NETWORKING & SYSTEM ADMINISTRATION LAB

Name: Manya Madhu

Roll No: 17

Batch: S2 RMCA B

Date: 31-02-2022

Experiment No.: 1

Aim

Familiarization of linux commands.

Procedure

1. cat -n command

This command is used to number all the output lines.

$ cat -n filename

Output:

```
student@S17:~/manya$ cat -n mm.txt
     1  manya madhu rollno 17
     2  puthuppallikkunnel h
     3  S2 MCA B
     4  13291
     5  NSA
     6  OOP
     7  DBMS
     8  VC
     9
    10  IPR
    11
    12  ACN
```

2. cat -b command

This command is used to number all the output lines except blank lines.

$ cat -b filename

Output:

```
student@S17:~/manya$ cat -b mm.txt
     1  manya madhu rollno 17
     2  puthuppallikkunnel h
     3  S2 MCA B
     4  13291
     5  NSA
     6  OOP
     7  DBMS
     8  VC

     9  IPR

    10  ACN
```

3. touch command

It is used to create a file without any content. The file created using touch command is empty. This command can be used when the user doesn't have data to store at the time of file creation.

$ touch filename

Output:

```
student@S17:~/manya$ touch empty.txt
```

4. echo command

This command in linux is used to display line of text/string that are passed as an argument.

$ echo …. >> filename

Output:

```
student@S17:~/manya$ echo this is an empty file >> empty.txt
```

5. Head command

The head command, as the name implies, print the top N numbers of data of the given input. By default, it prints the first 10 lines of the specified files.

$ head filename

Output:

```
student@S17:~/manya$ head mm.txt
manya madhu rollno 17
puthuppallikkunnel h
S2 MCA B
13291
NSA
OOP
DBMS
VC

IPR
```

$ head -number filename

Output:

```
student@S17:~/manya$ head -3 mm.txt
manya madhu rollno 17
puthuppallikkunnel h
S2 MCA B
```

6. Tail command

The tail command, as the name implies, print the last N number of data of the given input. By default it prints the last 10 lines of the specified files.

$ tail filename

Output:

```
student@S17:~/manya$ tail mm.txt
S2 MCA B
13291
NSA
OOP
DBMS
VC

IPR

ACN
```

$ tail -number filename

Output:

```
student@S17:~/manya$ tail -5 mm.txt
VC

IPR

ACN
```

7. cut command

This command for cutting out the sections from each line of files and writing the result to standard output. It can be used to cut parts of a line by byte position, character and field**.**

$ cut option filename

Output:

```
student@S17:~/manya$ cat > marks
maths 90
english 96
hindi 89
science 99
^Z
[1]+  Stopped                 cat > marks
student@S17:~/manya$  cut -d ' ' -f2 marks
90
96
89
99
```

$ cut -b number filename

This command specifies the byte number.

Output:

```
student@S17:~/manya$ cut -b 5 marks
s
i
i
n
```

$ cut –complement -c number filename

This command is used to display all columns except the column specified in the command.

Output:

```
student@S17:~/manya$ cut --complement -c 1 mark1
alayalam 99
nglish 90
aths 95
```

8. Paste command

Paste command is one of the useful commands in Unix or Linux operating system. It is used to join files horizontally (parallel merging) by outputting lines consisting of lines from each file specified, separated by **tab** as delimiter, to the standard output.

$ paste file1 file2

This command is used to append contents of two files in column order.

Output:

```
student@S17:~/manya$ paste marvel1 marvel2
iron man        harry
spider man      smarf
super man
```

$ paste file1 file2 > file3

It is used to append content of two files in column order and paste into another file.

Output:

```
student@S17:~/manya$ paste marvel1 marvel2 >marvel3
student@S17:~/manya$ cat marvel3
iron man        harry
spider man      smarf
super man       mickey
```

$ paste -d '-' file1 file2

It append contents of two files in column order separated by hyphen delimiter.

Output:

```
student@S17:~/manya$ paste -d '-' marvel1 marvel2
iron man-harry
spider man-smarf
super man-mickey
```

$ paste -d '—' file1 file2 file3

It helps to separate contents of 3 files with hyphen delimiter in column order.

Output:

```
student@S17:~/manya$ paste -d'%|' marvel1 marvel2 marvel1
iron man%harry|iron man
spider man%smarf|spider man
super man%mickey|super man
```

$ paste -s file1 file2

This command is used to append contents in two files in row order.

Output:

```
student@S17:~/manya$ paste -s marvel1 marvel2
iron man        spider man      super man
harry   smarf   mickey
```

9. more command

$ more filename

This command is used to view the text files in the command prompt, displaying one screen at a time in case the file is large.

Output:

```
Photo: What an operating system does: As a user, the last thing you want to be bothered with is all the nitty gritty of how a computer actually works.
 Typically, you work with user-friendly applications ("apps"), which run on top of the operating system. The operating system controls the computer's
basic operations with the help of an even more fundamental layer of software (actually firmware—software embedded in a computer's hardware) called the
 BIOS (Basic Input Output System), which is responsible for booting up your computer when you first switch it on.

Think of a computer as an office block where all the people inside are journalists busily putting together a newspaper. The journalists are trained to
--More--(35%)
```

$ more -s filename

This command is used to squeeze multiple blank lines into one single blank line.

Output:

```
student@T70:~/manya$ more -s f.txt
ould a bunch of ordinary people get together to design and build a space rocket that would take them to the moon? It sounds like a crazy idea, doesn't
 it? How could amateurs possibly cooperate to pull off something so intricate and complex? A few years ago, if you'd suggested thousands of enthusiast
ic geeks could work together to build a rival to Microsoft's mighty Windows operating system, people would have thought you crazy too. But that's exac
tly what's happened with Linux®, a freely available alternative to Windows developed mostly by brilliant young computer programmers in their spare tim
e. Let's take a closer look at the miracle of Linux: what is it, and why should you use it?

Photo: Tux the penguin is the mascot of the Linux operating system. Tux image by Larry Ewing published under a Creative Commons Licence.
```

$ more -2 filename

It is used to type the number lines to display per screen.

Output:

```
student@T70:~/manya$ more -2 f.txt
ould a bunch of ordinary people get together to design and build a space rocket that would take them to the moon? It sounds like a crazy idea, doesn't
 it? How could amateurs possibly cooperate to pull off something so intricate and complex? A few years ago, if you'd suggested thousands of enthusiast
--More--(3%)
```

$ more +2 filename

It is used to display after specified number of lines.

Output:

```
A typical computer architecture linking the hardware to the applications via the BIOS and the operating system.

Photo: What an operating system does: As a user, the last thing you want to be bothered with is all the nitty gritty of how a computer actually works.
 Typically, you work with user-friendly applications ("apps"), which run on top of the operating system. The operating system controls the computer's
basic operations with the help of an even more fundamental layer of software (actually firmware—software embedded in a computer's hardware) called the
 BIOS (Basic Input Output System), which is responsible for booting up your computer when you first switch it on.

Think of a computer as an office block where all the people inside are journalists busily putting together a newspaper. The journalists are trained to
--More--(35%)
```