

# An Overview of MS-DOS

---

## MS-DOS as an Operating System

**An operating system** consists of one or more programs that control the allocation and usage of hardware resources (e.g. memory, CPU time, disk space, peripheral devices) and function as an interface between the user, application programs, and the hardware.

Microsoft Corp. MS-DOS 1.0 came out in 1981. Since then there have been a number of upgraded versions, the last being Version 6 Release 22 (v6.22). Versions are upward compatible.

Unlike GUI (Graphical User Interface) Operating Systems like Windows XP, MS-DOS has a command-line interface. That is commands are typed at the so-called MS-DOS prompt

```
C:\DOS>
```

where *c:* identifies the *default drive* and *\DOS* identifies the *default path*. The *top level* directory, identified by a single *\* as in *c:\>* is called the *root directory*.

---

## Classifying MS-DOS Commands

MS-DOS commands fall roughly into three categories

### Environment Commands

which report on or affect the operating system environment. Examples are CLS (clear screen), TIME, DATE, VER (display MS-DOS version number), and HELP.

### File Manipulation Commands

which manipulate files. Examples are COPY, DEL (delete), TYPE (display file to screen) and, DIR (directory - or list all files in current directory).

### Utilities

which perform some useful function. Examples are FORMAT (format a diskette) and EDIT (invoke MS-DOS text editor).

---

## MS-DOS File Specifications

All files under MS-DOS are specified by a four part naming convention. MS-DOS File Manipulation commands identify files in this manner.

```
drive:\path\filename.ext
```

where

drive:

is a single letter identifying the device where the file is found (e.g. *c:*)

\path \

is the hierarchy of directories (folders) where the file is found. Directories are separated by \ (back-slash character)

filename

is the file name of from 1 to 8 letters and or digits. Blanks and certain special characters (e.g. \ or .) are not allowed.

.ext

is the file type or extension of up to 3 non-blank characters. File types or extensions are used to identify the type of file where certain naming conventions are used. For example, .exe identifies an executable file, .txt identifies an ASCII text file, and .doc and .wpd identifies MS-DOS Word and WordPerfect documents.

**Remember** all directory names and file names are limited to 8 characters (digits, or letters) although *MS-DOS running under Windows NT allows longer names*. Embedded blanks are NOT allowed and in general special characters are not allowed.

Example c:\dos\autoexec.bat

---

## Some MS-DOS File Manipulation Commands

Commands that reference files may either give the full file specification (drive:\path\filename.ext) or a partial file specification where the *default drive* and *pathname* values given by the MS-DOS prompt are used. The rule is *when the drive name and/or path is not given, the default values are used*.

For Example, full file specifications are given by

```
C:\DOS> copy a:\file1.txt c:\dos\file1.txt
```

Alternately, one can use the *default* values where the target of the copy command defaults to c:\dos.

```
C:\>DOS> copy a:\file1.txt file1.txt
```

---

In the commands given below, we adopt the convention that anything enclosed in square brackets, [ ], is optional. For example, [x:\path \] refers to an optional device and/or path.

```
copy [x:\path \]source.ext [x:\path \]target.ext  
copies source file to target file.
```

```
del [x:\path \]filename.ext  
deletes file
```

```
move [x:\path \]source.ext [x:\path \]target.ext  
moves source to target
```

```
type [x:\path \]filename.ext
```

displays contents of file to screen. Note, only works for ASCII text files.

```
rename [x:\path \]source target.ext
```

renames file. Note, target must be in same directory as source.

```
dir [x:\path \filename.ext]
  lists all files in directory.
```

---

## Working with Directories and Path Names

The following commands can be used to create and remove subdirectories and set the default drive and directory. Again anything in square brackets [ ] is optional.

- To change the default drive, at the MS-DOS prompt type the drive letter followed by a colon. For example, to make a: the default drive

```
C:\> a:
```

- To change the default path, use `chdir` or `cd`, the *change directory* command. For example, to make FILES, a subdirectory of DOS the default use

```
C:\DOS> chdir files
```

To make the root directory the default, use

```
C:\DOS> cd \
```

- To make (i.e. create) a directory use `mkdir` or `md`, the make directory command. That is

```
mkdir [\path\]directory
```

```
md [\path\]directory
```

- To remove a directory use `rmdir` or `rd`, the remove directory command. That is

```
rmdir [\path \]directory
```

```
rd [\path\]directory
```

Important! The directory must be empty before you remove it

- To display the directory *structure* use

```
tree [drive:/path/]
```

---

## Using the more command

One of the problems with MS-DOS commands is "screen overflow" - more material is displayed than can be held on screen. To get around this problem use the **more** command

To display the contents of a file one screen at a time use

```
c:\> more filename.txt
```

To display execute an MS-DOS display command in conjunction with more use `c:\> cmd | more`. For example

```
c:\> dir | more
```

The vertical bar "|" character is called a "pipe"

---

## Miscellaneous MS-DOS

- \* is used as a wildcard symbol. For example

```
c:\> dir h:\*.exe
```

will display all files in h:\ with an .exe extension

- To execute any application or run executable file (any file with an .exe or .com extension), simply type the file name at the MS-DOS prompt. This is like the *run* command in Windows.
- EDIT is a full-screen text editor that comes with MS-DOS. You can use EDIT to create and edit text files. (Note MS-DOS EDIT is somewhat similar to Note Pad in Windows).

```
C:\> edit hello.txt
```

- Use the Help command to display information about all MS-DOS commands.

```
C:\> help
```

- A single period "." and a double period ".." refer to the current directory and the parent directory respectively. They can be used to specify *relative paths*. For example

```
C:\DOS\FILES> chdir ..  
C:\DOS>
```

will change the default path to C:\DOS.