# LINUX BASICS NOTE

***ls list the items***

## -R – lists all the files in the sub directories. (All files listing, including subdirectories)

* **-a** – shows all files, including hidden ones, of the specific (current) directory.
* -d command in Linux is used to **list directories themselves**, rather than their contents.
* -F append
* -l Display results in long formats (L) [*This format provides detailed information about files, including permissions, number of links, owner, group, file size, and timestamps (modification time).]*
* -r Display the result in reverse order
* -s Sort the result by file size
* -t sort by modification time
* **-lh** – converts sizes to readable formats, such as **MB**, **GB**, **TB.**

*(R: Recursively list subdirectories. -a: Include hidden files (those starting with a dot). -l: Use a long listing format.-h: Human-readable sizes (e.g., KB, MB, GB).)* -R -alh

### Cd

***cd*** *changes the working directory to home directory*

***cd pathname*** *goes to the specific directory (folder*)

***cd ..*** *goes one step back to the directory (folder)*

***pwd*** *command prints your current working directory’s path*

mkdir

**-p** – creates a directory between two existing folders. For example, **mkdir -p Music/rock/Songs** creates a new **rock** directory.

* **-m** – sets the folder permissions. For instance, enter **mkdir -m777 directory** to create a directory with read, write, and execute permissions for all users.

### rmdir command to [delete an empty directory in Linux](https://www.hostinger.com/tutorials/how-to-remove-files-and-folders-using-linux-command-line/).

**rm command to permanently *delete files* within a directory**

To modify the command,

* **-i** – prompts a confirmation before deletion.
* **-f** – allows file removal without a confirmation.

## -r – deletes files and directories both. (File which is inside directory and related directory) empty, non-empty both

**cp command**

* Copying **one file** from the current directory to another folder. cp filename.txt /home/username/Documents
* Duplicating **multiple** files to a directory.

cp filename1.txt filename2.txt filename3.txt /home/username/Documents

* Copying **a file’s content to another within the same** directory. cp filename1.txt filename2.txt
* **Duplicating an entire directory**. Pass the **-R** flag followed by the source and destination directory:

cp -R /home/username/Documents /home/username/Documents backup

**mv command**

**mv** command to ***move or rename*** files and directories. mv filename.txt /home/username/Documents

to [**rename a file in Linux**](https://www.hostinger.com/tutorials/how-to-rename-files-in-linux/)with the **mv** command:

mv old\_filename.txt new\_filename.txt

# MOVE AND RENAME A DIRECTORY

mv old directory\_path new location/

e.g. mv USA/SD/VERMILLION/ newlocation/YANKTON

This moves the "VERMILLION" directory to "/new location/" and renames it to "YANKTON."

[**touch command**](https://www.hostinger.com/tutorials/linux-touch-command-with-useful-examples/)

The **touch** command create an empty file in a specific directory path touch /home/directory/path/file.txt

**Create Multiple files**: touch hw1.txt hw2.txt hw3.txt

## Create a new empty file with a specific timestamp:

touch -t 202402211920 myfile.txt

## Change the timestamp of an existing file to a specific time:

touch -t 202402211140 hw1.*txt [ to see modification time we use ls -l hw1.txt]*

## Updating the timestamp Recursively (touch with find command):

find /mnt/c/Users/MANOJ/Desktop/Computation -exec touch {} +

# ABOLUTE AND RELATIVE PATH

The absolute path starts from the root directory. E.g. /mnt/c/Users/MANOJ/Desktop/Computation/no Specifies the exact location of a file or directory from the root directory of the file system.

# RELATIVE PATH

specifies the location of a file or directory relative to the current working directory.

*Consider I want to see the properties of “2025” using ls command. I can use both absolute as well as relative path, both will give the same result. Consider I am at “manoj” directory.*

*Then by using absolute path : ls -1 /mnt/c/Users/MANOJ/Desktop/Computation/no/manoj/2 025*

*And by using relative path* ls -1 manoj/2025,, if you want to go from current directory to 2025 directory.

Both will give same result.

### diff command

The **diff** command compares two files’ content and outputs the differences. It is used to alter a program without modifying the code. Here’s the general format:

diff [option] file1 file2

Below are some acceptable options:

* **-c** – displays the difference between two files in a context form.
* **-u** – shows the output without redundant information.
* **-i** – makes the **diff**command case insensitive.

### [grep command](https://www.hostinger.com/tutorials/grep-command-in-linux-useful-examples/)

The**global regular expression** or **grep** command lets you find a word by searching the content of a file. This Linux command prints all lines containing the matching strings, which is useful for filtering large log files.

For example, to display lines containing **blue**in the **notepad.txt** file, enter:

grep blue notepad.txt

### File System Navigation

* **ls**: List directory contents.
  + Example: **ls -l** to list files in long format.
* **cd**: Change directory.
  + Example: **cd /path/to/directory** to change to a specific directory.
* **pwd**: Print working directory.
  + Example: **pwd** to display the current working directory.
* **mkdir**: Make directory.
  + Example: **mkdir new\_directory** to create a new directory.
* **rmdir**: Remove directory.
  + Example: **rmdir directory\_name** to remove an empty directory.
* **cp**: Copy files and directories.
  + Example: **cp file1 file2** to copy file1 to file2.
* **mv**: Move/rename files and directories.
  + Example: **mv file1 new\_location** to move file1 to a new location or **mv file1 new\_name** to rename file1.
* **rm**: Remove files and directories.
  + Example: **rm file1** to remove a file, or **rm -r directory1** to remove a directory recursively.
* **touch**: Create an empty file.
  + Example: **touch new\_file.txt** to create a new empty file.
* **ln**: Create links between files.
  + Example: **ln -s /path/to/file link\_name** to create a symbolic link.

### File Viewing/Editing

* **cat**: Concatenate and display file content.
  + Example: **cat file1** to display contents of file1.
* **more**: Display output one screen at a time.
  + Example: **more file1** to display contents of file1, advancing one screen at a time.
* **less**: A file viewer similar to **more** but allows scrolling backward.
  + Example: **less file1** to view contents of file1 with backward scrolling.
* **head**: Display the beginning of a file.
  + Example: **head file1** to display the first few lines of file1.
* **tail**: Display the end of a file.
  + Example: **tail file1** to display the last few lines of file1.
* **nano**: A simple text editor.
  + Example: **nano filename** to edit a file using the nano editor.
* **vim** or **vi**: A powerful text editor.
  + Example: **vim filename** to edit a file using the vim editor.

### File Permissions

* **chmod**: Change file permissions.
  + Example: **chmod 755 filename** to set file permissions to read, write, execute for owner, and read, execute for group and others.
* **chown**: Change file owner and group.
  + Example: **chown user:group filename** to change the owner and group of a file.
* **chgrp**: Change group ownership of a file.
  + Example: **chgrp groupname filename** to change the group ownership of a file.

### File Searching

* **find**: Search for files in a directory hierarchy.
  + Example: **find /path/to/search -name "filename"** to find a file by name.
* **locate**: Search a pre-built database of files.
  + Example: **locate filename** to quickly find a file by name.
* **grep**: Search text for patterns.
  + Example: **grep "pattern" file** to search for a pattern in a file.
* **which**: Locate a command.
  + Example: **which command\_name** to find the location of a command.

### File Compression/Archiving

* **tar**: Tape archive, used for file compression and archiving.
  + Example: **tar -cvzf archive.tar.gz /path/to/directory** to create a gzipped tarball.
* **gzip** and **gunzip**: Compress or decompress files.
  + Example: **gzip filename** to compress a file, or **gunzip filename.gz** to decompress.
* **zip** and **unzip**: Create or extract zip archives.
  + Example: **zip archive.zip file1 file2** to create a zip archive, or **unzip archive.zip** to extract.
* **bzip2** and **bunzip2**: Compress or decompress files using the bzip2 algorithm.
  + Example: **bzip2 filename** to compress a file, or **bunzip2 filename.bz2** to decompress.

### Process Management

* **ps**: Display information about active processes.
  + Example: **ps aux** to display all processes running on the system.
* **top**: Display and update sorted information about processes.
  + Example: **top** to see a dynamic view of system processes.
* **kill**: Terminate processes.
  + Example: **kill PID** to terminate a process with a specific PID.
* **pkill**: Send signals to processes based on name.
  + Example: **pkill process\_name** to send a signal to a process by name.
* **killall**: Kill processes by name.
  + Example: **killall process\_name** to kill all processes with a specific name.

### System Information

* **uname**: Print system information.
  + Example: **uname -a** to display all system information.
* **uptime**: Show how long the system has been running.
  + Example: **uptime** to display system uptime and load averages.
* **hostname**: Print or set system hostname.
  + Example: **hostname** to display the system hostname.
* **df**: Display free disk space.
  + Example: **df -h** to display disk space in human-readable format.
* **du**: Display disk usage.
  + Example: **du -sh /path/to/directory** to show the total disk usage of a directory.

### Network

* **ifconfig** or **ip**: Display and configure network interfaces.
  + Example: **ifconfig** or **ip addr show** to display network interface information.
* **ping**: Check connectivity to a remote host.
  + Example: **ping google.com** to ping google.com.
* **traceroute** or **mtr**: Trace the route to a host.
  + Example: **traceroute google.com** to trace the route to google.com.
* **netstat**: Print network connections, routing tables, interface statistics, etc.
  + Example: **netstat -tuln** to display listening ports.
* **ss**: Another utility to investigate sockets.
  + Example: **ss -tuln** to display listening ports.

### User/Group Management

* **useradd** and **adduser**: Add a new user to the system.
  + Example: **useradd username** or **adduser username** to add a new user.
* **passwd**: Change user password.
  + Example: **passwd username** to change the password for a user.
* **usermod**: Modify user account.
  + Example: **usermod -aG groupname username** to add a user to a group.
* **groupadd** and **addgroup**: Create a new group.
  + Example: **groupadd groupname** or **addgroup groupname** to create a new group.

### Package Management

* **apt**: Advanced Package Tool, for package management on Debian/Ubuntu systems.
  + Example: **apt update** to update the package list, or **apt install package\_name** to install a package.
* **dpkg**: Debian package manager.
  + Example: **dpkg -i package.deb** to install a .deb package.
* **yum** or **dnf**: Package manager for Red Hat based systems like CentOS and Fedora.
  + Example: **yum install package\_name** to install a package using yum.
* **rpm**: RPM package manager.
  + Example: **rpm -i package.rpm** to install an RPM package.

### System Control

* **shutdown**: Shutdown the system.
  + Example: **shutdown -h now** to shut down the system immediately.
* **reboot**: Reboot the system.
  + Example: **reboot** to reboot the system.
* **halt**: Halt the system.
  + Example: **halt** to halt the system.
* **init**: Change the system's runlevel.
  + Example: **init 0** to halt the system, or **init 6** to reboot.

### Miscellaneous

* **date**: Display or set the system date and time.
  + Example: **date** to display the current date and time.
* **echo**: Display text on the screen.
  + Example: **echo "Hello, World!"** to print "Hello, World!" to the screen.
* **history**: Display the command history.
  + Example: **history** to display the list of previously executed commands.
* **wget**: Download files from the web.
  + Example: **wget URL** to download a file from a URL.
* **curl**: Transfer data from or to a server.
  + Example: **curl URL** to retrieve the content of a URL.