# Experiment No. : 1

**Title:** Implementation of DevOps related Linux commands

# Objectives:

To implement various Linux commands related to DevOps.

# Theory:

# Commands:

# mkdir - to create new directories

# mkdir name

# mkdir name1 name2

# mkdir name/newname

# ls- to list all the files and folders present at that location

# cd - change directory

# cd nameofdirectory

# cd .. - to navigate back single directory

# cd - to navigate back to home location

# touch - to create empty files, (files with no data)

# touch filename

# echo - to enter data while creating the file.

# echo "your data" > filename

# cat - cat filename -> to check data present in the file

# echo "Shashank" >> q.txt - to append data

# cat >> a.txt - to append

# nano file - to open editor

# copying and moving files and folders.

# cp /home/ubuntu/t1/t13 /home/ubuntu/test1/

# cp -r /home/ubuntu/f1/f13 /home/ubuntu/test1/

# #mv command - cut paste

# mv /home/ubuntu/f11/ /home/ubuntu/test4/

# mv /home/ubuntu/f1/f11/ /home/ubuntu/test4/

# #delete files and folders

# rm 1.txt 2.txt - remove files

# rmdir f1 - remove empty folder

# rm -r f2 - if folder not empty

# #installing softwares and packages

# sudo apt-get update - to get minor updates in your system

# sudo apt-get install pacakgename - to install a package

# EXAMPLE : sudo apt-get install apache2 - to install apache2 in your system

grep - to extract standard ASCII text from a file

-i - case insensitive

-w - searching exact words

-i -w /-iw - case with words

-v - it discards/omits the pattern

pipeline - to redirect the output of a command to the next command specified.

ps -e | grep 'ssh'

mkdir kal | touch kalyan

ps(process status) - processes running on our system (ppid-parent process id) (stime-standard time)

ps -e --> to see all the background processes.

ps -f frontend

ps -alx/elx-all frontend n backend

man - to know about the documentation of a particular command/service

example: man command\_name

awk - works with files containing tables

NF - no. of fields/columns

NR - no. of rows

$0 - to print everything Example - awk '{print $0}' will print all your data in the file, i.e, whole table

awk '{print $1}' reports.txt - prints first column

awk '{print NR,$2,$3}' reports.txt - to add no. for number of rows in output

awk '{print $2,$3}' reports.txt - to print multiple columns

awk '{print NF}' reports.txt - to print no. of elments in the row

awk '{if (NR==2) print $0}' reports.txt - prints the 2nd row

awk '{if (NR==2) print $4}' reports.txt - prints the 4th coloumn of second row

ping - to reach out to the server.

alias - to create a nickname for a particular command

Syntax - alias name='command to be given alias'

to enter .bashrc file --> 1. sudo su

2. nano ~/.bashrc

Source ~/.bashrc

Example : alias linux='sudo apt-get update'

ls -l is used for all the permissions

ls -r seeing sub folders

ls -a for hidden files