**LINUX COMMANDS**

**Sudo -i -----** To go from ec2 user to root user

**SYSTEM COMMANDS**

**Uname ----** It is used to get to know to that the os.

**Uname –r -------** To kow the which kernel version we are using.

**Cat /etc/os-release ------** TO know which type of os falvour we are using.

**Cat** --- It is used to read the data in a file.

**Etc ---** It is a folder

**Os-release** ------- file

**Uptime**  ---- TO know how long the server has been running.

**Time** --- To know the present time

**Date** ---- to know current date and time

**Cal** --- It is used to know the present month

**Cal 2023** -- it gets the current year calander.

**Clear** --- It is used to clear the screen /(ctrl + l)

**Whoami** --- To know the current user

**W** --- default user

**Who** – default user but it gives the region

**History** --- to know the commands that we are used till now.

**Hostname** ------ It is used to see the IP adress.

**HARDWARE COMMANDS**

**cat /proc/cpuinfo** ------ To know the information about cpu / **lscpu**

**cat** ---- It is used to read the data in a file

p**roc** ------ processor folder

c**puinfo** ------file name

**cat /proc/meminfo**  ------- To know the information about memory information. / **free**

**free –m ----** It converts the information into megabytes

**df -h** --------- To know how much volume we have used in our server.

**fdisk –l**  ------ To know how many volumes we are using.

**dmesg** ------ To know the information about the bootup messages. ( The information is related to kernel.)

**Ps** ----- To see the list of processors.

**FILE COMMANDS**

**touch filename ---** To create a file we use touch command.

**ll ----** It is used to see the list of files n folders (Long List)

**touch aws azure gcp** --- It is used to create multiple files at a time.

**ll –t** --- It gives the files which are created recently.

**ll –r** --- It gives the files in reverse order.(r --- Recursive)

**ll** –a ---- It gives the hidden files also.

. ----- If any file starts with (.) then it is hidden file.

**ll –ltr , ll –trl, ll-rtl ---** To see the list of files.

**rm** **filename**----- It is used to remove the file.

**rm –f filename** ----- It is used to delete the file forcefully.

**Cat > filename** ---- To insert data into the file.

**Ctrl + d** ------- To savethe data.

**Cat** ----- It is used to read the data in a file.

**Cat >> filename** ----- TO add data second time.

**Cat myfile** ----- To see data in the file.

**Cp souce file destination file** ----- To copy dat afrom one file to another file.

**Mv souce file destination file** ---- moving from souce file to destination file.

**Cat source file destination file** ---- It copies dat from S to D

**Cat –n fileneame** --- It prints with the lines

**Tac filename** – it reverse the data.

**Rev filename** – It reverse the data.

**Mkdir foldername** ---- It is used to create a folder.

**Rmdir foldername** ------ It is used to remove the folder.

**rm –f \*** ---- It is used to delete all files.

**rmdir \*** ----- It removes all directories.

**rm –rf \*** ---- It is used to delete all files and folders.

**STEPS TO GO INSIDE A FOLDER**

* mkdir folder1
* md folder1 (CHANGE DIRECTORY)
* ll
* mkdir folder2
* cd folder2
* mkdir folder3
* cd folder3
* Mkdir folder4
* Cd folder4
* Cd .. ---- it takes one step back
* Pwd -----(Present working directory)

**/root/folder1/folder2/folder3/folder4**

Cd : Used to go to home directory.

Cd .. : used to go to one step back.

Cd - : used to go to previous folder.

HOW TO CREATE A FILE INSIDE A FOLDER

Touch myfile ----creating file in fiolder1.

touch folder1/lakshmi.txt --- It creates a text file inside the folder 1 without going inside the folder.

touch folder2/ajar.pdf

touch folder3/anitha.jpg

**ls folder3** ----- It gives the list of files inside the folder3 without going inside the folder.

**mkdir folder1/folder2/folder3/dharani** --- Creating a folder inside the folder3 from home directory.

**ls folder1/folder2/folder3/** ---- It shows the list of files/folders from home directory.

HOW TO COPY THE FILE FROM HOME DIRECTORY TO FOLDER

Cp source folder1

**mkdir -p tcs/infosys/ibm** ---- It automatically creates the folder inside the folder(PARENTING –p).

TO MODIFY DATA IN FILES WE USE EDITORS

**VIM EDITOR(VISUAL EDITOR)**

It is used to modify the files. It has 3 modes.

1. COMMAND MODE
2. INSERT MODE
3. SAVE & QUIT MODE

**1.COMMAND MODE :** It is the default mode in Vim editor.Command mode is used to copy delete and paste the lines in a file.

Yy : It is used to copy the line

P : print the copied content

3p : It is used to print the copied content 3 times.

3d : it is used to delete the line.

3dd : It is used to delete 3 lines.

gg : Goes to the first line of the file.

G : goes to the last line of a file.

:5 : goes to the 5th line of a file.

23gg : Goes to 23rd line of a file.

U :undo the data (TO remove modifications)

Ctrl + r : redo the changes

: set number : sets number to the file.

/word : used to find the word.

?word: finds the word in a file.

:%s/old word/new word --- It is used to replace the word

S ---- String.

:%s/oldword/newword/g ----- It is used change many same words.

INSERT MODE : It is used to insert/modify the data in a file.

i : It is used to go from command mode to insert mode.

esc : It is used to go from insert mode to command mode.

O : new line creates (up of the cursor)

o : creates new line (down of the cursor)

A : used to go to end of the line

I : starting of the line.

a : move one character ahead.

**SAVE & QUIT MODE : It is used to save data and quit from vim editor.**

:w – used to save data

:w! -- used to save the data forcefully.

:q – quit from vi editor

:q! -- forcefully quit

:wq – save & quit at a time

:wq! -- save & quit forcefully

Vim filename: To go to vim editor.

**FILE PERMISSIONS**

**-rw-r- - r- - 1 root root 0**

* : Type of the file

- : regular file

D : directory file

C : character file

B : blocked file

| : pipe file

PERMISSIONS

R : read : 4

W : write : 2

X : execute : 1

- : nothing : 0

1 ; ACL(Access control list) -- For file it is 1 , For folder it is 2

root root ------ Owners of the file

root : user

root : group

0 ---- data in kb

USER : rw- : 4+2+0 =6

GROUP : r - - = 4+0+0 =4

OTHERS : r - - = 4+0+0 = 4

**USERS**

**cat /etc/passwd --** To see list of users.

**Useradd username** ----- It creates new user.

**User:**

**devops:x: 1002:1002::/home/devops:/bin/bash**

**devops ------** username

X ----- We can store the user passwords in x

1002 ----- uid(user id)

1002 ----- gid (group id)

/home/devops ----- devops folder created in home folder.

/bin/bash ---- commands will store in this path.

**passwd username** --- It cretes password for the user.

**Su - username** --- It is used to go from root user to any user. If u login from any other user except root user it asks password.

**cd /hom**e --- All the user are in home.

**User del filename**  ---- It is used to del the user.

**cat /etc/group** --- list of groups.

**group add aws** ----- It is used to create our own group.

**If we create group users are not created. If we create users groups are created.**

**group del username ----** It is used to delete particular group.

CHANGING PERMISSIONS OF A FILE(OWNERS)

**chown username filename - - -** It is used to change the root of the file name.

**chgrp groupname filename - - -** It is used to change the group of the file name.

**chown username:groupname filename - - -** This command is used to change the user and group at a time.

**chmod 751 filename** : It is used to change the permissions of a file.

---- If the file colour changes then the file has executable permissions.

**find command** ----- It is used to know whether the file is present are not.

**find . -name “\*.txt”**

**find . -name “\*.filename”**

**grep (GLOBAL REGULAR EXPRESSION PRINT)**

**grep --- It is used word for searching.**

**grep – i “the” filename ---** It is used to find the word in the file.j

**grep –c “word” filename (-c ==== COUNT)**

**grep –n “word” filename (-n === USED FOR LINES)**

**grep –i “THE” filename (i== is used for capital and small letters to identify.)**

**Vim –o filename1 filename2 - - - -** We can compare two files in vim editor.

**Vim –O filename1 filename2 - - - -** We can compare two files in vim editor side by side. (-O --- Comparing the files)

**head filename** --- It prints the first 10 lines

**tail filename** ------ It prints the last 10 lines

**sed –n “range 3,14p” filename** ---- It prints the middile lines.

**more** ----- It is used to see the data for multiple files.

**nl filename** --- It gives the data along with numbers.

**Cat -n filename** --- It gives the data along with numbers.(it gives the number even the line is empty).

**cat -E filename** ---- prints the line along with $ symbol

$ -- represents the end of the line.

**sort filename** --- It is used to print in alphabetical order.

**sort -r filename**  ------- It is used to print in reverse order.

**nslookup google.com ----**It is used to get the IP adreess of a website.

**Alias t=touch** ---- It is the shortcut to cretae a file.

t = filename

t = filename2

**SHELL SCRIPTING**

**Scripting language :** Not doing manual work and not giving multiple commands to create file folders we use scripting language (TO Automate the tasks we are using shell scripting).

**Vim filename.sh** ---- Sh(shell).creating a file with .sh

**#! ------ Shabang**(Every bash file starts with #!)

**#! /bin/bash** ------- Whatever the commands we use they store in /bin/bash.

Commands

**Touch aws gcp**  --- Creating a file in bash

**Mkdir cloud-computing** --- Creating a folder in bash

To execute above files it should consists of executable permissions.

**Chmod +x filename.sh**

**./filename.sh** ---- To execute any file

**Echo** ---- it is used to print the line in bash

Echo “Hi everyone, we are learning DevOps”

echo -e “This is the \t second \n line”

\t --- space /// \n ------- new line.

VARIABLES

Variables are used to store a data.

**Name** = “Yamuna”

**echo** “Hi this is $name”

Name = “Dev”

Age = 22

Place =”hyderabad”

Course =”devops”

Fee =”12000”

Echo “hai my name is $name”

Echo “I am from $palce and my age is $age , I am learning $course and fee is $ fee.

TASK --- If we enter a number or name with that it should create file.

echo “Please enter a name that you want to create a file”

Read abc ------READ(storing) abc(variable)

Touch 4abc

TASK --- If we enter a number or name with that it should create folder.

echo “Please enter a name that you want to create a folder”

Read hey ------READ(storing) hey(variable)

Mkdir $hey

TASK --- If we enter a number or name with that it should delete a folder.

echo “Please enter a name that you want to delete a folder”

Read hey ------READ(storing) hey(variable)

rmdir $hey

Execute every time

**Bash one.sh / sh one.sh**----- to execte we use it

CONDITIONAL STTMENTS

Count =5

If [ $count –eq 5 ] (eq---Equal to)

Then

echo “your value is correct”

Fi

Count =5

If [ $count –eq 5 ] (eq---Equal to)

Then

echo “your value is correct”

Elif [ $count –gt 5 ]

then

Echo “ Your value is greater than count value”

Elif [ $count –lt 5 ]

Then

echo “ your value is less than count value

fi