Task on Linux:

==========

1) **Create user with name Techie and provide sudo access to user.**

1st I am creating the user name with Techie.

To create user CMD ----- useradd username

To set the password CMD ----- passwd username ------- which user you want to set the password.

```
[root@ip-172-31-93-132 ~]# sudo passwd Techie
Changing password for user Techie.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ip-172-31-93-132 ~]# su Techie
[Techie@ip-172-31-93-132 root]$ pwd
/root
[Techie@ip-172-31-93-132 root]$ sudo yum -y install git

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for Techie:
Techie is not in the sudoers file.  This incident will be reported.
```

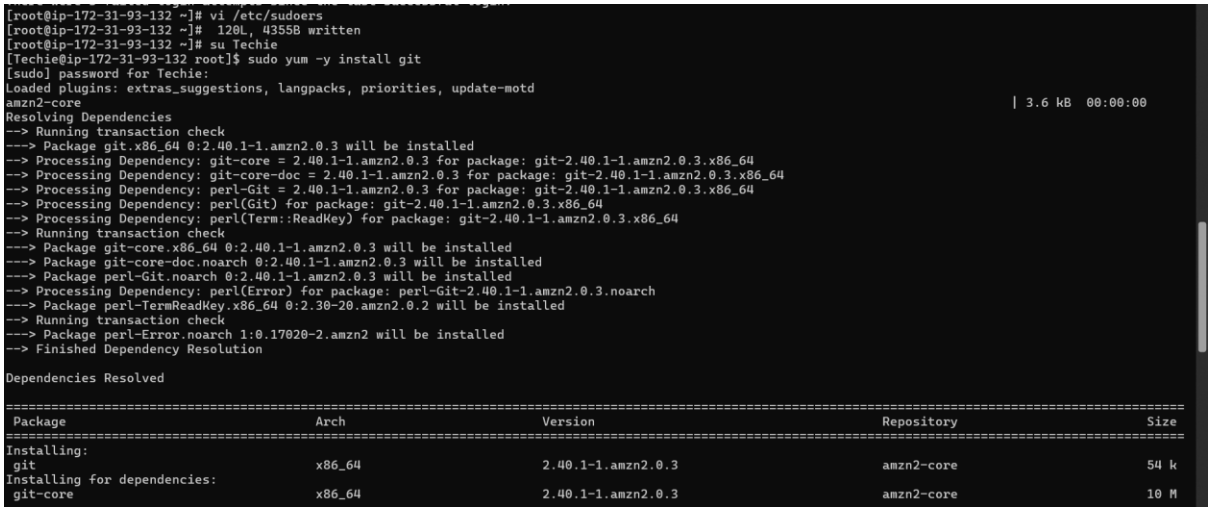
The above image show's one error that is ---- Techie user is not in the sudoers file. That means this user doesn't have the permissions to download any package we have to give the sudo permission to user.

Steps:

- Go to the root user CMD ----- sudo su –
- You are in the Techie user then you have to type CMD ----- exit after these go to the root user.
- Go to these file CMD ----- vi /etc/sudoers

```
#
# Adding HOME to env_keep may enable a user to run unrestricted
# commands via sudo.
#
# Defaults   env_keep += "HOME"

Defaults    secure_path = /sbin:/bin:/usr/sbin:/usr/bin

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
## systems).
## Syntax:
##
##      user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)       ALL

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS

## Allows people in group wheel to run all commands
%wheel  ALL=(ALL)       ALL

Techie ALL=(ALL)       ALL
## Same thing without a password
# %wheel        ALL=(ALL)       NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
## cdrom as root
# %users  ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom

## Allows members of the users group to shutdown this system
# %users  localhost=/sbin/shutdown -h now

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)
```

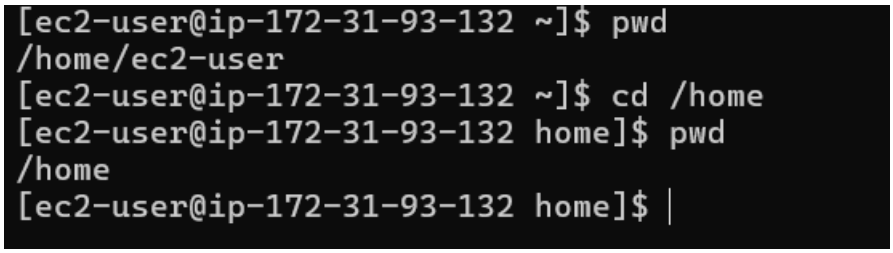
- Here you can see and add the Techie user all permission.
- Now install the git it will download.



The git is installed.

2) **Navigate to the home directory.**

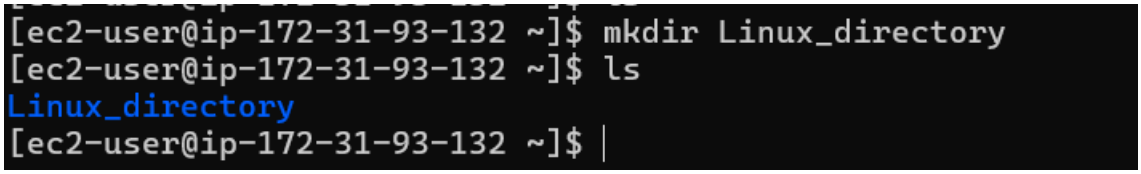
- First I am in ec2-user directory.
- To change the ec2-user directory to home directory use the below cmd.
- CMD ----- cd home ------- cd means change directory.
    - home is the directory



Now I am in home directory.

3**) Create a new directory.**

- To create new directory the CMD ---- mkdir directory name.



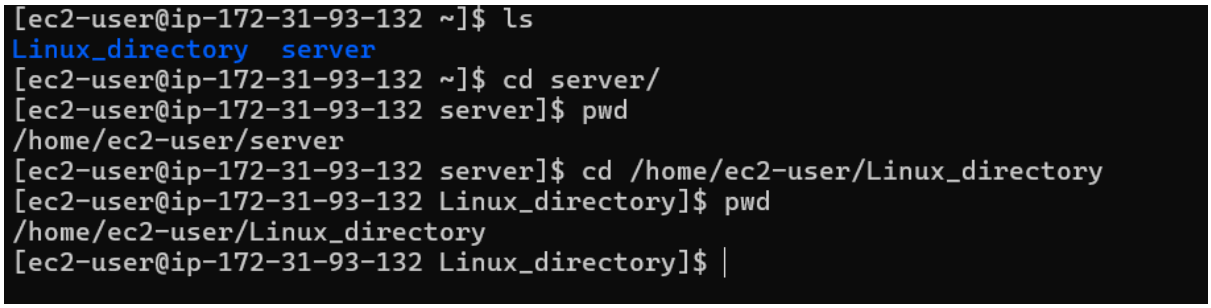
- The new directory will be crated.

4) List the contents of a directory.

First I went to directory then I am using ls cmd it will help us to List the contents of a directory.

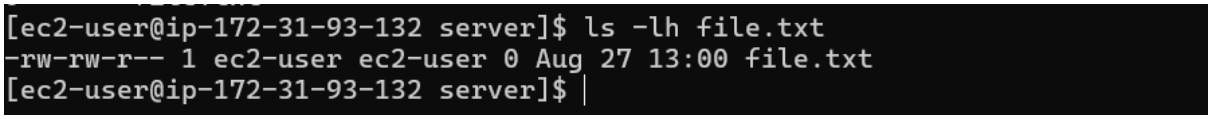
5) Change the current directory.

- I have two directories one is Linux_directory and second one is server .
- First I went to server directory then I am changed to Linux_directory.



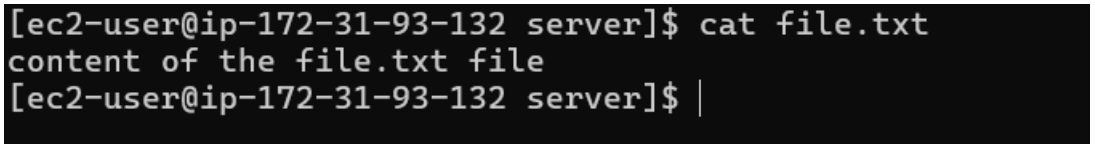
6) Create a new empty file.

- To create empty file we can use these command --- touch file name**.**
- With ls –lh command it will shows the all intermation of the file.
- Like file permissions, username, size of the file , date and time.



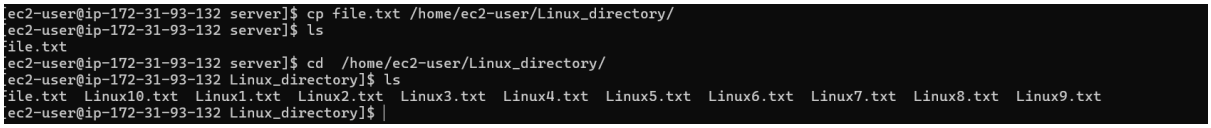
7) View the contents of a file.

To view the content of a file we can use CMD ----- cat file.txt



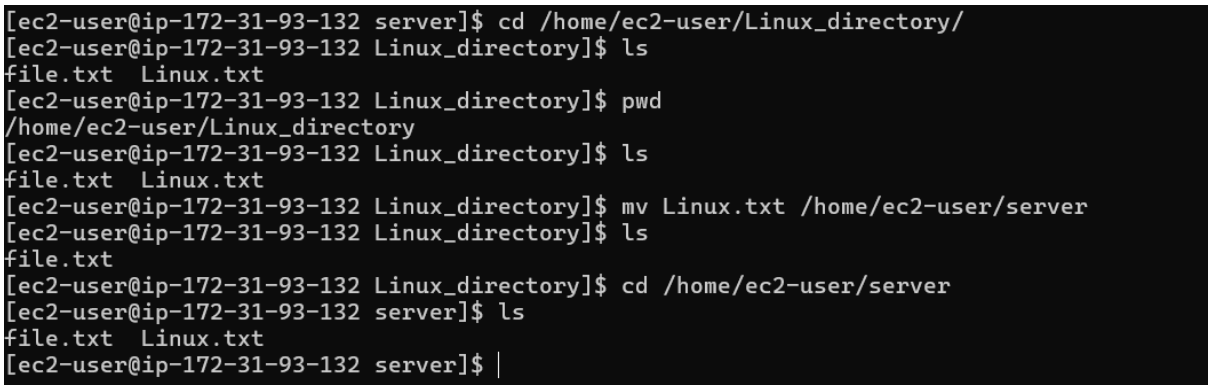
8) Copy a file to another location.

- We are in the same directory where the file has then no need to give current location otherwise give the location.
- CMD ------ cp file.txt /home/ec2-user/Linux_directory/
- cp --- copy the file one location to anethor location.
- /home/ec2-user/Linux_directory ----- these is the path of the where we want send.



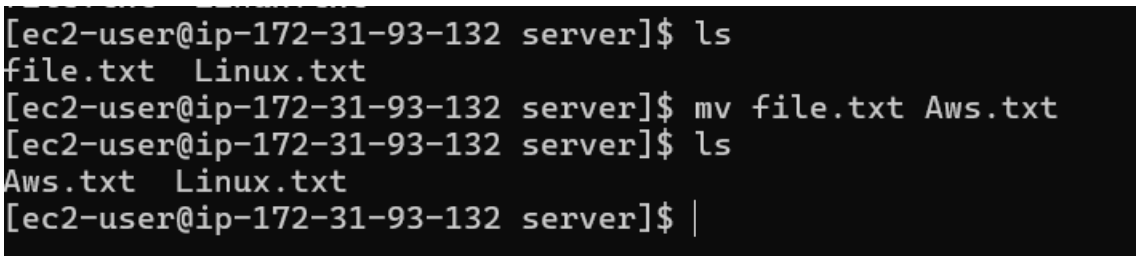
9) Move a file to another location.

- Here I am move a file one location to another location.
- Whenever we use mv cmd it will move to to one location to another location.
- It's means move the file in past directory no file .
- While using copy command it's will be available in two locations.



10) Rename a file.

- To rename a file also we can use mv command.



11) Delete a file.

- Empty file delete command -- rm filename

```
[ec2-user@ip-172-31-93-132 server]$ ls
Aws.txt  Linux.txt
[ec2-user@ip-172-31-93-132 server]$ rm Aws.txt
[ec2-user@ip-172-31-93-132 server]$ ls
Linux.txt
[ec2-user@ip-172-31-93-132 server]$
```

12) Grant or revoke permissions on a file or directory.

- To change the permissions we can use the CMD------ chmod 777 filename

- Chmod ------- change modification.
- file permissions are :

  read = 4, write = 2, execute = 1

  777 ------ these are file permission.

  -rwx-rwx-rwx 1 root root 0 Aug 27 12:35 Linux.txt

  1st ---7 --- User have all permission to accesses the file**.**

  2nd –7---- Group have all permission to accesses the file.

  3rd –7---- others have all permission to accesses the file

```
[ec2-user@ip-172-31-93-132 server]$ ls -ll
total 0
-rw-r--r-- 1 root root 0 Aug 27 12:35 Linux.txt
[ec2-user@ip-172-31-93-132 server]$ sudo chmod 777 Linux.txt
[ec2-user@ip-172-31-93-132 server]$ ls -ll
total 0
-rwxrwxrwx 1 root root 0 Aug 27 12:35 Linux.txt
[ec2-user@ip-172-31-93-132 server]$
```

13) View the current date and time.

- To view the current date and time CMD----- date

```
[ec2-user@ip-172-31-93-132 ~]$ date
Tue Aug 27 13:57:55 UTC 2024
[ec2-user@ip-172-31-93-132 ~]$
```

14) Check the system uptime.

This command shows how long the system has been running, along with the number of users currently logged in and the system load averages.

```
[ec2-user@ip-172-31-93-132 ~]$ uptime
 14:00:42 up  3:32,  7 users,  load average: 0.00, 0.00, 0.00
[ec2-user@ip-172-31-93-132 ~]$
```

15) View the running processes.

- To check the all running processes.

```
[ec2-user@ip-172-31-93-132 ~]$ ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 10:27 ?        00:00:00 /usr/lib/systemd/systemd --switched-root --system --deserialize 21
root         2     0  0 10:27 ?        00:00:00 [kthreadd]
root         3     2  0 10:27 ?        00:00:00 [rcu_gp]
root         4     2  0 10:27 ?        00:00:00 [rcu_par_gp]
root         6     2  0 10:27 ?        00:00:00 [kworker/0:0H-ev]
root         8     2  0 10:27 ?        00:00:00 [mm_percpu_wq]
root         9     2  0 10:27 ?        00:00:00 [rcu_tasks_rude_]
root        10     2  0 10:27 ?        00:00:00 [rcu_tasks_trace]
root        11     2  0 10:27 ?        00:00:00 [ksoftirqd/0]
root        12     2  0 10:27 ?        00:00:00 [rcu_sched]
root        13     2  0 10:27 ?        00:00:00 [migration/0]
root        15     2  0 10:27 ?        00:00:00 [cpuhp/0]
root        17     2  0 10:27 ?        00:00:00 [kdevtmpfs]
root        18     2  0 10:27 ?        00:00:00 [netns]
root        20     2  0 10:27 ?        00:00:00 [kauditd]
root       299     2  0 10:27 ?        00:00:00 [khungtaskd]
root       300     2  0 10:27 ?        00:00:00 [oom_reaper]
root       301     2  0 10:27 ?        00:00:00 [writeback]
root       303     2  0 10:27 ?        00:00:00 [kcompactd0]
root       304     2  0 10:27 ?        00:00:00 [ksmd]
root       305     2  0 10:27 ?        00:00:00 [khugepaged]
root       330  3228  0 12:32 ?        00:00:00 sshd: ec2-user [priv]
ec2-user   348   330  0 12:32 ?        00:00:00 sshd: ec2-user@pts/4
ec2-user   349   348  0 12:32 pts/4    00:00:00 -bash
root       361     2  0 10:27 ?        00:00:00 [kintegrityd]
root       363     2  0 10:27 ?        00:00:00 [kblockd]
root       364     2  0 10:27 ?        00:00:00 [blkcg_punt_bio]
root       401  3228  0 12:42 ?        00:00:00 sshd: ec2-user [priv]
ec2-user   419   401  0 12:42 ?        00:00:00 sshd: ec2-user@pts/5
ec2-user   470   419  0 12:42 pts/5    00:00:00 -bash
root       518  3228  0 13:00 ?        00:00:00 sshd: ec2-user [priv]
ec2-user   536   518  0 13:00 ?        00:00:00 sshd: ec2-user@pts/6
ec2-user   537   536  0 13:00 pts/6    00:00:00 -bash
root       717     2  0 10:27 ?        00:00:00 [xen-balloon]
root       723     2  0 10:27 ?        00:00:00 [tpm_dev_wq]
root       729     2  0 10:27 ?        00:00:00 [md]
root       732     2  0 10:27 ?        00:00:00 [edac-poller]
root       737     2  0 10:27 ?        00:00:00 [watchdogd]
postfix    763  3102  0 13:44 ?        00:00:00 pickup -l -t unix -u
```

16) Kill a running process.

- First I am installing the httpd
- To install CMD---- yum –y install httpd

```
[root@ip-172-31-93-132 ~]# yum -y install httpd
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.62-1.amzn2.0.2 will be installed
--> Processing Dependency: httpd-filesystem = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-tools = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.7.2-1.amzn2 will be installed
---> Package apr-util.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.3-1.amzn2.0.1 for package: apr-util-1.6.3-1.amzn2.0.1.x86_64
---> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
---> Package httpd-filesystem.noarch 0:2.4.62-1.amzn2.0.2 will be installed
---> Package httpd-tools.x86_64 0:2.4.62-1.amzn2.0.2 will be installed
---> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
---> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.2 will be installed
--> Running transaction check
---> Package apr-util-bdb.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================================================
 Package                   Arch            Version                    Repository                          Size
================================================================================================================
Installing:
 httpd                     x86_64          2.4.62-1.amzn2.0.2         amzn2-core                         1.4 M
Installing for dependencies:
 apr                       x86_64          1.7.2-1.amzn2              amzn2-core                         130 k
 apr-util                  x86_64          1.6.3-1.amzn2.0.1          amzn2-core                         101 k
 apr-util-bdb              x86_64          1.6.3-1.amzn2.0.1          amzn2-core                          22 k
 generic-logos-httpd       noarch          18.0.0-4.amzn2             amzn2-core                          19 k
 httpd-filesystem          noarch          2.4.62-1.amzn2.0.2         amzn2-core                          25 k
 httpd-tools               x86_64          2.4.62-1.amzn2.0.2         amzn2-core                          89 k
```

- Next start the httpd sevice CMD---- systemctl start httpd
- Check the status httpd service CMD ---- systemctl status httpd

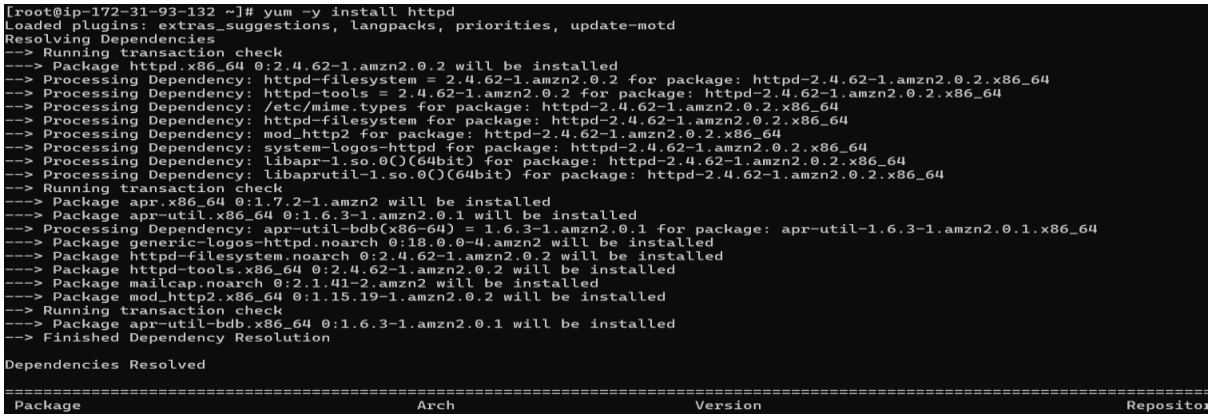
```
[root@ip-172-31-93-132 ~]# systemctl start httpd
[root@ip-172-31-93-132 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Tue 2024-08-27 14:18:48 UTC; 8s ago
     Docs: man:httpd.service(8)
 Main PID: 1055 (httpd)
   Status: "Processing requests..."
   CGroup: /system.slice/httpd.service
           ├─1055 /usr/sbin/httpd -DFOREGROUND
           ├─1056 /usr/sbin/httpd -DFOREGROUND
           ├─1057 /usr/sbin/httpd -DFOREGROUND
           ├─1058 /usr/sbin/httpd -DFOREGROUND
           ├─1059 /usr/sbin/httpd -DFOREGROUND
           └─1060 /usr/sbin/httpd -DFOREGROUND

Aug 27 14:18:48 ip-172-31-93-132.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Aug 27 14:18:48 ip-172-31-93-132.ec2.internal systemd[1]: Started The Apache HTTP Server.
```

- Check the processes is running or not. To check CMD----- ps -ef | grep httpd
- The process is running then you need to kill the process.
- CMD ---- kill -9 1055
- Kill ---- these will kill the process.
- -9 --- is the forcefully kill the process.
- 1055 --- is the process id of the httpd service.
- After you can check the httpd service is running or not. CMD---- systemctl status httpd.
-

```
[root@ip-172-31-93-132 ~]# ps -ef | grep httpd
root      1055     1  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1056  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1057  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1058  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1059  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1060  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
root      1105   931  0 14:19 pts/0    00:00:00 grep --color=auto httpd
[root@ip-172-31-93-132 ~]# ps -ef | grep httpd
root      1055     1  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1056  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1057  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1058  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1059  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    1060  1055  0 14:18 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
root      1157   931  0 14:19 pts/0    00:00:00 grep --color=auto httpd
[root@ip-172-31-93-132 ~]# kill -9 1055
[root@ip-172-31-93-132 ~]# ps -ef | grep httpd
root      1168   931  0 14:20 pts/0    00:00:00 grep --color=auto httpd
[root@ip-172-31-93-132 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: failed (Result: signal) since Tue 2024-08-27 14:20:11 UTC; 15s ago
     Docs: man:httpd.service(8)
  Process: 1055 ExecStart=/usr/sbin/httpd $OPTIONS -DFOREGROUND (code=killed, signal=KILL)
 Main PID: 1055 (code=killed, signal=KILL)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:   0 B/sec"

Aug 27 14:18:48 ip-172-31-93-132.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Aug 27 14:18:48 ip-172-31-93-132.ec2.internal systemd[1]: Started The Apache HTTP Server.
Aug 27 14:20:11 ip-172-31-93-132.ec2.internal systemd[1]: httpd.service: main process exited, code=killed, status=9/KILL
Aug 27 14:20:11 ip-172-31-93-132.ec2.internal systemd[1]: Unit httpd.service entered failed state.
Aug 27 14:20:11 ip-172-31-93-132.ec2.internal systemd[1]: httpd.service failed.
[root@ip-172-31-93-132 ~]#
```

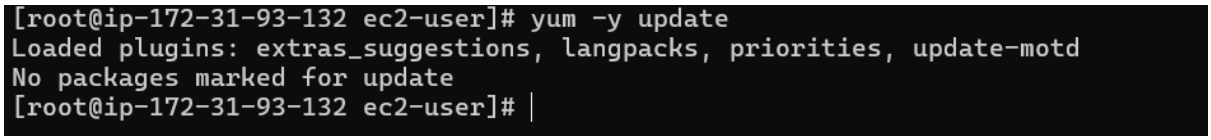
17) Install a package using the package manager (e.g., apt or yum).

- Using the yum package manager I am install the httpd server.



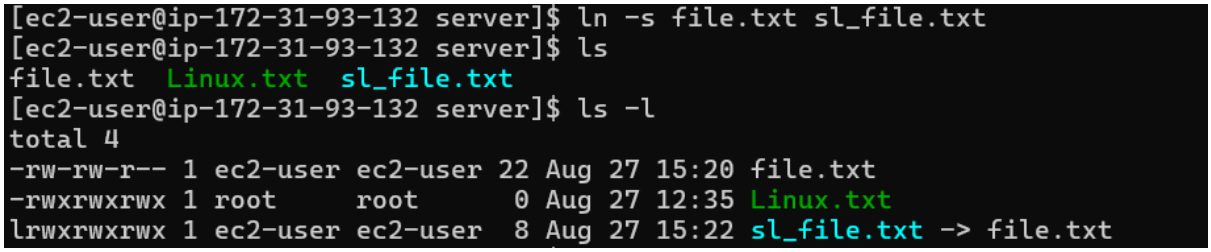
18) Update the system packages.

- To update the system packages CMD --- yum –y update
- My server already updated that's why these showing like.



19) Create a symbolic link.

- To create a symbolic link or soft link we have to use the below command
- CMD ----- ln –s file_name softlink_name
- **`ln`**: The command used to create links in Unix/Linux.
- **`-s`**: This option specifies that the link should be symbolic (as opposed to a hard link).
- **`file.txt`**: The target file that the symbolic link will point to.
- **`sl_file.txt`**: The name of the symbolic link being created.



20) Search for files using the find command.

- Instead of manually searching through folders, you can quickly find files by name.

```
[root@ip-172-31-93-132 /]# find / -name file.txt
/home/ec2-user/Linux_directory/file.txt
/home/ec2-user/server/file.txt
[root@ip-172-31-93-132 /]#
```

21) Compress and decompress files using tar.

Compress:

Her I am 3 directories are achieving Names are --- Mahesh, Ramesh, Suresh

The CMD ------ tar -cvf directories_archive.tar Mahesh Ramesh Suresh

These command will be created the tar file.

```
[ec2-user@ip-172-31-93-132 ~]$ ls
Linux_directory  server
[ec2-user@ip-172-31-93-132 ~]$ cd Linux_directory/
[ec2-user@ip-172-31-93-132 Linux_directory]$ ls
Mahesh  Ramesh  Suresh
[ec2-user@ip-172-31-93-132 Linux_directory]$ cd Ramesh/
[ec2-user@ip-172-31-93-132 Ramesh]$ ls
file.txt    linux14.txt  linux19.txt  linux23.txt  linux28.txt  linux32.txt  linux37.txt  linux41.txt  linux46.txt  linux50.txt  linux9.txt
linux10.txt  linux15.txt  linux1.txt   linux24.txt  linux29.txt  linux33.txt  linux38.txt  linux42.txt  linux47.txt  linux5.txt
linux11.txt  linux16.txt  linux20.txt  linux25.txt  linux2.txt   linux34.txt  linux39.txt  linux43.txt  linux48.txt  linux6.txt
linux12.txt  linux17.txt  linux21.txt  linux26.txt  linux30.txt  linux35.txt  linux3.txt   linux44.txt  linux49.txt  linux7.txt
linux13.txt  linux18.txt  linux22.txt  linux27.txt  linux31.txt  linux36.txt  linux40.txt  linux45.txt  linux4.txt   linux8.txt
[ec2-user@ip-172-31-93-132 Ramesh]$ cd ..
[ec2-user@ip-172-31-93-132 Linux_directory]$ tar -cvf directories_archive.tar Mahesh Ramesh Suresh
Mahesh/
Ramesh/
Ramesh/file.txt
Ramesh/linux10.txt
Ramesh/linux11.txt
Ramesh/linux12.txt
Ramesh/linux13.txt
Ramesh/linux14.txt
Ramesh/linux15.txt
Ramesh/linux16.txt
Ramesh/linux17.txt
Ramesh/linux18.txt
Ramesh/linux19.txt
Ramesh/linux1.txt
Ramesh/linux20.txt
Ramesh/linux21.txt
Ramesh/linux22.txt
Ramesh/linux23.txt
```

Decompress:

- It means extract the file.
- We want extract the another location.
- I will move to present location to another location the CMD ----- mv directories_archive.tar /home/ec2-user/server/
- Next I went to these location using these CMD---- cd /home/ec2-user/server/
- Ther I am extracting the tar file Cmd ----- tar -xvf directories_archive.tar
- `-x`: Extracts files from the archive.
- `-v`: Verbose mode; shows the progress of extraction by listing the files being extracted.
- `-f`: Specifies the filename of the archive to work with.

```
[ec2-user@ip-172-31-93-132 Linux_directory]$ ls
directories_archive.tar  Mahesh  Ramesh  Suresh
[ec2-user@ip-172-31-93-132 Linux_directory]$ mv directories_archive.tar  /home/ec2-user/server/
[ec2-user@ip-172-31-93-132 Linux_directory]$ cd ..
[ec2-user@ip-172-31-93-132 ~]$ ls
Linux_directory  server
[ec2-user@ip-172-31-93-132 ~]$ cd server/
[ec2-user@ip-172-31-93-132 server]$ ls
directories_archive.tar  file.txt  Linux.txt  sl_file.txt
[ec2-user@ip-172-31-93-132 server]$ tar -xvf directories_archive.tar
Mahesh/
Ramesh/
Ramesh/file.txt
Ramesh/linux10.txt
Ramesh/linux11.txt
Ramesh/linux12.txt
Ramesh/linux13.txt
Ramesh/linux14.txt
Ramesh/linux15.txt
Ramesh/linux16.txt
Ramesh/linux17.txt
Ramesh/linux18.txt
Ramesh/linux19.txt
Ramesh/linux1.txt
Ramesh/linux20.txt
Ramesh/linux21.txt
Ramesh/linux22.txt
Ramesh/linux23.txt
Ramesh/linux24.txt
Ramesh/linux25.txt
Ramesh/linux26.txt
Ramesh/linux27.txt
Ramesh/linux28.txt
Ramesh/linux29.txt
```

22) Monitor system resources with top or htop.

- To monitor the system resources we can use the top and htop commands.
- **top :** Displays real-time system information like CPU, memory usage, and running processes.
- **htop:** An enhanced version of `top` with a colorful and interactive interface.

23) Create and manage user groups.

- **Here first I am checking the groups.**
- **To use CMD --- tail -n 5 /etc/group --- these command shows the last five lines in the group.**
- **After I am created one Devops group with command :---- groupadd Devops**
- **You will show in the below the image.**

```
[root@ip-172-31-93-132 ~]#  tail -n 5 /etc/group
tcpdump:x:72:
ec2-user:x:1000:
Techie:x:1001:
apache:x:48:
Ramesh:x:1002:
[root@ip-172-31-93-132 ~]# groupadd Devops
[root@ip-172-31-93-132 ~]# tail -n 5 /etc/group
ec2-user:x:1000:
Techie:x:1001:
apache:x:48:
Ramesh:x:1002:
Devops:x:1003:
```

Add a User to a Group

- **Command**: `sudo usermod -aG [groupname] [username]`
- **Description**: Adds a user to the specified group.

```
[root@ip-172-31-93-132 ~]# usermod -aG Devops Ramesh
[root@ip-172-31-93-132 ~]# tail -n 5 /etc/group
ec2-user:x:1000:
Techie:x:1001:
apache:x:48:
Ramesh:x:1002:
Devops:x:1003:Ramesh
```

Remove a User from a Group

- **Command**: `sudo deluser [username] [groupname]` or `sudo gpasswd -d [username] [groupname]`
- **Description**: Removes a user from the specified group.

Delete a Group

- **Command**: `sudo groupdel [groupname]`
- **Description**: Deletes the specified group.

```
[root@ip-172-31-93-132 ~]# gpasswd -d Ramesh Devops
Removing user Ramesh from group Devops
[root@ip-172-31-93-132 ~]# groupdel Devops
[root@ip-172-31-93-132 ~]# tail -n 5 /etc/group
tcpdump:x:72:
ec2-user:x:1000:
Techie:x:1001:
apache:x:48:
Ramesh:x:1002:
```

24) Set up SSH password less authentication.

25) Monitor log files using tail or grep.

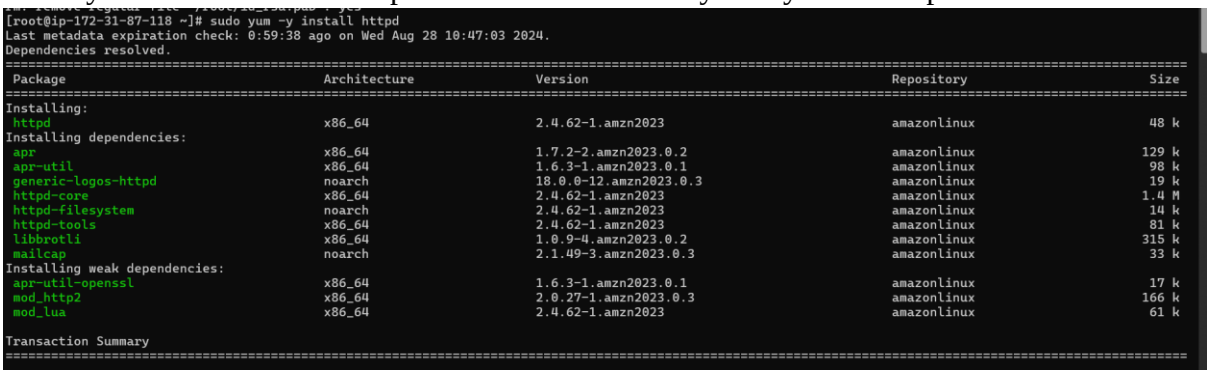
Monitor a Log File in Real-Time Use the `-f` option to follow the log file and see new entries as they are added.

CMD -- tail -f /var/log/yum.log

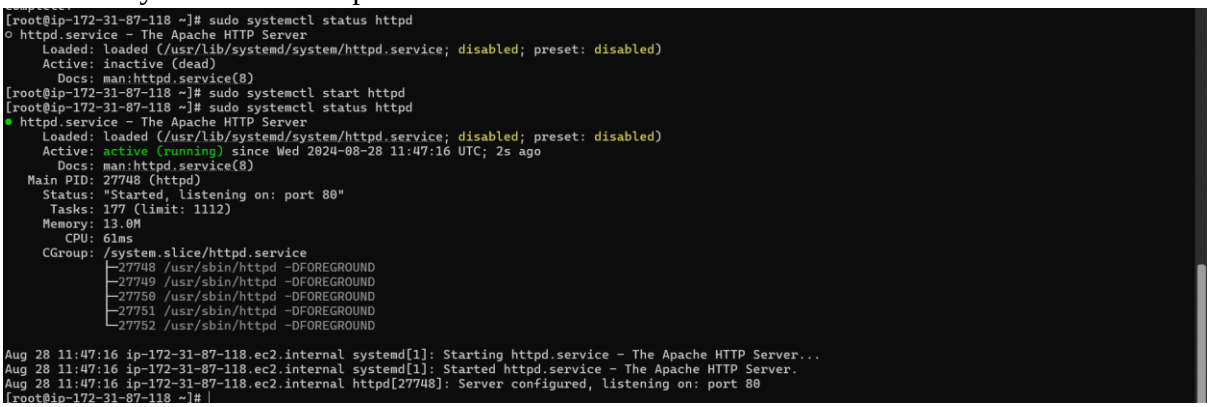
```
[root@ip-172-31-93-132 /]# tail -f /var/log/yum.log
Aug 27 14:18:20 Installed: apr-1.7.2-1.amzn2.x86_64
Aug 27 14:18:20 Installed: apr-util-1.6.3-1.amzn2.0.1.x86_64
Aug 27 14:18:20 Installed: apr-util-bdb-1.6.3-1.amzn2.0.1.x86_64
Aug 27 14:18:20 Installed: httpd-tools-2.4.62-1.amzn2.0.2.x86_64
Aug 27 14:18:20 Installed: httpd-filesystem-2.4.62-1.amzn2.0.2.noarch
Aug 27 14:18:20 Installed: generic-logos-httpd-18.0.0-4.amzn2.noarch
Aug 27 14:18:20 Installed: mailcap-2.1.41-2.amzn2.noarch
Aug 27 14:18:20 Installed: mod_http2-1.15.19-1.amzn2.0.2.x86_64
Aug 27 14:18:21 Installed: httpd-2.4.62-1.amzn2.0.2.x86_64
Aug 28 06:14:33 Installed: htop-2.0.2-1.amzn2.0.2.x86_64
```

26) Set up a web server (e.g., Apache or Nginx).

- To setup a web server we need to launch the ec2 instance.
- There you need to install the apache server CMD ---- yum -y install httpd



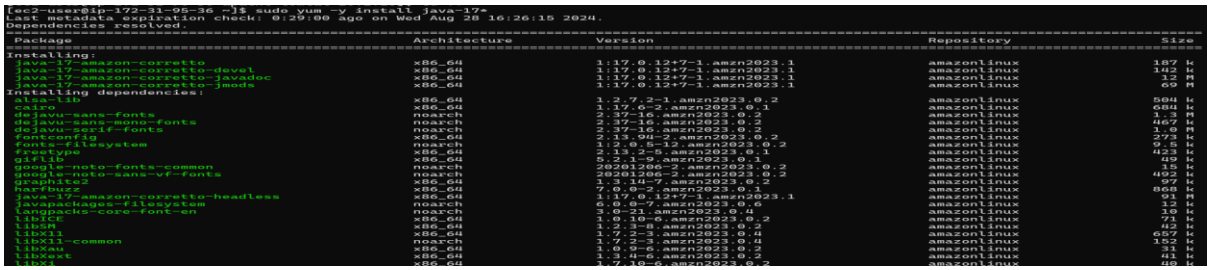
- You need to check the status of httpd Cmd ---- systemctl status httpd
- Its active running no need to worry but it's dead you need to start
- CMD---- systemctl start httpd



27) Configure and secure a MySQL Database.

28) Set up a Application Server (e.g.,Apache Tomcat)

- To setup a apache tomcat server we need to Ec2-server.
- In ec2 server we have install the java and apache tomcat.
- To install java CMD---- sudo yum -y install java-17* ----it will download the java -17 version



- CMD---- wget https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz
- Wget to download the links from the browser.
- After you need to extract the tar.gz file

```
[ec2-user@ip-172-31-95-36 ~]$ wget https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz
--2024-08-28 16:46:05--  https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13056103 (12M) [application/x-gzip]
Saving to: 'apache-tomcat-10.1.28.tar.gz'

apache-tomcat-10.1.28.tar.gz        100%[=================================================================>]  12.45M  --.-KB/s    in 0.08s

2024-08-28 16:46:05 (147 MB/s) - 'apache-tomcat-10.1.28.tar.gz' saved [13056103/13056103]

[ec2-user@ip-172-31-95-36 ~]$ ls
apache-tomcat-10.1.28.tar.gz
[ec2-user@ip-172-31-95-36 ~]$ tar -xvf apache-tomcat-10.1.28.tar.gz
apache-tomcat-10.1.28/conf/
apache-tomcat-10.1.28/conf/catalina.policy
apache-tomcat-10.1.28/conf/catalina.properties
apache-tomcat-10.1.28/conf/context.xml
apache-tomcat-10.1.28/conf/jaspic-providers.xml
apache-tomcat-10.1.28/conf/jaspic-providers.xsd
apache-tomcat-10.1.28/conf/logging.properties
apache-tomcat-10.1.28/conf/server.xml
apache-tomcat-10.1.28/conf/tomcat-users.xml
apache-tomcat-10.1.28/conf/tomcat-users.xsd
apache-tomcat-10.1.28/conf/web.xml
apache-tomcat-10.1.28/bin/
apache-tomcat-10.1.28/lib/
apache-tomcat-10.1.28/logs/
apache-tomcat-10.1.28/temp/
apache-tomcat-10.1.28/webapps/
apache-tomcat-10.1.28/webapps/ROOT/
apache-tomcat-10.1.28/webapps/ROOT/WEB-INF/
apache-tomcat-10.1.28/webapps/docs/
apache-tomcat-10.1.28/webapps/docs/META-INF/
apache-tomcat-10.1.28/webapps/docs/WEB-INF/
apache-tomcat-10.1.28/webapps/docs/WEB-INF/jsp/
apache-tomcat-10.1.28/webapps/docs/annotationapi/
```

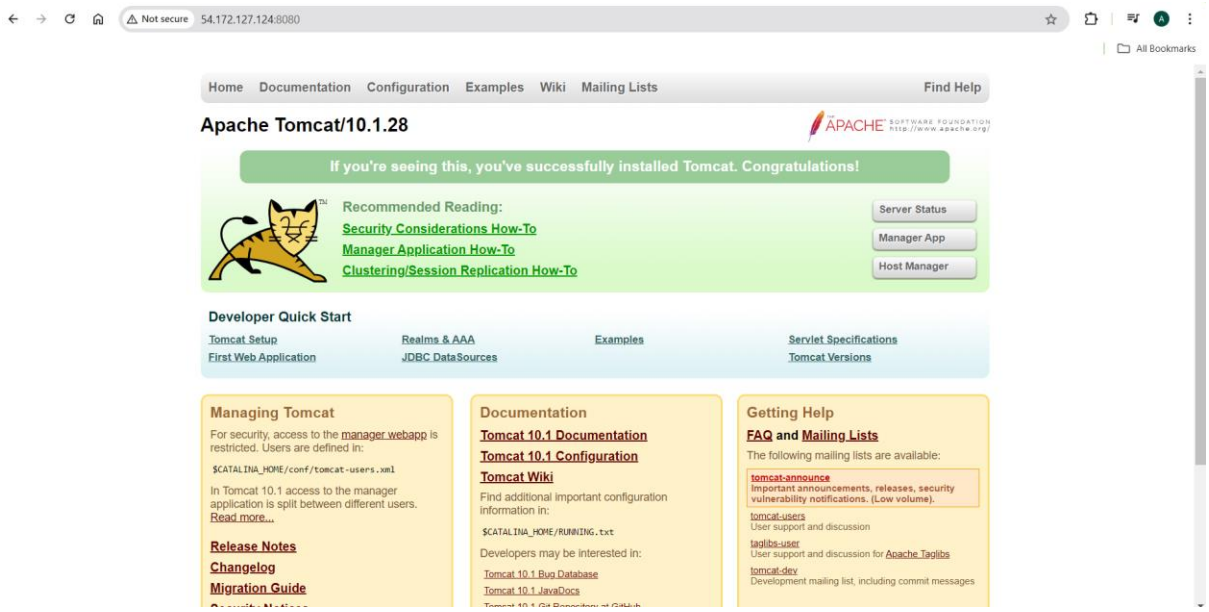
- Next you have to go the file after you have to go bin dir.

Steps :

- CMD ---- cd bin
- There you have to start the tomcat server CMD ---- ./startup.sh
- To check the port numbers of the apachie tomcat CMD ----- sudo netstat -ntpl

```
[ec2-user@ip-172-31-95-36 ~]$ ls
apache-tomcat-10.1.28  apache-tomcat-10.1.28.tar.gz
[ec2-user@ip-172-31-95-36 ~]$ cd apache-tomcat-10.1.28/
[ec2-user@ip-172-31-95-36 apache-tomcat-10.1.28]$ ls
BUILDING.txt  CONTRIBUTING.md  LICENSE  NOTICE  README.md  RELEASE-NOTES  RUNNING.txt  bin  conf  lib  logs  temp  webapps  work
[ec2-user@ip-172-31-95-36 apache-tomcat-10.1.28]$ cd bin/
[ec2-user@ip-172-31-95-36 bin]$ ls
bootstrap.jar       ciphers.sh                commons-daemon-native.tar.gz  daemon.sh     migrate.bat       shutdown.sh          tool-wrapper.bat
catalina-tasks.xml  commons-daemon-native.tar.gz  digest.bat                migrate.sh    startup.bat       tool-wrapper.sh
catalina.bat        commons-daemon.jar        digest.sh     setclasspath.bat  startup.sh        version.bat
catalina.sh         configtest.bat            makebase.bat  setclasspath.sh   tomcat-juli.jar   version.sh
ciphers.bat         configtest.sh             makebase.sh   shutdown.bat      tomcat-native.tar.gz
[ec2-user@ip-172-31-95-36 bin]$ ./startup.sh
Using CATALINA_BASE:   /home/ec2-user/apache-tomcat-10.1.28
Using CATALINA_HOME:   /home/ec2-user/apache-tomcat-10.1.28
Using CATALINA_TMPDIR: /home/ec2-user/apache-tomcat-10.1.28/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /home/ec2-user/apache-tomcat-10.1.28/bin/bootstrap.jar:/home/ec2-user/apache-tomcat-10.1.28/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[ec2-user@ip-172-31-95-36 bin]$ sudo netstat -ntpl
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      2171/sshd: /usr/sbi
tcp6       0      0 :::22                   :::*                    LISTEN      2171/sshd: /usr/sbi
tcp6       0      0 :::8080                 :::*                    LISTEN      28205/java
tcp6       0      0 127.0.0.1:8005          :::*                    LISTEN      28205/java
[ec2-user@ip-172-31-95-36 bin]$
```

- The below we can see the apache tomcat service.



29) create a service file for Apache Tomcat.(Should execute by using systemtctl command)

The above process we have to do now we want create the one file and

30) Print specific columns from a delimited file.

- Heate first iam creating the file with some columns ex: iD, name, age, email, mobile number. Like that
- To create file I am using these CMD--- echo -e "Name,Age,Location,Score\nJohn,25,New York,85\nJane,30,Los Angeles,90\nDoe,22,Chicago,88" > data.txt
- To Print specific columns from a delimited file I am using awk cmd.
- Cmd -----awk -F',' '{print $2, $4}' data.txt
- The below see the out of the cmd.

```
[ec2-user@ip-172-31-95-36 ~]$ cat data.txt
[ec2-user@ip-172-31-95-36 ~]$ echo -e "Name,Age,Location,Score\nJohn,25,New York,85\nJane,30,Los Angeles,90\nDoe,22,Chicago,88" > data.txt
[ec2-user@ip-172-31-95-36 ~]$ cat data.txt
Name,Age,Location,Score
John,25,New York,85
Jane,30,Los Angeles,90
Doe,22,Chicago,88
[ec2-user@ip-172-31-95-36 ~]$ cut -d',' -f2,4 data.txt
Age,Score
25,85
30,90
22,88
[ec2-user@ip-172-31-95-36 ~]$ ^C
[ec2-user@ip-172-31-95-36 ~]$ cut -d',' -f2,4 data.txt
Age,Score
25,85
30,90
22,88
[ec2-user@ip-172-31-95-36 ~]$ awk -F',' '{print $2, $4}' data.txt
Age Score
25 85
30 90
22 88
[ec2-user@ip-172-31-95-36 ~]$
```

31) Filter and print lines based on a specific pattern or condition.

CMD------ grep 'John' data.txt

What It Does:

- This command searches for the exact pattern "John" in the `data.txt` file. The search is case-sensitive, meaning it will only match "John" and not variations like "john", "JOHN", or "JoHn".

CMD------- grep -v 'John' data.txt

Command Breakdown:

- **`grep`**: The command used to search for patterns in a file.
- **`-v`**: This option inverts the match, meaning it will select lines that do **not** match the given pattern.
- **`'John'`**: The pattern to search for.
- **`data.txt`**: The file in which to search for the pattern.

```
[ec2-user@ip-172-31-95-36 ~]$ grep 'John' data.txt
John,25,New York,85
[ec2-user@ip-172-31-95-36 ~]$ grep -v 'John' data.txt
Name,Age,Location,Score
Jane,30,Los Angeles,90
Doe,22,Chicago,88
[ec2-user@ip-172-31-95-36 ~]$ cat data.txt
Name,Age,Location,Score
John,25,New York,85
Jane,30,Los Angeles,90
Doe,22,Chicago,88
```

32) Calculate and print the average, sum, or other statistics of a column.

- To calculate the sum of the "Score" column CMD------ `awk -F',' '{sum += $4} END {print "Sum:", sum}' data.txt`
- **`-F','`**: Sets the field separator to a comma.
- **`sum += $4`**: Adds the value in the 4th column (`$4`) to the variable `sum`.
- **`END {print "Sum:", sum}`**: After processing all lines, it prints the total sum.

```
[ec2-user@ip-172-31-95-36 ~]$ awk -F',' '{sum += $4} END {print "Sum:", sum}' data.txt
Sum: 263
[ec2-user@ip-172-31-95-36 ~]$ awk -F',' '{sum += $4; count++} END {if (count > 0) print "Average:", sum/
count}' data.txt
Average: 65.75
```

33) Perform string manipulation, such as extracting substrings or changing case.

- **Extracting Substrings**
- *Example: Extract the first three characters of each line*
- Using `cut`:
- `cut -c 1-3 data.txt`

- **`-c 1-3`**: Extracts characters from position 1 to 3 from each line in `data.txt`.

```
[ec2-user@ip-172-31-95-36 ~]$ cut -c 1-2 data.txt
Na
Jo
Ja
Do
[ec2-user@ip-172-31-95-36 ~]$
```

- **Changing Case**
- *Example: Convert all text to uppercase*
- Using `tr`:
- `cat data.txt | tr '[:lower:]' '[:upper:]'`

- **`tr '[:lower:]' '[:upper:]'`**: Translates all lowercase letters to uppercase.

```
[ec2-user@ip-172-31-95-36 ~]$ cat data.txt
Name,Age,Location,Score
John,25,New York,85
Jane,30,Los Angeles,90
Doe,22,Chicago,88
[ec2-user@ip-172-31-95-36 ~]$ cat data.txt | tr '[:lower:]' '[:upper:]'
NAME,AGE,LOCATION,SCORE
JOHN,25,NEW YORK,85
JANE,30,LOS ANGELES,90
DOE,22,CHICAGO,88
[ec2-user@ip-172-31-95-36 ~]$
```

34) Count the occurrences of a specific pattern in a file.

Using `sed`

- `sed` can also be used, but it's more complex. This is an alternative approach using `sed` with pattern space:
- *Example: Count occurrences of "John" in a file*
- `sed -n 's/John/&/gp' data.txt | wc -l`
- **`-n`**: Suppresses automatic printing of pattern space.
- **`s/John/&/gp`**: Substitutes "John" with itself and prints the matching lines.
- **`wc -l`**: Counts the number of lines printed by `sed`.

35) Sort lines based on a specific field or column.

```
[ec2-user@ip-172-31-95-36 ~]$ cat data.txt
Name,Age,Location,Score
John,25,New York,85
Jane,30,Los Angeles,90
Doe,22,Chicago,88
[ec2-user@ip-172-31-95-36 ~]$ sort -t',' -k2,2n data.txt
Name,Age,Location,Score
Doe,22,Chicago,88
John,25,New York,85
Jane,30,Los Angeles,90
[ec2-user@ip-172-31-95-36 ~]$
```

36) Merge multiple files based on a common field or column.

- Fist I am create two file with different file names.
- Later iam attach the two file with the below cmd---- join -t',' -1 1 -2 1 file1.txt file2.txt
- `-t','`: Specifies the delimiter as a comma.
- `-1 1`: Specifies that the join field in the first file is the first column.
- `-2 1`: Specifies that the join field in the second file is the first column.

```
[ec2-user@ip-172-31-95-36 ~]$ cat file1.txt
1,suresh,25
2,Ramesh,30
3,sai,22

[ec2-user@ip-172-31-95-36 ~]$ cat file2.txt
1,85
2,90
3,88


[ec2-user@ip-172-31-95-36 ~]$ join -t',' -1 1 -2 1 file1.txt file2.txt
1,suresh,25,85
2,Ramesh,30,90
3,sai,22,88


[ec2-user@ip-172-31-95-36 ~]$
```

37) Substitute text in a file using search and replace.

Using `sed`

- The `sed` (stream editor) command is commonly used for text substitution.

- *Basic Syntax*
- `sh`
- `Copy code`
- `sed 's/search_pattern/replacement_text/' file.txt`
- **`s/search_pattern/replacement_text/`**: Substitutes `search_pattern` with `replacement_text`.
- By default, `sed` prints the modified lines to the standard output.

These is my actual file now.

```
[ec2-user@ip-172-31-95-36 ~]$ sudo vi ramesh
[ec2-user@ip-172-31-95-36 ~]$ cat ramesh
Ramesh is John is a software engineer.
John lives in New York.
he joined at pg
```

These is the after the enter the command my the content will be change.

```
[ec2-user@ip-172-31-95-36 ~]$ sed 's/John/Jonathan/' ramesh
Ramesh is Jonathan is a software engineer.
Jonathan lives in New York.
he joined at pg

[ec2-user@ip-172-31-95-36 ~]$
```

38) Delete specific lines based on a pattern or line number.

Using `awk`

- *Delete Lines Based on a Pattern*
- To delete lines that contain a specific pattern:
- `awk '!/pattern/' file.txt`
- **`!/pattern/`**: Prints lines that do not match the `pattern`.

```
[ec2-user@ip-172-31-95-36 ~]$ cat file1.txt
1,suresh,25
2,Ramesh,30
3,sai,22

[ec2-user@ip-172-31-95-36 ~]$ awk '!/suresh/' file1.txt
2,Ramesh,30
3,sai,22
```

39) Append or insert text before or after a specific pattern or line.

Using `sed`

Insert Text Before a Specific Pattern

To insert text before a line containing a specific pattern:

```
sh
Copy code
sed '/pattern/i\text_to_insert' file.txt
```

- **`/pattern/`**: Matches lines containing the `pattern`.
- **`i\`**: Inserts text before the matched line.

```
[ec2-user@ip-172-31-95-36 ~]$ cat ramesh
Ramesh is John is a software engineer.
John lives in New York.
he joined at pg

[ec2-user@ip-172-31-95-36 ~]$ sed '/John/i\New Line is added' ramesh
New Line is added
Ramesh is John is a software engineer.
New Line is added
John lives in New York.
he joined at pg

[ec2-user@ip-172-31-95-36 ~]$
```

40) Print only specific lines from a file.

- Using se commnd I am print only specific lines from a file.
- CMD ---- sed -n '3p' Ramesh
- : Suppresses automatic printing of lines.
- `Np`: Prints line number `N`.

```
[ec2-user@ip-172-31-95-36 ~]$ cat ramesh
Ramesh is John is a software engineer.
John lives in New York.
he joined at pg

[ec2-user@ip-172-31-95-36 ~]$ sed -n '3p' ramesh
he joined at pg
[ec2-user@ip-172-31-95-36 ~]$
```

41) Delete leading or trailing whitespace from lines.

- *Remove Leading Whitespace*
- To remove leading whitespace:
- sh
- Copy code
- `awk '{$1=$1; print}' file.txt`
- **`{$1=$1; print}`**: Reassigns the first field to itself, which effectively trims leading whitespace.

```
[ec2-user@ip-172-31-95-36 ~]$ cat ramesh
    Ramesh is John is a software engineer.
John lives in New York.
         he joined at pg

[ec2-user@ip-172-31-95-36 ~]$ awk '{$1=$1; print}' ramesh
Ramesh is John is a software engineer.
John lives in New York.
he joined at pg

[ec2-user@ip-172-31-95-36 ~]$
```

42) Edit files in-place, making changes directly to the file.

Using `perl`

- Perl can also edit files in place with the `-i` option.
- *Syntax*
- sh
- Copy code
- `perl -i -pe 's/pattern/replacement/' file.txt`
- **`-i`**: Edits the file in place.
- **`-pe`**: Processes each line of the file (loop over lines, print each after applying the code).

```
[ec2-user@ip-172-31-95-36 ~]$ cat ramesh
    Ramesh is John is a software engineer.
John lives in New York.
         he joined at pg

[ec2-user@ip-172-31-95-36 ~]$ perl -i -pe 's/Ramesh is John is a software engineer/suresh is a software/
' ramesh
[ec2-user@ip-172-31-95-36 ~]$ cat ramesh
    suresh is a software.
John lives in New York.
         he joined at pg

[ec2-user@ip-172-31-95-36 ~]$
```

43) Join multiple lines into a single line or split a line into multiple lines.

First in my file 3 lines of matter is there after entering the we seen in single line.

- **`awk`**: The command-line utility for pattern scanning and processing.
- **`{printf "%s ", $0}`**: For each line of input:

- `printf "%s "`: Prints the current line (`$0`) followed by a space.
- This ensures that each line is concatenated with a space, effectively joining them into a single line.
- **`file.txt`**: The input file containing multiple lines.

```
[ec2-user@ip-172-31-95-36 ~]$ cat ramesh
    suresh is a software.
John lives in New York.
        he joined at pg

[ec2-user@ip-172-31-95-36 ~]$ awk '{printf "%s ", $0}' ramesh
    suresh is a software. John lives in New York.          he joined at pg  [ec2-user@ip-172-31-95-36 ~]$
```

44) Copy file from Linux to windows machine

CMD ---- scp -i .\Linux.pem ec2-user@54.172.127.124:/home/ec2-user/ramesh C:\Users\ramee\Downloads\

HERE WE HAVE copy file from Linux to windows machine. Through ec2 machine we can't do it will shows below error.

```
[ec2-user@ip-172-31-95-36 ~]$ scp -i .\Linux.pem ec2-user@54.172.127.124:/home/ec2-user/ramesh C:\Users\ramee\downloads
Warning: Identity file .Linux.pem not accessible: No such file or directory.
ssh: Could not resolve hostname c: Name or service not known
Connection closed
[ec2-user@ip-172-31-95-36 ~]$
```

Enter in local terminal it will done.

```
PS C:\Users\ramee\downloads> scp -i .\Linux.pem ec2-user@54.172.127.124:/home/ec2-user/ramesh C:\Users\ramee\Downloads\
ramesh                                                                                        100%   76     0.3KB/s   00:00
PS C:\Users\ramee\downloads>
```

```
PS C:\Users\ramee\downloads> ls .\ramesh


    Directory: C:\Users\ramee\downloads


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        29-08-2024     15:08             76 ramesh


PS C:\Users\ramee\downloads>
```

45) 5 use cases for AWK and 5 use cases for sed

Text Extraction by Column:

- **Use Case:** Extract specific columns from a file or output.
- **Example:** To print the first and third columns of a file (`file.txt`

- This is the action that `awk` will perform on each line of the file.
- **`print`**: This tells `awk` to output the specified fields.
- **`$1`**: This refers to the first field (or column) in the current line. Fields are separated by whitespace by default.
- **`$3`**: This refers to the third field in the current line.
- **`$1, $3`**: By specifying `$1` and `$3`, you're telling `awk` to print the first and third fields, separated by a space.

```
[ec2-user@ip-172-31-95-36 ~]$ cat ramesh
    suresh is a software.
John lives in New York.
         he joined at pg

[ec2-user@ip-172-31-95-36 ~]$ awk '{print $1, $3}' ramesh
suresh a
John in
he at

[ec2-user@ip-172-31-95-36 ~]$
```

Conditional Text Processing:

- **Use Case:** Perform actions based on specific conditions.
- **Example:** Print lines where the value in the second column is greater than 100