Linux commands:

To check the logged-in user is root or normal use:

echo $EUID → gitve zero as output if user is root user otherwise normal user id

**Remove commands:**

**rm -r** /home/nraghu/naga → this will delete the naga folder (if it’s not worked use -f)

or **rm -fr** /*home/*nraghu/naga → this will forcefully delete the naga folder

**rm -r** /*home*/nraghu/naga/raghu.txt → this will delete the raghu.txt

cat /*etc*/shells → give what and all shell will support in your system

which bash → give where the executable file is available in system

**echo command:**

**\* echo "Raghu M N" >> Nagendra.txt → >> will append the content(Raghu M N) into Nagendra.txt**

**\* echo "Ravi M N" > Nagendra.txt → > will overwrite the content of Nagendra.txt**

**listing the files commands:**

**ls** → list the file and directories except hidden files

**ls -l**  → list files and directories with owner and permission details

**ls -a** -->list the file and directories with hidden files

**ls -al or ll** → list all files and directories with hidden files and their permissions

-rwxr-xr-x 1 nraghu nraghu 53480440 Dec 19 10:59 beam.jit

**ls -lh** → give list of files with time

-rwxr-xr-x 1 nraghu nraghu 52M Dec 19 10:59 beam.jit

**Copy commands:**

**cp file\_name folder\_name** → copy the file into folder

**cp -r folder\_a/folder\_b folder\_dest** → copy the folder\_b into insde of folder\_dest directory

if folder\_a contains the 10 folder we can copy at a time and keep it all copied file inside of folder\_dest:

**cp -r folder\_a/\* folder\_dest**

**Grep commands:**

**grep -rin word path\_directory** → it will search the word in entire given directory(in every files)

**ls | grep -i word** -→ignore the case sensitive and search for that word in current ditectory

**Sed command:**

**:%s/old\_string/new\_string/g** → it will replace the all old\_string with new\_string in opened file

ex: :%s/raghu/ravi/g → wherever raghu word is there, get replaced with ravi

**:s/old\_string/new\_string/g** → it will replace the first appear of old\_string with new\_string in opened file

**Chown and Chmod commands:**

**Grep commands:**

**grep -rin word path\_directory** → it will search the word in entire given directory(in every files)

**ls | grep -i word** -→ignore the case sensitive and search for that word in current ditectory

**awk command:**

The awk command is used for text processing in Linux. Although, the sed command is also used for text processing, but it has some limitations, so the awk command becomes a handy option for text processing. It provides powerful control to the data.

The options can be:

**\* -f program files:**It reads the source code of the script written on the awk command

**\* -F fs:** It is used as the input field separator.

**List students with the specified pattern.**

**awk ' /CS/{print}' student.txt** → in student.txt scan for the student name container the CS and it his name

If we specify the column number on this command, it will print that line only

**awk ‘{print $1,$5}’student.txt**

**awk -F"<test" '{print $2}' student.txt**  -→in student.txt search for the <test and print next part starting after <test

ex: <testsuites> → print the suites>

**set -e and set +e command in shell script:**

The set -e command is used to enable the "exit on error" behavior, which causes the script to terminate immediately if any command returns a non-zero exit status (indicating an error). This is the default behavior for shell scripts.

The set +e command is used to disable the "exit on error" behavior, allowing the script to continue executing even if a command returns a non-zero exit status.

The set -e is very impatant to use in jenkins jobs.

**cat command:**

The cat(concatenate) command is mainly used to view the content of the file.

The cat command also used for folowing cases:

\*Display the text file or any file(yaml file) content → **cat raghu.txt**

\*Read text file → **Name=`cat raghu.txt | grep -i name`**

\*create the new text file → **cat > raghu.txt**

\*file concatenation → **cat score.txt name.txt > report.txt**

To view file: cat -n /*etc*/passwd → show content with line number

Create file: cat > Nagendra.txt

To view big file: more filename and less filename

combine two or more file: cat score.txt name.txt > report.txt

**head command:** head command is used read the first 10 line content by default, if you don’t pass the number

head Names.txt → show only first 10 lines

head -15 Names.txt --> show first 15 lines

head -1 Names.txt → only first line

**tail command:** opposite of head, read the last ten files of file by default

tail Names.txt → show only last 10 lines

tail -3 Names.txt → show only last 3 lines

tail -1 Names.txt → last line

**ps command:**

ps → give the processes running by current user in that terminal and terminal (bash), by default give bash and ps → bash is opned terminal,ps is commad running in that terminal

start any container in that terminal then do ps,you will get below output

PID TTY TIME CMD

1718962 pts/16 00:00:00 bash → terminal

1755239 pts/16 00:00:00 docker --→ process

1755957 pts/16 00:00:00 ps → processor