

Department of Computer Engineering

Batch: B2

Name: Ashwera Hasan

Roll: 16010124107

TITLE: Exploring basic Commands of UNIX: Shell, Processes, Files

AIM: To Explore basic commands for handling File system under Unix/Linux using shell scripts.(Creating groups, chown , chmod , directory name, tty , diff, umask).

Expected Outcome of Experiment:

CO 1. To introduce basic concepts and functions of operating systems.

Books/ Journals/ Websites referred:

1. Silberschatz A., Galvin P., Gagne G. “Operating Systems Principles”, Willey Eight edition.
2. Achyut S. Godbole , Atul Kahate “Operating Systems”, McGraw Hill Third Edition.
3. Sumitabha Das “ UNIX Concepts & Applications”, McGraw Hill Second Edition

Pre Lab/ Prior Concepts:

An operating system (OS) is a resource manager. It takes the form of a set of software routines that allow users and application programs to access system resources (e.g. the CPU, memory, disks, modems, printers network cards etc.) in safe efficient and abstract way.

- The operating system kernel is in direct control of the underlying hardware. The kernel provides low-level device, memory and processor management functions (e.g. dealing with interrupts from hardware devices, sharing the processor among multiple programs, allocating memory for programs etc.)
- Basic hardware-independent kernel services are exposed to higher-level programs through a library of system calls (e.g. services to create a file, begin execution of a program, or open a logical network connection to another computer).
- Application programs (e.g. word processors, spreadsheets) and system utility programs (simple but useful application programs that come with the operating system, e.g. programs which find text inside a group of files) make use of system calls. Applications and system utilities are launched using a shell (a textual command line interface) or a graphical user interface that provides direct user interaction.

Operating systems can be distinguished from one another by the system calls, system utilities and user interface they provide, as well as by the resource scheduling policies implemented by the kernel.

UNIX has been a popular OS for more than two decades because of its multi-user, multi-tasking environment, stability, portability and powerful networking capabilities.

Linux is a free open source UNIX OS for PCs.

Linux has all of the components of a typical OS :

Department of Computer Engineering

- **Kernel**

The Linux kernel includes device driver support for a large number of PC hardware devices (graphics cards, network cards, hard disks etc.), advanced processor and memory management features, and support for many different types of file systems. In terms of the services that it provides to application programs and system utilities, the kernel implements most BSD and SYSV system calls, as well as the system calls described in the POSIX.1 specification.

The kernel (in raw binary form that is loaded directly into memory at system startup time) is typically found in the file `/boot/vmlinuz`, while the source files can usually be found in `/usr/src/linux`.

- **Shells and GUIs**

Linux supports two forms of command input: through textual command line shells similar to those found on most UNIX systems (e.g. `sh` - the Bourne shell, `bash` - the Bourne again shell and `csh` - the C shell) and through graphical interfaces (GUIs) such as the KDE and GNOME window managers.

- **System Utilities**

Virtually every system utility that you would expect to find on standard implementations of UNIX has been ported to Linux. This includes commands such as `ls`, `cp`, `grep`, `awk`, `sed`, `bc`, `wc`, `more`, and so on. These system utilities are designed to be powerful tools that do a single task extremely well (e.g. `grep` finds text inside files while `wc` counts the number of words, lines and bytes inside a file). Users can often solve problems by interconnecting these tools instead of writing a large monolithic application program.

- **Application programs**

Linux distributions typically come with several useful application programs as standard. Examples include the `emacs` editor, `xv` (an image viewer), `gcc` (a C compiler), `g++` (a C++ compiler), `xfig` (a drawing package), `latex` (a powerful typesetting language) and `soffice` (StarOffice, which is an MS-Office style clone that can read and write Word, Excel and PowerPoint files).

Description of Unix Commands and options:

Department of Computer Engineering

A. Basic and Important UNIX (Linux/macOS) Commands

1. Directory and File Management- ls, cp, rm , mv,
2. Viewing and Editing Files- cat, echo, more , less , head, tail , grep
3. Searching and Permissions- chmod, chown , chgrp

B. Process and System Management

1. Process Control- ps, top, kill
2. System Information- hostname, w, uname

C. Advanced and Useful UNIX Commands

1. Administrative and Remote Operations- whoami , id, groups, passwd , who, last
2. Searching and Processing Files- find, cut, sort
3. System Monitoring and Disk Management- history, top, df, du

Screen Shot of Implemented Commands:

A. Basic and Important UNIX (Linux/macOS) Commands

Command: ls

Use: Used to list all files and directories in the terminal.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ ls
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ mkdir Folder1
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ mkdir Folder2
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ ls
Folder1  Folder2
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$
```

Command: cp

Use: Transfers the content of file1 to file2. The first file followed by cp is the source file and second file is the destination file.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/AshweraHasan/Folde
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat file1.txt
ashwera is a rockstar
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat file2.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cp file1.txt file2.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat file2.txt
ashwera is a rockstar
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat file1.txt
ashwera is a rockstar
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: rm

Use: Delete the file name/folder that follows rm keyword. Rm* removes everything under this folder.

Screenshot:

Department of Computer Engineering

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ls
file2.txt  fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ rm file2.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ls
fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: mv

Use: When used in syntax mv [oldname] [newname], the command renames the file. When used in syntax mv source destination, it is used to move a file/folder from one location to another.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ls
file1.txt  file2.txt  file3.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ mv file1.txt fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ls
file2.txt  file3.txt  fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

```
Moving file3 contents to ashweraCopy directory
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cd..
cd..: command not found
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cd..
cd..: command not found
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cd ..
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ cd ..
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ cd
aditya/  .cache/  Desktop/  .gnupg/  manit/    my1.txt/  new2/     .pki/     .ssh/     Videos/
AshweraCopy/  .config/  Documents/  JInKaChiraag/  monaco/   my2.txt/  Paras/    Public/    Templates/
AshweraHasan/  demo1/   Downloads/  .local/    Music/    new1/     Pictures/  snap/      test1/
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ cd AshweraCopy
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraCopy$ pwd
/home/kjsce/AshweraCopy
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraCopy$ cd ..
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ cd AshweraHasan
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ cd Folder1
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ mv file3.txt /home/kjsce/AshweraCopy
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ls
file2.txt  fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cd ..
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ cd ..
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ cd AshweraCopy
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraCopy$ ls
file3.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraCopy$
```

Command: touch, cat, echo

Use: Touch updates the access and modification times of each file, if no time is mentioned, touch takes the current time. Cat is used to open the file for read or write. Echo is used to write to this file; anything followed by echo is written into the file.

Screenshot:

Department of Computer Engineering

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/AshweraHasan/Folde
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ touch file2.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ls
file1.txt  file2.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/AshweraHasan/Folde
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ls
file1.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ echo "ashwera is a rockstar" > file1.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ echo "Stay geeky"
Stay geeky
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat file1.txt
ashwera is a rockstar
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: less

Use: Used to open a file. Once the file is open in 'less', we can navigate through it using various commands

Screenshot:

```
ashwera is a rockstar
fileONE.txt (END)
```

Command: more

Use: Based on the cmd following more -, this command displays the text in a file.

Screenshot:

```
fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ more -d fileONE.txt
ashwera is a rockstar
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ more -f fileONE.txt
ashwera is a rockstar
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: head

Use: By default, head prints the first 10 lines of the file. When followed by -n num, it prints the first num lines. There are other commands associated with head, too.

Screenshot:

Department of Computer Engineering

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ touch fileTWO.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ls
fileONE.txt  fileTWO.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat fileTWO.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ echo -e "January\nFebruary\nMaruary\n">>fileTWO.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ head fileTWO.txt
January
February
Maruary

kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ head -n 1 fileTWO.txt
January
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: tail

Use: By default, tail prints the last 10 lines of the file. When followed by -n num, it prints the last num lines. There are other commands associated with tail, too.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ tail fileTWO.txt
January
February
Maruary

kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ tail -n 1 fileTWO.txt
Maruary

kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ tail -n 21 fileTWO.txt
January
February
Maruary

kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ tail -n 2 fileTWO.txt
Maruary
```

Command: grep

Use: It is used to search for specific words, phrases, or patterns inside text files, and shows the matching lines on the screen.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ grep "uary" fileTWO.txt
January
February
Maruary
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ grep "hi" fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: chmod

Use: The chmod (change mode) command is used to set or modify file and directory permissions

Screenshot:

Department of Computer Engineering

Command: chown

Use: Changes the owner of the file/folder.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ chown ashwera fileONE.txt
chown: invalid user: 'ashwera'
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ chown kjsce fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: chgrp

Use: The chgrp command in Linux is used to change the group ownership of a file or directory.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ sudo chgrp ashwera fileONE.txt
[sudo] password for kjsce:
```

B. Process and System Management

Command: ps

Use: The ps command in Linux is used to display information about the currently running processes on the system.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ ps
  PID TTY          TIME CMD
  6267 pts/0        00:00:00 bash
 29707 pts/0        00:00:00 ps
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: kill

Use: The kill command in Linux is used to send signals to processes for managing their execution, typically to terminate them

Screenshot:

Department of Computer Engineering

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ cd AshweraHasan
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ ps
  PID TTY          TIME CMD
 31058 pts/0    00:00:00 bash
 31095 pts/0    00:00:00 ps
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$
```

after kill command, the terminal closed. Upon opening, we can see that the new bash process has a different PID.

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ id
uid=1000(kjsce) gid=1000(kjsce) groups=1000(kjsce),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),100(users),114(lpadmin)
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: top

Use: Shows a dynamic, real-time view of running processes and kernel-managed tasks in Linux

Screenshot:

```
top - 15:51:15 up 1:51, 1 user, load average: 0.38, 0.53, 0.64
Tasks: 521 total, 2 running, 519 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.5 us, 0.5 sy, 0.0 ni, 97.9 id, 0.1 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 31762.8 total, 23384.6 free, 4989.9 used, 4740.6 buff/cache
MiB Swap: 8192.0 total, 8192.0 free, 0.0 used. 26772.8 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3351	kjsce	20	0	4782680	359184	141004	R	11.6	1.1	3:34.46	gnome-shell
34904	kjsce	20	0	4148960	191280	124188	S	9.3	0.6	0:00.28	gnome-character
34900	kjsce	20	0	1593708	130564	101960	S	5.0	0.4	0:00.15	nautilus
34905	kjsce	20	0	1141216	121632	94896	S	4.7	0.4	0:00.14	gnome-clocks
31046	kjsce	20	0	8488800	61572	43076	S	4.0	0.2	0:13.33	gnome-terminal-
3107	kjsce	20	0	25.1g	198472	135100	S	2.3	0.6	3:10.96	Xorg
34909	kjsce	20	0	1040704	110068	86424	S	2.3	0.3	0:00.07	gnome-control-c
3500	kjsce	20	0	462592	12864	7044	S	1.3	0.0	0:38.29	ibus-daemon
3020	kjsce	20	0	11184	6952	4680	S	1.0	0.0	0:03.01	dbus-daemon
34897	kjsce	20	0	14544	5952	3672	R	0.7	0.0	0:00.03	top
1769	root	20	0	336512	19552	16428	S	0.3	0.1	0:12.54	NetworkManager
1838	syslog	20	0	222564	6036	4596	S	0.3	0.0	0:00.28	rsyslogd
1993	root	20	0	2384252	44268	30492	S	0.3	0.1	0:09.35	containerd
2983	kjsce	20	0	21464	12912	9612	S	0.3	0.0	0:00.90	systemd
3010	kjsce	20	0	316696	10612	9336	S	0.3	0.0	0:00.88	gnome-keyring-d
3067	kjsce	20	0	536560	7588	6820	S	0.3	0.0	0:00.33	xdg-document-po
3285	kjsce	20	0	9612	5392	4700	S	0.3	0.0	0:00.65	dbus-daemon
3309	kjsce	20	0	314260	8336	7348	S	0.3	0.0	0:00.20	gvfsd
3696	kjsce	20	0	347900	32248	18568	S	0.3	0.1	0:08.84	ibus-extension-
3754	kjsce	20	0	2408612	109608	83972	S	0.3	0.3	0:01.70	snapd-desktop-i
3783	kjsce	20	0	309772	6784	6120	S	0.3	0.0	0:00.13	gvfs-goa-volume
3788	kjsce	20	0	389376	8196	7092	S	0.3	0.0	0:00.44	gvfs-afc-volume
3877	kjsce	39	19	947616	29872	18808	S	0.3	0.1	0:02.22	tracker-miner-f
4084	kjsce	20	0	627992	15132	12272	S	0.3	0.0	0:00.93	xdg-desktop-por
6499	kjsce	20	0	49.1g	274060	135380	S	0.3	0.8	2:46.49	chrome
11104	root	20	0	0	0	0	I	0.3	0.0	0:04.31	kworker/u112:4-i915
29544	kjsce	20	0	1392.4g	117784	92660	S	0.3	0.4	0:00.75	chrome
30530	root	20	0	0	0	0	I	0.3	0.0	0:01.31	kworker/u112:3-i915
34901	kjsce	20	0	527032	17312	15272	S	0.3	0.1	0:00.01	gnome-calculato
1	root	20	0	23744	14852	9504	S	0.0	0.0	0:02.15	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pool_workqueue_release
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-rcu_gp
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-sync_wq
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-kvfree_rcu_reclaim
7	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-slub_flushwq

Department of Computer Engineering

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ chmod 745 fileONE.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: who, whoami, w

Use: who command is used to find out the following information :

- Time of last system boot
- Current run level of the system
- List of logged-in users and more.

It is basically the concatenation of the strings "who","am","i" as whoami. It displays the username of the current user when this command is invoked.

W provides information about the users currently logged into the system and their activities.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ who
kjsce  seat0      2026-01-14 13:59 (login screen)
kjsce  :1          2026-01-14 13:59 (:1)
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ whoami
kjsce
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ w
15:20:29 up 1:20, 1 user, load average: 0.58, 0.64, 0.55
USER   TTY      FROM             LOGIN@   IDLE   JCPU   PCPU   WHAT
kjsce  tty2    -                13:59    1:20m  1:58   0.05s /usr/libexec/gnome-session-binary --session=ubuntu
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: hostname

Use: hostname command in Linux is used to obtain the DNS (Domain Name System) name.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ hostname
kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: uname

Use: The uname command writes to standard output the name of the operating system that you are using.

Screenshot:

Department of Computer Engineering

```
kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ uname
Linux
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Advanced and Useful UNIX Commands

Command: find

Use: Finds the file and returns its path.

Screenshot:

```
find: /Folder1: no such file or directory
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ find -name *.txt
./Folder1/fileONE.txt
./Folder1/fileTWO.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$
```

Command: cut

Use: Cuts from a file based on bytes, characters, etc.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cut -b 1,2,3 fileTWO.txt
Jan
Feb
Mar
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: sort

Use: Print the content of the file in sorted order.

Screenshot:

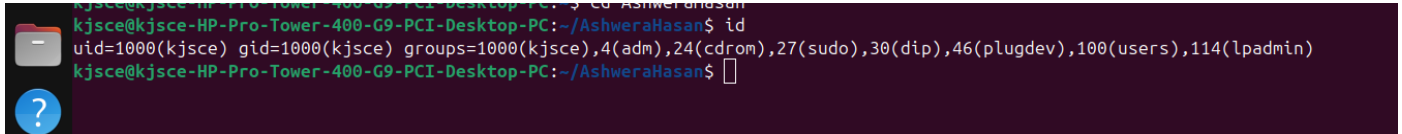
```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ sort fileTWO.txt
February
January
Maruary
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Department of Computer Engineering

Command: id

Use: a basic Linux command used to confirm the identity of a specified Linux user.

Screenshot:



```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/AshweraHasan$ id
uid=1000(kjsce) gid=1000(kjsce) groups=1000(kjsce),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),100(users),114(lpadmin)
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/AshweraHasan$
```

Command: history

Use: The history command in Linux displays a list of commands that were previously entered in the terminal

Screenshot:

Department of Computer Engineering

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ history
1  update-grub
2  sudo su
3  cl
4  dl
5  lc
6  ls
7  pwd
8  mkdir demo1
9  ls
10 cd demo1
11 cd ..
12 cd demo1
13 touch file1.txt
14 ls
15 touch file2.txt
16 ls
17 echo "Hello Linux">file1.txt
18 cat file1.txt
19 cp file1.txt file2.txt
20 cat file2.txt
21 mv file2.txt ../
22 rm file2.txt
23 ls ..
24 rm file2.txt
25 cd ..
26 rm -r file2.txt
27 touch file2.txt
28 ls
29 rm file2.txt
30 cd demo1
31 touch file2.txt
32 cat file2.txt
33 head file2.txt
34 head -n 7 file2.txt
35 tail file2.txt
36 tail -n 1 file2.txt
37 tail -n 10 file2.txt
38 history
39 where python
40 hack karde
41 cd --help
42 touch file1.txt
```

Command: last

Use: Used to display a list of all users who have logged in and out since the creation of the '/var/log/wtmp' file.

Screenshot:

Department of Computer Engineering

```

NO PLAN.
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ las
kjsce      :1      :1      Wed Jan 14 13:59      still logged in
kjsce      seat0    login screen Wed Jan 14 13:59      still logged in
reboot     system boot 6.14.0-37-generi Wed Jan 14 13:59      still running
kjsce      :1      :1      Wed Jan 14 13:53 - down (00:00)
kjsce      seat0    login screen Wed Jan 14 13:53 - down (00:00)
reboot     system boot 6.14.0-37-generi Wed Jan 14 13:53 - 13:54 (00:01)
kjsce      :1      :1      Wed Jan 14 16:48 - down (-3:57)
kjsce      seat0    login screen Wed Jan 14 16:48 - down (-3:57)
reboot     system boot 6.14.0-37-generi Wed Jan 14 16:48 - 12:51 (-3:56)
kjsce      :1      :1      Wed Jan 14 16:38 - down (00:04)
kjsce      seat0    login screen Wed Jan 14 16:38 - down (00:04)
reboot     system boot 6.14.0-37-generi Wed Jan 14 16:38 - 16:43 (00:05)
kjsce      :1      :1      Tue Jan 13 17:34 - down (-5:07)
kjsce      seat0    login screen Tue Jan 13 17:34 - down (-5:07)
reboot     system boot 6.14.0-37-generi Tue Jan 13 17:34 - 12:26 (-5:07)
kjsce      :1      :1      Mon Jan 12 18:21 - down (-2:46)
kjsce      seat0    login screen Mon Jan 12 18:21 - down (-2:46)
reboot     system boot 6.14.0-37-generi Mon Jan 12 18:21 - 15:34 (-2:46)
kjsce      :1      :1      Fri Jan 9 16:17 - down (01:24)
kjsce      seat0    login screen Fri Jan 9 16:17 - down (01:24)
reboot     system boot 6.14.0-37-generi Fri Jan 9 16:17 - 17:41 (01:24)
kjsce      :1      :1      Fri Jan 9 14:13 - down (01:45)
kjsce      seat0    login screen Fri Jan 9 14:13 - down (01:45)
reboot     system boot 6.14.0-37-generi Fri Jan 9 14:12 - 15:58 (01:45)
kjsce      :1      :1      Thu Jan 8 16:11 - down (01:46)
kjsce      seat0    login screen Thu Jan 8 16:11 - down (01:46)
reboot     system boot 6.8.0-31-generic Thu Jan 8 16:11 - 17:57 (01:46)
kjsce      :1      :1      Wed Jan 7 14:17 - down (01:39)
kjsce      seat0    login screen Wed Jan 7 14:17 - down (01:39)
reboot     system boot 6.8.0-31-generic Wed Jan 7 14:17 - 15:57 (01:40)
kjsce      :1      :1      Wed Jan 7 16:47 - down (-3:56)
kjsce      seat0    login screen Wed Jan 7 16:47 - down (-3:56)
reboot     system boot 6.8.0-31-generic Wed Jan 7 16:47 - 12:51 (-3:55)
kjsce      :1      :1      Tue Jan 6 20:42 - down (00:48)
kjsce      seat0    login screen Tue Jan 6 20:42 - down (00:48)
reboot     system boot 6.8.0-31-generic Tue Jan 6 20:41 - 21:30 (00:49)
kjsce      :1      :1      Mon Jan 5 20:18 - 15:30 (-4:48)
  
```

D. Other Linux Commands

Command: pwd

Use: Writes the full path name of current directory to standard output.

Department of Computer Engineering

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ cd AshweraHasan
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ pwd
/home/kjsce/AshweraHasan
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$
```

Command: cd

Use: Used to change the working directory to specified directory/path. cd.. returns to the parent folder of current folder.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ cd Folder1
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: mkdir

Use: Creates directory in current path

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ mkdir Folder3
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ ls
Folder1 Folder2 Folder3
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$
```

Command: rmdir

Use: Removes the directory that follows this command. Returns error output when directory does not exist.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ rmdir Folder3
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ rmdir Ashwera
rmdir: failed to remove 'Ashwera': No such file or directory
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$ ls
Folder1 Folder2
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan$
```

Command: cmp

Use: Is used to compare the content of two files. Returns the byte/line where the two files differ at the first instance and returns nothing if the files are same.

Department of Computer Engineering

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/Ashwera
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ touch file3.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat file3.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ echo "hello world!" >> file3.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat file3.txt
hello world!
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cat file1.txt
ashwera is a rockstar
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cmp file1.txt file3.txt
file1.txt file3.txt differ: byte 1, line 1
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ cmp file1.txt file2.txt
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: useradd

Use: useradd is a command in Linux that is used to add user accounts to your system.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ sudo useradd ashwera
[sudo] password for kjsce:
```

Command: passwd

Use: The passwd command is the standard Linux utility for managing user passwords and account access.

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ passwd
Changing password for kjsce.
Current password:
```

Command: groupadd

Use: The groupadd command creates new groups.

Screenshot:

Department of Computer Engineering

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ sudo groupadd kjsce
[sudo] password for kjsce:
C
adSorry, try again.
[sudo] password for kjsce:
^Csudo: 2 incorrect password attempts
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: finger

Use: The 'finger' command is used to fetch detailed information about users

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ finger kjsce
finger: /dev//seat0: No such file or directory
Login: kjsce                               Name: KJSCE
Directory: /home/kjsce                     Shell: /bin/bash
On since Wed Jan 14 13:59 (IST) on seat0 from login screen
On since Wed Jan 14 13:59 (IST) on :1 from :1 (messages off)
No mail.
No Plan.
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: lpr

Use: Sends the file to connected printer to print

Screenshot:

```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ lpr -p fileONE.txt
lpr: Error - No default destination.
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: date

Use: Displays the current date and time.

Screenshot:

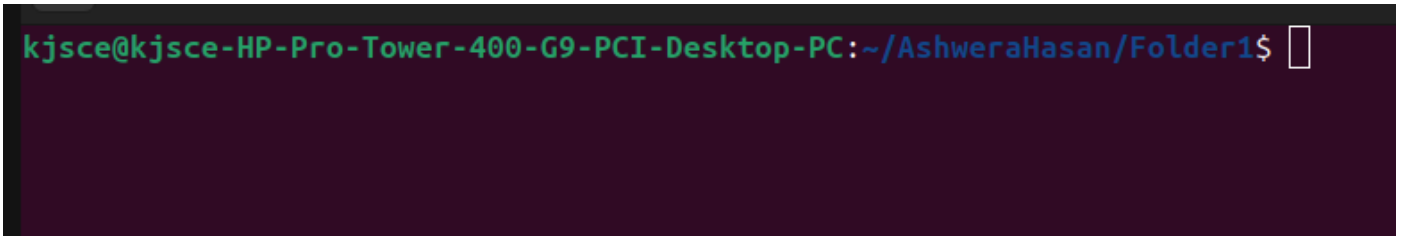
```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ date
Wed Jan 14 03:39:22 PM IST 2026
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: clear

Department of Computer Engineering

Use: Clears the terminal

Screenshot:




```
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: man

Use: Used to display the user manual of any command that we can run on the terminal.

Screenshot:



```
MAN(1) Manual pager utils MAN(1)
NAME
man - an interface to the system reference manuals

SYNOPSIS
man [man options] [[section] page ...] ...
man -k [apropos options] regexp ...
man -K [man options] [section] term ...
man -f [whatis options] page ...
man -l [man options] file ...
man -w|-W [man options] page ...

DESCRIPTION
man is the system's manual pager. Each page argument given to man is normally the name of a program, utility or function. The manual page associated with each of these arguments is then found and displayed. A section, if provided, will direct man to look only in that section of the manual. The default action is to search in all of the available sections following a pre-defined order (see DEFAULTS), and to show only the first page found, even if page exists in several sections.

The table below shows the section numbers of the manual followed by the types of pages they contain.

1 Executable programs or shell commands
2 System calls (functions provided by the kernel)
3 Library calls (functions within program libraries)
4 Special files (usually found in /dev)
5 File formats and conventions, e.g. /etc/passwd
6 Games
7 Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7), man-pages(7)
8 System administration commands (usually only for root)
9 Kernel routines [Non standard]

A manual page consists of several sections.

Conventional section names include NAME, SYNOPSIS, CONFIGURATION, DESCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUE, ERRORS, ENVIRONMENT, FILES, VERSIONS, STANDARDS, NOTES, BUGS, EXAMPLE, AUTHORS, and SEE ALSO.

The following conventions apply to the SYNOPSIS section and can be used as a guide in other sections.

bold text      type exactly as shown.
italic text     replace with appropriate argument.
[-abc]         any or all arguments within [ ] are optional.
-a|-b          options delimited by | cannot be used together.
argument ...   argument is repeatable.
[expression] ... entire expression within [ ] is repeatable.

Manual page man(1) line 1 (press h for help or q to quit)
```

Command: du

Use: It measures the disk space occupied by files or directories

Screenshot:

Department of Computer Engineering

```
man: can't resolve man7/groff_man.7
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ du
12      .
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Command: df

Use: Used to get a report of the total, used and available space on mounted file systems.

Screenshot:

```
12      .
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
tmpfs            3252508      2544   3249964    1% /run
/dev/nvme0n1p7 100205640 25521004 69548252   27% /
tmpfs            16262532    308908 15953624    2% /dev/shm
tmpfs             5120         12     5108    1% /run/lock
efivarfs         246         159        83   66% /sys/firmware/efi/efivars
/dev/nvme0n1p1   262144    124332   137812   48% /boot/efi
tmpfs            3252504      232   3252272    1% /run/user/1000
kjsce@kjsce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/AshweraHasan/Folder1$
```

Conclusion:

This experiment helped me learn the basic linux commands on the terminal.

Post Lab Multiple Choice Questions

1. What does the -r option do when used with the rm command?
 - a) Removes files recursively
 - b) Removes files forcefully
 - c) Removes files interactively
 - d) Removes files silently

Answer: Removes files recursively

2. Which command is used to move a file from one location to another?
 - a) cp

Department of Computer Engineering

- b) mv
- c) rm
- d) touch

Answer: mv

3. What is the purpose of the -n option in the cat command?

- a) Displays the file contents in reverse order
- b) Displays the file contents with line numbers
- c) Displays the file contents in uppercase
- d) Displays the file contents in lowercase

Answer: Displays the file contents with line numbers

4. Which command is used to search for a specific pattern in a file?

- a) find
- b) grep
- c) locate
- d) chmod

Answer: grep

Department of Computer Engineering

Post Lab Descriptive Questions

1. Explain how do you read and interpret syntax of any OS command.
To read and interpret the syntax of any OS command, we first look at the command name to know what it does. Next, we check the options or flags that modify its behavior. Finally, we see the arguments to know which files, folders, or resources the command will act on.
2. Explain different functions of the operating systems.
 - Process Management – Handles running programs and manages CPU time.
 - Memory Management – Keeps track of memory usage and allocates space to programs.
 - File Management – Manages files and directories, including storage and retrieval.
 - Device Management – Controls and coordinates hardware devices like printers and disks.
3. What are the default permissions assigned by Unix for Directory.

In Unix, the default permissions for a directory are usually `rwxr-xr-x` (755)

- Owner: `rw` → can read, write, and enter the directory
- Group: `r-x` → can read and enter, but not write
- Others: `r-x` → can read and enter, but not write

This allows the owner full control, while others can only view and access files inside.

Date: 14/01/2026

Signature of faculty in-charge