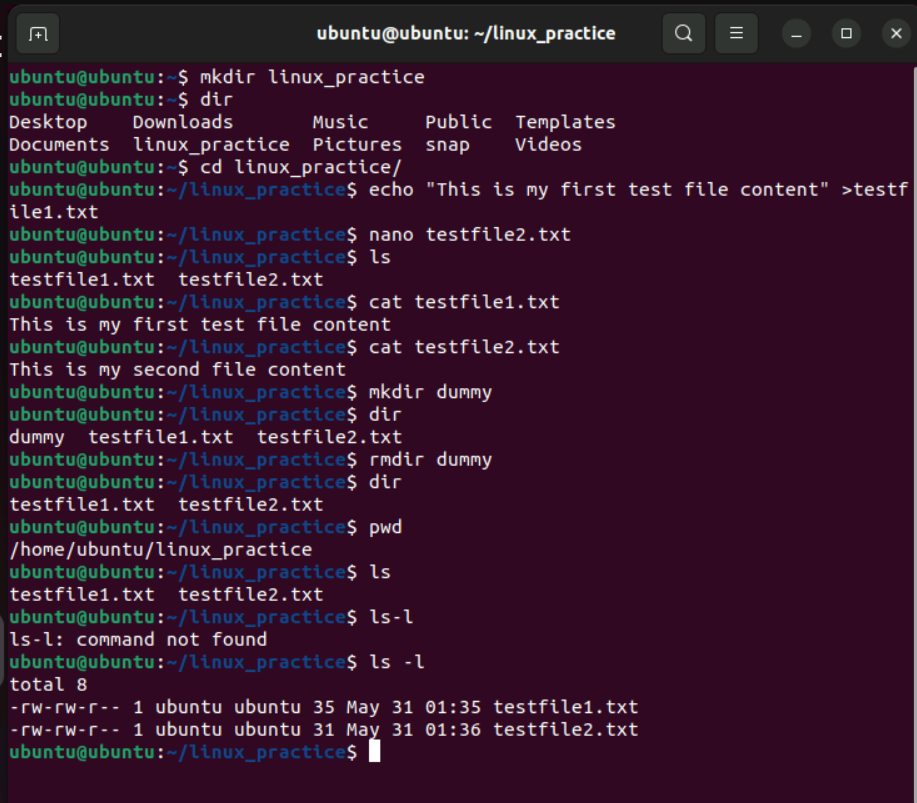
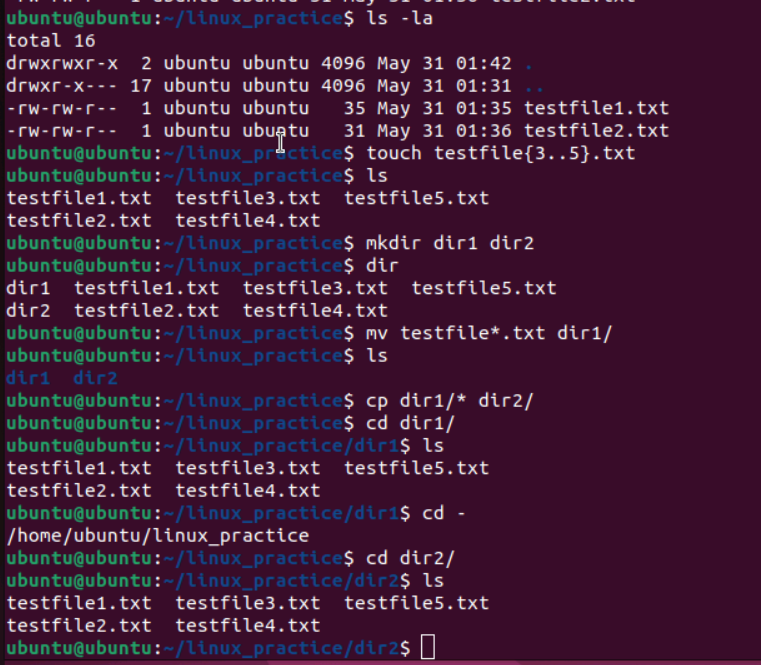
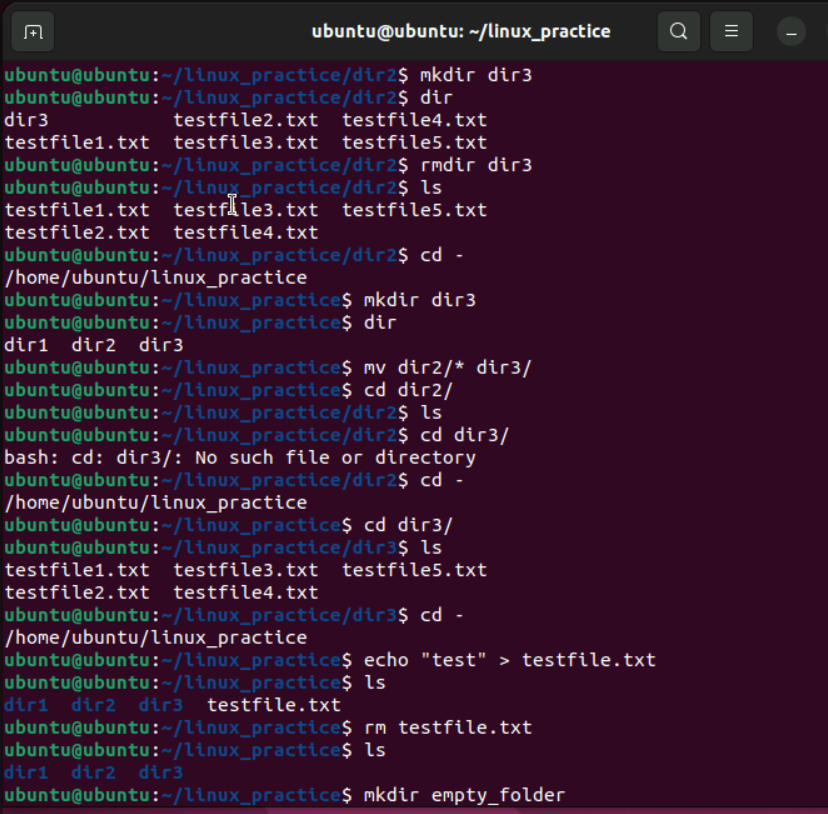
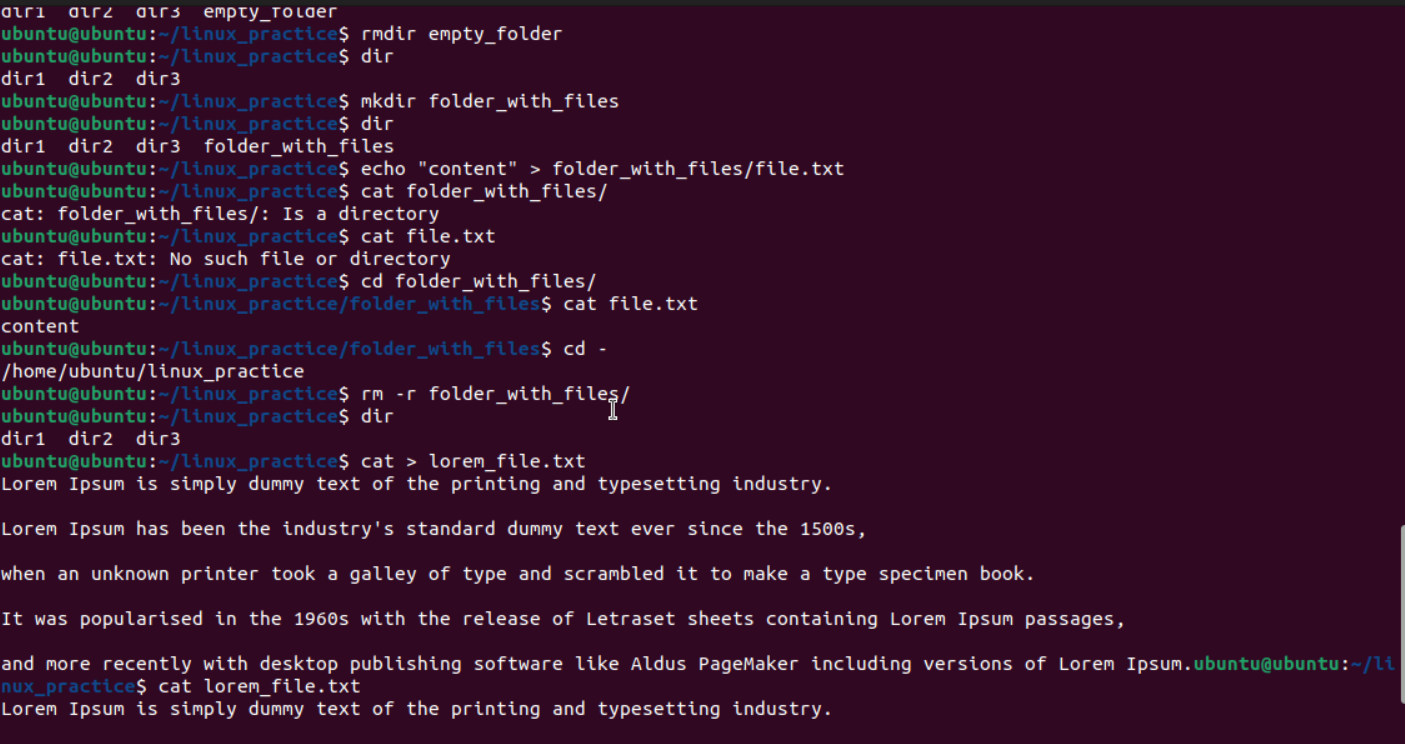
**Day 5**: **28th May 2025**

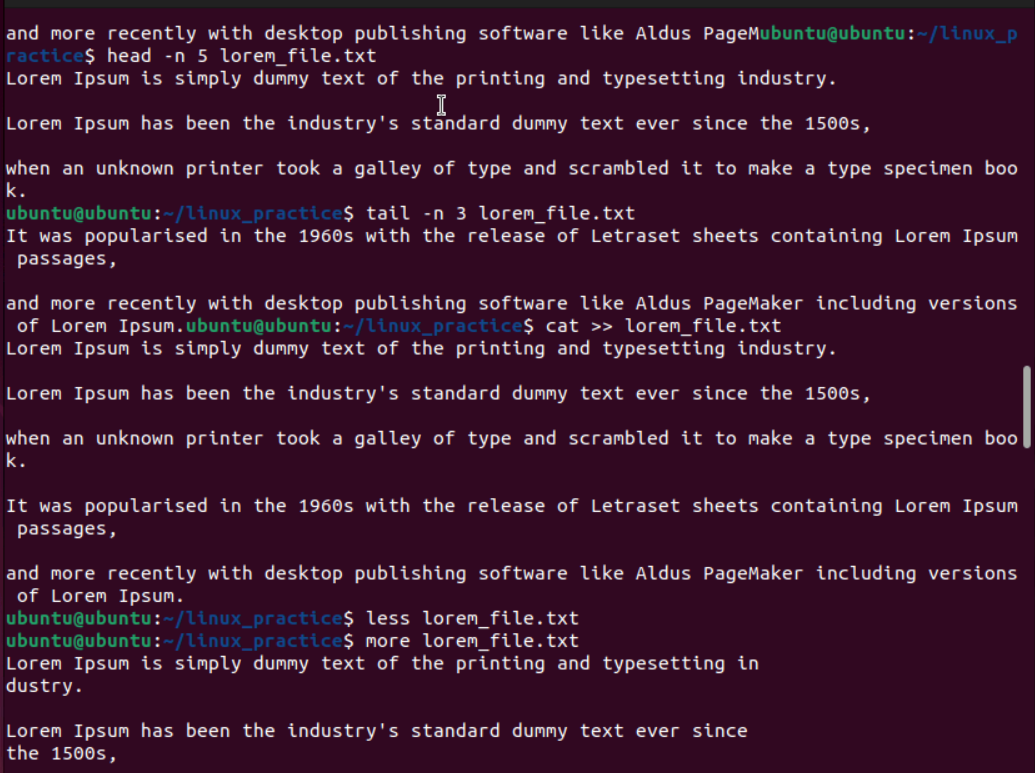
**Linux Fundamentals:**

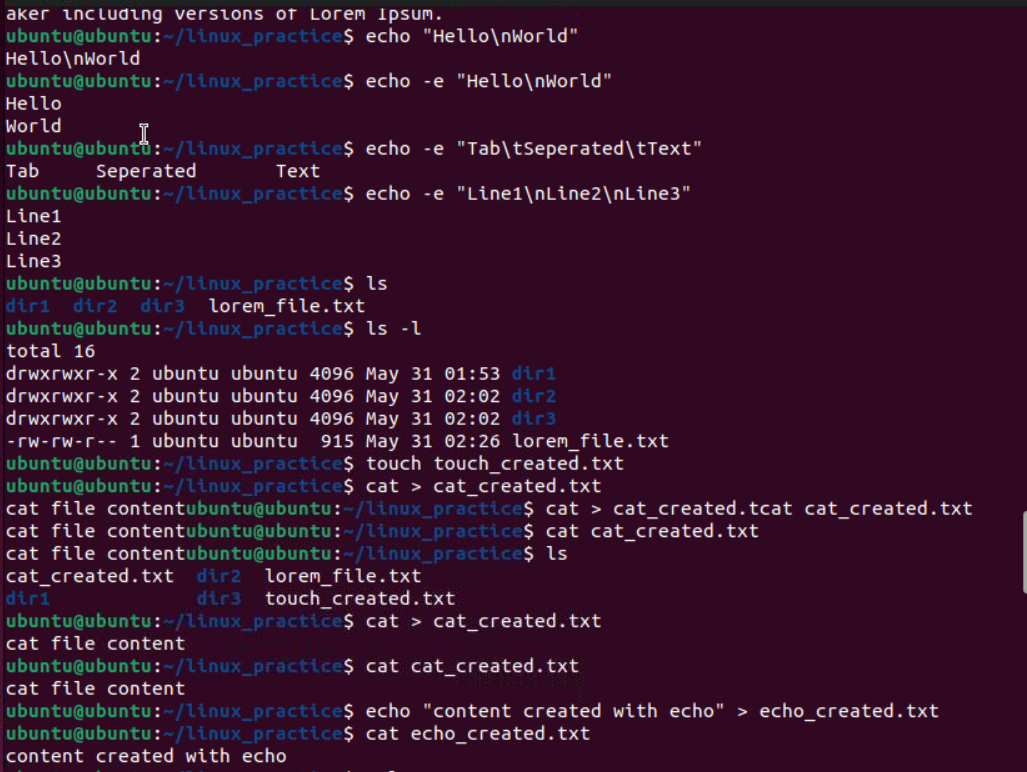
****

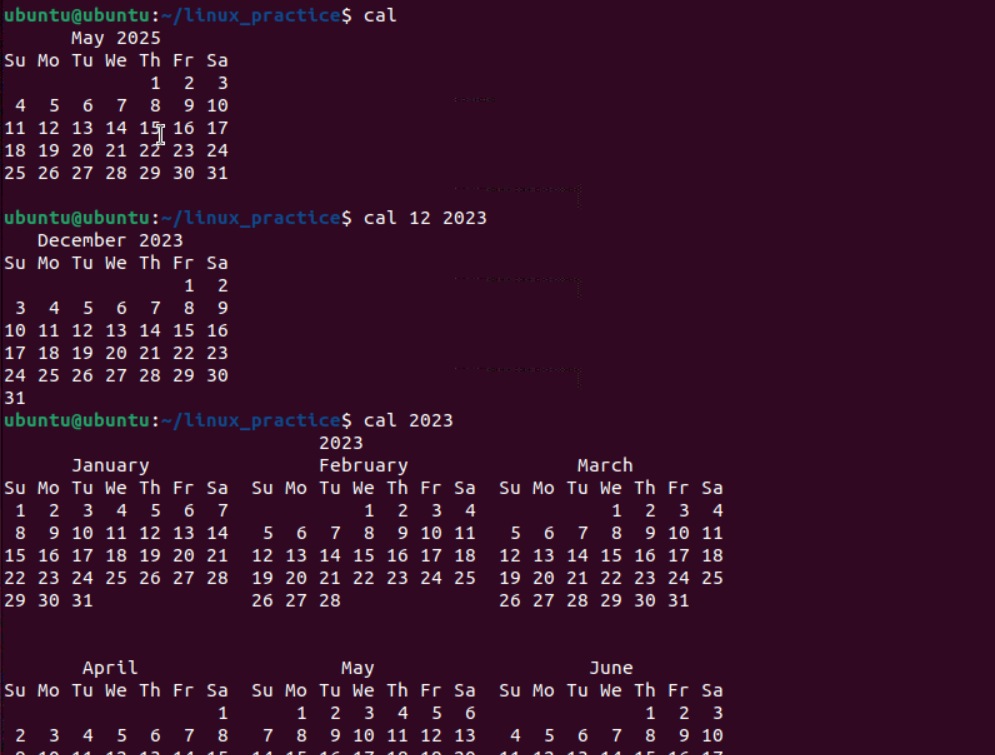


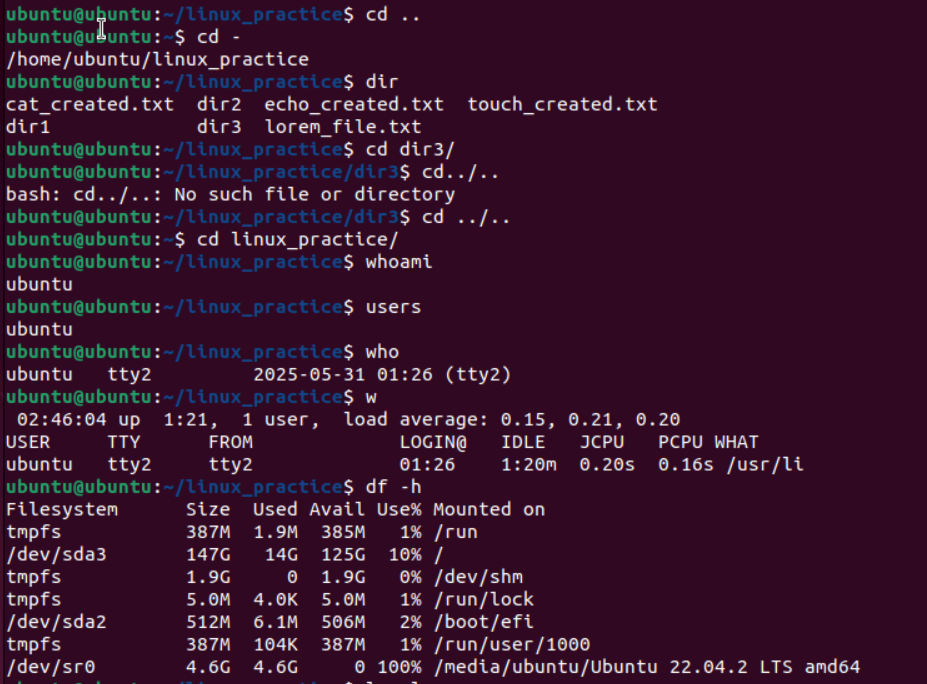


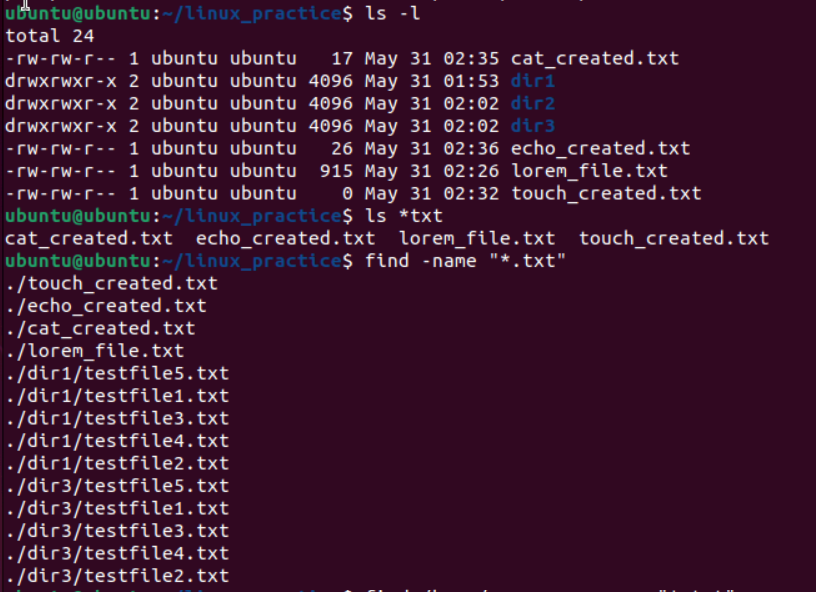


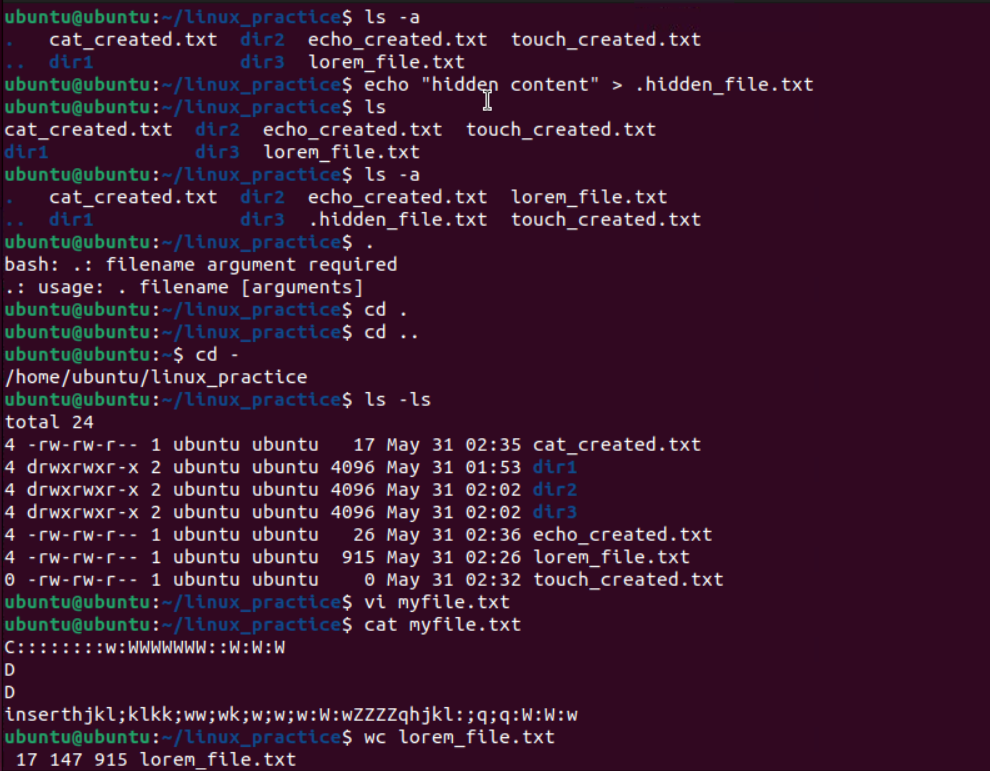


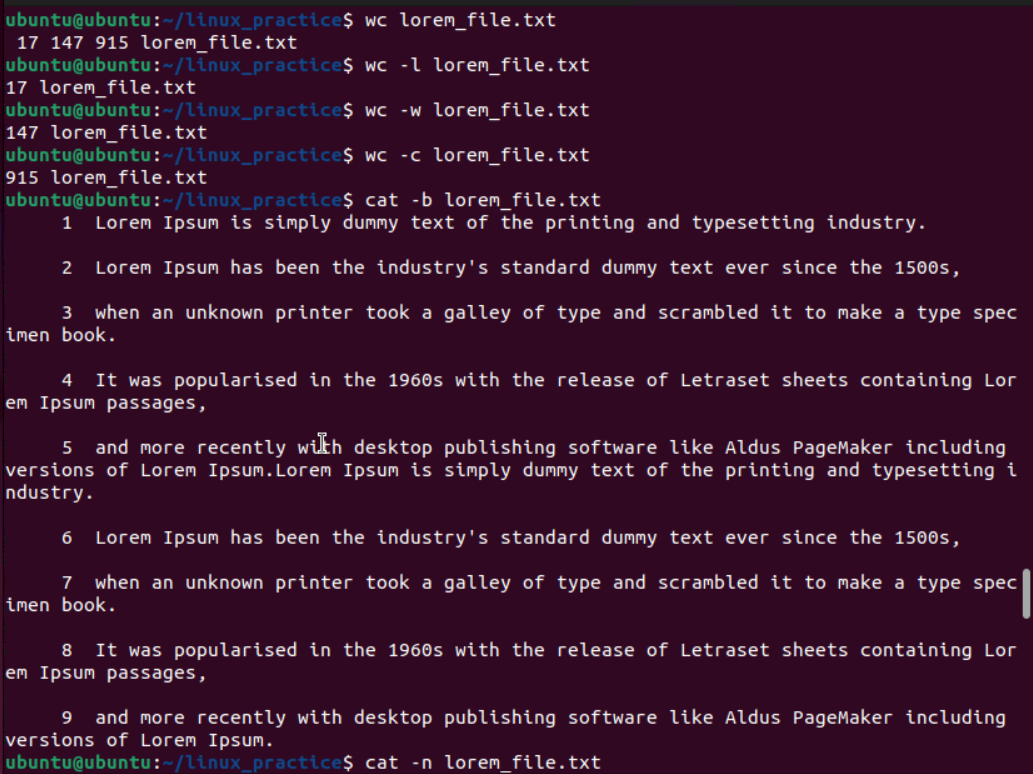


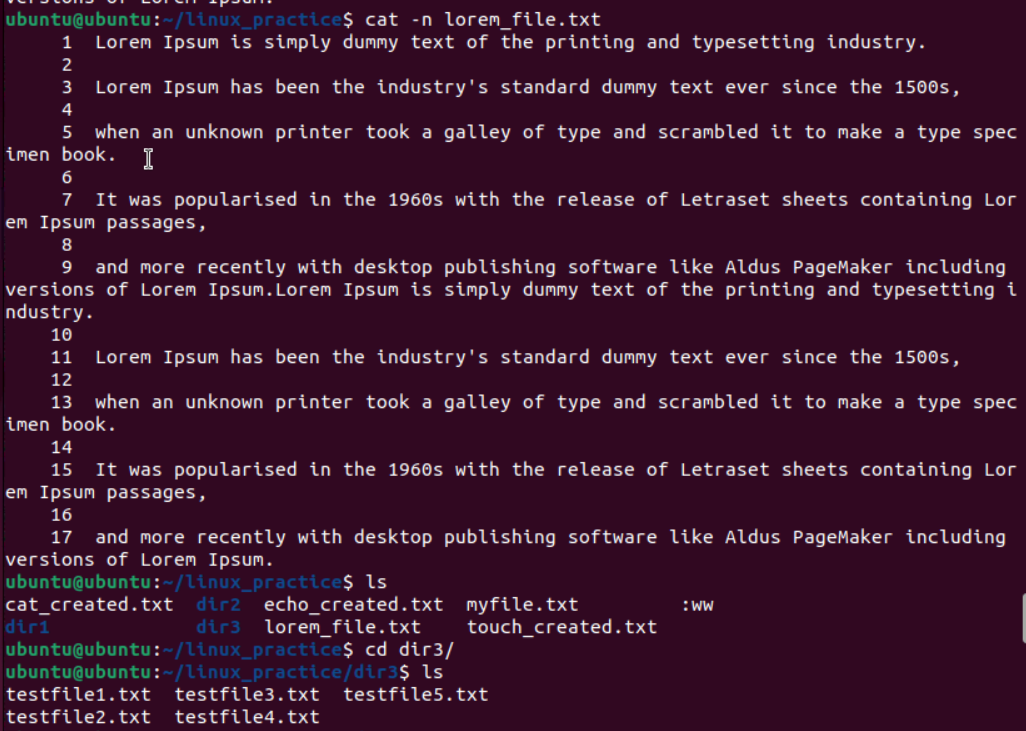


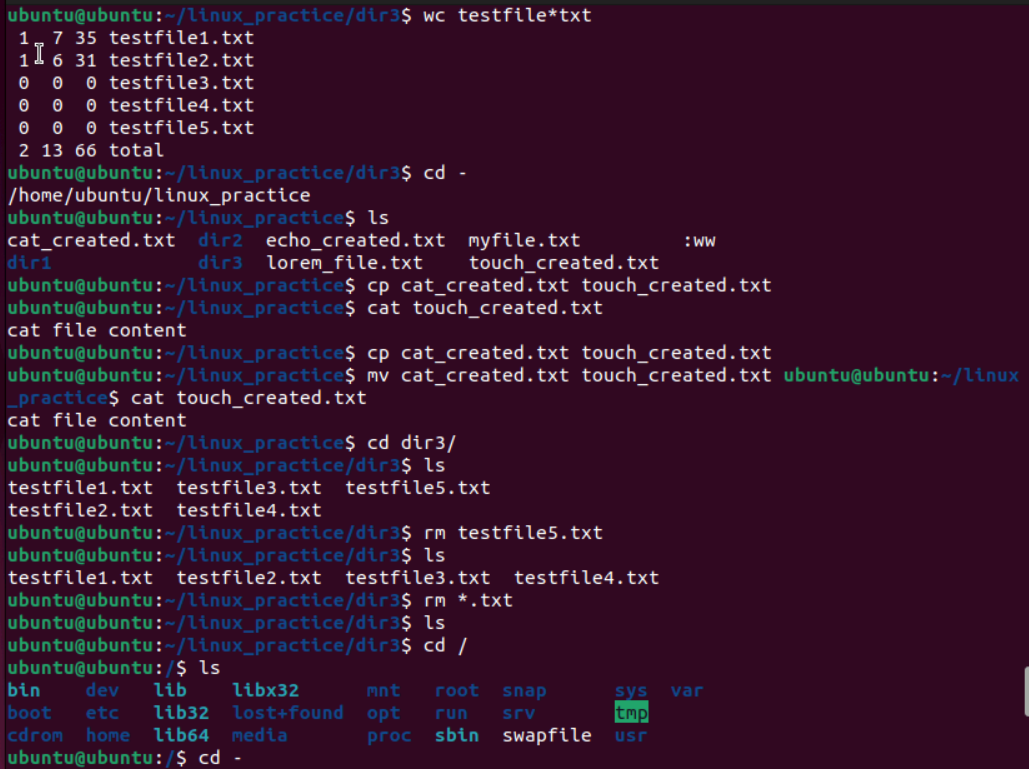


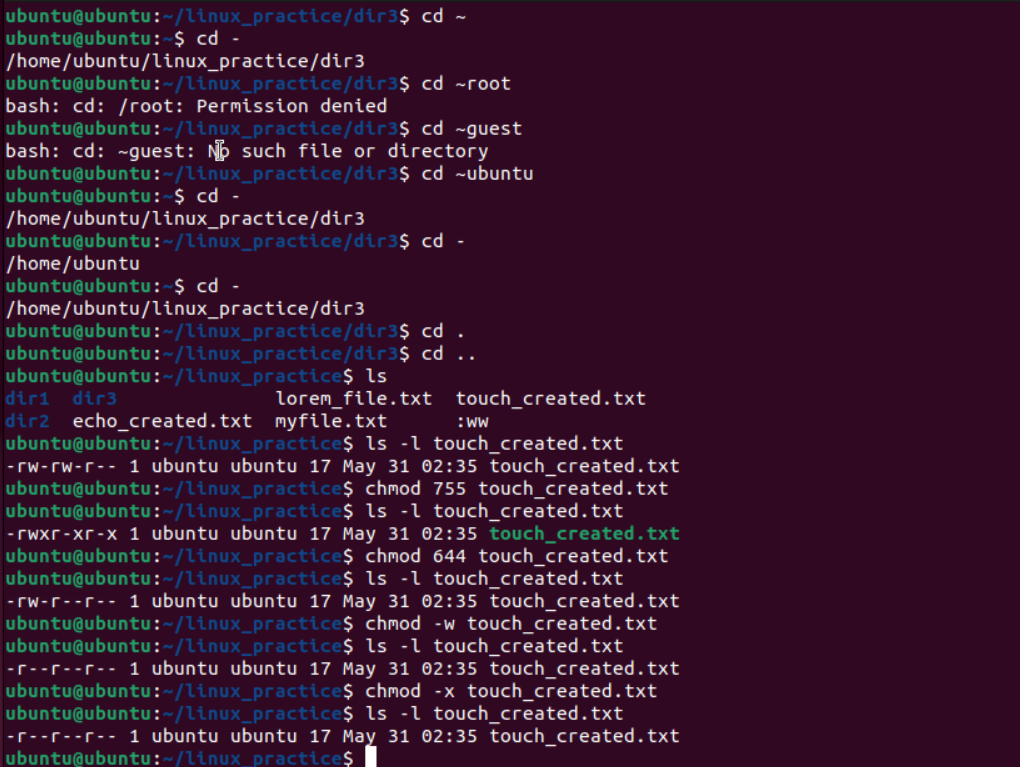
****











**Task 1: Create Directory**

**Command: mkdir**

**What it does:** Creates a new directory/folder  
**Usage:** mkdir directory\_name

mkdir "Linux Practice"

# OR (without spaces, easier to work with)

mkdir Linux\_Practice

**Note:** Use quotes for names with spaces, or use underscore/dash instead

**Task 2: Change Directory**

**Command: cd**

**What it does:** Changes current working directory  
**Usage:** cd directory\_name

cd "Linux Practice"

# OR

cd Linux\_Practice

**Try:** Use Tab key for auto-completion!

**Task 3: Create File and Add Content**

**Command: nano or echo >**

**What it does:** Creates file and adds content

# Method 1: Using nano editor

nano TestFile1.txt

# Type your content, then Ctrl+X, Y, Enter to save

# Method 2: Using echo

echo "This is my first test file content" > TestFile1.txt

**Try both methods** and see which you prefer!

**Task 4: Create and Delete Folder**

**Commands: mkdir and rmdir**

**What it does:** Creates then removes empty directory

# Create folder

mkdir Dummy

# Delete empty folder

rmdir Dummy

# If folder has files, use:

# rm -r Dummy

**Note:** rmdir only works on empty directories!

**Task 5: Check Working Directory**

**Command: pwd**

**What it does:** Prints current working directory path

pwd

**Output example:** /home/username/Linux\_Practice

**Task 6: List Files and Directories**

**Command: ls**

**What it does:** Lists all files and directories in current location

ls

# For detailed view:

ls -l

# For all files including hidden:

ls -la

**Task 7: Create Multiple Files**

**Command: touch**

**What it does:** Creates empty files

# Method 1: One by one

touch TestFile2.txt TestFile3.txt TestFile4.txt TestFile5.txt TestFile6.txt

# Method 2: Using brace expansion

touch TestFile{2..6}.txt

**Try both methods!** The second is faster for sequential files.

**Task 8: Copy Files Between Directories**

**Command: cp**

**What it does:** Copies files from one location to another

# First create directories

mkdir Dir1 Dir2

# Move some files to Dir1

mv TestFile\*.txt Dir1/

# Copy all files from Dir1 to Dir2

cp Dir1/\* Dir2/

**Note:** \* is a wildcard meaning "all files"

**Task 9: Move Files**

**Command: mv**

**What it does:** Moves files (cuts and pastes)

# Create Dir3

mkdir Dir3

# Move all files from Dir2 to Dir3

mv Dir2/\* Dir3/

# Check Dir2 is empty

ls Dir2/

**Difference:** cp copies, mv moves (original location becomes empty)

**Task 10: Difference between rm and rmdir**

| **Command** | **Purpose** | **Example** | **What it removes** |
| --- | --- | --- | --- |
| rm | Remove files | rm file.txt | Files only |
| rm -r | Remove directories with content | rm -r folder/ | Folders with files |
| rmdir | Remove empty directories | rmdir emptyfolder/ | Empty folders only |

# Test rm

echo "test" > testfile.txt

rm testfile.txt

# Test rmdir

mkdir empty\_folder

rmdir empty\_folder

# Test rm -r

mkdir folder\_with\_files

echo "content" > folder\_with\_files/file.txt

rm -r folder\_with\_files

**Task 11: Create File with cat**

**Command: cat >**

**What it does:** Creates file and allows typing content

cat > lorem\_file.txt

# Now type your Lorem Ipsum text (2-3 paragraphs)

# Press Ctrl+D when finished to save and exit

**Example content to type:**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

**Task 12: View Top of File**

**Command: head**

**What it does:** Shows first 10 lines of file (by default)

head lorem\_file.txt

# Show specific number of lines

head -n 5 lorem\_file.txt

**Try different numbers** with -n option!

**Task 13: View Bottom of File**

**Command: tail**

**What it does:** Shows last 10 lines of file (by default)

tail lorem\_file.txt

# Show specific number of lines

tail -n 3 lorem\_file.txt

**Task 14: Add More Content and Use less**

**Commands: cat >> and less**

# Add more content to existing file

cat >> lorem\_file.txt

# Type 5-6 more paragraphs, then Ctrl+D

# View file page by page

less lorem\_file.txt

**In less:** Use Space for next page, q to quit, arrow keys to navigate

**Task 15: Compare less and more**

| **Command** | **Navigation** | **Features** |
| --- | --- | --- |
| less | Forward & backward | More features, search, better navigation |
| more | Forward only | Basic, shows percentage |

more lorem\_file.txt

# Press Space for next page, q to quit

less lorem\_file.txt

# Press Space (next), b (back), q (quit), / (search)

**Task 16: Echo with -e Option**

**Command: echo -e**

**What it does:** Enables interpretation of backslash escape sequences

# Without -e

echo "Hello\nWorld"

# With -e

echo -e "Hello\nWorld"

echo -e "Tab\tSeparated\tText"

echo -e "Line1\nLine2\nLine3"

**Common escape sequences:** \n (newline), \t (tab), \\ (backslash)

**Task 17: Difference between ls and ls -l**

# Simple list

ls

# Detailed list with permissions, size, date

ls -l

**ls -l shows:**

* File permissions
* Number of links
* Owner name
* Group name
* File size
* Last modification date
* File name

**Task 18: Create Files with Different Commands**

**Using touch**

touch touch\_created.txt

**Purpose:** Creates empty file or updates timestamp

**Using cat**

cat > cat\_created.txt

# Type content, then Ctrl+D

**Purpose:** Creates file with content

**Using echo**

echo "Content created with echo" > echo\_created.txt

**Purpose:** Creates file with single line content

**Task 19: Display Calendar**

**Command: cal**

**What it does:** Shows calendar

# Current month

cal

# Specific month/year

cal 12 2023

# Full year

cal 2023

**Task 20: Go Back One Directory**

**Command: cd ..**

**What it does:** Moves to parent directory

cd ..

# Go back multiple levels

cd ../../

**Task 21: Check Current User**

**Command: whoami**

**What it does:** Shows current username

whoami

**Task 22: Check System Users**

# Show logged-in users

users

# Show who is logged in with details

who

# Show detailed user activity

w

**Differences:**

* users: Simple list of usernames
* who: Users with login time and terminal
* w: Users with current activity and system load

**Task 23: Check Disk Space**

**Command: df -h**

**What it does:** Shows disk space usage in human-readable format

df -h

**Shows:** Filesystem, Size, Used, Available, Use%, Mounted on

**Task 24: File Type Prefixes**

**Command: ls -l**

**What it does:** Shows file types with prefixes

ls -l

# Create examples of different file types

touch regular\_file.txt # - (regular file)

mkdir directory\_example # d (directory)

ln -s regular\_file.txt link\_example # l (symbolic link)

**File Type Prefixes:**

* - Regular file
* d Directory
* l Symbolic link
* b Block device
* c Character device
* p Named pipe
* s Socket

**Task 25: Find Files Ending with .txt**

**Command: find or ls**

**What it does:** Locates files with specific patterns

# Using find

find . -name "\*.txt"

# Using ls with wildcard

ls \*.txt

# Find in specific directory

find /home/username -name "\*.txt"

**Task 26: Show Hidden Files**

**Command: ls -a**

**What it does:** Shows all files including hidden ones

ls -a

# Create a hidden file to test

echo "hidden content" > .hidden\_file.txt

ls -a

**Note:** Hidden files start with . (dot)

**Task 27: Difference between . and ..**

| **Symbol** | **Meaning** | **Example** |
| --- | --- | --- |
| . | Current directory | cp file.txt . (copy to current dir) |
| .. | Parent directory | cd .. (go to parent dir) |

# Show current and parent directory

ls -la

# You'll see . and .. in the listing

**Task 28: Using Vi Editor**

**Command: vi**

**What it does:** Opens Vi text editor

vi myfile.txt

**Vi Commands:**

* i - Insert mode (start typing)
* Esc - Exit insert mode
* :w - Save file
* :q - Quit
* :wq or Shift+ZZ - Save and quit
* h,j,k,l - Navigate left, down, up, right

**Practice:** Create a file, add content, save and exit

**Task 29: Count Words in File**

**Command: wc**

**What it does:** Counts lines, words, characters, and bytes

wc lorem\_file.txt

**Output columns:**

1. **Lines** - Number of lines
2. **Words** - Number of words
3. **Characters** - Number of characters
4. **Filename** - File name

# Specific counts

wc -l lorem\_file.txt # Lines only

wc -w lorem\_file.txt # Words only

wc -c lorem\_file.txt # Characters only

**Task 30: cat -b Command**

**Command: cat -b**

**What it does:** Shows file content with line numbers (non-empty lines only)

cat -b lorem\_file.txt

**Difference from cat -n:** -b numbers only non-empty lines, -n numbers all lines

**Task 31: wc with Multiple Files**

**Command: wc file1 file2 ...**

**What it does:** Shows word count for multiple files

wc TestFile1.txt TestFile2.txt TestFile3.txt

# Shows count for each file plus total

**Task 32: Copy File Content**

**Command: cp**

**What it does:** Copies content from one file to another

cp source\_file.txt destination\_file.txt

# Example

cp lorem\_file.txt lorem\_copy.txt

**Task 33: Rename File**

**Command: mv**

**What it does:** Renames/moves files

mv oldname.txt MYFILENEW.txt

# Example

mv lorem\_file.txt MYFILENEW.txt

**Note:** mv can both move and rename files

**Task 34: Delete Multiple Files**

**Command: rm**

**What it does:** Removes multiple files at once

# Delete specific files

rm file1.txt file2.txt file3.txt

# Delete all .txt files

rm \*.txt

# Interactive deletion (asks confirmation)

rm -i \*.txt

**Task 35: Root Directory**

**Command: cd /**

**What it does:** Changes to root directory (top of filesystem)

cd /

ls

**Linux Directory Structure:**

* /bin - Essential binaries
* /boot - Boot files
* /dev - Device files
* /etc - Configuration files
* /home - User home directories
* /lib - System libraries
* /media - Removable media
* /mnt - Mount points
* /opt - Optional software
* /proc - Process information
* /root - Root user's home
* /sbin - System binaries
* /tmp - Temporary files
* /usr - User programs
* /var - Variable data

**Task 36: Go to Home Directory**

**Command: cd ~**

**What it does:** Changes to user's home directory

cd ~

# OR simply

cd

**Task 37: Go to Different User's Home**

**Command: cd ~username**

**What it does:** Changes to specified user's home directory

cd ~root

cd ~ubuntu

cd ~guest

**Note:** You need appropriate permissions to access other users' directories

**Task 38: File Permissions (chmod)**

**Command: chmod**

**What it does:** Changes file permissions

# View current permissions

ls -l file.txt

# Permission format: rwxrwxrwx (owner, group, others)

# r=read(4), w=write(2), x=execute(1)

# Examples

chmod 755 file.txt # rwxr-xr-x

chmod 644 file.txt # rw-r--r--

chmod +x script.sh # Add execute permission

chmod -w file.txt # Remove write permission

**Common Permission Numbers:**

* 755 - rwxr-xr-x (executable files)
* 644 - rw-r--r-- (regular files)
* 777 - rwxrwxrwx (full permissions)
* 600 - rw------- (private files)

**Quick Command Reference**

| **Task** | **Command** | **Purpose** |
| --- | --- | --- |
| **Create** | mkdir, touch, cat > | Make directories/files |
| **Navigate** | cd, pwd, ls | Move around filesystem |
| **View** | cat, less, more, head, tail | Display file content |
| **Copy/Move** | cp, mv | Copy and move files |
| **Delete** | rm, rmdir | Remove files/directories |
| **Search** | find, ls \*.ext | Locate files |
| **Info** | wc, ls -l, whoami, df -h | Get file/system information |
| **Edit** | vi, nano | Text editors |