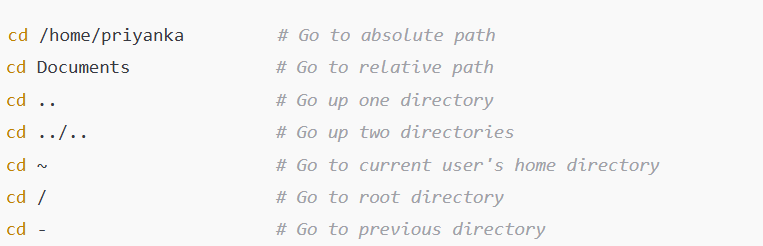
Linux Essential Commands

1. cd – Change Directory



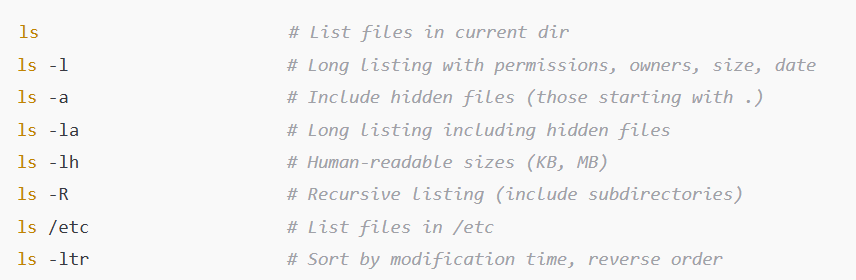
* + Examples:



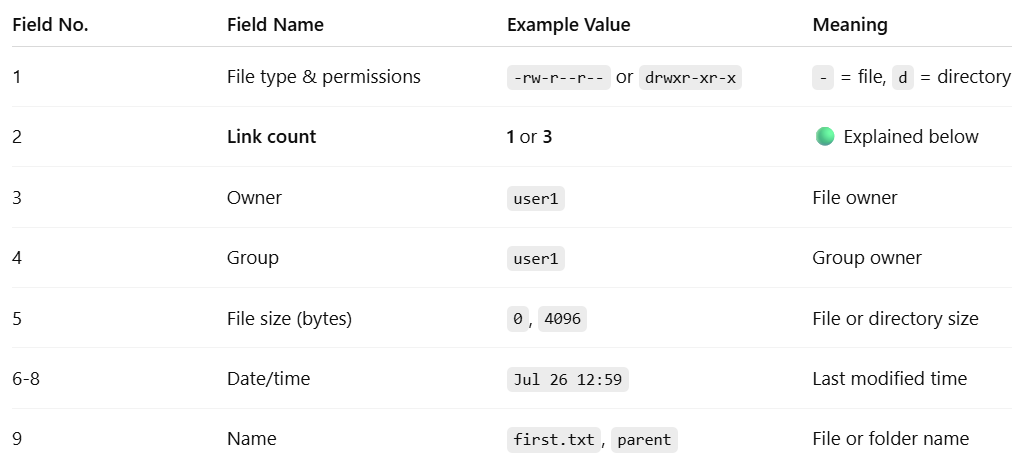
1. LS – List Directory Content



Examples:



Example: **ls -l**

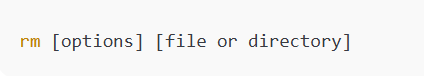


For a File: The **1** means: **1 hard link** to the file.

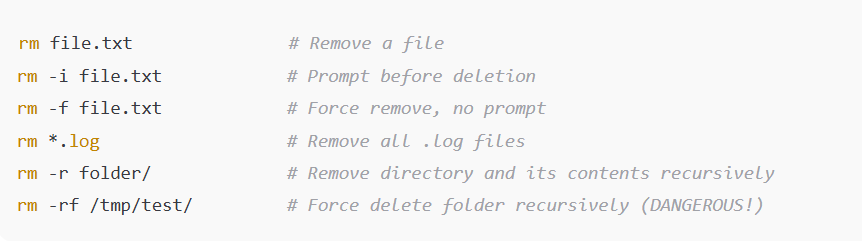
For a Directory: 2 + N

* 1 = The . (self) entry inside the directory
* 1 = The .. entry inside **each** of its **subdirectories**

1. rm – Remove Files or Directories



Examples:



1. sudo – Run Commands as Root (Superuser)

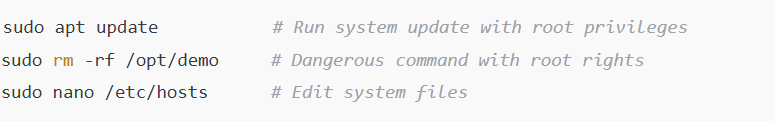


Examples:

1. To check what groups are created in linux

$ groups

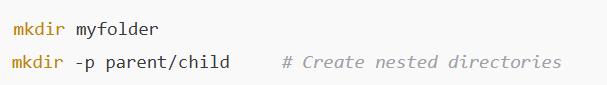
1. Users – Normal vs. Sudo (Admin)



1. pwd – Print Working Directory



1. mkdir – Make directory

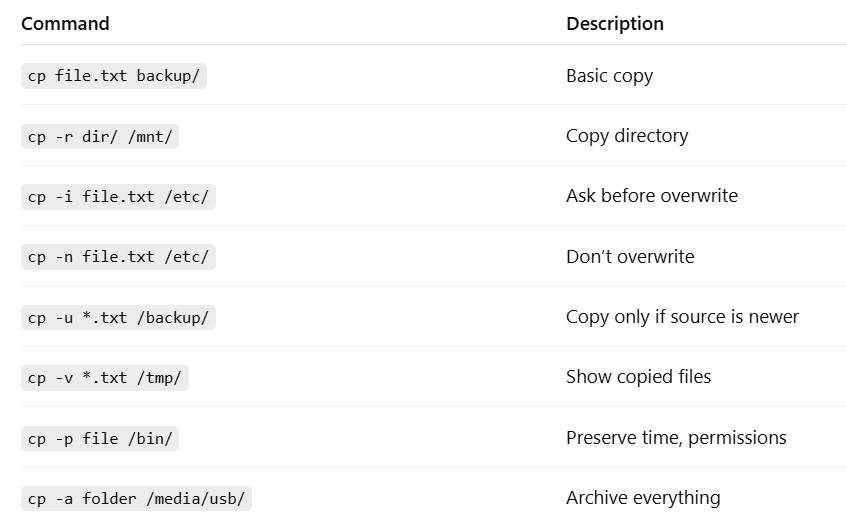


1. touch – Create file

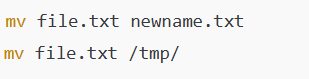


1. cp – copy files





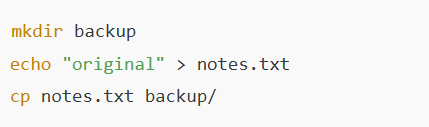
1. mv – move or rename

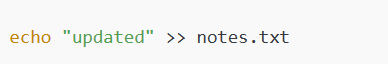


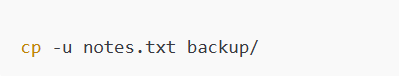
Practice on cp command:



* + **cp** – copy files
  + **-u** – **update mode**: only copy the file **if the source is newer than the destination** or if the file does **not exist** in the destination.
  + It **compares modification timestamps** of the source and destination files.
  + **\*.txt** – wildcard to match all .txt files in the current directory
  + **/backup/** – target directory where the files will be copied







This will **overwrite** the one in /backup/ **because it's older**. If you run again **without changes**, it **won’t copy again**.



* **cp** – Copy a file
* **-p** – **Preserve file attributes**
* **file** – Source file
* **/bin/** – Destination directory (commonly requires sudo)

When you use the -p (preserve) option, it retains the following attributes from the original file:

| **Attribute** | **Description** |
| --- | --- |
| **Modification time** | The mtime timestamp when the file was last modified |
| **Access time** | The atime timestamp when the file was last read |
| **Ownership** | The file’s **user** and **group** owner |
| **Permissions** | Read, write, execute permissions (e.g., rwx) |

**Without -p**

cp file /bin/

* The **new file** in /bin/ will:
  + Have the **current timestamp**
  + Be **owned by the user who ran the command**
  + Use the **default permissions** from the system

**Use Cases for -p**

* Backups where timestamps matter
* Deploying files with precise permissions (scripts, binaries)
* File auditing or compliance needs

As DevOps Engineer, daily basis common tasks like monitoring, troubleshooting, deployment, and system management.

**File and Directory Operations**

