|  |  |
| --- | --- |
| **Pwd** | Displays the current working directory of the terminal |
| **/** | Root directory |
| **Echo** | command that writes its arguments to standard output |
| **Su** | used to switch to root user ( so that super user permissions can be used to execute commands) |
| **Su username** | used to switch to a different user |
| **Sudo** | Executes only that command with root/super user privileges |
| **Clear** | This command is used to clear the terminals screen. |

|  |  |
| --- | --- |
| **Ls** | **Lists ALL the contents in the current working directory** |
| **Ls –l** | **Lists all the contents along with its owner settings,permissions and timestamp** |
| **Ls -a** | **Lists all the hidden contents in the specified directory** |
| **Ls –author** | **Lists the contents in the specified directory along with its owner** |
| **Ls –S** | **Lists all the contents in the specified directory by size** |
| **Ls \*.html** | **Lists all the contents on the directory of a particular format** |
| **Ls –LS > file.txt** | **Copies the result of ls command into a text file.** |
|  |  |

|  |  |
| --- | --- |
| **Cd** | **Changes the directory to the home directory** |
| **Cd ~** | **This also changes the directory to home directory** |
| **Cd /** | **Changes the directory to root directory** |
| **Cd ..** | **Changes the directory to its parent directory** |
| **Cd ‘xx yy’** | **We specify the folder name in inverted commas because there is a space in folder name** |

|  |  |
| --- | --- |
| **Cat** | **This command is used to display the content of text files and concatenate several files into one** |
| **Cat -b** | **This is used to add line numbers to non bank lines** |
| **Cat –n** | **This is used to add line numbers to all lines** |
| **Cat –s** | **This is used to squeeze blank lines into one line** |
| **Cat –E** | **Show $ at the end of the line** |
|  |  |
|  |  |

**Cp This command is used to copy files and directories.**

|  |  |
| --- | --- |
| **Cp -i** | **Enters interactive mode; CLI asks before overwriting files** |
| **Cp -n** | **Does not overwrite the file** |
| **Cp –u** | **Updates the destination file only when source file is different from destination file.** |
| **Cp -R** | **Recursive copy; copies even hidden files** |
| **Cp –v** | **Verbose; prints informative messages** |
| **Cp** | **Cp filename path** |
|  |  |

**Grep**

**Grep options file1.txt🡪** Returns results for matching values

Grep –I options file1.tx🡪 Returns the results for case insensitive strings.

**Grep –n options file1.txt 🡪 returns the matching strings along with their line numbers**

**Grep –v options file1.txt🡪 returns the result of lines not matching the search string**

**Mkdir: Used to create a new directory**

**Mkdir –p 🡪 creates a new parent directory and a sub directory**

**Mkdir - - parents 🡪 this is also used for the same process**

**Mkdir –p file1/ {f2, f3, f4}🡪 used to create multiple subdirectories inside the new parent directory.**

**rmdir: used to remove the specified directory**

**rmdir -p 🡪 removes both parent and child directory**

**rmdir –pv🡪 removes all parent and sub directories along with the verbose**

**rm –r🡪 removes even non empty directories**

**rm –rp 🡪 removes non empty directories including parent and subdirectories.**

**How to add a user?**

1. Sudo useradd username
2. Sudo passwd username
3. su username
4. password

**How to add a user to a group**

sudo groupadd groupname

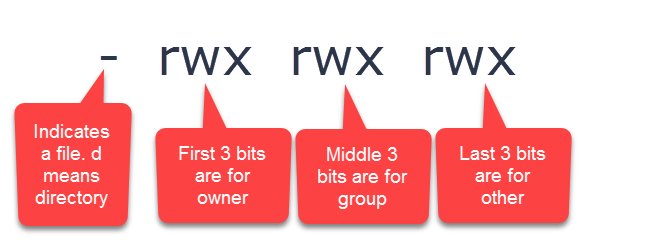
sudo usermod –a –G groupname username

**Adding user in wheel**

usermod –a –G wheel username

**Sort command to sort the results of a search either alphabetically or numerically**

**Files,file contents and directories can be sorted.**



**There are three levels of permissions in Linux: owner, group and other. The owner is the user who owns the file/folder, the group includes other users in the file’s group and other just represents all other users who are not the owner or in the group.**

The permissions are written as follows: the first bit is either a dash or the letter d. **Dash** means it’s a file and **d** stands for directory. Note that the first bit can also be an **l** if the file name is a link. Next, there are three groups of three bits. The first bit in each group is for read, the second bit is for write and the third bit is for execute. The first three bits are for the owner, the second three bits are for the group and the third three bits are for other.

If you see a dash in place of a letter, it means that the owner, group or all other users do not have that permission. In the example above, the owner, group and everyone else has read write and execute permissions.

If you look at the output from the ls -l command, you’ll notice that my practice text file has the following permissions:

-rw-rw-rw-

This means that everyone only has read/write permissions for the file. Here’s another example:

drwxr--r--

Looking at the first bit, we can see that the permissions are for a directory. The owner has read/write/execute permissions, but the group and other users only have read permission