Student Name: Manish Gangole

Course Name: C.S.E. (IoT CS BC)

Year: S.E.

Roll No.: 53

PRN No.: 221111028

Course code: CSL403

Semester: IV

## Experiment Evaluation Sheet

## Experiment No.: 1

## Experiment Name:
## Explore usage of basic Linux commands

| Sr No. | Evaluation Criteria | Marks (Out of 9) | Performance Date | Correction Date and Signature of Instructor |
|--------|--------------------|--------------------|------------------|---------------------------------------------|
| 1 | **Experiment Performance** | | | |
| 2 | **Journal Performance** | | | |
| 3 | **Punctuality** | | | |
| | **Total** | | | |

**Aim :**  To explore basic usage of Linux commands like mkdir, cd, pwd, touch, cat chown, chmod etc.

**Software required :** Terminal

## Theory :

The Linux command is a utility of the Linux operating system. All basic and advanced tasks can be done by executing commands. The commands are executed on the Linux terminal. The terminal is a command-line interface to interact with the system, which is similar to the command prompt in the Windows OS. Commands in Linux are case-sensitive.

Linux provides a powerful command-line interface compared to other operating systems such as Windows and MacOS. We can do basic work and advanced work through its terminal. We can do some basic tasks such as creating a file, deleting a file, moving a file, and more. In addition, we can also perform advanced tasks such as administrative tasks (including package installation, user management), networking tasks (ssh connection), security tasks, and many more.

**1. pwd Command**
The pwd command is used to display the location of the current working directory.

2. mkdir Command
The mkdir command is used to create a new directory under any directory.

**3. ls Command**
The ls command is used to display a list of content of a directory.

**4. cd Command**
The cd command is used to change the current directory.

**5. touch Command**
The touch command is used to create empty files. We can create multiple empty files by executing it once.

**6. cat Command**
The cat command is a multi-purpose utility in the Linux system. It can be used to create a file, display content of the file, copy the content of one file to another file, and more.

**7. chown Command**
The chown command is used to change the owner of the file.

**8. chgrp Command**
The chgrp command is used to change group owner of the file.

**9. chmod Command**
The chmod command is used to chang the read write and execute permissions of a file.

**Output :**

```
MANISH@UBUNTU:~$ cd Desktop
MANISH@UBUNTU:~/Desktop$ mkdir OSLAB
MANISH@UBUNTU:~/Desktop$ cd OSLAB
MANISH@UBUNTU:~/Desktop/OSLAB$ pwd
/home/MANISH/Desktop/OSLAB
MANISH@UBUNTU:~/Desktop/OSLAB$ touch file.txt
MANISH@UBUNTU:~/Desktop/OSLAB$ cat file.txt
Hello UBUNTU
MANISH@UBUNTU:~/Desktop/OSLAB$ ls -l
total 4
-rw-rw-r-- 1 MANISH MANISH 13 Jan 22 14:28 file.txt
MANISH@UBUNTU:~/Desktop/OSLAB$ sudo chown root file.txt
[sudo] password for MANISH:
MANISH@UBUNTU:~/Desktop/OSLAB$ ls -l
total 4
-rw-rw-r-- 1 root MANISH 13 Jan 22 14:28 file.txt
MANISH@UBUNTU:~/Desktop/OSLAB$ sudo chgrp root file.txt
MANISH@UBUNTU:~/Desktop/OSLAB$ ls -l
total 4
-rw-rw-r-- 1 root root 13 Jan 22 14:28 file.txt
MANISH@UBUNTU:~/Desktop/OSLAB$ sudo chmod 744 file.txt
MANISH@UBUNTU:~/Desktop/OSLAB$ ls -l
total 4
-rwxr--r-- 1 root root 13 Jan 22 14:28 file.txt
```

**ii) Program to illustrate Open and Close system call:**
The gcc command is the GNU Compiler Collection and is used to compile C programs. It compiles the C program in the file and produces an executable file using the -o option.
The './' specifies the current directory and executes the compiled program.
These commands are essential for compiling and executing C programs in a Linux environment

```c
#include <stdio.h>
#include <stdlib.h>
int main() {
        int num;
        FILE *fptr;
        fptr = fopen("num.txt","w");
        if(fptr == NULL) {
                printf("Error!");
                exit(1);
        }
        printf("Enter num: ");
        scanf("%d",&num);
        fprintf(fptr,"%d\n",num);
        fclose(fptr);
        return 0;
}
```

**Output :**

```
~/OSLab/02Practical$ cat num.txt
~/OSLab/02Practical$ gcc file.c -o file
~/OSLab/02Practical$ ./file
 Enter num: 69
/OSLab/02Practical$ cat num.txt
69
```

**Conclusion :**

With this practical we understood how to use bacic Linux commands like mkdir, cd, pwd, touch, cat chown, chmod etc.