipsis (...)	Parameter that can be repeated several times in a command line
	Between brackets ([])	Optional items
	Between braces ({}); choices separated by pipe (). Example: {even odd}	Set of choices from which the user must choose only one
	Courier font	Code or program output

MODE

Configures system devices.

Serial port: **MODE COM***n*[:] [BAUD=*b*] [PARITY=*p*] [DATA=*d*] [STOP=*s*]
 [*to*=on|off] [*xon*=on|off] [*odsr*=on|off]
 [*octs*=on|off] [*dtr*=on|off|hs]
 [*rts*=on|off|hs|tg] [*idsr*=on|off]

Device Status: **MODE** [*device*] [/STATUS]

Redirect printing: **MODE** LPT*n*[:]=COM*n*[:]

Select code page: **MODE** CON[:] CP SELECT=*yyy*

Code page status: **MODE** CON[:] CP [/STATUS]

Display mode: **MODE** CON[:] [COLS=*c*] [LINES=*n*]

Typematic rate: **MODE** CON[:] [RATE=*r* DELAY=*d*]