1. User Identity Verification

Command:

```
whoami groups
```

Explanation:
This command displays my currently logged-in username and all Linux groups associated with my user account.

---

2. Workspace Validation

Command:

```
pwd ls -l
```

Explanation:
`pwd` shows the present working directory and `ls -l` lists all files and directories with permissions, ownership, and size.

---

3. Environment Confirmation File

Command:

```
echo "Linux user environment verified" > user_info.txt
```

Explanation:
This command creates a file named `user_info.txt` and writes the verification message into it.

---

4. File Integrity Check

Command:

```
wc -c user_info.txt
```

Explanation:
This command displays the total number of characters present in the file.

---

5. Learning the Tools

Command:

```
man mdir
```

Explanation:
One useful option is `-b`, which displays files in bare format without extra information, useful for scripting.

---

6. Home Directory Inspection

Command:

```
ls ~ | sort
```

Explanation:
This lists all files in the home directory and sorts them alphabetically.

---

7. Log Investigation

Command:

```
grep "admin" log.txt
```

Explanation:
This command searches for the word “admin” in `log.txt` and displays only matching lines.

---

8. System Information Check

Command:

```
uname -r
```

Explanation:
This command displays the Linux kernel version currently running.

---

9. Network Connectivity Test

Command:

```
ping -c 4 www.google.com
```

Explanation:
This command verifies network connectivity by sending ICMP packets to Google.

---

10. System Health Awareness

Command:

```
uptime
```

Explanation:
This command shows how long the system has been running, number of logged-in users, and system load averages.

QUESTION 2 – File & Directory Management

1. Project Workspace Setup

Command:

```
mkdir ~/documents
```

Explanation:
Creates a directory named `documents` inside the home directory.

---

2. File Creation

Command:

```
cd ~/documents touch plan.txt
```

Explanation:
Creates an empty file named `plan.txt`.

---

3. Content Addition

Command:

```
echo "Complete project documentation by Friday." > plan.txt
```

Explanation:
Adds sample project text into the file.

---

4. File Metadata Verification

Command:

```
ls -l plan.txt
```

Explanation:
Displays permissions, ownership, file size, and modification time.

---

5. File Duplication

Command:

```
cp plan.txt plan_copy.txt
```

Explanation:
Creates a duplicate copy of the original file.

---

6. Directory Renaming

Command:

```
cd ~ mv documents project_documents
```

Explanation:
Renames the directory to better reflect project usage.

---

7. Archival Structure

Command:

```
mkdir project_documents/archive
```

Explanation:
Creates a subdirectory for archived files.

---

8. File Organization

Command:

```
mv project_documents/plan_copy.txt project_documents/archive/
```

Explanation:
Moves the copied file into the archive directory.

---

9. Recursive Listing

Command:

```
ls -R project_documents
```

Explanation:
Displays the complete directory structure recursively.

---

10. Path Verification

Command:

```
realpath project_documents/archive/plan_copy.txt
```

Explanation:
Shows the absolute path of the moved file.

QUESTION 3 – Links & Disk Usage

1. File Creation

Command:

```
echo "Sample data for link testing" > ~/sample_data.txt
```

Explanation:
Creates a file with sample text content.

---

2. Hard Link Creation

Command:

```
ln sample_data.txt sample_hard.txt
```

Explanation:
Creates a hard link pointing to the same inode.

---

3. Symbolic Link Creation

Command:

```
ln -s sample_data.txt sample_soft.txt
```

Explanation:
Creates a symbolic link pointing to the original file.

---

4. Inode Verification

Command:

```
ls -i sample_data.txt sample_hard.txt sample_soft.txt
```

Explanation:
Displays inode numbers of all three files.

---

5. Inode Analysis

Explanation:
`sample_data.txt` and `sample_hard.txt` share the same inode because hard links reference the same file data.

---

6. File Metadata Inspection

Command:

```
ls -l sample_data.txt
```

Explanation:
Displays permissions, owner, size, and timestamps.

---

7. Disk Usage Check

Command:

```
du -sh ~
```

Explanation:
Shows total disk usage of the home directory.

---

8. File Size Overview

Command:

```
ls -lh ~
```

Explanation:
Lists file sizes in human-readable format.

---

9. Link Deletion Test

Command:

```
rm sample_soft.txt ls sample_data.txt
```

Explanation:
Deleting the symbolic link does not affect the original file.

---

10. Disk Utility Demonstration

Command:

```
df -h du -ah ~ | head
```

Explanation:
`df` shows filesystem usage, while `du` displays directory-wise disk usage.

QUESTION 4 – System Monitoring

1. System Uptime Verification

Command:

```
uptime
```

Explanation:
Displays system running time since last boot.

---

2. User Process Listing

Command:

```
ps -u $USER
```

Explanation:
Lists all processes owned by the current user.

---

3. CPU Usage Analysis

Command:

```
top
```

Explanation:
Displays real-time CPU usage; highest consumer appears at the top.

---

4. Background Process Execution

Command:

```
sleep 300 & jobs
```

Explanation:
Runs a command in the background and confirms it is running.

---

5. Process Priority Management

Command:

```
renice 10 -p <PID>
```

Explanation:
Changes the priority of a running process.

---

6. Memory Usage Monitoring

Command:

```
free -h
```

Explanation:
Displays system memory usage in human-readable format.

---

7. Disk Space Inspection

Command:

```
df -h ~
```

Explanation:
Shows disk usage of the filesystem containing the home directory.

---

8. Shell Identification

Command:

```
echo $SHELL
```

Explanation:
Displays the currently used shell.

---

9. Output Redirection

Command:

```
uname -a > system_report.txt
```

Explanation:
Redirects system information into a file.

---

10. Disk Usage Visualization

Command:

```
nodu ~
```

Explanation:
Displays a visual, interactive representation of disk usage.