



## MS-DOS Commands



Objectives:

[Internal commands](#)

[External commands](#)

[Input/Output Redirection and Filter commands](#)

[Examples](#)

Let's discuss MS-DOS commands from COMMAND.COM. It's a command processor which works as an interface between you, the user, and DOS. It basically interprets what you have typed at the DOS prompt and processes them. Commands can be categorized into INTERNAL (memory resident) and EXTERNAL (utilities / programs) types.

---

### INTERNAL commands [ [Top](#) ]

Internal commands are memory resident commands. They are resident in the memory when the COMMAND.COM is loaded in the boot up process. Table 1 below shows the list of internal commands.

BREAK	DEL or ERASE	MKDIR or MD	SHIFT
CALL	DIR	PATH	TIME
CHCP	ECHO	PAUSE	TYPE
CHDIR or CD	EXIT	PROMPT	VER
CLS	FOR	REM	VERIFY
COPY	GOTO	RENAME or REN	VOL
CTTY	IF	RMDIR or RD	
DATE	LOADHIGH or LH	SET	

Table 1 -- List of Internal Commands.

(click [here](#) to view the brief description of these commands)

---

### EXTERNAL commands [ [Top](#) ]

**External commands are MS-DOS utilities / programs. These are the .EXE or .COM programs located on your hard drive. They are normally placed under C:\DOS, the default directory. MS-DOS will load external commands if and only if you instruct to execute them at the DOS prompt and /or in a batch program.**

Click [here](#) to view the brief description of external commands.

---

## **Input/Output Redirection and Filter Commands** [ [Top](#) ]

So far, you might have learned how to use some commands in their standard form, e.g. to display the content of a text file, to view a directory listing or to view the tree structure of directories. You can view the output directly on the screen.

Using MS-DOS command in standard form, you type in your instruction at the command prompt, through the keyboard. It's a *standard input*. Then, MS-DOS will process your input and display the result on the screen. It's a *standard output*. Up to this point, the job is simply done.

Now, if you want to record a directory listing for further reference or get a hard copy immediately, you need to instruct MS-DOS not to display the result on the screen but to save it to a file or print it out directly. It's called *output redirection*.

On the other hand, if you want to process a job by using data on an existing file, you need to instruct MS-DOS to use data from a given file. It's called *input redirection*.

In some cases, moreover, you may need to process a job by executing more than one command. By the way, the output from the first command will not be displayed on the screen but to be redirected into the input of the next command. It involves both output and input redirection. The logical connection between these two commands is called *pipng*.

Piping is associated with filter commands: *find*, *sort* and *more*. "They are called filter commands because they work much like a filter in a water system, which takes incoming water, changes it in some way, and sends it along the system (a good explanation from Van Wolverton's *Running MS-DOS*)." The application of these filter commands are listed as follow. They can be used simply in an input redirection or through a logical piping connection.

- *Find* searches input data for a text string.
- *Sort* arranges input data in order.
- *More* displays output one screen at a time.

The following special characters are needed in Input/Output redirection and Piping:

>	<u>Redirects</u> standard output of a command or program <u>to a device or a file.</u> [Output redirection]	MS-DOS will create the named file if it does not exist or overwrite it (without prompting to confirm) if it already exists.
>>	<u>Adds</u> standard output of a command or program <u>to a file.</u> [Output redirection]	MS-DOS will create the named file if it does not exist.
<	<u>Feeds data</u> from a file as the input of a command or program. (i.e. to read in input from a named file) [Input redirection]	<ul style="list-style-type: none"> <li>• The data from a given file substitute a standard input from keyboard.</li> </ul>
	Feeds the <u>output</u> of a command or program directly into the input of a filter command. [Piping]	<ul style="list-style-type: none"> <li>• It applies to logical connection between two commands or programs only. (i.e. if the output of the first command cannot be used as an input for the next command, it's not a logical connection and therefore is not applicable.)</li> <li>• The connection is called a <i>pipe</i>.</li> <li>• Piping is used together with filter commands: Sort, Find and More.</li> </ul>

**Examples** [ [Top](#) ]

**This section will demonstrate those commands frequently asked by the students. Before I continue, let me advise you that **don't try to hard memorise all the examples from this paper only but to practise on the computer until you fully understand the concepts.****

Commands & its	Explanation
----------------	-------------

syntax	
<b>Output redirection</b>	
(1) C:\> <b>DIR dos\*.com</b> > <b>comfile.txt</b>	<ul style="list-style-type: none"> <li>• Redirects a directory listing of all .com files in the DOS directory to a file called comfile.txt in the current directory.</li> </ul>
(2) C:\> <b>TYPE</b> <b>readme.txt</b> > <b>prn</b>	<ul style="list-style-type: none"> <li>• Redirects the content of the file called readme.txt to a print device. (prn is a reserved name in MS-DOS. It refers to a print device.)</li> <li>• Same result as using the PRINT command.i.e. C:\&gt;PRINT readme.txt</li> </ul>
<b>Output redirection</b>	
(3a) C:\> <b>PATH /?</b> > <b>path.hlp</b> (3b) C:\> <b>APPEND /?</b> >> <b>path.hlp</b>	<ul style="list-style-type: none"> <li>• In (3a), redirects the content of the built-in help program on PATH command to a file called path.hlp.</li> <li>• If the file does not exist, DOS will create it for you. If it already exists, DOS will overwrite the existing content by the new content.</li> </ul>
	<ul style="list-style-type: none"> <li>• In (3b), adds the content of the built-in help program on APPEND command to a file called path.hlp.</li> <li>• If the file does not exist, DOS will create it for you.</li> </ul>
<b>Input redirection</b>	
(4) C:\> <b>MORE</b> < <b>c:\dos\readme.txt</b>	<ul style="list-style-type: none"> <li>• The MORE command reads in input from a file called readme.txt under C:\DOS directory.</li> <li>• It displays the output one screen at a time (to give you sufficient time to view the content). You can press any key to continue the display.</li> </ul>
<b>Output / Input redirection</b>	
(5) C:\> <b>SORT /R</b> < <b>phone1.txt</b> > <b>sort_ph.txt</b>	<ul style="list-style-type: none"> <li>• The SORT command reads in input from a file called phone1.txt and arranges the data in an descending order.</li> <li>• Then, it redirects the output to a file called sort_ph.txt</li> </ul>
(6) C:\> <b>SORT</b> < <b>phone2.txt</b> > <b>prn</b>	<ul style="list-style-type: none"> <li>• The SORT command reads in input from a file called phone2.txt and arranges the data in an ascending order.</li> <li>• Then, it redirects the output to a print device.</li> </ul>

## Piping

(7) C:\>**DIR** | **FIND /C**  
**"04-16-00"**

- Feeds a directory listing (output of DIR) directly into the input of the FIND filter to count how many files and directories created on 04-16-00.

(8) C:\>**FIND /I**  
**"marketing" phone3.txt**  
| **SORT**

- Searches the given file phone3.txt for all lines containing the text string of "marketing".
- Then, feeds the output directly into the input of SORT filter to arrange all lines in ascending order.
- Additional note for filter commands:
  - Basically, all filter commands are used in association with a input redirection (<) or piping (|).
  - But, FIND command can also open a file for processing, i.e. the file phone3.txt in this example.

