**OS LAB ASSIGNMENT 2**

**Part 1**

1.pwd

Purpose: Print the current working directory.

Command:

Bash

Pwd

Output:

/home/username

(Displays the absolute path of the current directory.)

2.cd

Purpose: Change the current directory.

Command:

Bash

Cd /path/to/directory

Output:

(No output if successful. Use pwd to verify the change.)

3. ls

Purpose: List files and directories in the current directory.

Command:

Bash

Ls

Output:

File1.txt file2.txt directory1 directory2

(Lists the contents of the current directory.)

4.mkdir

Purpose: Create a new directory.

Command:

Bash

Mkdir new\_directory

Output:

(No output if successful. Use ls to verify the creation.)

5. rm

Purpose: Remove files or directories.

Command:

Bash

Rm file.txt

Output:

(No output if successful. Use ls to verify the deletion.)

6.touch

Purpose: Create an empty file or update the timestamp of an existing file.

Command:

Bash

Touch newfile.txt

Output:

(No output if successful. Use ls to verify the creation.)

7.hostname

Purpose: Display the system’s hostname.

Command:

Bash

Hostname

Output:

Ubuntu

(Displays the name of the system.)

8.cat

Purpose: Display the contents of a file.

Command:

Bash

Cat file.txt

Output:

This is the content of file.txt.

(Displays the file’s content.)

9. chmod

Purpose: Change file permissions.

Command:

Bash

Chmod 755 script.sh

Output:

(No output if successful. Use ls -l to verify the permissions.)

10. echo

Purpose: Display a message or write to a file.

Command:

Bash

Echo “Hello, World!”

Output:

Hello, World

(Displays the message.)

11. grep

Purpose: Search for a pattern in a file.

Command:

Bash

Grep “pattern” file.txt

Output:

This line contains the pattern.

(Displays lines containing the pattern.)

12. fgrep

Purpose: Search for fixed strings in a file.

Command:

Bash

Fgrep “fixed\_string” file.txt

Output:

This line contains the fixed\_string.

(Displays lines containing the fixed string.)

13. mv

Purpose: Move or rename files/directories.

Command:

Bash

Mv oldfile.txt newfile.txt

Output:

(No output if successful. Use ls to verify the move/rename.)

14. cp

Purpose: Copy files or directories.

Command:

Bash

Cp file.txt copyfile.txt

Output:

(No output if successful. Use ls to verify the copy.)

15. more

Purpose: View file content page by page.

Command:

Bash

More largefile.txt

Output:

Content of largefile.txt (displayed one page at a time).

(Displays the file content interactively.)

16.less

\*Purpose:\* View file content with backward navigation.

Command:

Bash

Less largefile.txt

Output:

Content of largefile.txt (with navigation support).

(Displays the file content interactively.)

17. wc

Purpose: Count lines, words, and characters in a file.

Command:

Bash

Wc file.txt

Output:

10 50 300 file.txt

(Displays the number of lines, words, and characters.)

18. awk

Purpose: Process and analyze text files.

Command: bash

Awk ‘{print $1}’ file.txt

Output:

FirstColumnValue1

FirstColumnValue2

(Displays the first column of each line.)

19. sed

Purpose: Stream editor for text manipulation.

Command:

Bash

Sed ‘s/old/new/’ file.txt

Output:

This is the new content.

(Replaces “old” with “new” in the file.)

20. tail

Purpose: Display the last part of a file.

Command:

Bash

Tail file.txt

Output:

Last 10 lines of file.txt.

(Displays the last 10 lines by default.)

**PART 2**

1. How to navigate to a specific directory?

Use the cd command followed by the directory path : bash

cd /path/to/directory

2.How to see detailed information about files and directories using ls?

Use the -l option with ls: bash

ls -la

For more details, including hidden files, use:

bash

ls -la

3.How to create multiple directories in Linux using mkdir command?

Use the mkdir command with directory names separated by spaces:bash

mkdir dir1 dir2 dir3

To create nested directories, use the -p option : bash

mkdir -p parent/child/grandchild

4.How to remove multiple files at once with rm?

Specify the filenames separated by spaces: bash

rm file1.txt file2.txt file3.txt

5.Can rm be used to delete directories?

Yes, but you need to use the -r (recursive) option:

bash

rm -r directory\_name

6.How Do You Copy Files and Directories in Linux?

Use the cp command:

- To copy a file:

bash

cp source\_file destination\_file

- To copy a directory (recursively):

bash

cp -r source\_directory destination\_directory

7.How to Rename a file in Linux Using mv Command

Use the mv command:

bash

mv old\_filename new\_filename

8.How to Move Multiple files in Linux Using mv Command

Specify the files and the destination directory:

bash

mv file1.txt file2.txt file3.txt /destination\_directory/

9.How to Create Multiple Empty Files by Using touch Command in Linux

Use the touch command with filenames separated by spaces:

bash

touch file1.txt file2.txt file3.txt

10.How to View the Content of Multiple Files in Linux

Use the cat command:

bash

cat file1.txt file2.txt file3.txt

11.How to Create a file and add content in Linux Using cat Command

Use cat with output redirection:

bash

cat > newfile.txt

Type the content, then press Ctrl+D to save and exit.

12.How to Append the Contents of One File to the End of Another File using cat

Use cat with the >> operator:

bash

cat file1.txt >> file2.txt

13.How to use cat command if the file has a lot of content and can’t fit in the terminal.

Use a pager like less or more:

bash

cat largefile.txt | less

14.How to Merge Contents of Multiple Files Using cat Command

Use cat with output redirection:

bash

cat file1.txt file2.txt > mergedfile.txt

15.How to use cat Command to Append to an Existing File

Use cat with the >> operator:

bash

cat new\_content.txt >> existing\_file.txt

16.What is “chmod 777”, “chmod 755”, and “chmod +x” or “chmod a+x”?

- chmod 777: Grants read, write, and execute permissions to everyone (owner, group, others).

- chmod 755: Grants full permissions to the owner and read/execute permissions to group and others.

- chmod +x or chmod a+x: Adds execute permission for all users.

17.How to find the number of lines that matches the given string/pattern

Use grep with the -c option:

bash

grep -c "pattern" filename

18.How to display the files that contain the given string/pattern

Use grep with the -l option:

bash

grep -l "pattern"

19.How to show the line number of file with the line matched

Use grep with the -n option:

bash

grep -n "pattern" filename

20.How to match the lines that start with a string using grep

Use the ^ anchor:

bash

grep "^pattern" filename

21.Can the sort command be used to sort files in descending order by default?

No, by default sort sorts in ascending order. Use the -r option for descending order:

bash

sort -r filename

22.How can I sort a file based on a specific column using the sort command?

Use the -k option to specify the column number:

bash

sort -k 2 filename

Replace 2 with the desired column number.