**DISK OPERATING SYSTEM (DOS)**

It is a system software, which is closely associated with the computer hardware and provides the interface between the user and resources of the computer, such as central processing unit, memory, files and I/O devices. In simple words, we can say that, it is a program which supervises and controls the operation of a computer.

As DOS was written by Microsoft Corporation, usually it is called MS-DOS. IBM has been licensed to use and sell the same DOS with their computer. In that case, when it marketed by IBM, the DOS is called PC-DOS, there are many operating systems, like UNIX, OS/2, VMS, etc., but DOS is the most popular operating system.

Knowledge about DOS is very essential for computer users, otherwise it is not possible to use the computer efficiently. It is not essential to have a thorough knowledge about DOS, some of the basic commands will be sufficient for most of the purposes. The commands are discussed in this chapter. The recent versions of MS-DOS contain extensive online help. You can refer to it any time by typing HELP at the command prompt. There are many functions performed by DOS, few of them are listed below.:

1. Coordinates the input/output devices such as monitor, printers, plotters, disk drivers, etc.
2. Configures the computer hardware for required purposes.
3. Maintains an orderly system of files on the disk.
4. Customizes for international use (you can change the keyboard layout, character set and

even the language according to your choice within a given set).

1. Manages computer’s memory.
2. Enables the user to load and execute the programs.

**SETTING UP DOS**

As soon as you switch on the computer, two special built-in programs are executed from the start up program called bootstrap loader. The first does a self testing of the computer to see that things are in the working order and second one starts up DOS. The bootstrap operation works in two stages in a file named COMMAND.COM and made memory resident. These commands are called internal commands. You cannot see them in a directory list as these are part of COMMAND. COM file. The rest of the commands are called external commands, and are stored in a separate file individually. They occupy the memory only when you call them

**Internal commands**

As said earlier, the internal commands are stored in **COMMAND.COM** file and these will always be present in the memory. You can access these commands irrespective of the directory in which you are working. Before discussing the **DOS** commands, it is essential to know how the information is organized in the computer. As you know, a complete collection of related data is called a file. The data here refers to any item of information that you want to save in a computer memory. For example, a **FORTRAN** program file, a letter, a mailing list, etc. Each individual file will have a specific name. The file name consists of two parts, file name and extensions separated by a period (.), e.g., **COMMAND. COM**. Though the use of extension is not compulsory for user’s file, they will be useful in many ways. Some of the application software/languages assign a specific extension to the file name given by the user automatically. For example, **FORTRAN** will assign “for” as extension to all program. No two files can have the same names and extension in the same directory. A group of such files constitutes a directory. Usually, all the files related to software are stored in a single directory. If you need to group user files and store them in another **directory**, you can always do so by creating a subdirectory. This is particularly a good practice because you are separating user files from the software files, thus protecting the software files from getting corrupted unknowingly. With this brief introduction to files and directories, we shall now move on to the internal commands.

**DIR:** This command displays a list of files and subdirectories that all in the directory you specify.

*Syntax Dir [drive] [Path] [File name] [/P] [/W]*

Example: dir C:\ws7\macros\ws6macro.exe Displays the file WS6MACROS.EXE in the subdirectory ‘MACROS of WS7 directory which is in drive C. /P and /W are called switches. /P displays one screen of listing at a time. To see next screen press any key. /w displays the listing in wide format with as many as five files/directory names on each line**.**

**Using Wildcards:** The symbol\* (asterisk) and ? (question mark) are called wildcards in DOS. You can use them to display a listing of subsets of files and subdirectories. The following examples illustrate the use of wildcards.

Example: 1. dir C:\ws7\*.txt, displays all the files and subdirectories with extensions, .txt in the parent directory ‘w7’ in ‘C’ drive.

Example 2. dir C:\ws7\.txt, displays all the files and subdirectories whose names being with ‘ws’ irrespective of the trailling (following) characters in the file name of parent directory ‘ws7’ in ‘C’ drive. **CD or CHDIR**, displays the name of the current directory or changes the current directory to the specified directory.

**Syntax 1.** *CD [drive] [Path], or 2, cd…, or 3. cd/* in case of 1, the directory will be changed to the one you specify for ex. cd C:\dos changes the directory to the directory named ‘dos’. If you are not changing the drive you need not specify the drive name. In case 2, you will be changing back to parent directory from subdirectory. In case 3, you will be changing the root directory from current directory. The root directory is the top of the directory hierarchy for a drive.

**MD (Make Directory):** Using this command, you can create a directory or a subdirectory under a parent directory. Syntax ***MD*** *[drive:] [Path] [directory name] [drive:]* Specifies the drive on which you want to create a ‘new directory’. [Path] [directory name] Specifies the name and location of the new directory. For example, to create a new directory named **FORTRAN** on drive A: simply type the following and press enter key. **MD A:\FORTRAN**

**RD (Remove Directory):** Using this command, you can delete a directory provided the directory is empty, i.e., it does not contain any files in it.

Syntax **RD** *[drive:] [Path] [directory name]* The components of this syntax has the same meaning as in MD.

**Copy:** Copy one or more files to the location you specify. Syntax *Copy [source] [destination].* Source:  Specifies the location and name of a file or a set of files from which you want to copy. For example, if you want to copy a file named readme.txt under ‘ws7’ directory in drive C: you can specify it as follows. C:\ws7\readme.txt

**CLS:** This command clears the screen. The cleared screen will have command prompt and cursor only at the top left corner. Syntax *CLS*

**REN OR RENAME**: Changes the name of the files you specify. Syntax *REN [Drive:] [Path] [Filename1 Filename2]* You can also make use of wildcards to change the name of a group of files that matches your specification You cannot change name of a directory or subdirectory and move the files across the drives using the command.

**Exercise:**

1. Distinguish between internal and external DOS commands.

2. What are wildcards in DOS?

3. Explain the booting process.

4. Give syntax and explain:

(a) MD

(b) RD

(c) COPY

(d) CLS

(e) REN

5. Write a short note on ‘Function Keys’.

**WINDOWS OPERATING SYSTEM**

*Definition:* Windows is a series of operating systems developed by Microsoft. Each version of Windows includes a graphical user interface, with a desktop that allows users to view files and folders in windows. For the past two decades, Windows has been the most widely used operating system for personal computers PCs.

The earliest versions of the Microsoft Windows operating system were really just applications that ran on top of the MS-DOS operating system. *Modern versions of Windows (Windows 95 and greater) no longer run on top of DOS, but do retain a DOS-like user interface called the "MS-DOS Prompt" or "Command Prompt".* The command prompt allows Windows users to issue DOS commands to the system.

*DOS commands are useful to Windows users for a variety of reason*s. Many tasks that can be performed in the Windows graphical environment can be performed much more quickly and efficiently by utilizing DOS commands. Also, most bootable floppy disks will boot the system directly to a DOS environment.

Users of any Windows system can invoke a command prompt by performing the following steps:

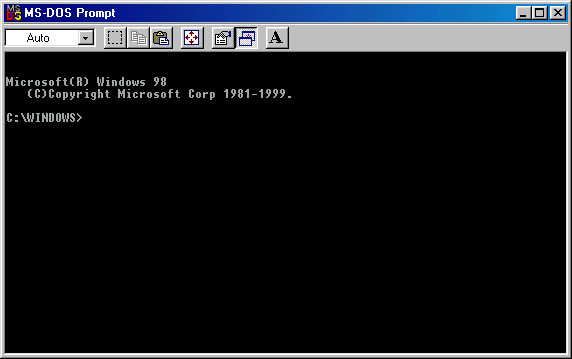
1. Click Start.
2. Select Run.
3. In the "Open:" field, type command.
4. Click OK.

A command prompt window is generally white or gray text on a black background. The prompt itself usually takes the following form:

X:\PATH>

where "X" represents the letter of the active drive, and "PATH" refers to the present working directory. Here's an example of how a DOS command prompt commonly looks:

C:\Windows>\_



Commands are typed and echoed to the screen at the cursor's location. Usually commands consist of a keyword, followed by optional flags, followed by a path. More will be explained about syntax and additional flags under each specific command.

**DIR**: Displays a list of files and subdirectories in a directory.

**SYNTAX:** DIR [drive:][path][filename] [/P] [/W] [/A[[:]attributes]] [/O[[:]sortorder]] [/S] [/B] [/L] [/V][drive:][path][filename] Specifies drive, directory, and/or files to list.

* /P Pauses after each screenful of information.
* /W Uses wide list format.
* /A displays only files with these attributes:

D Directories

R Read-only files

H Hidden files

A Files ready for archiving

S System files

- Prefix meaning not: can be added to above attributes

* /O List by files in sorted order, sortorder:

N By name (alphabetic)

S By size (smallest first)

E By extension (alphabetic)

D By date & time (earliest first)

G Group directories first

A By Last Access Date (earliest first)

- Prefix to reverse order: put in front of above

* /S Displays files in specified directory and all subdirectories.
* /B Uses bare format (no heading information or summary).
* /L Uses lowercase.
* /V Verbose mode.

**EXAMPLES**

* dir - Lists all files and directories in the directory that you are currently in.
* dir /s - Lists the files in the directory that you are in and all sub directories after that directory.
* dir /p - If the directory has a lot of files and you cannot read all the files you can use this command and it will display all files one page at a time.
* dir /w - Lists just the files and directories, going horizontal, and taking as little as space needed.
* dir /a:-s - This would list all the files that aren't system files in the current directory.

**CD:**  Changes the directory.

*Syntax:* CD [drive:][path]

CD[..]

EXAMPLES

* cd \ - Changes current directory to the highest level of the current drive (the root directory).
* cd .. - Goes up one directory. For example if you are within the C:\WINDOWS\COMMAND> directory this would take you to C:\WINDOWS>
* cd windows - Changes to the Windows subdirectory of the current directory (if it exists).

Note: If you want to change to a different drive, CD will not switch to it. CD only changes directories in the current drive. To change drives, type the drive letter followed by a colon. i.e. cd d: will not switch to the d drive, but d: will switch drives.

**COPY:** Copies one or more files to another location. If trying to copy files longer the 8 characters, DOS will truncate them with normal COPY. Also COPY will not move all files, it will leave Hidden and System files alone. Use XCOPY instead to solve both of these issues.

*Syntax:* COPY [/A | /B] source [/A | /B] [+ source [/A | /B] [+ ...]] [destination] [/A | /B]] [/V] [/Y | /-Y]

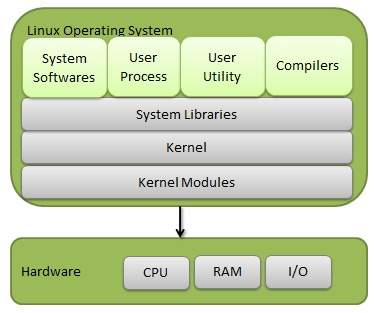
* source Specifies the file or files to be copied.
* /A Indicates an ASCII text file.
* /B Indicates a binary file.
* destination Specifies the directory and/or filename for the new file(s).
* /V Verifies that new files are written correctly.
* /Y Suppresses prompting to confirm you want to overwrite an existing destination file.
* /-Y Causes prompting to confirm you want to overwrite an existing destination file.
* To append files, specify a single file for destination, but multiple files for source (using wildcards or file1+file2+file3 format).

EXAMPLES

* copy \*.\* a: - This would copy all files in the current directory to the floppy disk in drive a:
* copy autoexec.bat c:\windows - Copies autoexec.bat from the current directory to the Windows directory of the C: drive.
* copy file1.txt+file2.txt+file3.txt newfile.txt /v - This would add together (concatenate) file1.txt, file2.txt and file3.txt and store the results in newfile.txt, then verify that it copied correctly.

**LINUX OPERATING SYSTEM**

Linux is one of popular version of UNIX operating System. It is open source as its source code is freely available. It is free to use. Linux was designed considering UNIX compatibility. Its functionality list is quite similar to that of UNIX.



**Basic Features:**

Following are some of the important features of Linux Operating System.

* Portable − Portability means software can works on different types of hardware in same way. Linux kernel and application programs supports their installation on any kind of hardware platform.
* Open Source − Linux source code is freely available and it is community based development project. Multiple teams work in collaboration to enhance the capability of Linux operating system and it is continuously evolving.
* Multi-User − Linux is a multiuser system means multiple users can access system resources like memory/ ram/ application programs at same time.
* Multiprogramming − Linux is a multiprogramming system means multiple applications can run at same time.
* Hierarchical File System − Linux provides a standard file structure in which system files/ user files are arranged.
* Shell − Linux provides a special interpreter program which can be used to execute commands of the operating system. It can be used to do various types of operations, call application programs. etc.
* Security − Linux provides user security using authentication features like password protection/ controlled access to specific files/ encryption of data.

**List of Basic UNIX/Linux shell commands**

Following is just the very basic list of some useful commands.

Files and directories

* pwd: shows current directory.
* cd directoryname: makes directoryname your current directory. cd with no parameters
* switches to your home directory
* ls directoryname: lists contents of directories. Use ls -l for more information about the files.
* You can limit the list with wildcards (e.g., “ls /home/mydirectory/\*.txt”)
* mkdir directoryname: creates a new directory.
* cp source destination: makes a copy of a file named “source” to “destination”.
* cp –r source destination: copies a directory and its content
* mv source destination: moves a file or directory.
* rm filenamelist: removes/deletes file(s). Be careful with wildcards.
* rm –r directory: removes directory (-ies) including its content

**Command Line Shell**

Shell can be accessed by user using a command line interface. A special program called **Terminal** in linux/macOS or **Command Prompt** in Windows OS is provided to type in the human readable commands such as “cat”, “ls” etc. and then it is being execute. The result is then displayed on the terminal to the user.

There are several shells are available for Linux systems like –

* + [BASH (Bourne Again SHell)](https://en.wikipedia.org/wiki/Bash_(Unix_shell)) – It is most widely used shell in Linux systems. It is used as default login shell in Linux systems and in macOS. It can also be installed on Windows OS.
  + [CSH (C SHell)](https://en.wikipedia.org/wiki/C_shell) – The C shell’s syntax and usage are very similar to the C programming language.
  + [KSH (Korn SHell)](https://en.wikipedia.org/wiki/Korn_shell) – The Korn Shell also was the base for the POSIX Shell standard specifications etc.

Each shell does the same job but understand different commands and provide different built in functions.

**Advantages of shell scripts**

* + The command and syntax are exactly the same as those directly entered in command line, so programmer do not need to switch to entirely different syntax
  + Writing shell scripts are much quicker
  + Quick start
  + Interactive debugging etc.

**Disadvantages of shell scripts**

* + Prone to costly errors, a single mistake can change the command which might be harmful
  + Slow execution speed
  + Design flaws within the language syntax or implementation
  + Not well suited for large and complex task
  + Provide minimal data structure unlike other scripting languages. Etc

**Displaying the file contents on the terminal:**

* [**cat**](https://www.geeksforgeeks.org/cat-command-linux-examples/) : It is generally used to concatenate the files. It gives the output on the standard output.
* [**more**](https://www.geeksforgeeks.org/more-command-in-linux-with-examples/) : It is a filter for paging through text one screenful at a time.
* [**less**](https://www.geeksforgeeks.org/less-command-linux-examples/) : It is used to viewing the files instead of opening the file.Similar to morecommand but it allows backward as well as forward movement.
* [**head**](https://www.geeksforgeeks.org/head-command-linux-examples/) : Used to print the first N lines of a file. It accepts N as input and the default value of N is 10.
* [**tail**](https://www.geeksforgeeks.org/tail-command-linux-examples/) : Used to print the last N-1 lines of a file. It accepts N as input and the default value of N is 10.

**File and Directory Manipulation Commands:**

* [**mkdir**](https://www.geeksforgeeks.org/mkdir-command-in-linux-with-examples/) : Used to create a directory if not already exist. It accepts directory name as input parameter.
* [**cp**](https://www.geeksforgeeks.org/cp-command-linux-examples/) : This command will copy the files and directories from source path to destination path. It can copy a file/directory with new name to the destination path. It accepts source file/directory and destination file/directory.
* [**mv**](https://www.geeksforgeeks.org/mv-command-linux-examples/) : Used to move the files or directories. This command’s working is almost similar to cp command but it deletes copy of file or directory from source path.
* [**rm**](https://www.geeksforgeeks.org/rm-command-linux-examples/) : Used to remove files or directories.
* [**touch**](https://www.geeksforgeeks.org/touch-command-in-linux-with-examples/) : Used to create or update a file.

**Extract, sort and filter data Commands:**

* [**grep**](https://www.geeksforgeeks.org/grep-command-in-unixlinux/) : This command is used to search for the specified text in a file.
* **grep with Regular Expressions :** Used to search for text using specific regular expressions in file.
* [**sort**](https://www.geeksforgeeks.org/sort-command-linuxunix-examples/) : This commands is used to sort the contents of files.
* [**wc**](https://www.geeksforgeeks.org/wc-command-linux-examples/) : Used to count the number of characters, words in a file.
* [**cut**](https://www.geeksforgeeks.org/cut-command-linux-examples/) : Used to cut a specified part of a file.

**Basic Terminal Navigation Commands:**

* [**ls**](https://www.geeksforgeeks.org/practical-applications-ls-command-linux/) : To get the list of all the files or folders.
* [**cd**](https://www.geeksforgeeks.org/cd-command-in-linux-with-examples/) : Used to change the directory.
* [**du**](https://www.geeksforgeeks.org/du-command-linux-examples/) : Show disk usage.
* [**pwd**](https://www.geeksforgeeks.org/pwd-command-in-linux-with-examples/) : Show the present working directory.
* [**man**](https://www.geeksforgeeks.org/man-command-in-linux-with-examples/) : Used to show the manual of any command present in Linux.
* **rmdir** : It is used to delete a directory if it is empty.
* **ln file1 file2** : Creates physical link.
* **ln -s file1 file2** : Creates symbolic link.

**File Permissions Commands:**The *chmod*and *chown*commands are used to control access to files in UNIX and Linux systems.

* **chown** : Used to change the owner of file.
* [**chmod**](https://www.geeksforgeeks.org/chmod-command-linux/) : Used to modify the access/permission of a user.