File & Directory Management in Linux

Efficient file and directory management is essential for working effectively in any Linux environment. It incorporates organizing, navigating, and manipulating files and directories to maintain order and streamline workflows. Mastery of these operations allows users to access data quickly, manage projects systematically, and perform system tasks accurately, whether handling everyday documents, scripts, or configuration files.

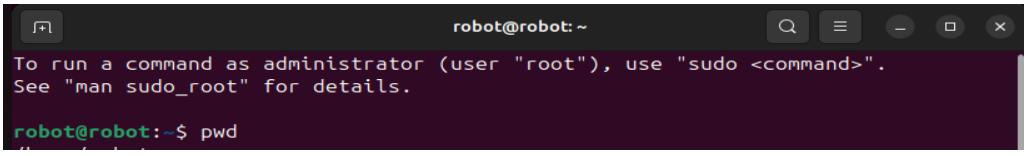
File and Directory Management commands for Linux

Command	Purpose
pwd	Displays the current directory
cd	Changes directory
ls	Lists files and directories
mkdir	Create s directory
rm	Deletes file
rmdir	Deletes directory
touch	Creates file
cp	Copies file
mv	Moves or rename file
less	Scrolls through file content
head	Views first few lines
tail	Views last few lines
grep	Searches text patterns

1.pwd (Print Working Directory) Command

The **pwd** commands shows the full absolute path of the current working directory in the terminal. This is useful for confirming the location in the filesystem before running file or directory operations.

Command:



Output:

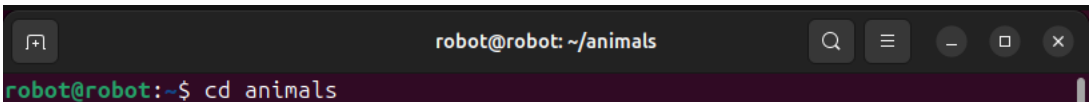
2. cd (Change Directory) Command

The cd command in Linux is used to navigate between directories in the filesystem. It changes the current working directory of the terminal session.

Syntax:

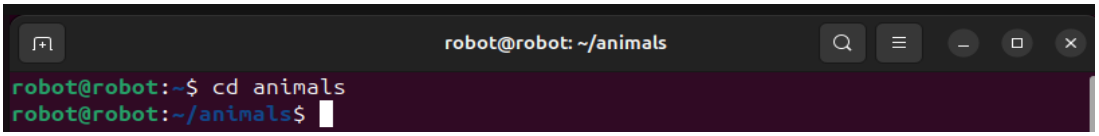
```
cd [directory]
```

Command:



Output:

You will see that the terminal is navigated to **animals** directory.



3. ls Command

The ls (list) command is used to list the contents of a directory in Linux. It displays the files and subdirectories contained within the specified directory or, if no directory is specified, the current working directory.

Syntax:

```
ls [directory]
```

Command:

Output:

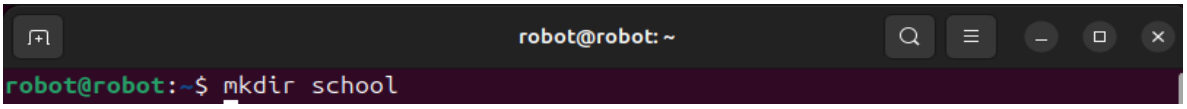
4. mkdir command

The mkdir (make directory) is used to create new directories in the Linux filesystem.

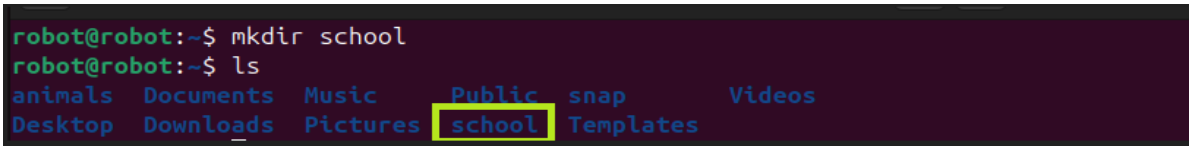
Syntax:

```
mkdir [directory_name]
```

Command:



Output:



5. rm Command

The rm (remove) command is used to delete files in the Linux filesystem.

Syntax:

```
rm [file_name]
```

Command:

Output:

You will see that **cat** file is gone.

6. rmdir Command

The rmdir (remove directory) is used to delete empty directories in the Linux filesystem.

Syntax:

```
rmdir [directory_name]
```

Command:

```
robot@robot:~$ cd animals
robot@robot:~/animals$ ls
dog  insects  lion  movies  zebra
robot@robot:~/animals$ rmdir insects
```

Output:

You will see that **insects** directory is gone.

```
robot@robot:~/animals$ rmdir insects
robot@robot:~/animals$ ls
dog  lion  movies  zebra
```

7. touch Command

The touch command is used to create new empty files or update the timestamp of existing files in the Linux filesystem.

Syntax:

```
touch [file_name]
```

Command:

```
robot@robot:~/animals$ touch shark
```

Output:

```
robot@robot:~/animals$ touch shark
robot@robot:~/animals$ ls
dog  lion  movies  shark  zebra
```

8. cp Command

The cp (copy) command is used to copy files or directories within the Linux filesystem.

Syntax:

```
cp [source] [destination]
```

Command:

```
robot@robot:~/animals$ ls
shark
robot@robot:~/animals$ cp shark whale
```

Output:

```
robot@robot:~/animals$ ls
shark  whale
robot@robot:~/animals$ cat whale
hello, hello, hello, hello, hello
```

9. mv Command

The mv command in Linux is used to move or rename files and directories.

Syntax:

```
mv [source_fiile] [destination_file]
```

Command:

```
robot@robot:~/animals$ ls
shark  whale
robot@robot:~/animals$ mv whale chicken
```

Output:

```
robot@robot:~/animals$ ls
chicken  shark
robot@robot:~/animals$ cat chicken
hello, hello, hello, hello, hello
```

10. less Command

The less command in Linux is used to views the contents of a file one screen at a time. Unlike editors, it does not allow changing the file – it is strictly for viewing.

Syntax:

```
less [filename]
```

or

```
less [pathname]
```

Command:

```
robot@robot:~$ less /var/log/syslog
```

Output:

```
2025-08-19T03:03:20.892236+00:00 robot systemd[1]: Mounted dev-hugepages.mount -
 Huge Pages File System.
2025-08-19T03:03:20.892589+00:00 robot kernel: Linux version 6.14.0-28-generic (
buildd@lcy02-amd64-079) (x86_64-linux-gnu-gcc-13 (Ubuntu 13.3.0-6ubuntu2~24.04)
13.3.0, GNU ld (GNU Binutils for Ubuntu) 2.42) #28~24.04.1-Ubuntu SMP PREEMPT_DY
NAMIC Fri Jul 25 10:47:01 UTC 2 (Ubuntu 6.14.0-28.28~24.04.1-generic 6.14.8)
2025-08-19T03:03:20.904315+00:00 robot kernel: Command line: BOOT_IMAGE=/boot/vm
linuz-6.14.0-28-generic root=UUID=ccca9bb5-2a85-4b61-95fd-523bb9d033e5 ro quiet
splash
2025-08-19T03:03:20.904322+00:00 robot kernel: KERNEL supported cpus:
2025-08-19T03:03:20.904322+00:00 robot kernel:   Intel GenuineIntel
2025-08-19T03:03:20.904323+00:00 robot kernel:   AMD AuthenticAMD
2025-08-19T03:03:20.904323+00:00 robot kernel:   Hygon HygonGenuine
2025-08-19T03:03:20.904324+00:00 robot kernel:   Centaur CentaurHauls
2025-08-19T03:03:20.904327+00:00 robot kernel:   zhaoxin   Shanghai
2025-08-19T03:03:20.904328+00:00 robot kernel: BIOS-provided physical RAM map:
2025-08-19T03:03:20.904328+00:00 robot kernel: BIOS-e820: [mem 0x000000000000000
0-0x000000000009fbff] usable
2025-08-19T03:03:20.904329+00:00 robot kernel: BIOS-e820: [mem 0x000000000009fc0
0-0x000000000009ffff] reserved
2025-08-19T03:03:20.904329+00:00 robot kernel: BIOS-e820: [mem 0x00000000000f000
0-0x00000000000fffff] reserved
2025-08-19T03:03:20.904330+00:00 robot kernel: BIOS-e820: [mem 0x000000000010000
/var/log/syslog
```

11. head Command

The head command in Linux is used to display the first part of a file, typically the first 10 lines by default. It is useful for quickly checking the beginning of files.

Syntax:

```
head [filename]
```

or

```
head [pathname]
```

Command:

```
robot@robot:~$ head /var/log/syslog
```

Output:

```
robot@robot:~$ head /var/log/syslog
2025-08-19T03:03:20.892236+00:00 robot systemd[1]: Mounted dev-hugepages.mount -
 Huge Pages File System.
2025-08-19T03:03:20.892589+00:00 robot kernel: Linux version 6.14.0-28-generic (
buildd@lcy02-amd64-079) (x86_64-linux-gnu-gcc-13 (Ubuntu 13.3.0-6ubuntu2~24.04)
13.3.0, GNU ld (GNU Binutils for Ubuntu) 2.42) #28~24.04.1-Ubuntu SMP PREEMPT_DY
NAMIC Fri Jul 25 10:47:01 UTC 2 (Ubuntu 6.14.0-28.28~24.04.1-generic 6.14.8)
2025-08-19T03:03:20.904315+00:00 robot kernel: Command line: BOOT_IMAGE=/boot/vm
linuz-6.14.0-28-generic root=UUID=ccca9bb5-2a85-4b61-95fd-523bb9d033e5 ro quiet
splash
2025-08-19T03:03:20.904322+00:00 robot kernel: KERNEL supported cpus:
2025-08-19T03:03:20.904322+00:00 robot kernel:   Intel GenuineIntel
2025-08-19T03:03:20.904323+00:00 robot kernel:   AMD AuthenticAMD
2025-08-19T03:03:20.904323+00:00 robot kernel:   Hygon HygonGenuine
2025-08-19T03:03:20.904324+00:00 robot kernel:   Centaur CentaurHauls
2025-08-19T03:03:20.904327+00:00 robot kernel:   zhaoxin   Shanghai
2025-08-19T03:03:20.904328+00:00 robot kernel: BIOS-provided physical RAM map:
```

12. tail Command

The tail command in Linux is used to display the last part of a file, typically the last 10 lines by default.

Syntax:

```
tail [filename]
```

or

```
tail [pathname]
```

Command:

```
robot@robot:~$ tail /var/log/syslog
```

Output:

```
robot@robot:~$ tail /var/log/syslog
2025-08-19T20:37:48.786445-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 33 with keysym 33 (keycode c).
2025-08-19T20:37:48.786478-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 37 with keysym 37 (keycode 10).
2025-08-19T20:37:48.786500-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 34 with keysym 34 (keycode d).
2025-08-19T20:37:48.786527-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 39 with keysym 39 (keycode 12).
2025-08-19T20:37:48.786539-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 38 with keysym 38 (keycode 11).
2025-08-19T20:37:48.786549-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 37 with keysym 37 (keycode 10).
2025-08-19T20:37:48.786566-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 39 with keysym 39 (keycode 12).
2025-08-19T20:37:48.786576-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 35 with keysym 35 (keycode e).
2025-08-19T20:37:48.786586-04:00 robot gnome-shell[1872]: Window manager warning
: Overwriting existing binding of keysym 36 with keysym 36 (keycode f).
2025-08-19T20:38:13.391826-04:00 robot systemd[1]: fprintd.service: Deactivated
successfully.
```

13. grep Command

The grep command in Linux is used to search for specific text patterns within files. It prints the lines that match the given pattern, making it useful for quickly finding information in large files.

Syntax:

grep [pattern] [filename]

Or

grep [pattern] [pathname]

Command:

```
robot@robot:~/animals$ ls
chicken  shark
robot@robot:~/animals$ grep "ell" shark
```

Output:

```
robot@robot:~/animals$ grep "ell" shark
hello, hello, hello, hello, hello
```