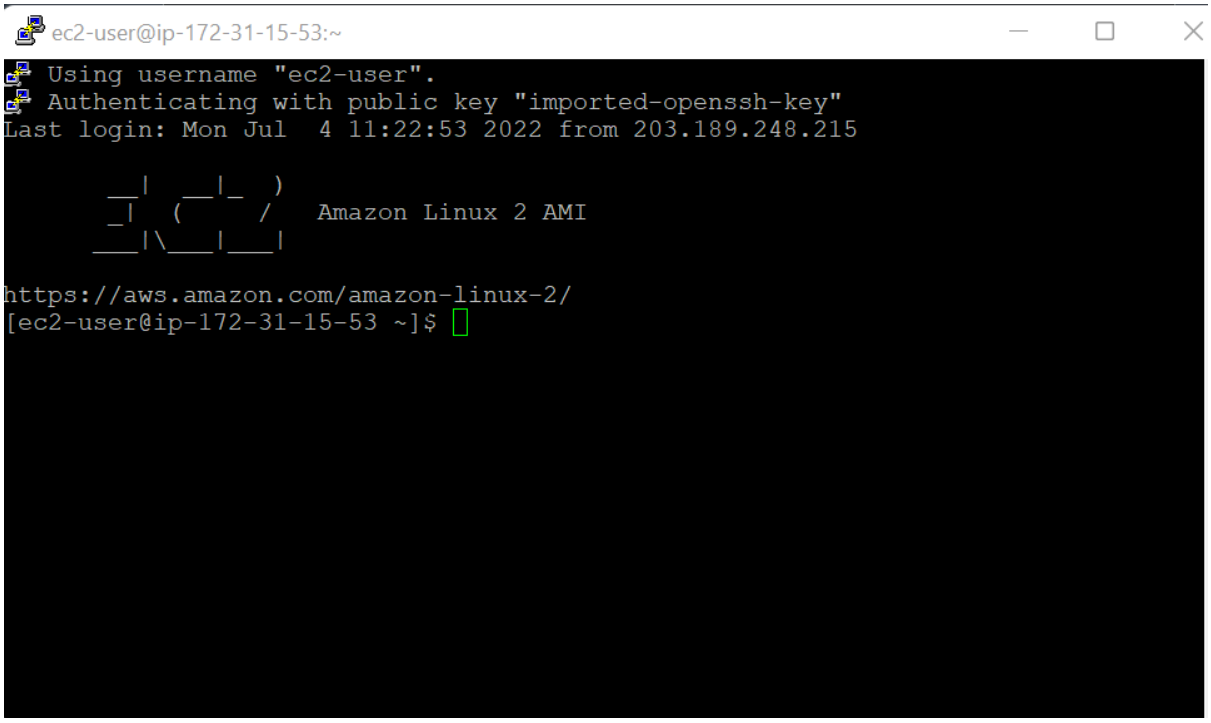
Linux Commands Lab:- By Kishan

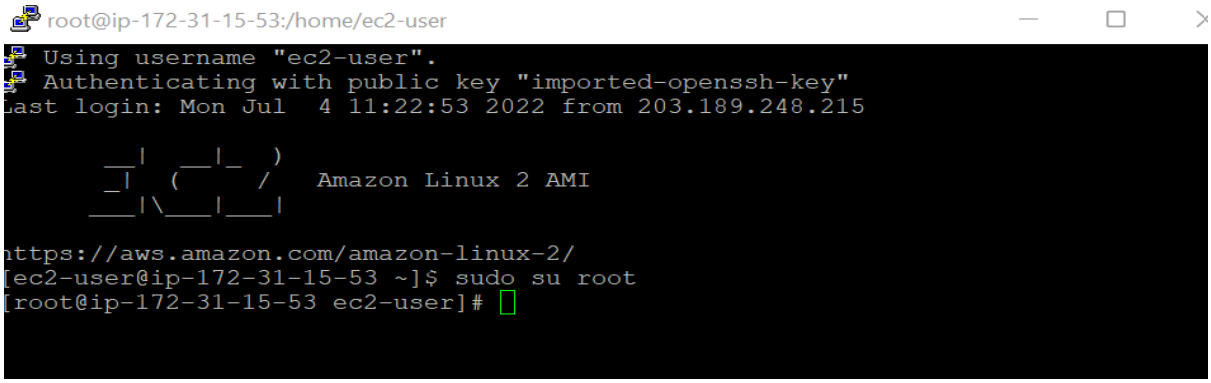
First quickly login to your running ec2-instance using any of the method like ec2-connect, ssh or Putty. (Refer to doc learn how to launch ec2 instance)

You will get Linux terminal in end to work on.



Note:- currently you are logged in as ec2-user. So you have to first login as super user root.

Command:- sudo su root



Now we are logged in as root user.

Lab1:-

1. Command: ls
   Will list all the files and directory in current directory

```
root@ip-172-31-15-53:/home/ec2-user
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
Last login: Mon Jul  4 11:22:53 2022 from 203.189.248.215

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-15-53 ~]$ sudo su root
[root@ip-172-31-15-53 ec2-user]# ls
[root@ip-172-31-15-53 ec2-user]#
```

   Currently empty directory, no file present.

   <u>To create file in linux using cat command</u>

2. Command: cat > file1
   Will give you space to write something, once you are done you can use “ctrl+d” to close.

```
root@ip-172-31-15-53:/home/ec2-user
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
Last login: Mon Jul  4 11:22:53 2022 from 203.189.248.215

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-15-53 ~]$ sudo su root
[root@ip-172-31-15-53 ec2-user]# ls
[root@ip-172-31-15-53 ec2-user]# cat > file1
Hi
Hello
bye
[root@ip-172-31-15-53 ec2-user]#
```

Command: ls
To check file is created or not
Command: cat file1
To read the content of file1

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-15-53 ~]$ sudo su root
[root@ip-172-31-15-53 ec2-user]# ls
[root@ip-172-31-15-53 ec2-user]# cat > file1
Hi
Hello
bye
[root@ip-172-31-15-53 ec2-user]# ls
file1
[root@ip-172-31-15-53 ec2-user]# cat file1
Hi
Hello
bye
[root@ip-172-31-15-53 ec2-user]#
```

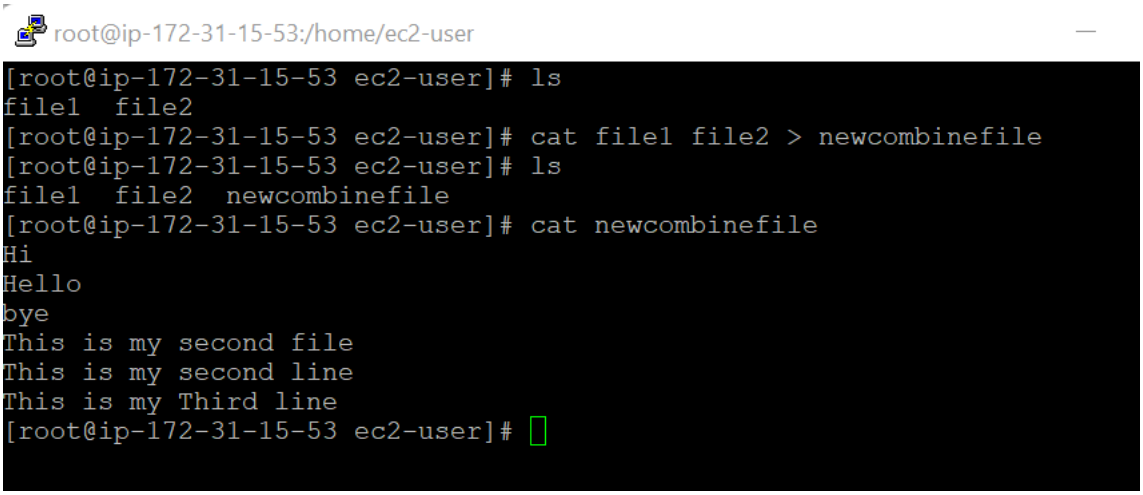
Similarly try to create one more file with name “file2” and add some content and see the content using cat command.

```
[root@ip-172-31-15-53 ec2-user]# ls
file1
[root@ip-172-31-15-53 ec2-user]# cat > file2
This is my second file
[root@ip-172-31-15-53 ec2-user]# ls
file1  file2
[root@ip-172-31-15-53 ec2-user]# cat file2
This is my second file
[root@ip-172-31-15-53 ec2-user]# cat file1
Hi
Hello
bye
[root@ip-172-31-15-53 ec2-user]#
```

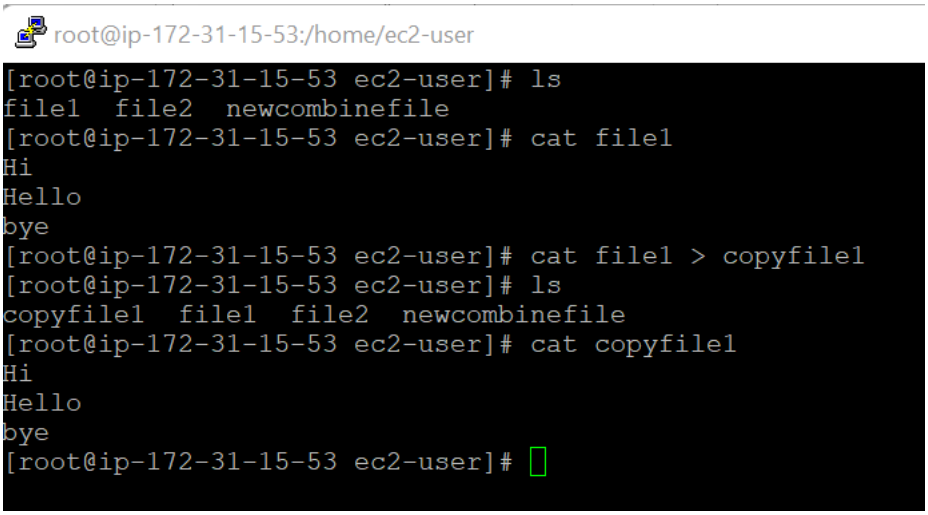
3. Command: cat >> file2
   You can add more lines in the file2 using this command. Previously in file2 we have single line, now we will add two more line and print the content.
   You can see after adding now we have in total three lines.

```
[root@ip-172-31-15-53 ec2-user]# cat >> file2
This is my second line
This is my Third line
[root@ip-172-31-15-53 ec2-user]# cat file2
This is my second file
This is my second line
This is my Third line
[root@ip-172-31-15-53 ec2-user]#
```

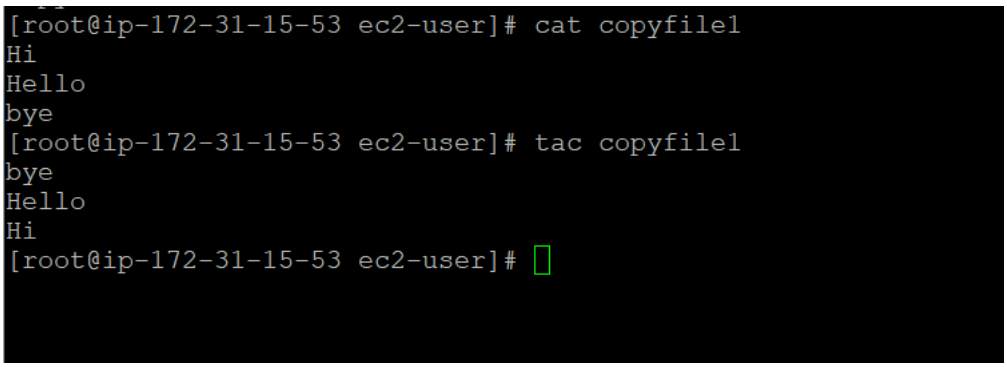
4. Command: cat file1 file2 > newcombinefile
   Will copy the content of file1 and file 2 in “newcombinefile”



5. Command: cat file1 > copyfile1
   Will copy the data of “file1” to “copyfile1”



6. Command: tac copyfile1
   Will invert the data of file and show the output. Last line will shown first and first line will show in last.



<u>To create file in linux using “touch” command</u>

7. Command: touch file3 file4 file5
   Will create three empty files with name “file3”, “file4”, “file5”.

```
[root@ip-172-31-15-53 ec2-user]# ls
copyfile1  file1  file2  newcombinefile
[root@ip-172-31-15-53 ec2-user]# touch file3 file4 file5
[root@ip-172-31-15-53 ec2-user]# ls
copyfile1  file1  file2  file3  file4  file5  newcombinefile
[root@ip-172-31-15-53 ec2-user]#
```

   To check content use command :- cat file3
   Nothing will shown because there is no content in newly create “file3”

```
[root@ip-172-31-15-53 ec2-user]# ls
copyfile1  file1  file2  file3  file4  file5  newcombinefile
[root@ip-172-31-15-53 ec2-user]# cat file3
[root@ip-172-31-15-53 ec2-user]#
```

8. Command: stat file1
   Will show the timestamp of file “file1”.

```
[root@ip-172-31-15-53 ec2-user]# stat file1
  File: 'file1'
  Size: 13              Blocks: 8          IO Block: 4096   regular file
Device: ca01h/51713d    Inode: 12585388    Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2022-07-04 11:32:45.825310619 +0000
Modify: 2022-07-04 11:33:16.669473727 +0000
Change: 2022-07-04 11:33:16.669473727 +0000
 Birth: -
[root@ip-172-31-15-53 ec2-user]#
```

   Now to change the all timestamp(access,modify,change)
   run command: touch file1
   and then again run command: stat file1 {to check whether timestamp changed or not}

```
[root@ip-172-31-15-53 ec2-user]# stat file1
  File: 'file1'
  Size: 13              Blocks: 8          IO Block: 4096   regular file
Device: ca01h/51713d    Inode: 12585388    Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2022-07-04 11:32:45.825310619 +0000
Modify: 2022-07-04 11:33:16.669473727 +0000
Change: 2022-07-04 11:33:16.669473727 +0000
 Birth: -
[root@ip-172-31-15-53 ec2-user]# touch file1
[root@ip-172-31-15-53 ec2-user]# stat file1
  File: 'file1'
  Size: 13              Blocks: 8          IO Block: 4096   regular file
Device: ca01h/51713d    Inode: 12585388    Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2022-07-04 11:57:59.289314172 +0000
Modify: 2022-07-04 11:57:59.289314172 +0000
Change: 2022-07-04 11:57:59.289314172 +0000
 Birth: -
[root@ip-172-31-15-53 ec2-user]#
```

9. Command: touch -a file2
   Will change the only access timestamp of file2.
   Then run command: stat file2 {To check access timestamp changed}

```
[root@ip-172-31-15-53 ec2-user]# touch -a file2
[root@ip-172-31-15-53 ec2-user]# stat file2
  File: 'file2'
  Size: 68              Blocks: 8          IO Block: 4096   regular file
Device: ca01h/51713d    Inode: 12585389    Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2022-07-04 12:00:48.918208603 +0000
Modify: 2022-07-04 11:41:06.771948916 +0000
Change: 2022-07-04 12:00:48.918208603 +0000
 Birth: -
[root@ip-172-31-15-53 ec2-user]#
```

10. Command: touch -m file2
    Will change the modify timestamp.
    Then run command: stat file2 {To check modify timestamp changed}

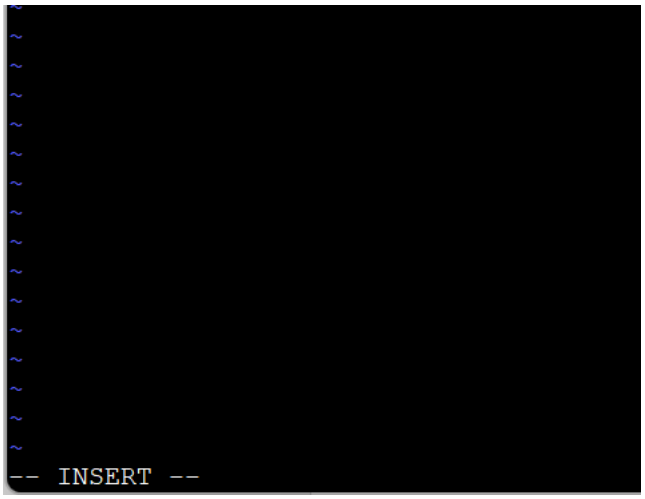
```
[root@ip-172-31-15-53 ec2-user]# touch -m file2
[root@ip-172-31-15-53 ec2-user]# stat file2
  File: 'file2'
  Size: 68              Blocks: 8          IO Block: 4096   regular file
Device: ca01h/51713d    Inode: 12585389    Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2022-07-04 12:00:48.918208603 +0000
Modify: 2022-07-04 12:03:04.766926943 +0000
Change: 2022-07-04 12:03:04.766926943 +0000
 Birth: -
[root@ip-172-31-15-53 ec2-user]#
```

<u>To create file in linux using "vi" command</u>

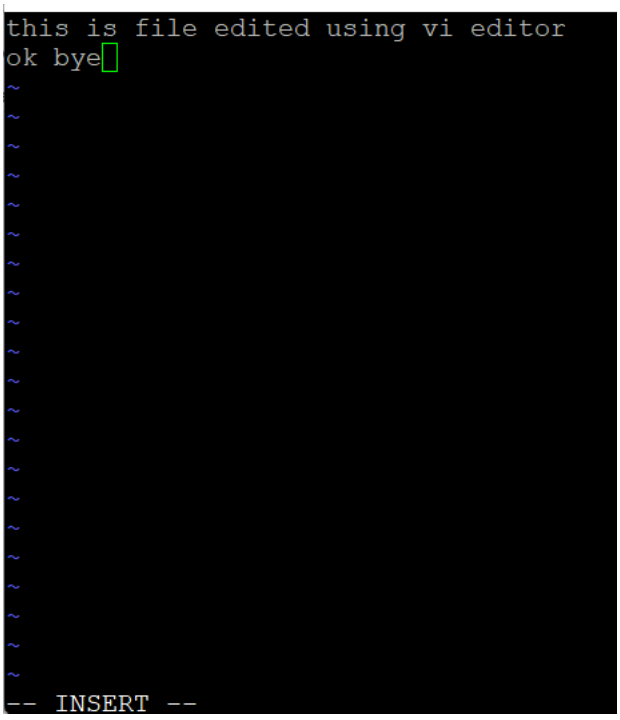
11. Command: vi file5
    Will open file5 in vi editor where we can edit fie and then save it.

```
[root@ip-172-31-15-53 ec2-user]# vi file5
[root@ip-172-31-15-53 ec2-user]#
```

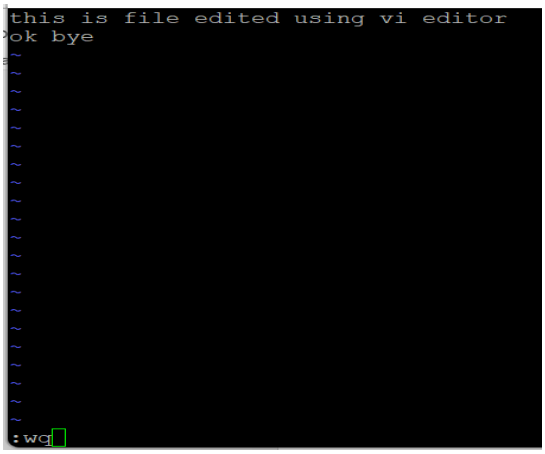
Press “I” then you will see file5 written downside will replace with insert.



Now type whatever content you want to keep.



Now press “ESC” button the press :wq and press enter.



Use command: cat file5

To see whatever we written in “file5” using vi editor

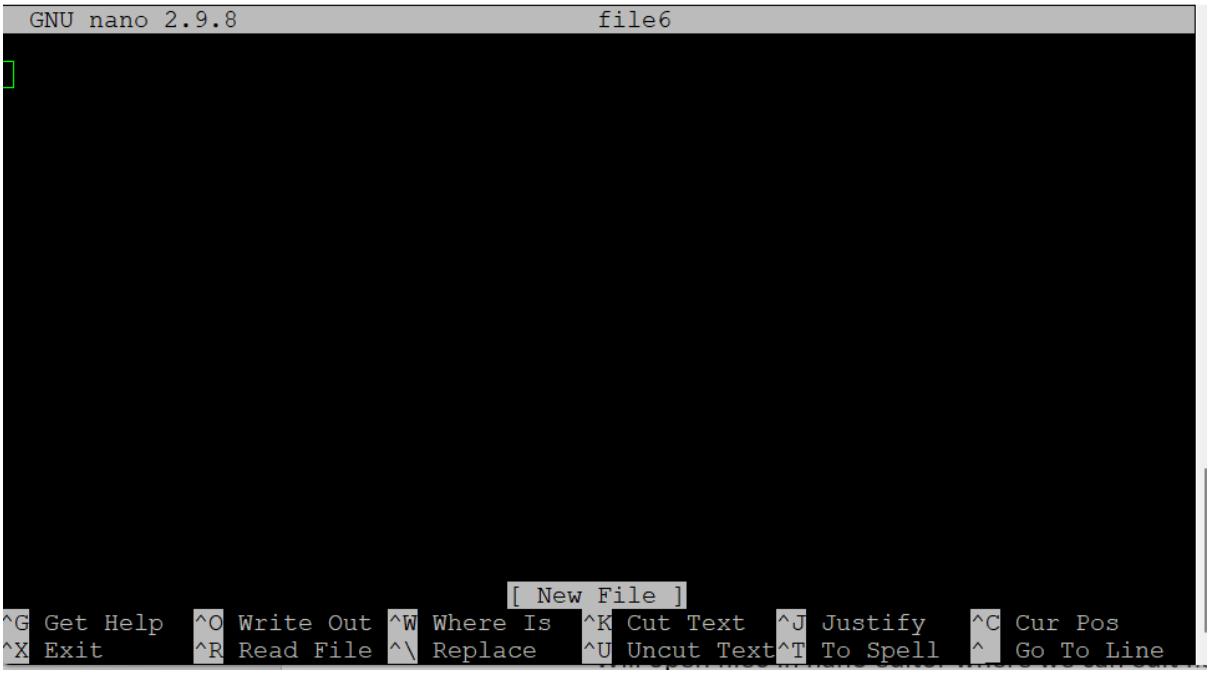
```
[root@ip-172-31-15-53 ec2-user]# cat file5
this is file edited using vi editor
ok bye
[root@ip-172-31-15-53 ec2-user]#
```

<u>To create file in linux using “nano” command</u>

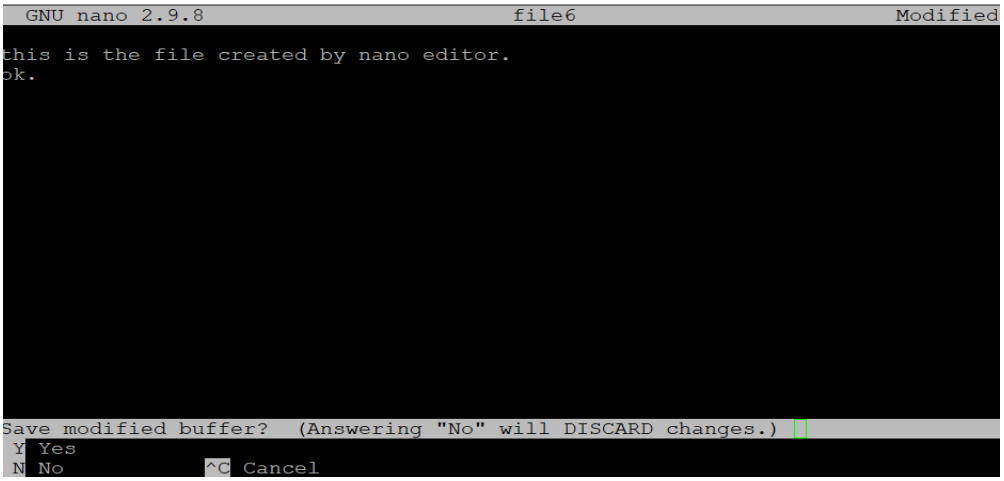
12. Command: nano file6

    Will open file6 in nano editor where we can edit fie and then save it.

```
[root@ip-172-31-15-53 ec2-user]# nano file6
[root@ip-172-31-15-53 ec2-user]#
```



Write whatever you want then press “ctrl+x” and then “y” to save



Use command: cat file6

To see whatever we written using nano editor.

```
[root@ip-172-31-15-53 ec2-user]# cat file6
this is the file created by nano editor.
ok.
[root@ip-172-31-15-53 ec2-user]#
```

13. Command: ls -l

    Will list all the files we create till now with detailed info.

```
[root@ip-172-31-15-53 ec2-user]# ls -l
total 24
-rw-r--r-- 1 root root 13 Jul  4 11:47 copyfile1
-rw-r--r-- 1 root root 13 Jul  4 11:57 file1
-rw-r--r-- 1 root root 68 Jul  4 12:03 file2
-rw-r--r-- 1 root root  0 Jul  4 11:52 file3
-rw-r--r-- 1 root root  0 Jul  4 11:52 file4
-rw-r--r-- 1 root root 43 Jul  4 12:11 file5
-rw-r--r-- 1 root root 45 Jul  4 12:17 file6
-rw-r--r-- 1 root root 81 Jul  4 11:44 newcombinefile
[root@ip-172-31-15-53 ec2-user]#
```

14. Command: clear

    Write clear and then press enter it will clear the screen

15. Command: ll

    Will list all the files with detailed info same as “ls -l command”

```
[root@ip-172-31-15-53 ec2-user]# ll
total 24
-rw-r--r-- 1 root root 13 Jul  4 11:47 copyfile1
-rw-r--r-- 1 root root 13 Jul  4 11:57 file1
-rw-r--r-- 1 root root 68 Jul  4 12:03 file2
-rw-r--r-- 1 root root  0 Jul  4 11:52 file3
-rw-r--r-- 1 root root  0 Jul  4 11:52 file4
-rw-r--r-- 1 root root 43 Jul  4 12:11 file5
-rw-r--r-- 1 root root 45 Jul  4 12:17 file6
-rw-r--r-- 1 root root 81 Jul  4 11:44 newcombinefile
[root@ip-172-31-15-53 ec2-user]#
```

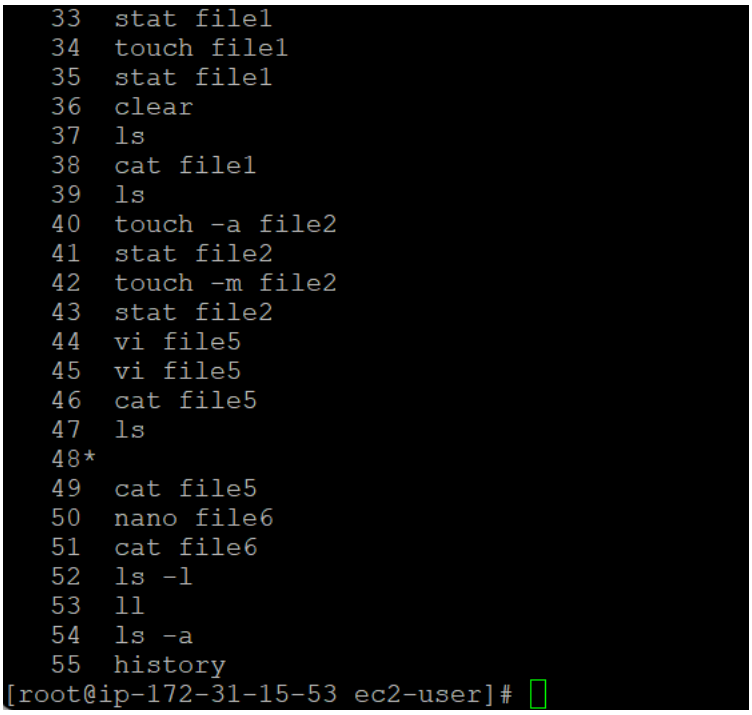
16. Command: ls -a

    Will list all the files including hidden files.

```
[root@ip-172-31-15-53 ec2-user]# ls -a
.   .bash_history  .bash_profile  copyfile1  file2  file4  file6          .ssh
..  .bash_logout   .bashrc        file1      file3  file5  newcombinefile
[root@ip-172-31-15-53 ec2-user]#
```

17. Command: history

Will list all the command we used in this lab.



LAB1 COMPLETED.

LAB2:

Login to ec2-instance. Use “sudo su root” to login as root

```
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
Last login: Mon Jul  4 11:25:48 2022 from 203.189.248.215

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-15-53 ~]$
```

1. Command: ls
   To list all the files {if you are using the same ec2-instance then it will show all the files whatever we created in Lab1}

```
[ec2-user@ip-172-31-15-53 ~]$ ls
copyfile1  file1  file2  file3  file4  file5  file6  newcombinefile
[ec2-user@ip-172-31-15-53 ~]$
```

2. Command: rm -rvf *
   Will delete all the files created previously in this directory.

```
[ec2-user@ip-172-31-15-53 ~]$ ls
copyfile1  file1  file2  file3  file4  file5  file6  newcombinefile
[ec2-user@ip-172-31-15-53 ~]$ rm -rvf *
removed 'copyfile1'
removed 'file1'
removed 'file2'
removed 'file3'
removed 'file4'
removed 'file5'
removed 'file6'
removed 'newcombinefile'
[ec2-user@ip-172-31-15-53 ~]$ ls
[ec2-user@ip-172-31-15-53 ~]$
```

3. Command: touch file1 file2 file3
   Will create three empty file with name “file1”, “file2”, “file3”.

```
[root@ip-172-31-15-53 ec2-user]# touch file1 file2 file3
[root@ip-172-31-15-53 ec2-user]# ls
file1  file2  file3
[root@ip-172-31-15-53 ec2-user]#
```

4. Command: mkdir dir1
   Will create a directory with name dir1

```
[root@ip-172-31-15-53 ec2-user]# mkdir dir1
[root@ip-172-31-15-53 ec2-user]# ls
dir1  file1  file2  file3
[root@ip-172-31-15-53 ec2-user]#
```

5. Command: mkdir dir2 dir3
   Will create two directory with name “dir2” and “dir3”

```
[root@ip-172-31-15-53 ec2-user]# mkdir dir2 dir3
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  file1  file2  file3
[root@ip-172-31-15-53 ec2-user]#
```

6. Command: ls -l
   Will show the list in detailed form.
   All the directory list is starting with “dr….” d for directory
   All the file list is starting with “-r….” – for file

```
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwxr-xr-x 2 root root 6 Jul  4 12:35 dir1
drwxr-xr-x 2 root root 6 Jul  4 12:36 dir2
drwxr-xr-x 2 root root 6 Jul  4 12:36 dir3
-rw-r--r-- 1 root root 0 Jul  4 12:33 file1
-rw-r--r-- 1 root root 0 Jul  4 12:33 file2
-rw-r--r-- 1 root root 0 Jul  4 12:33 file3
[root@ip-172-31-15-53 ec2-user]#
```

7. Command: touch .file4
   Will create a hidden file with name “file4”
   To list file command: ls
   To list file including hidden file command: ls -a

```
[root@ip-172-31-15-53 ec2-user]# touch .file4
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  file1  file2  file3
[root@ip-172-31-15-53 ec2-user]# ls -a
.   .bash_history  .bash_profile  dir1  dir3   file2  .file4
..  .bash_logout   .bashrc        dir2  file1  file3  .ssh
[root@ip-172-31-15-53 ec2-user]#
```

8. Command: mkdir .dir4
   Will create a hidden directory with name “dir4”

```
[root@ip-172-31-15-53 ec2-user]# mkdir .dir4
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  file1  file2  file3
[root@ip-172-31-15-53 ec2-user]# ls -a
.   .bash_history  .bash_profile  dir1  dir3   file1  file3   .ssh
..  .bash_logout   .bashrc        dir2  .dir4  file2  .file4
[root@ip-172-31-15-53 ec2-user]#
```

9. Command: ls -al
   Will show all the files and directory including hidden ones.

```
[root@ip-172-31-15-53 ec2-user]# ls -al
total 16
drwx------ 7 ec2-user ec2-user 197 Jul  4 12:42 .
drwxr-xr-x 3 root     root      22 Jul  2 07:17 ..
-rw------- 1 ec2-user ec2-user  35 Jul  4 12:28 .bash_history
-rw-r--r-- 1 ec2-user ec2-user  18 Jul 15  2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15  2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15  2020 .bashrc
drwxr-xr-x 2 root     root       6 Jul  4 12:35 dir1
drwxr-xr-x 2 root     root       6 Jul  4 12:36 dir2
drwxr-xr-x 2 root     root       6 Jul  4 12:36 dir3
drwxr-xr-x 2 root     root       6 Jul  4 12:42 .dir4
-rw-r--r-- 1 root     root       0 Jul  4 12:33 file1
-rw-r--r-- 1 root     root       0 Jul  4 12:33 file2
-rw-r--r-- 1 root     root       0 Jul  4 12:33 file3
-rw-r--r-- 1 root     root       0 Jul  4 12:41 .file4
drwx------ 2 ec2-user ec2-user  29 Jul  2 07:17 .ssh
[root@ip-172-31-15-53 ec2-user]#
```

10. Command: mkdir -p dir5/dira/dirb/dirc
    Will create a parent directory with name dir5, and inside dir5 it will create a directory dira, inside directory dira it will create dirb and so on.

```
[root@ip-172-31-15-53 ec2-user]# mkdir -p dir5/dira/dirb/dirc
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  dir5  file1  file2  file3
[root@ip-172-31-15-53 ec2-user]#
```

11. Command: cd dir5
    To change current directory, it will forward us to “dir5” where we will able to “dira” using command: ls

```
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  dir5  file1  file2  file3
[root@ip-172-31-15-53 ec2-user]# cd dir5
[root@ip-172-31-15-53 dir5]# ls
dira
[root@ip-172-31-15-53 dir5]#
```

12. Command: pwd
    Pwd-print working directory will show the current path.

```
[root@ip-172-31-15-53 dir5]# pwd
/home/ec2-user/dir5
[root@ip-172-31-15-53 dir5]#
```

13. Command: cd ..
    To go to parent directory

```
[root@ip-172-31-15-53 dir5]# pwd
/home/ec2-user/dir5
[root@ip-172-31-15-53 dir5]# cd ..
[root@ip-172-31-15-53 ec2-user]# pwd
/home/ec2-user
[root@ip-172-31-15-53 ec2-user]#
```

14. Command: cd /

    Will forward you to root directory

```
[root@ip-172-31-15-53 ec2-user]# cd /
[root@ip-172-31-15-53 /]# ls
bin   dev  home  lib64  media  opt   root  sbin  sys  usr
boot  etc  lib   local  mnt    proc  run   srv   tmp  var
[root@ip-172-31-15-53 /]#
```

    These file structure we already discussed.

15. Command: cd home/ec2-user

    Will land you to ec2-user directory inside home.

```
[root@ip-172-31-15-53 /]# cd home/ec2-user/
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  dir5  file1  file2  file3
[root@ip-172-31-15-53 ec2-user]#
```

16. Command: vi file1

    Will open the file1 in editor, press “I” to insert data, add some data and then press “esc” then “:wq”

    And use command: cat file1 to see the data.

```
Adding data in file 1 using vi editor
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

```
[root@ip-172-31-15-53 ec2-user]# cat file1
Adding data in file 1 using vi editor
[root@ip-172-31-15-53 ec2-user]#
```

17. Command: cp file1 file2

    And then press y.

    Will copy data of “file1” to “file2”.

    Use command: cat file2 to see the data

```
[root@ip-172-31-15-53 ec2-user]# cp file1 file2
cp: overwrite 'file2'? y
```

```
[root@ip-172-31-15-53 ec2-user]# cp file1 file2
cp: overwrite 'file2'? y
[root@ip-172-31-15-53 ec2-user]# cat file2
Adding data in file 1 using vi editor
[root@ip-172-31-15-53 ec2-user]#
```

18. Command: cp file2 dir1

    Will copy the file2 inside directory "dir1"

```
[root@ip-172-31-15-53 ec2-user]# cp file2 dir1
[root@ip-172-31-15-53 ec2-user]# cd dir1
[root@ip-172-31-15-53 dir1]# ls
file2
[root@ip-172-31-15-53 dir1]# cat file2
Adding data in file 1 using vi editor
[root@ip-172-31-15-53 dir1]#
```

19. Command: cd ..

    Will again move you to parent directory i.e /home/ec2-user

```
[root@ip-172-31-15-53 dir1]# cd ..
[root@ip-172-31-15-53 ec2-user]#
```

20. Command: mv file1 dir2

    Will move file1 from /home/ec2-user to /home/ec2-user/dir2

```
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  dir5  file1  file2  file3
[root@ip-172-31-15-53 ec2-user]# mv file1 dir2
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  dir5  file2  file3
[root@ip-172-31-15-53 ec2-user]# cd dir2
[root@ip-172-31-15-53 dir2]# ls
file1
[root@ip-172-31-15-53 dir2]# pwd
/home/ec2-user/dir2
[root@ip-172-31-15-53 dir2]#
```

21. Command: cd ..

    Will again move you to parent directory i.e /home/ec2-user

```
/home/ec2-user/dir2
[root@ip-172-31-15-53 dir2]# cd ..
[root@ip-172-31-15-53 ec2-user]#
```

22. Command: mv dir2 dir1

    Will move directory “dir2” to inside “dir1”

```
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dir3  dir5  file2  file3
[root@ip-172-31-15-53 ec2-user]# mv dir2 dir1
[root@ip-172-31-15-53 ec2-user]# cd dir1
[root@ip-172-31-15-53 dir1]# ls
dir2  file2
[root@ip-172-31-15-53 dir1]#
```

23. Command: cd ..

    Will again move you to parent directory i.e /home/ec2-user

24. Command: mv dir1 newdir1

    Will rename the directory “dir1” to “newdir1”

```
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir3  dir5  file2  file3
[root@ip-172-31-15-53 ec2-user]# mv dir1 newdir1
[root@ip-172-31-15-53 ec2-user]# ls
dir3  dir5  file2  file3  newdir1
[root@ip-172-31-15-53 ec2-user]#
```

25. Command: mv file2 .file2

    Will make file “file2” as a hidden file

```
[root@ip-172-31-15-53 ec2-user]# ls
dir3  dir5  file2  file3  newdir1
[root@ip-172-31-15-53 ec2-user]# mv file2 .file2
[root@ip-172-31-15-53 ec2-user]# ls
dir3  dir5  file3  newdir1
[root@ip-172-31-15-53 ec2-user]# ls -a
.    .bash_history  .bash_profile  dir3   dir5    file3   newdir1
..   .bash_logout   .bashrc        .dir4  .file2  .file4  .ssh
[root@ip-172-31-15-53 ec2-user]#
```

26. Command: rm -rf newdir1

    Will delete the directory “newdir1”

```
[root@ip-172-31-15-53 ec2-user]# ls
dir3  dir5  file3  newdir1
[root@ip-172-31-15-53 ec2-user]# rm -rf newdir1
[root@ip-172-31-15-53 ec2-user]# ls
dir3  dir5  file3
[root@ip-172-31-15-53 ec2-user]#
```

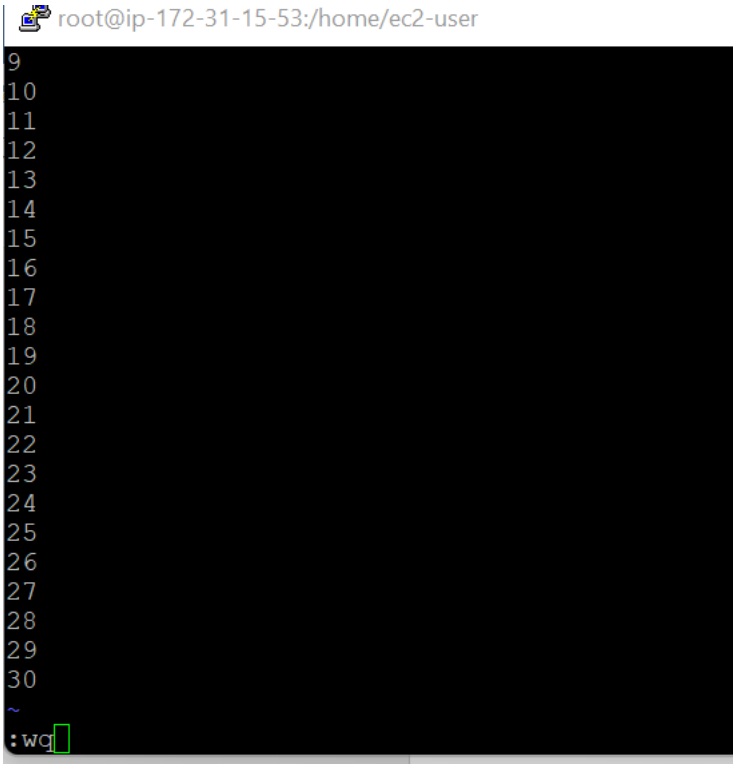
27. Command: rm -rf file3

    Will delete the file “file3”

```
[root@ip-172-31-15-53 ec2-user]# ls
dir3  dir5  file3
[root@ip-172-31-15-53 ec2-user]# rm -rf file3
[root@ip-172-31-15-53 ec2-user]# ls
dir3  dir5
[root@ip-172-31-15-53 ec2-user]#
```

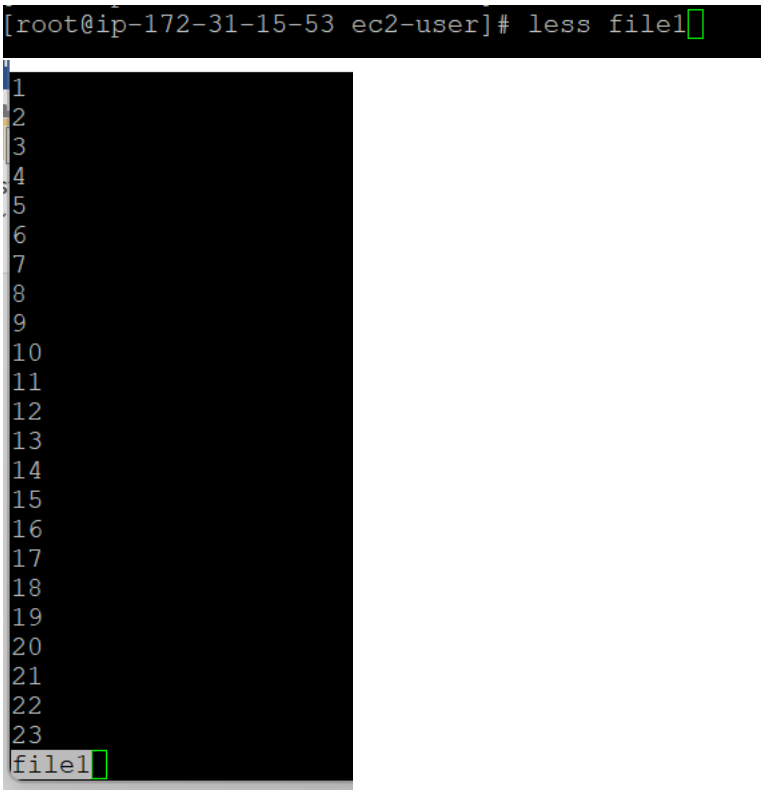
28. Command: vi file1

    Will open vi editor and add 1 to 30 serially.



29. Command: less file1

    Will show the content in 1 page and then press “q” to exit.



    Press “q” to exit from here

30. Command: head file1

    Will show the first 10 line(by default)

```
[root@ip-172-31-15-53 ec2-user]# head file1
1
2
3
4
5
6
7
8
9
10
[root@ip-172-31-15-53 ec2-user]#
```

31. Command: head file1 -n 6

    Will show the first 6 line

```
[root@ip-172-31-15-53 ec2-user]# head file1 -n 6
1
2
3
4
5
6
[root@ip-172-31-15-53 ec2-user]#
```

32. Command: tail file1

    Will show the last 10 line

```
[root@ip-172-31-15-53 ec2-user]# tail file1
21
22
23
24
25
26
27
28
29
30
[root@ip-172-31-15-53 ec2-user]#
```

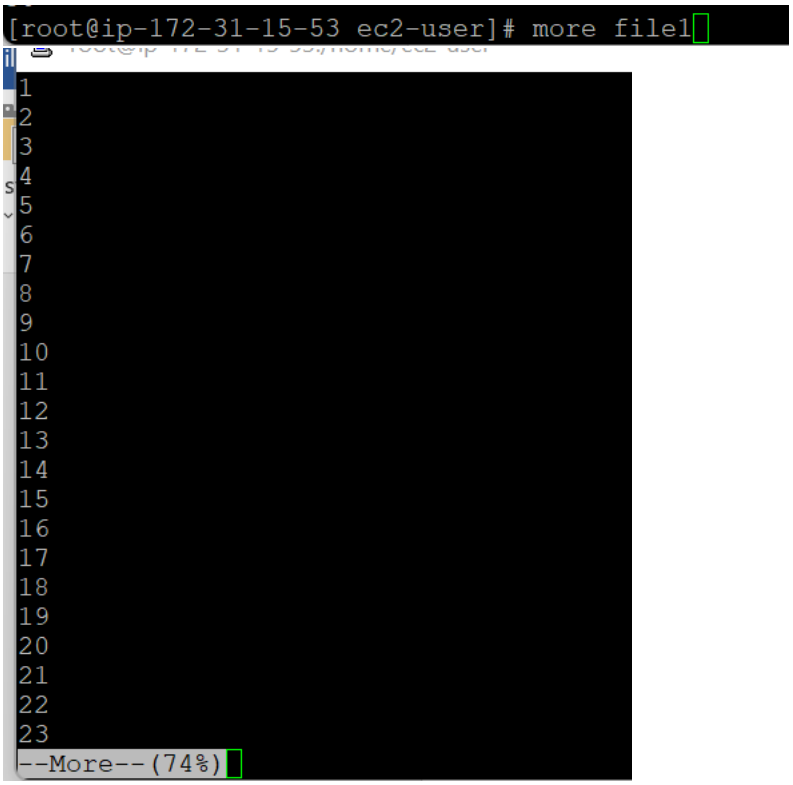
33. Command: tail file1 -n 4

    Will show the last 4 line

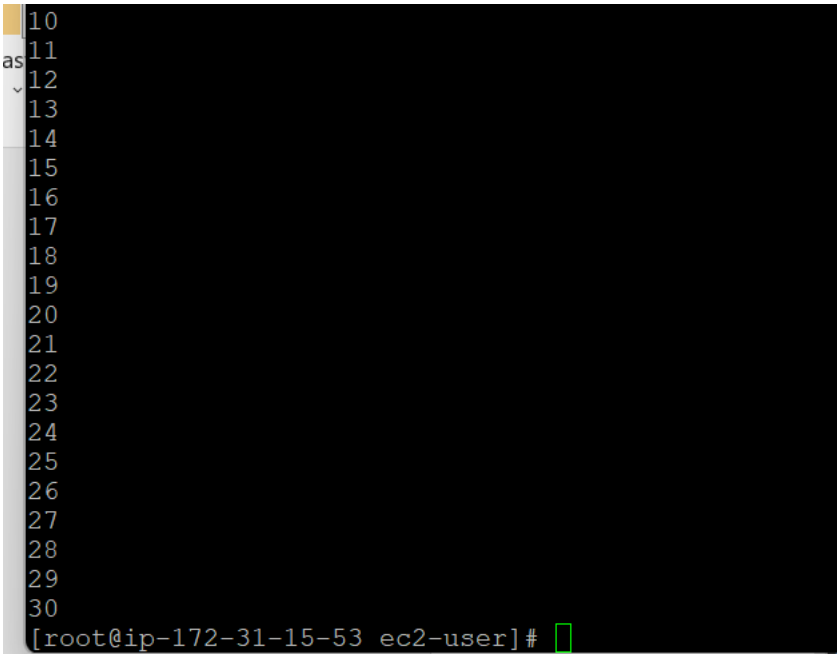
```
[root@ip-172-31-15-53 ec2-user]# tail file1 -n 4
27
28
29
30
[root@ip-172-31-15-53 ec2-user]#
```

34. Command: more file1

Will show all lines in form of pages.



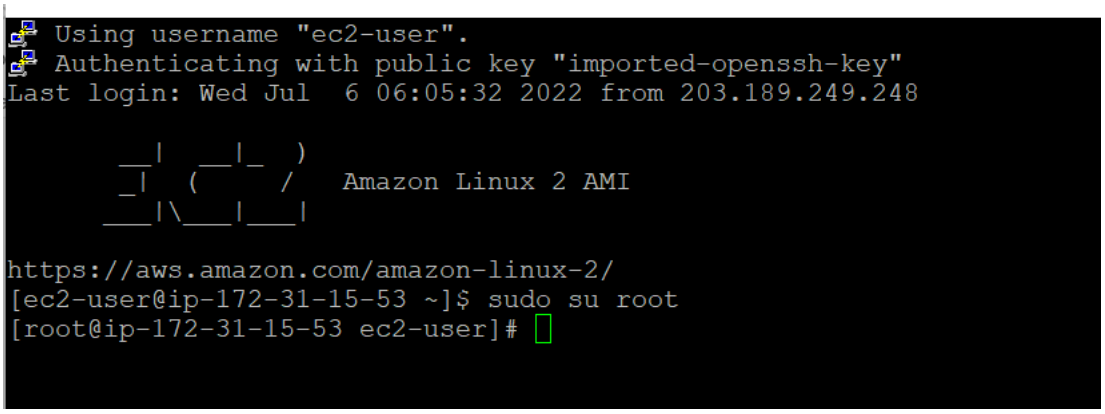
Press enter enter you will see it will go down pages and after 30 it will quit.



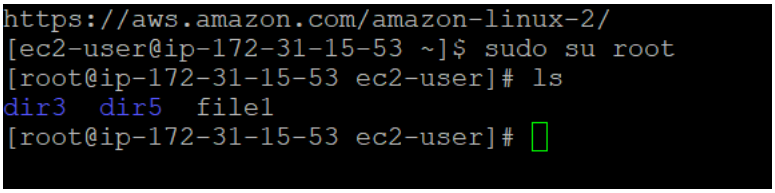
Lab2 finished……

LAB 3:

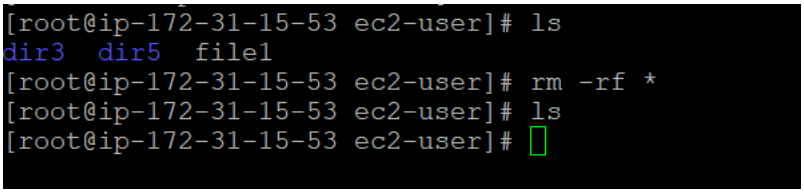
1. Command: sudo su root
   To login as root user.



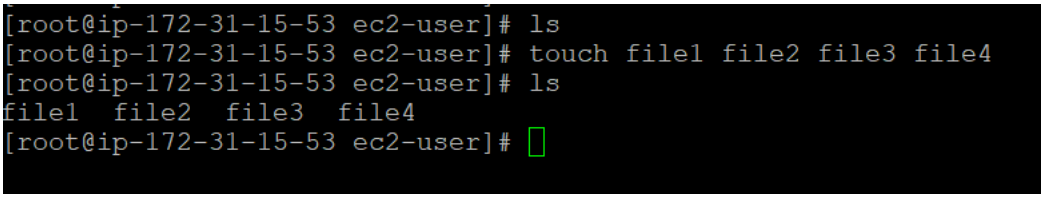
2. Command: ls
   To list previously created file. Will list the file created in previous lab.



3. Command: rm -rf *
   To remove all the file and directory created in previous lab.



4. Command: touch file1 file2 file3 fil4
   Will create four empty files.



5. Command: mkdir dira dirb dirc
   Will create 3 empty directory

```
[root@ip-172-31-15-53 ec2-user]# ls
file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]# mkdir dira dirb dirc
[root@ip-172-31-15-53 ec2-user]# ls
dira  dirb  dirc  file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]#
```

6. Command: yum update -y
   Will update all the software/packages installed in this o.s
   No package marked for update means already updated.

```
[root@ip-172-31-15-53 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 3.7 kB     00:00
No packages marked for update
[root@ip-172-31-15-53 ec2-user]#
```

7. Command: mkdir -p dir1/dir2/dir3
   Will create dir 3 inside dir2 which will be inside dir1.

```
[root@ip-172-31-15-53 ec2-user]# ls
dira  dirb  dirc  file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]# mkdir -p dir1/dir2/dir3
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dira  dirb  dirc  file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]#
```

8. Command: tree
   Will say command not found

```
[root@ip-172-31-15-53 ec2-user]# tree
bash: tree: command not found
[root@ip-172-31-15-53 ec2-user]#
```

9. Command: yum install tree -y
   Will install tree package in system.

```
[root@ip-172-31-15-53 ec2-user]# yum install tree -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 3.7 kB     00:00
Resolving Dependencies
--> Running transaction check
---> Package tree.x86_64 0:1.6.0-10.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package     Arch          Version                    Repository         Size
```

10. Command: tree

    Will show the structure of files in tree form.

```
[root@ip-172-31-15-53 ec2-user]# tree
.
├── dir1
│   └── dir2
│       └── dir3
├── dira
├── dirb
├── dirc
├── file1
├── file2
├── file3
└── file4

6 directories, 4 files
[root@ip-172-31-15-53 ec2-user]#
```

11. Command: grep root /etc/passwd

    will find word “root” in file /etc/passwd

```
[root@ip-172-31-15-53 ec2-user]# grep root /etc/passwd
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/sbin/nologin
[root@ip-172-31-15-53 ec2-user]#
```

12. Command: cat >> file1

    Add some lines in file name file1

```
[root@ip-172-31-15-53 ec2-user]# cat >> file1
hello
hi
software
devops
aws
ubuntu
python
java
norton
[root@ip-172-31-15-53 ec2-user]#
```

13. Command: sort file1

    Will arrange the line in alphabetical order.

```
[root@ip-172-31-15-53 ec2-user]# sort file1
aws
devops
hello
hi
java
norton
python
software
ubuntu
[root@ip-172-31-15-53 ec2-user]#
```

14. Command: hostname

    Will give the host name of your system

```
[root@ip-172-31-15-53 ec2-user]# hostname
ip-172-31-15-53.ap-south-1.compute.internal
[root@ip-172-31-15-53 ec2-user]#
```

15. Command: hostname -i

    Will show the ip address of system.

```
[root@ip-172-31-15-53 ec2-user]# hostname -i
172.31.15.53
[root@ip-172-31-15-53 ec2-user]#
```

16. Command: ifconfig

    To get detailed info about your system ip, mac address and other networking details.

```
[root@ip-172-31-15-53 ec2-user]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 172.31.15.53  netmask 255.255.240.0  broadcast 172.31.15.255
        inet6 fe80::850:1aff:febd:66c  prefixlen 64  scopeid 0x20<link>
        ether 0a:50:1a:bd:06:6c  txqueuelen 1000  (Ethernet)
        RX packets 2695  bytes 763069 (745.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2303  bytes 253570 (247.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@ip-172-31-15-53 ec2-user]#
```

17. Command: cat /etc/os-release

    Will gives details about release version of o.s

```
[root@ip-172-31-15-53 ec2-user]# cat /etc/os-release
NAME="Amazon Linux"
VERSION="2"
ID="amzn"
ID_LIKE="centos rhel fedora"
VERSION_ID="2"
PRETTY_NAME="Amazon Linux 2"
ANSI_COLOR="0;33"
CPE_NAME="cpe:2.3:o:amazon:amazon_linux:2"
HOME_URL="https://amazonlinux.com/"
[root@ip-172-31-15-53 ec2-user]#
```

18. Command: yum install httpd -y

    Will install httpd package in your o.s

```
[root@ip-172-31-15-53 ec2-user]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 3.7 kB     00:00
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.53-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.53-1.amzn2 for package: httpd-2.4.
3-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.53-1.amzn2 for package: httpd
2.4.53-1.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.53-1.amzn2
x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.53-1.amzn2.x86_64
```

19. Command: service httpd status or systemctl status httpd

    Will show the status of httpd/apache package.

    Both command will show the same output.

```
[root@ip-172-31-15-53 ec2-user]# service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor prese
t: disabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)
[root@ip-172-31-15-53 ec2-user]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor prese
t: disabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)
[root@ip-172-31-15-53 ec2-user]#
```

20. Command: service httpd start or systemctl start httpd

    Will start the httpd service

    Both command will give same output.

```
[root@ip-172-31-15-53 ec2-user]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-15-53 ec2-user]#
```

21. Command: systemctl status httpd

    Will show the status of httpd package.

```
[root@ip-172-31-15-53 ec2-user]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor prese
t: disabled)
   Active: active (running) since Wed 2022-07-06 07:19:08 UTC; 1min 13s ago
     Docs: man:httpd.service(8)
 Main PID: 3926 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes se
rved/sec:    0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─3926 /usr/sbin/httpd -DFOREGROUND
           ├─3927 /usr/sbin/httpd -DFOREGROUND
           ├─3928 /usr/sbin/httpd -DFOREGROUND
           ├─3929 /usr/sbin/httpd -DFOREGROUND
           ├─3930 /usr/sbin/httpd -DFOREGROUND
           └─3931 /usr/sbin/httpd -DFOREGROUND

Jul 06 07:19:08 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Start...
Jul 06 07:19:08 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Start...
Hint: Some lines were ellipsized, use -l to show in full.
[root@ip-172-31-15-53 ec2-user]#
```

22. Command: yum update httpd -y

    Will update the httpd package.

    Currently it is up to date.

```
[root@ip-172-31-15-53 ec2-user]# yum update httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 3.7 kB     00:00
No packages marked for update
[root@ip-172-31-15-53 ec2-user]#
```

23. Command: yum remove httpd -y

    Will remove the httpd package from system.

```
[root@ip-172-31-15-53 ec2-user]# yum remove httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-mo
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.53-1.amzn2 will be erased
--> Processing Dependency: httpd-mmn = 20120211x8664 for package: mo
 19-1.amzn2.0.1.x86_64
```

24. Command: yum install httpd -y

    To install again

```
[root@ip-172-31-15-53 ec2-user]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.53-1.amzn2 will be installed
--> Processing Dependency: mod_http2 for package: httpd-2.4.53-1.
--> Running transaction check
---> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be insta
```

25. Command: service httpd start

    To start the apache/httpd service

```
[root@ip-172-31-15-53 ec2-user]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-15-53 ec2-user]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled;
t: disabled)
   Active: active (running) since Wed 2022-07-06 07:25:12 UTC; 7s a
     Docs: man:httpd.service(8)
 Main PID: 4124 (httpd)
   Status: "Processing requests..."
   CGroup: /system.slice/httpd.service
           └─4124 /usr/sbin/httpd -DFOREGROUND
```

26. Command: service httpd stop

    To stop the httpd service.

```
[root@ip-172-31-15-53 ec2-user]# service httpd stop
Redirecting to /bin/systemctl stop httpd.service
[root@ip-172-31-15-53 ec2-user]# systemctl  status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor prese
t: disabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)

Jul 06 07:22:52 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Stopp...
Jul 06 07:22:53 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Stopp...
Jul 06 07:25:12 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Start...
Jul 06 07:25:12 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Start...
Jul 06 07:26:22 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Stopp...
Jul 06 07:26:23 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Stopp...
Jul 06 07:27:01 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Start...
Jul 06 07:27:01 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Start...
```

27. Command: systemctl enable httpd –now or chkconfig httpd on

    Will enable the httpd service so even you will restart the system service will also restart automatically.

```
[root@ip-172-31-15-53 ec2-user]# systemctl enable httpd --now
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service t
o /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-15-53 ec2-user]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset
: disabled)
   Active: active (running) since Wed 2022-07-06 07:30:42 UTC; 6s ago
     Docs: man:httpd.service(8)
 Main PID: 4368 (httpd)
   Status: "Processing requests..."
   CGroup: /system.slice/httpd.service
           ├─4368 /usr/sbin/httpd -DFOREGROUND
           ├─4369 /usr/sbin/httpd -DFOREGROUND
           ├─4370 /usr/sbin/httpd -DFOREGROUND
           ├─4371 /usr/sbin/httpd -DFOREGROUND
           ├─4372 /usr/sbin/httpd -DFOREGROUND
           └─4373 /usr/sbin/httpd -DFOREGROUND

Jul 06 07:30:42 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Start...
Jul 06 07:30:42 ip-172-31-15-53.ap-south-1.compute.internal systemd[1]: Start...
Hint: Some lines were ellipsized, use -l to show in full.
[root@ip-172-31-15-53 ec2-user]#
```

28. Command: yum list installed

    Will show all the installed packages in this o.s

```
[root@ip-172-31-15-53 ec2-user]# yum list installed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
GeoIP.x86_64                          1.5.0-11.amzn2.0.2            installed
PyYAML.x86_64                         3.10-11.amzn2.0.2             installed
acl.x86_64                            2.2.51-14.amzn2               installed
acpid.x86_64                          2.0.19-9.amzn2.0.1            installed
amazon-linux-extras.noarch            2.0.1-1.amzn2                 installed
amazon-linux-extras-yum-plugin.noarch 2.0.1-1.amzn2                 installed
amazon-ssm-agent.x86_64               3.1.1188.0-1.amzn2            installed
apr.x86_64                            1.7.0-9.amzn2                 @amzn2-core
apr-util.x86_64                       1.6.1-5.amzn2.0.2             @amzn2-core
apr-util-bdb.x86_64                   1.6.1-5.amzn2.0.2             @amzn2-core
```

29. Command: which tree

    Will show the path of installed package

```
[root@ip-172-31-15-53 ec2-user]# which tree
/bin/tree
[root@ip-172-31-15-53 ec2-user]# which ansible
/usr/bin/which: no ansible in (/sbin:/bin:/usr/sbin:/usr/bin)
[root@ip-172-31-15-53 ec2-user]#
```

    If not installed will show not in.

30. Command: echo “hello”

    Will show hello as output.

```
[root@ip-172-31-15-53 ec2-user]# echo "hello"
hello
[root@ip-172-31-15-53 ec2-user]#
```

31. Command: echo “hello devops” > echotextfile

    Will write “hello devops” in “echotextfile”

```
[root@ip-172-31-15-53 ec2-user]# echo "hello devops" > echotextfile
[root@ip-172-31-15-53 ec2-user]# cat echotextfile
hello devops
[root@ip-172-31-15-53 ec2-user]#
```

    LAB3 Finished.

LAB 4:

1. Command: sudo su root
   To login as root user.

```
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
Last login: Wed Jul  6 06:05:32 2022 from 203.189.249.248

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-15-53 ~]$ sudo su root
[root@ip-172-31-15-53 ec2-user]#
```

2. Command: ls
   To list previously created file. Will list the file created in previous lab.

```
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dira  dirb  dirc  echotextfile  file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]#
```

3. Command: rm -rf *
   To remove all the file and directory created in previous lab.

```
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dira  dirb  dirc  echotextfile  file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]# rm -rf *
[root@ip-172-31-15-53 ec2-user]# ls
[root@ip-172-31-15-53 ec2-user]#
```

4. Command: touch file1 file2 file3 fil4
   Will create four empty files.

```
[root@ip-172-31-15-53 ec2-user]# ls
[root@ip-172-31-15-53 ec2-user]# touch file1 file2 file3 file4
[root@ip-172-31-15-53 ec2-user]# ls
file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]#
```

5. Command: mkdir dir1 dir2
   Will create two directory

```
[root@ip-172-31-15-53 ec2-user]# mkdir dir1 dir2
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]#
```

6. Command: mkdir -p dirx/diry/dirz
   Will create directory inside directory

```
[root@ip-172-31-15-53 ec2-user]# mkdir -p dirx/diry/dirz
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  file1  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]#
```

7. Command: tree
   Will show file and directory in tree structure

```
[root@ip-172-31-15-53 ec2-user]# tree
.
├── dir1
├── dir2
├── dirx
│   └── diry
│       └── dirz
├── file1
├── file2
├── file3
└── file4

5 directories, 4 files
[root@ip-172-31-15-53 ec2-user]#
```

8. Command: useradd kishan
   It will create a user with name “kishan”

```
[root@ip-172-31-15-53 ec2-user]# useradd kishan
[root@ip-172-31-15-53 ec2-user]#
```

9. Command: cat /etc/passwd
   To check whether user “kishan” is created or not.

```
ec2-user:x:1000:1000:EC2 Default User:/home/ec2-user:/bin/bash
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
kishan:x:1001:1001::/home/kishan:/bin/bash
[root@ip-172-31-15-53 ec2-user]#
```

10. Command: groupadd devops
    It will create a group with name “devops”

```
[root@ip-172-31-15-53 ec2-user]# groupadd devops
[root@ip-172-31-15-53 ec2-user]#
```

11. Command: cat /etc/group
    To check whether group “devops” is created or not.

```
ec2-user:x:1000:
apache:x:48:
kishan:x:1001:
devops:x:1002:
[root@ip-172-31-15-53 ec2-user]#
```

Note: “kishan” we created as user but it is also visible in group because user kishan was not added in group so by default linux create a group with same name as of user.

12. Command: useradd john
    Command: useradd jack
    To add two user with name john and jack.

```
[root@ip-172-31-15-53 ec2-user]# useradd john
[root@ip-172-31-15-53 ec2-user]# useradd jack
[root@ip-172-31-15-53 ec2-user]#
```

13. Commad: gpasswd -a kishan devops
    Add user kishan previously created to group devops.

```
[root@ip-172-31-15-53 ec2-user]# gpasswd -a kishan devops
Adding user kishan to group devops
[root@ip-172-31-15-53 ec2-user]#
```

14. Command: gpasswd -M john,jack devops
    Will add multiple user to group devops

```
[root@ip-172-31-15-53 ec2-user]# gpasswd -M john,jack devops
[root@ip-172-31-15-53 ec2-user]#
```

15. Command: cat /etc/group
    Will show these two user added in devops group.

```
devops:x:1002:john,jack
john:x:1003:
jack:x:1004:
[root@ip-172-31-15-53 ec2-user]#
```

16. Command: cat >> file1
    ADD some line in this file

```
[root@ip-172-31-15-53 ec2-user]# cat >> file1
Hi
Hello
Kishan
Bye
[root@ip-172-31-15-53 ec2-user]# cat file1
Hi
Hello
Kishan
Bye
[root@ip-172-31-15-53 ec2-user]#
```

17. Command: ln -s file1 softlinkfile1
    Will create a soft link of file1 with name “softlinkfile1”

```
[root@ip-172-31-15-53 ec2-user]# ln -s file1 softlinkfile1
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  file1  file2  file3  file4  softlinkfile1
[root@ip-172-31-15-53 ec2-user]#
```

18. Command: cat softlinkfile1

    Will show the exact same content as of file1

```
[root@ip-172-31-15-53 ec2-user]# cat softlinkfile1
Hi
Hello
Kishan
Bye
[root@ip-172-31-15-53 ec2-user]#
```

19. Command: ls -l

    See the details of softlink file.

```
[root@ip-172-31-15-53 ec2-user]# ls -l
total 4
drwxr-xr-x 2 root root  6 Jul  6 08:28 dir1
drwxr-xr-x 2 root root  6 Jul  6 08:28 dir2
drwxr-xr-x 3 root root 18 Jul  6 08:30 dirx
-rw-r--r-- 1 root root 21 Jul  6 08:49 file1
-rw-r--r-- 1 root root  0 Jul  6 08:28 file2
-rw-r--r-- 1 root root  0 Jul  6 08:28 file3
-rw-r--r-- 1 root root  0 Jul  6 08:28 file4
lrwxrwxrwx 1 root root  5 Jul  6 08:51 softlinkfile1 -> file1
[root@ip-172-31-15-53 ec2-user]#
```

20. Command: cat >> softlinkfile1

    Add some line and it will be added in file1 directly.

```
[root@ip-172-31-15-53 ec2-user]# cat >> softlinkfile1
this line is added from softlink file.
[root@ip-172-31-15-53 ec2-user]# cat file1
Hi
Hello
Kishan
Bye
this line is added from softlink file.
[root@ip-172-31-15-53 ec2-user]#
```

21. Command: rm -rf file1

    Delete file1 after this softlink will not work as main file “file1” deleted.

    Softlinkfile1 in red color means not accessible.

```
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwxr-xr-x 2 root root  6 Jul  6 08:28 dir1
drwxr-xr-x 2 root root  6 Jul  6 08:28 dir2
drwxr-xr-x 3 root root 18 Jul  6 08:30 dirx
-rw-r--r-- 1 root root  0 Jul  6 08:28 file2
-rw-r--r-- 1 root root  0 Jul  6 08:28 file3
-rw-r--r-- 1 root root  0 Jul  6 08:28 file4
lrwxrwxrwx 1 root root  5 Jul  6 08:51 softlinkfile1 -> file1
[root@ip-172-31-15-53 ec2-user]#
```

22. Command: rm -rf softlinkfile1

    Will remove softlinkfile1 as it was of no use.

```
[root@ip-172-31-15-53 ec2-user]# rm -rf softlinkfile1
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  file2  file3  file4
[root@ip-172-31-15-53 ec2-user]#
```

23. Command: cat >> file2

    ADD some line in this file

```
[root@ip-172-31-15-53 ec2-user]# cat >> file2
Hi
adding data in file2
will create hardlink for this file
ok
bye
[root@ip-172-31-15-53 ec2-user]# cat file2
Hi
adding data in file2
will create hardlink for this file
ok
bye
[root@ip-172-31-15-53 ec2-user]#
```

24. Command: ln file2 hardlinkfile2

    Will create a hardlink of file2 with name “hardlinkfile2”

```
[root@ip-172-31-15-53 ec2-user]# ln file2 hardlinkfile2
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  file2  file3  file4  hardlinkfile2
[root@ip-172-31-15-53 ec2-user]#
```

```
[root@ip-172-31-15-53 ec2-user]# ls -l
total 8
drwxr-xr-x 2 root root  6 Jul  6 08:28 dir1
drwxr-xr-x 2 root root  6 Jul  6 08:28 dir2
drwxr-xr-x 3 root root 18 Jul  6 08:30 dirx
-rw-r--r-- 2 root root 66 Jul  6 09:01 file2
-rw-r--r-- 1 root root  0 Jul  6 08:28 file3
-rw-r--r-- 1 root root  0 Jul  6 08:28 file4
-rw-r--r-- 2 root root 66 Jul  6 09:01 hardlinkfile2
[root@ip-172-31-15-53 ec2-user]#
```

    Note: No color change in case of hardlink as it is exactly same as file similar to backup.

25. Command: cat hardlinkfile2

    Will show the exact content as of file2.

```
[root@ip-172-31-15-53 ec2-user]# cat hardlinkfile2
Hi
adding data in file2
will create hardlink for this file
ok
bye
[root@ip-172-31-15-53 ec2-user]#
```

26. Command: cat >> hardlinkfile2

    Add some more line and it will reflect in file2 also

```
[root@ip-172-31-15-53 ec2-user]# cat >> hardlinkfile2
this line added by hardlink
[root@ip-172-31-15-53 ec2-user]# cat file2
Hi
adding data in file2
will create hardlink for this file
ok
bye
this line added by hardlink
[root@ip-172-31-15-53 ec2-user]#
```

27. Command: rm -rf file2

    Will remove file 2 but hardlink will still be accessible.

```
[root@ip-172-31-15-53 ec2-user]# rm -rf file2
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  file3  file4  hardlinkfile2
[root@ip-172-31-15-53 ec2-user]# cat hardlinkfile2
Hi
adding data in file2
will create hardlink for this file
ok
bye
this line added by hardlink
[root@ip-172-31-15-53 ec2-user]#
```

28. Command: tree

    Will show the tree structure

```
[root@ip-172-31-15-53 ec2-user]# tree
.
├── dir1
├── dir2
├── dirx
│   └── diry
│       └── dirz
├── file3
├── file4
└── hardlinkfile2

5 directories, 3 files
[root@ip-172-31-15-53 ec2-user]#
```

29. Command: tar -cvf dirx.tar dirx #tape archieve

    Will create a tar file. Create verbose force

```
[root@ip-172-31-15-53 ec2-user]# tar -cvf dirx.tar dirx
dirx/
dirx/diry/
dirx/diry/dirz/
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  dirx.tar  file3  file4  hardlinkfile2
[root@ip-172-31-15-53 ec2-user]#
```

30. Command: ls -l

    Will do detailed list

```
[root@ip-172-31-15-53 ec2-user]# ls -l
total 16
drwxr-xr-x 2 root root     6 Jul  6 08:28 dir1
drwxr-xr-x 2 root root     6 Jul  6 08:28 dir2
drwxr-xr-x 3 root root    18 Jul  6 08:30 dirx
-rw-r--r-- 1 root root 10240 Jul  6 09:43 dirx.tar
-rw-r--r-- 1 root root     0 Jul  6 08:28 file3
-rw-r--r-- 1 root root     0 Jul  6 08:28 file4
-rw-r--r-- 1 root root    94 Jul  6 09:06 hardlinkfile2
[root@ip-172-31-15-53 ec2-user]#
```

31. Command: gzip dirx.tar

    Will compress the tar file

    File size reduced from 10240 to 154 bytes.

```
[root@ip-172-31-15-53 ec2-user]# ls -l
total 16
drwxr-xr-x 2 root root     6 Jul  6 08:28 dir1
drwxr-xr-x 2 root root     6 Jul  6 08:28 dir2
drwxr-xr-x 3 root root    18 Jul  6 08:30 dirx
-rw-r--r-- 1 root root 10240 Jul  6 09:43 dirx.tar
-rw-r--r-- 1 root root     0 Jul  6 08:28 file3
-rw-r--r-- 1 root root     0 Jul  6 08:28 file4
-rw-r--r-- 1 root root    94 Jul  6 09:06 hardlinkfile2
[root@ip-172-31-15-53 ec2-user]# gzip dirx.tar
[root@ip-172-31-15-53 ec2-user]# ls -l
total 8
drwxr-xr-x 2 root root   6 Jul  6 08:28 dir1
drwxr-xr-x 2 root root   6 Jul  6 08:28 dir2
drwxr-xr-x 3 root root  18 Jul  6 08:30 dirx
-rw-r--r-- 1 root root 154 Jul  6 09:43 dirx.tar.gz
-rw-r--r-- 1 root root   0 Jul  6 08:28 file3
-rw-r--r-- 1 root root   0 Jul  6 08:28 file4
-rw-r--r-- 1 root root  94 Jul  6 09:06 hardlinkfile2
[root@ip-172-31-15-53 ec2-user]#
```

32. Command: gunzip dirx.tar.gz

    Will unzip the file

```
[root@ip-172-31-15-53 ec2-user]# gunzip dirx.tar.gz
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  dirx.tar  file3  file4  hardlinkfile2
[root@ip-172-31-15-53 ec2-user]#
```

33. Command: rm -rf dirx

    Will delete first dirx

```
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  dirx.tar  file3  file4  hardlinkfile2
[root@ip-172-31-15-53 ec2-user]# rm -rf dirx
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx.tar  file3  file4  hardlinkfile2
[root@ip-172-31-15-53 ec2-user]#
```

34. Command: tar -xvf dirx.tar dir1
    Will extract the dir from tarfiles

```
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx.tar  file3  file4  hardlinkfile2
[root@ip-172-31-15-53 ec2-user]# tar -xvf dirx.tar
dirx/
dirx/diry/
dirx/diry/dirz/
[root@ip-172-31-15-53 ec2-user]# ls
dir1  dir2  dirx  dirx.tar  file3  file4  hardlinkfile2
[root@ip-172-31-15-53 ec2-user]#
```

35. Command: wget https://download-ib01.fedoraproject.org/pub/epel/7/aarch64/Packages/a/ansible-2.8.5-1.el7.noarch.rpm
    Will download ansible

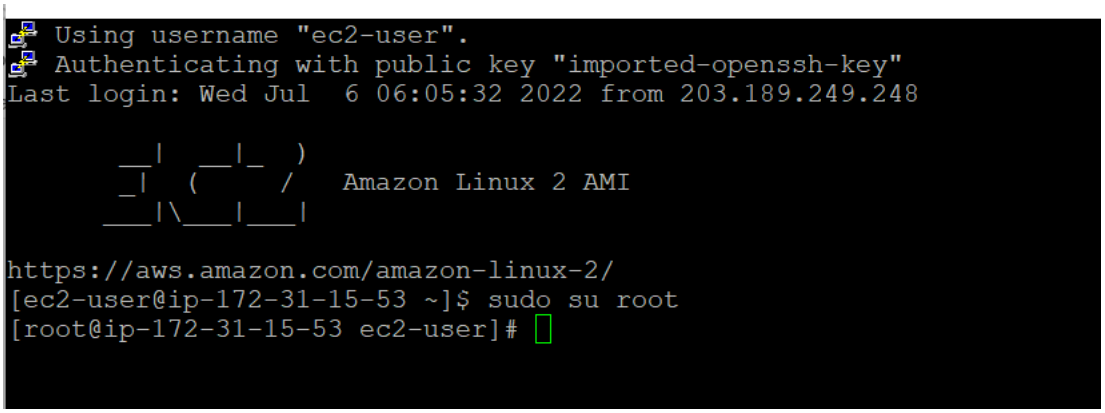
```
[root@ip-172-31-15-53 ec2-user]# wget https://download-ib01.fedoraproject.org/pu
b/epel/7/aarch64/Packages/a/ansible-2.8.5-1.el7.noarch.rpm
--2022-07-06 09:58:34--  https://download-ib01.fedoraproject.org/pub/epel/7/aarc
h64/Packages/a/ansible-2.8.5-1.el7.noarch.rpm
Resolving download-ib01.fedoraproject.org (download-ib01.fedoraproject.org)... 1
52.19.134.145, 2600:2701:4000:5211:dead:beef:fe:fed6
Connecting to download-ib01.fedoraproject.org (download-ib01.fedoraproject.org)|
152.19.134.145|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 15452459 (15M) [application/x-rpm]
Saving to: 'ansible-2.8.5-1.el7.noarch.rpm'

77% [============================>        ] 11,968,512  1.78MB/s  eta 3s
```

```
[root@ip-172-31-15-53 ec2-user]# ls
ansible-2.8.5-1.el7.noarch.rpm  dir2  dirx.tar  file4
dir1                            dirx  file3     hardlinkfile2
```

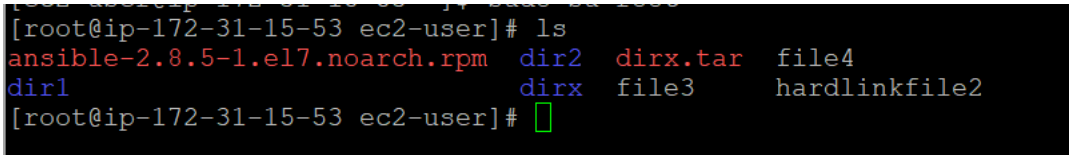
Lab4 finished.

LAB 5

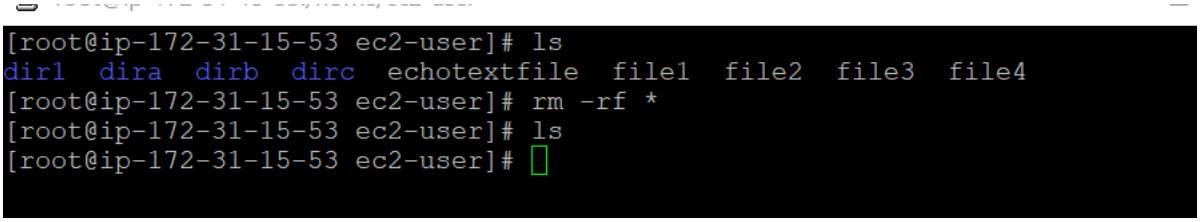
1. Command: sudo su root
   To login as root user.



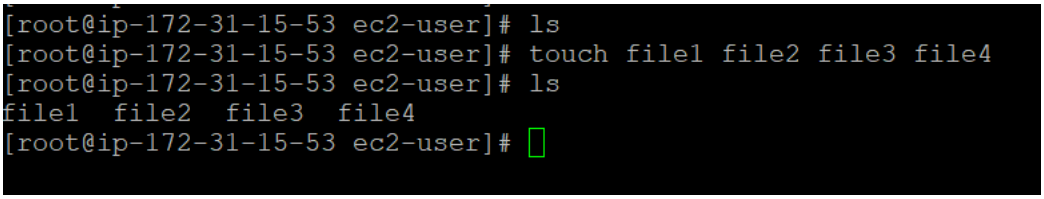
2. Command: ls
   To list previously created file. Will list the file created in previous lab.



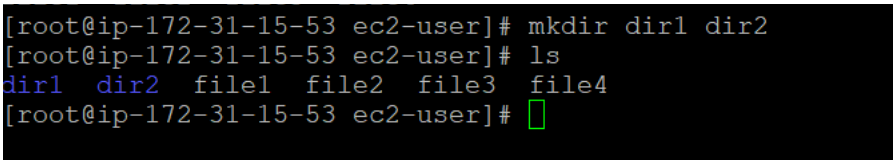
3. Command: rm -rf *
   To remove all the file and directory created in previous lab.



4. Command: touch file1 file2 file3 fil4
   Will create four empty files.



5. Command: mkdir dir1 dir2
   Will create two directory



6. Command: useradd roy
   Will create a user with name “roy”

```
[root@ip-172-31-15-53 ec2-user]# useradd roy
[root@ip-172-31-15-53 ec2-user]# cat /etc/passwd
```

```
jack:x:1003:1004::/home/jack:/bin/bash
roy:x:1004:1005::/home/roy:/bin/bash
[root@ip-172-31-15-53 ec2-user]#
```

7. Command: groupadd linux
   Will create a group with name linux

```
[root@ip-172-31-15-53 ec2-user]# groupadd linux
```

```
jack:x:1004:
roy:x:1005:
linux:x:1006:
[root@ip-172-31-15-53 ec2-user]#
```

8. Command: ls -l
   To see detailed list
   In case of directory: drwxr-xr-x (755) is default permission
   In case of file: -rw-r—r—(644) is default permission
   The number after permission denotes number of hardlink, for dir2 for file 1.
   Root and root denotes owner and group of firectory and file.
   6 and 0 is the respective directory and file size in bytes.
   Jul 6 is date 12:34 is time.

```
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwxr-xr-x 2 root root 6 Jul  6 12:34 dir1
drwxr-xr-x 2 root root 6 Jul  6 12:34 dir2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file1
-rw-r--r-- 1 root root 0 Jul  6 12:34 file2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file3
-rw-r--r-- 1 root root 0 Jul  6 12:34 file4
[root@ip-172-31-15-53 ec2-user]#
```

9. Command: chmod 777 dir1
   Will change the permission of dir1 and will change it to rwx for everyone that means owner,group and others.

```
[root@ip-172-31-15-53 ec2-user]# chmod 777 dir1
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwxrwxrwx 2 root root 6 Jul  6 12:34 dir1
drwxr-xr-x 2 root root 6 Jul  6 12:34 dir2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file1
-rw-r--r-- 1 root root 0 Jul  6 12:34 file2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file3
-rw-r--r-- 1 root root 0 Jul  6 12:34 file4
[root@ip-172-31-15-53 ec2-user]#
```

10. Command: chmod 700 dir1

    Will give rwx permission to owner and group,others will don't have any permission.

```
[root@ip-172-31-15-53 ec2-user]# chmod 700 dir1
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwx------ 2 root root 6 Jul  6 12:34 dir1
drwxr-xr-x 2 root root 6 Jul  6 12:34 dir2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file1
-rw-r--r-- 1 root root 0 Jul  6 12:34 file2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file3
-rw-r--r-- 1 root root 0 Jul  6 12:34 file4
[root@ip-172-31-15-53 ec2-user]#
```

11. Command: chmod 755 file1

    Will give rwx permission to owner, rx to group , rx to others.

```
[root@ip-172-31-15-53 ec2-user]# chmod 755 file1
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwx------ 2 root root 6 Jul  6 12:34 dir1
drwxr-xr-x 2 root root 6 Jul  6 12:34 dir2
-rwxr-xr-x 1 root root 0 Jul  6 12:34 file1
-rw-r--r-- 1 root root 0 Jul  6 12:34 file2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file3
-rw-r--r-- 1 root root 0 Jul  6 12:34 file4
[root@ip-172-31-15-53 ec2-user]#
```

12. Command: chmod g=r,o=rw dir2

    Will give read permission to group and rw permission to others.

```
[root@ip-172-31-15-53 ec2-user]# chmod g=r,o=rw dir2
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwx------ 2 root root 6 Jul  6 12:34 dir1
drwxr--rw- 2 root root 6 Jul  6 12:34 dir2
-rwxr-xr-x 1 root root 0 Jul  6 12:34 file1
-rw-r--r-- 1 root root 0 Jul  6 12:34 file2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file3
-rw-r--r-- 1 root root 0 Jul  6 12:34 file4
[root@ip-172-31-15-53 ec2-user]#
```

13. Command: chmod u=r,g=rwx,o=x file4

    Will give read permission to owner, rwx to group and x to others.

```
[root@ip-172-31-15-53 ec2-user]# chmod u=r,g=rwx,o=x file4
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwx------ 2 root root 6 Jul  6 12:34 dir1
drwxr--rw- 2 root root 6 Jul  6 12:34 dir2
-rwxr-xr-x 1 root root 0 Jul  6 12:34 file1
-rw-r--r-- 1 root root 0 Jul  6 12:34 file2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file3
-r--rwx--x 1 root root 0 Jul  6 12:34 file4
[root@ip-172-31-15-53 ec2-user]#
```

14. Command: chmod u+w,g-x,o+rw file4

Will add w permission to owner, will remove x permission from group and will add rw permission to others.

```
[root@ip-172-31-15-53 ec2-user]# chmod u+w,g-x,o+rw file4
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwx------ 2 root root 6 Jul  6 12:34 dir1
drwxr--rw- 2 root root 6 Jul  6 12:34 dir2
-rwxr-xr-x 1 root root 0 Jul  6 12:34 file1
-rw-r--r-- 1 root root 0 Jul  6 12:34 file2
-rw-r--r-- 1 root root 0 Jul  6 12:34 file3
-rw-rw-rwx 1 root root 0 Jul  6 12:34 file4
[root@ip-172-31-15-53 ec2-user]#
```

15. Command: chown roy dir1

Will change the owner of dir1 from root to roy

```
[root@ip-172-31-15-53 ec2-user]# chown roy dir1
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwx------ 2 roy  root 6 Jul  6 12:34 dir1
drwxr--rw- 2 root root 6 Jul  6 12:34 dir2
-rwxr-xr-x 1 root root 0 Jul  6 12:34 file1
```

16. Command: chgrp linux file1

Will change the group of file1 from root to linux

```
[root@ip-172-31-15-53 ec2-user]# chgrp linux file1
[root@ip-172-31-15-53 ec2-user]# ls -l
total 0
drwx------ 2 roy  root  6 Jul  6 12:34 dir1
drwxr--rw- 2 root root  6 Jul  6 12:34 dir2
-rwxr-xr-x 1 root linux 0 Jul  6 12:34 file1
-rw-r--r-- 1 root root  0 Jul  6 12:34 file2
-rw-r--r-- 1 root root  0 Jul  6 12:34 file3
-rw-rw-rwx 1 root root  0 Jul  6 12:34 file4
[root@ip-172-31-15-53 ec2-user]#
```

Lab5 finished…..