Assignment 6
Linux Programming

Name : Yathish N R

Class: 3B CY

Roll No: 66

USN : ENG25CY1007

1. **Which command is used to list the contents of a directory? Justify with proper example**

The command used to list the contents of a directory is **ls**.

Justification and Example:

The ls command is short for **"list"** and is fundamental for navigation. By default, it lists files and directories in the current working directory.
Command: ls -lF

Example Output:
total 4
-rw-r--r-- I user group 0 Sep I I0:00 notes.txt drwxr-xr-x 2 user group 4096 Sep I I0:05 reports/
Justification: The options used (-l for long listing format, -F to append indicators like / for directories) provide detailed and easily readable output.

2. **Write the command to create a new directory named I23test_dir.**

The command used to create a new directory is **mkdir** (make directory).
- **Command:**

mkdir I23test_dir

3. **What is the purpose of the sed command? Justify with proper example.**

The purpose of the **sed** (Stream Editor) command is to **filter and transform text** from an input stream (like a file or piped output). It is most commonly used for **substituting** (find and replace) text based on regular expressions.

Example: Replacing "error" with "warning" in a log file.

Syntax: sed 's/pattern/replacement/g' <filename> sed 's/error/warning/g' application.log

- **Justification:** This command reads application.log, globally (g) substitutes every occurrence of the string "error" with "warning", and prints the result to standard output. (The original file remains unchanged unless redirected).

4. **Which distinct command is used to display one-line descriptions of any commands.**

The distinct command used to display one-line descriptions of any command is **whatis**.
- **Justification:** The whatis command searches the manual page (man page) database for the exact command name and displays the brief description found in the NAME section of its man page. It is equivalent to using man -f.

5. Write the command to create an empty file named "notes.txt".

The simplest command to create an empty file is **touch**.

Command:

touch notes.txt

Justification: If notes.txt does not exist, touch creates it as an empty file. If it already exists, it merely updates the file's access and modification timestamps.

6. Differentiate between grep and awk commands with an example

Feature	**grep** (Global Regular Expression Print)	**awk** (Aho, Weinberger, and Kernighan)
Primary Purpose	**Filtering** lines that match a specific pattern (regular expression).	**Processing and reporting** based on fields/columns in a text file.
Input Unit	Operates on entire **lines** of text.	Operates on lines, but processes data as **records** and **fields** (columns).
Complexity	Simple, fast, and focused on pattern matching.	Powerful scripting language capable of arithmetic, control flow (if/else), and complex logic.

Example:

Task	grep Command	awk Command
Find Users	Find all lines (users) containing "bash" shell: grep 'bash' /etc/passwd	
Print Users		Print *only* the username (1st field) and shell (7th field) for users using 'bash': awk -F: '$7 ~ /bash/ {print $1, $7}' /etc/passwd

7. Write the command to give read, write, and execute permission to the owner of A file script.sh.

The command used is **chmod** (change mode), utilizing the symbolic mode to specify the permissions.

Command:

chmod u=rwx script.sh

Justification:

- u: Specifies the **user** (owner).
- =rwx: **Sets** the exact permissions: Read, Write, and Execute.

8. **How is chown different from chgrp? Give one example for each.**

Both commands manage file ownership, but they affect different entities:

chown (Change Owner): Used to change the **user ownership** of a file or directory.

Example: Change the owner of data.txt to the user alice. chown alice data.txt

chgrp (**Change Group**)**:** Used to change the **group ownership** of a file or directory.
Example: Change the group of data.txt to the group staff. chgrp staff data.txt

9. A user complains that they cannot execute a file even though it exists in their directory. How would you troubleshoot this using ls -l, chmod, and whoami?

If a user cannot execute a file, the problem is almost certainly related to permissions.

Step	Command	Purpose in Troubleshooting
1. Identify User	whoami	Confirms the username of the person executing the command to determine which permission column (Owner, Group, or Others) applies.
2. Check Permissions	ls -l script.sh	Displays the long listing, showing the file's current permissions, its owner, and its group. The execution bit (x) must be set for the user category identified in Step 1.
3. Fix Permissions	chmod +x script.sh	If the execution bit is missing, this command (or chmod 700 if the user is the owner) adds the execution permission for *all* categories, allowing the user to run the file.

10. Design a command pipeline to: find all .log files modified in the last 2 days in /var/log, display them on screen, and save the results into a file recent_logs.txt using tee command. tee

This pipeline uses find to locate the files, pipes the results to tee to print them, and simultaneously saves the output.

find /var/log -type f -mtime -2 -name "*.log" | tee recent_logs.txt
find /var/log: Starts searching in the /var/log directory. ● **-type f**: Only look for **files**.
-mtime -2: Find files modified in the **last 2 days**.
-name "*.log": Match files ending with .log.
|: Pipes the list of filenames to the next command.
tee recent_logs.txt: Reads the input, **displays it on the screen** (Standard Output), and **saves the identical output** to the file recent_logs.txt.