Name= shaik Raakin
Class =3rd sem
Branch = cybersecurity
usn=ENG24CY0192
Roll no:64
section:A
Assignment : 6

1Q)Which command is used to list the contents of a directory? Justify with proper example.
sol)
The ls command is the standard utility to display the contents of directories in Linux. It shows files, folders, and other items in the specified directory.
will list everything in /home/user/Documents. You can use options like ls -l for detailed info or ls -a to see hidden files.
Example
raakin@DESKTOP-0FFMSPJ:~$ ls
newc output.txt snap uni_linx

2Q)Write the command to create a new directory named 123test_dir.
sol)
To create a new directory, use the mkdir command.
Example:
raakin@DESKTOP-0FFMSPJ:~$ ls
123test_dir newc output.txt snap uni_linx

3Q)What is the purpose of the sed command? Justify with proper example.
sol)
Sed (stream editor) is used for parsing and transforming text in Linux. It works efficiently with large files or input streams.
Example
sed 's/old/new/g' file.txt
replaces every occurrence of “old” with “new” in file.txt. It’s used widely for text substitution, deletion, insertion, and more, making it essential for automation and scripting.

4Q)Which distinct command is used to display one-line descriptions of any commands?
sol)
The whatis command is used to display one line description for all commands

Example
raakin@DESKTOP-0FFMSPJ:~$ whatis mkdir

Output
raakin@DESKTOP-0FFMSPJ:~$ whatis mkdir
mkdir (1) - make directories
mkdir (2) - create a directory

5Q)Write the command to create an empty file named “notes.txt”
sol)
The command “touch” is used to create an empty file named “notes.txt “
Example
raakin@DESKTOP-0FFMSPJ:~$ ls
123test_dir newc notes.txt output.txt snap uni_linx

6Q)Differentiate between grep and awk commands with an example
sol)
grep is a command-line tool for searching plain-text data for lines matching a specific pattern.
Example
raakin@DESKTOP-0FFMSPJ:~$ grep "bytes" output.txt
PING www.google.com (142.250.67.228) 56(84) bytes of data.
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=1 ttl=118 time=73.1 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=2 ttl=118 time=30.6 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=3 ttl=118 time=41.9 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=4 ttl=118 time=31.2 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=5 ttl=118 time=65.3 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=6 ttl=118 time=29.0 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=7 ttl=118 time=96.8 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=8 ttl=118 time=60.7 ms

64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=9 ttl=118 time=48.3 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=10 ttl=118 time=30.2 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=11 ttl=118 time=30.9 ms
64 bytes from bom07s24-in-f4.1e100.net (142.250.67.228): icmp_seq=12 ttl=118 time=59.9 ms
finds all lines containing “bytes”

awk is a powerful programming language and command-line tool for processing and analyzing text, especially columnar data.
Example:
raakin@DESKTOP-0FFMSPJ:~$ awk '{print $2}' output.txt
www.google.com
bytes
bytes
bytes
bytes
bytes
bytes
bytes
bytes
bytes
bytes
bytes
bytes
prints the second column of every line. Unlike grep, awk can process fields, perform arithmetic, and handle more complex text processing.

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

sol)

By using the command chmod:
Example:
chmod u+rwx script.sh
Here, U stands for user (owner), and +rwx gives read, write, and execute permissions.

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

sol)

chown is used to change the owner (user) of a file or directory, and it can also change its group at the same time.
Example
chown alice file.txt

Chrgp changes only the group associated with a file or directory.
Example
chgrp developers file.txt
changes the group to developers

9Q)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?
sol)
First, start by examining if execute permission is set on the file by typing ls -l. If execute permission is not set, you can add execute permission by entering chmod +x filename. Next, use whoami to confirm that the user has ownership of the file or is a member of a group that has execute permission. If user or group permissions are incorrectly set, any of the three types of permission can be modified by using chmod, chown, or chgrp.

10Q)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.
sol)
To achieve this, combine find and tee.
Example
find /var/log -name "*.log" -mtime -2 | tee recent_logs.txt
Here, find locates .log files that were modified in the last 2 days, tee shows the results on screen and writes them to recent_logs.txt simultaneously.
Example
raakin@DESKTOP-0FFMSPJ:~$ cat recent_logs.txt
/var/log/apt/history.log
/var/log/apt/term.log
/var/log/unattended-upgrades/unattended-upgrades-dpkg.log
/var/log/unattended-upgrades/unattended-upgrades.log
/var/log/fontconfig.log
/var/log/auth.log
/var/log/alternatives.log
/var/log/kern.log
/var/log/dpkg.log