LINUX COMMANDS

KEYWORDS

append = adding new information in particular existing file.

override = deletes the previous saved data and overrides with new one.

" ; " (semicolon)*:-* Use to write the 2 different commands in one line

        EX:- *command 1* ; *command 2*

Shell Basics

Commands that are entered at the shell prompt have three basic parts:

- **Command** to run.
- **Options** to adjust the behavior of the command
- **Arguments**, which are typically targets of the command.

COMMANDS

- ***ssh(****secure shell****) :-*** आपल्याला दुसऱ्या workplace ला जायचं असेल तर
  - EX.:- **student@workstation => student@servera**
- ***passwd :-*** आपल्याला current workplace चा password change करायचा असेल तर
- ***whoami :-*** To display the name of user.
- ***date :-*** Use to display the current Date and Time

- ***EX:-***
  ```
  [user@host ~]$ date
  Sun Feb 27 08:32:42 PM EST 2022
  ```

- ***file :-*** Use to display the file's data-type
  - *ex:-*
    ```
    [user@host ~]$ file /home
    /home: directory
    ```

    ```
    [user@host ~]$ file /etc/passwd
    /etc/passwd: ASCII text
    ```
- ***cat :-*** Use to display the contents of those file
  - *ex:-*
    ```
    [user@host ~]$ cat file1 file2
    Hello World!!
    Introduction to Linux commands.
    ```

- ***head:-*** Use to display the starting 10 lines of any file.

```
[user@host ~]$ head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
```
Ex:-

- ***tail:-*** Use to display the last 10 lines of any file.

```
[user@host ~]$ tail -n 3 /etc/passwd
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:980:978::/run/gnome-initial-setup/:/sbin/nologin
dnsmasq:x:979:977:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin
```
Ex:-

(tail **-n 3** means numbers of lines want to display)

- ***wc:-*** To display the number of lines$_{(-l)}$,words$_{(-w)}$,characters$_{(-c)}$ of your file.

- ***history:-*** Use to show the list of commands which we have executed.

```
[user@host ~]$ history
...output omitted...
   23  clear
   24  who
   25  pwd
   26  ls /etc
   27  uptime
   28  ls -l
   29  date
   30  history
```
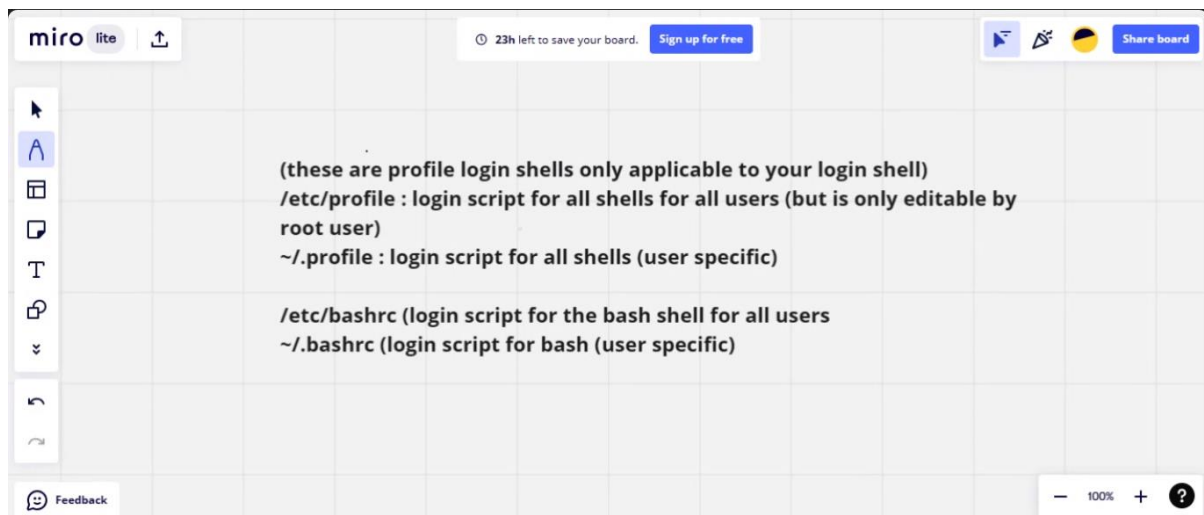***Ex:-***

Under this command, we can also access the previous commands
Like this- (Using "**!**")

```
[user@host ~]$ !ls
ls -l
total 0
drwxr-xr-x. 2 student student 6 Feb 27 19:24 Desktop
...output omitted...
[user@host ~]$ !26
ls /etc
abrt                   hosts               pulse
adjtime                hosts.allow         purple
aliases                hosts.deny          qemu-ga
...output omitted...
```

-

IN Vim

yank/put = copy/paste

(these are profile login shells only applicable to your login shell)
/etc/profile : login script for all shells for all users (but is only editable by root user)
~/.profile : login script for all shells (user specific)

/etc/bashrc (login script for the bash shell for all users
~/.bashrc (login script for bash (user specific)


```
[student@workstation ~]$ COUNT=100
[student@workstation ~]$ echo COUNT
COUNT
[student@workstation ~]$ echo $COUNT
100
[student@workstation ~]$ file1=/tmp/data
[student@workstation ~]$ ls -l $
```

Here are 50 basic Linux commands that are useful for navigating and managing your system:

1. **pwd** - Print working directory.

2. **ls** - List directory contents.

3. **cd** - Change directory.

4. **touch** - To create a file without any content.

5. **cat** - Concatenate and display file content.

6. **cp** - Copy files or directories.

7. **mv** - Move or rename files or directories.

8. **rm** - Remove files or directories.

9. **mkdir** - Create a new directory.

10. **rmdir** - Remove an empty directory.

11. **echo** - Display a line of text or a variable value.

12. **nano** - A simple text editor.

13. **vi** - A powerful text editor.

14. **chmod** - Change file or directory permissions.

15. **chown** - Change file or directory owner and group.

16. **find** - Search for files in a directory hierarchy.

17. **grep** - Search text using patterns.

18. **man** - Display the manual for a command.

19. **ps** - Display information about running processes.

20. **kill** - Terminate processes by PID.

21. **top** - Display and update sorted information about processes.

22. **df** - Report file system disk space usage.

23. **du** - Estimate file space usage.

24. **free** - Display memory usage.

25. **uname** - Print system information.

26. **uptime** - Tell how long the system has been running.

27. **whoami** - Display the current user.

28. **sudo** - Execute a command as another user, typically the superuser.

29. **apt-get** - Package handling utility for Debian-based distributions.

30. **yum** - Package manager for RPM-based distributions.

31. **tar** - Archive files.

32. **zip** - Package and compress (archive) files.

33. **unzip** - Extract compressed files.

34. **wget** - Retrieve files from the web.

35. **curl** - Transfer data from or to a server.

36. **ssh** - OpenSSH client (remote login program).

37. **scp** - Secure copy (remote file copy program).

38. **rsync** - Remote file and directory synchronization.

39. **hostname** - Show or set the system's host name.

40. **ping** - Send ICMP ECHO_REQUEST to network hosts.

41. **netstat** - Print network connections, routing tables, interface statistics, masquerade connections, and multicast

memberships.

42. **ifconfig** - Configure a network interface.

43. **ip** - Show/manipulate routing, devices, policy routing, and tunnels.

44. **iptables** - Administration tool for IPv4 packet filtering and NAT.

45. **systemctl** - Control the systemd system and service manager.

46. **journalctl** - Query and display messages from the journal.

47. **crontab** - Schedule periodic background jobs.

48. **sudo su** - allows us to switch to a different user and execute one or more commands in the shell without logging out from our

current session

49. **mount** - Mount a file system.

50. **umount** - Unmount a file system.

These commands form the basis of interacting with a Linux system and performing various administrative tasks.

Goodluck from Vallabh Deshpande..(●'‿'●)!!