Su and Sudo Command:

su username (SU generally we call it Switch user)

But su stands for substitute user

```
simran@ubuntu:~$ su user2
Password:
$ ls
Desktop    Downloads  Music     Public  temp       Videos
Documents  folder18   Pictures  snap    Templates
$ exit
simran@ubuntu:~$ exit
```

User **– user2**

```
simran@ubuntu:~$ su user2
Password:
$ ls
Desktop    Downloads  Music     Public  temp       Videos
Documents  folder18   Pictures  snap    Templates
$ exit
simran@ubuntu:~$ su -user2
Try 'su --help' for more information.
simran@ubuntu:~$
simran@ubuntu:~$ su - user2
Password:
$ ls
folder1
$
```

So by **'– '** we are specifying we want complete environment variable reset

Without '–' the environment variable remains the same

su root or **su –** both will get you switch to root user account

As an root user you can directly switch to any other user without providing any password

```
root@ubuntu:~# su simran
simran@ubuntu:/root$
```

And by using '**exit'** you can come outside of the current shell or user

SUDO:

- Sudo-super user do
- It just provides the administrative rights to any other user
- Like if we just try to install apache as other user
- Here you will see the simran user can't use sudo command

```
simran@ubuntu:/root$ sudo apt-get install apache
[sudo] password for simran:
simran is not in the sudoers file.  This incident will be reported.
simran@ubuntu:/root$
```

- Now in order to check details of sudo users we can see the details in file /etc/sudoers

less /etc/sudoers

- Here we can see there is a group "**sudo**" in Ubuntu which includes users those can use sudo

```
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d
(END)
```

- If I check the details of user "simran" then we see that it is not the part of **sudo** group

```
root@ubuntu:~#
root@ubuntu:~# su simran
simran@ubuntu:/root$ id
uid=1000(simran) gid=1000(simran) groups=1000(simran),999(vboxsf)
simran@ubuntu:/root$
```

- Now we will add “simran” user to this sudo group

Usermod -aG sudo simran

```
simran@ubuntu:/root$ su root
Password:
root@ubuntu:~# usermod -aG sudo simran
root@ubuntu:~# id simran
uid=1000(simran) gid=1000(simran) groups=1000(simran),27(sudo),999(vboxsf)
root@ubuntu:~#
```

If switch to user”simran” and try to use sudo command

```
simran@ubuntu:~$ sudo apt-get install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
```

How to remove users access for sudo command
First remove the simran user from group “sudo”

gpasswd -d simran sudo

```
Abort.
simran@ubuntu:/root$ id
uid=1000(simran) gid=1000(simran) groups=1000(simran),27(sudo),999(vboxsf)
simran@ubuntu:/root$ su -
Password:
root@ubuntu:~# gpasswd -d simran sudo
Removing user simran from group sudo
root@ubuntu:~# id simran
uid=1000(simran) gid=1000(simran) groups=1000(simran),999(vboxsf)
root@ubuntu:~#
```

Monitor User commands:

1. who - tell you about currently logged in user

   who -H →Prints column headers.

   who -u → Shows idle time and more details.

```
simran@ubuntu:~$ who
simran   tty2         2023-11-28 13:24 (tty2)
simran@ubuntu:~$ who -H
NAME     LINE         TIME             COMMENT
simran   tty2         2023-11-28 13:24 (tty2)
simran@ubuntu:~$ who -u
simran   tty2         2023-11-28 13:24 00:17        2532 (tty2)
simran@ubuntu:~$
```

2. last Command:
   The last command shows a list of last logged-in users. Common options include:

last
last -n 2 → tell about last 2 logged in session

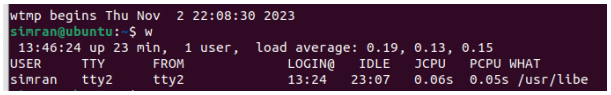
```
simran   tty2         tty2             Fri Nov  3 19:37 - down   (03:35)
reboot   system boot  6.2.0-26-generic Fri Nov  3 19:36 - 23:12  (03:36)
simran   tty2         tty2             Fri Nov  3 14:03 - crash  (05:32)
reboot   system boot  6.2.0-26-generic Fri Nov  3 14:02 - 23:12  (09:10)
simran   tty2         tty2             Fri Nov  3 13:40 - crash  (00:21)
reboot   system boot  6.2.0-26-generic Fri Nov  3 13:39 - 23:12  (09:33)
simran   tty2         tty2             Fri Nov  3 13:22 - crash  (00:17)
reboot   system boot  6.2.0-26-generic Fri Nov  3 13:20 - 23:12  (09:52)
simran   tty2         tty2             Thu Nov  2 22:11 - crash  (15:09)
reboot   system boot  6.2.0-26-generic Thu Nov  2 22:08 - 23:12 (1+01:04)

wtmp begins Thu Nov  2 22:08:30 2023
simran@ubuntu:~$ last -n 2
simran   tty2         tty2             Tue Nov 28 13:24   still logged in
reboot   system boot  6.2.0-37-generic Tue Nov 28 13:23   still running

wtmp begins Thu Nov  2 22:08:30 2023
simran@ubuntu:~$
```

3. w Command:

The w command provides information about currently logged-in users and their activities. Common options include:

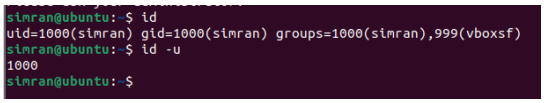


4. id Command:

The id command displays information about the user and their group. Common options include:

Id

Id -u → displays only user id



id -g → print only group iD

Process Management & System Monitoring:

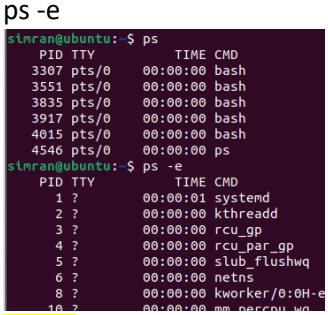
1. **ps** Command:

The ps command is used to display information about active processes. Common options include:

“TTY" refers to the terminal type associated with a particular process.

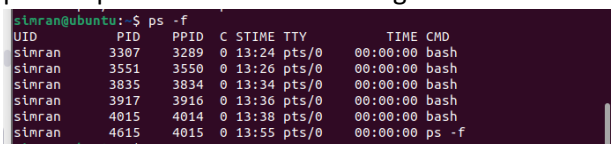
ps

ps -e



pts/0, which indicates that the process with PID 1234 is associated with the terminal pts/0 (pseudo-terminal slave).

ps -f → provides full format listing



ps -aux → gives detailed information about active processes