

EXPERIMENT NO. 1: Practice basic bash, IO redirection, pipe, file and directory operations, searching files, process management and vi/nano editor commands.

1. **PWD** :The pwd command stands for “print working directory,” and it outputs the absolute path of the directory you’re in.

```
suraj@surajpandit:~$ pwd
/home/suraj
suraj@surajpandit:~$
```

2. **LS** : It allows you to list the contents of the directory you want (the current directory by default), including files and other nested directories.
 - **Ls ..** : list files of previous directory.
 - **Ls ../../** : To move back 2 directories and list files.
 - **Ls -l** : list files with information about them.

```
suraj@surajpandit:~$ ls
Desktop  Documents  Downloads  file.txt  Music  myfile1.txt  myfile2.txt  os  osst  Pictures  Public  snap  Templates  Videos
suraj@surajpandit:~$ ls ../../
bin  boot  cdrom  dev  etc  home  lib  lib32  lib64  libx32  lost+found  media  mnt  opt  proc  root  run  sbin  snap  srv  swapfile  sys  tmp  usr  var
suraj@surajpandit:~$ ls -l
total 56
drwxr-xr-x  2 suraj suraj 4096 Sep 15 22:50 Desktop
drwxr-xr-x  2 suraj suraj 4096 Oct  5 18:16 Documents
drwxr-xr-x  2 suraj suraj 4096 Sep 15 18:39 Downloads
-rw-rw-r--  1 suraj suraj   29 Oct  5 17:36 file.txt
drwxr-xr-x  2 suraj suraj 4096 Sep 15 18:39 Music
-rw-rw-r--  1 suraj suraj   46 Oct  5 18:13 myfile1.txt
-rw-rw-r--  1 suraj suraj   46 Oct  5 18:14 myfile2.txt
drwxrwxr-x  2 suraj suraj 4096 Oct  5 17:55 os
drwxrwxr-x  4 suraj suraj 4096 Oct  5 17:01 osst
drwxr-xr-x  3 suraj suraj 4096 Oct  5 17:39 Pictures
drwxr-xr-x  2 suraj suraj 4096 Sep 15 18:39 Public
drwx----- 4 suraj suraj 4096 Sep 15 18:43 snap
drwxr-xr-x  2 suraj suraj 4096 Sep 15 18:39 Templates
drwxr-xr-x  2 suraj suraj 4096 Sep 15 18:39 Videos
```

3. **CD:** It refers to “change directory” and, as its name suggests, switches you to the directory you’re trying to access.
- **Cd ~** : Change directory to home directory.
 - **Cd /** : Change directory to root directory.

```
suraj@surajpandit: ~/Desktop
suraj@surajpandit:/$ cd ~
suraj@surajpandit:~$ cd Desktop
suraj@surajpandit:~/Desktop$
```

4. **CAT** : It lets you create, view, and concatenate files directly from the terminal.
- **Cat filename.txt** : For printing the content in a file.
 - **Cat > filename.txt** : Create a text file and write into it.
 - **Cat >> filename.txt**: Appending the content of a file.
 - **Cat file1 file2 > file3**: Combining the 2 files and storing the combined data in 3rd file.
 - **Cat -s file.txt** : It will delete multiple empty lines in output.
 - **Cat -b file.txt** : For numbering the lines except blank lines.
 - **Cat -e file.txt** : It will add \$ character at the end of each line.

```
suraj@surajpandit: ~
suraj@surajpandit:~$ cat >abc.txt
hello world
suraj@surajpandit:~$ cat >>abc.txt
operating system
suraj@surajpandit:~$ cat abc.txt
hello world
operating system
suraj@surajpandit:~$ cat -b abc.txt
 1 hello world
 2 operating system
suraj@surajpandit:~$ cat -e abc.txt
hello world $
operating system $
suraj@surajpandit:~$ cat abc.txt >new.txt
suraj@surajpandit:~$ cat new.txt
hello world
operating system
suraj@surajpandit:~$
```

5. **mkdir** : To create folders in the shell, we use the **mkdir** command.

- **mkdir abc** : will create a directory named abc.
- **mkdir -p n1/{n2/n3}**: Creating n1 as parent directory and n2 and n3 folder in it.

```
suraj@surajpandit:~$ mkdir app
suraj@surajpandit:~$ ls
abc.txt  app  Desktop  Documents  Downloads  Music  new  os  osst  Pictures  Public  snap  Templates  Videos
suraj@surajpandit:~$ mkdir -p n1/{n2,n3}
suraj@surajpandit:~$ ls n1
n2  n3
suraj@surajpandit:~$
```

6. **rmdir** : The **rmdir** command removes the directory, specified by the Directory parameter, from the system. The directory must be empty before you can remove it.

- **rmdir -pv n1/n2/n3** : Remove the requested directories recursively and display procedure.

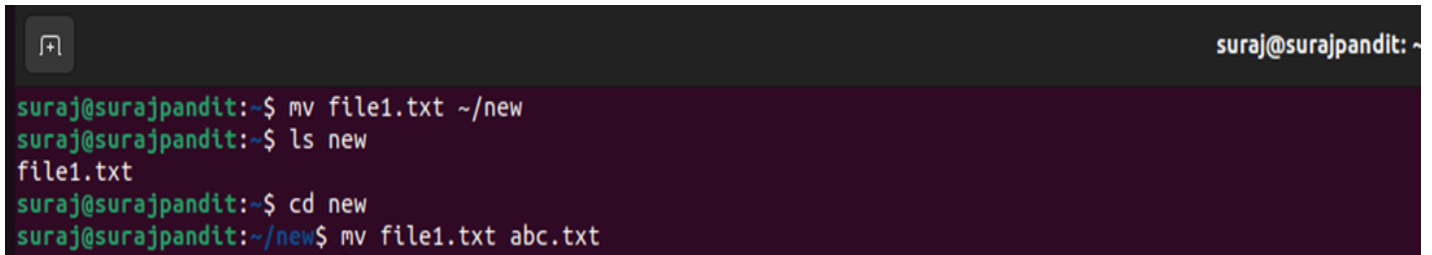
```
suraj@surajpandit:~$ rmdir -pv n1/n2/n3
rmdir: removing directory, 'n1/n2/n3'
rmdir: removing directory, 'n1/n2'
rmdir: removing directory, 'n1'
```

7. **rm** : **rm** command is used to remove objects such as files and directories.

- **rm [file]** : Remove file or directory.
- **rm -i file**: Confirms from user before removing file.
- **rm -r [directory]** : Removes all the directories which are non-empty too.

```
suraj@surajpandit:~$ rm new.txt
suraj@surajpandit:~$ rm -i ab.txt
rm: remove regular file 'ab.txt'? Y
suraj@surajpandit:~$ ls
app  Desktop  Documents  Downloads  Music  n1  new  os  osst  Pictures  Public  snap  Templates  Videos
suraj@surajpandit:~$ rm -r n1
suraj@surajpandit:~$ ls
app  Desktop  Documents  Downloads  Music  new  os  osst  Pictures  Public  snap  Templates  Videos
suraj@surajpandit:~$
```

8. **mv** : You use the mv command to move (or rename) files and directories through your file system.
- **mv file.txt home/document** : It move the file form current directory[home] also k/as source to an destination[document].
 - **mv file.txt new.txt** : It will rename the file name to new.
 - **mv -i [file] [new.txt]** :It will ask the user for configuration before moving a file that would overwrite the existing file.

A terminal window with a dark background and light green text. The window title bar shows a window icon on the left and the text 'suraj@surajpandit: ~' on the right. The terminal content shows a series of commands and their outputs: 'suraj@surajpandit:~\$ mv file1.txt ~/new', 'suraj@surajpandit:~\$ ls new' followed by 'file1.txt' on the next line, 'suraj@surajpandit:~\$ cd new', and 'suraj@surajpandit:~/new\$ mv file1.txt abc.txt'.

```
suraj@surajpandit:~$ mv file1.txt ~/new
suraj@surajpandit:~$ ls new
file1.txt
suraj@surajpandit:~$ cd new
suraj@surajpandit:~/new$ mv file1.txt abc.txt
```