Question 3 (2024eb03003):

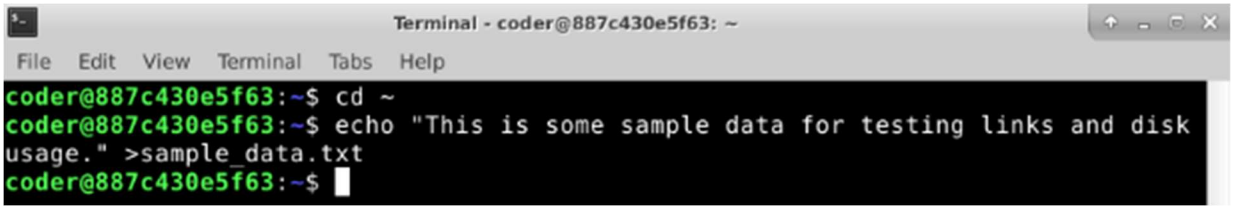
You have been asked to understand how Linux manages files using links and disk usage information.As part of your role, you will perform the following operations within your own user space.

1. File Creation

Create a file named sample_data.txt in your home directory and add some sample text to it.

cd ~ (cd ~ ensures we are in home directory)

echo "This is some sample data for testing links and disk usage." > sample_data.txt (echo "..." > sample_data.txt creates sample_data.txt and writes that sample line into it)

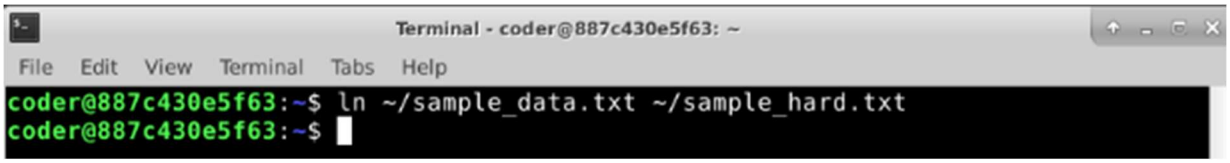


2. Hard Link Creation

Create a hard link to sample_data.txt named sample_hard.txt.

ln ~/sample_data.txt ~/sample_hard.txt

- The first path is the existing file (sample_data.txt).
- The second path is the new hard link name (sample_hard.txt) that points to the same inode and data as the original file.

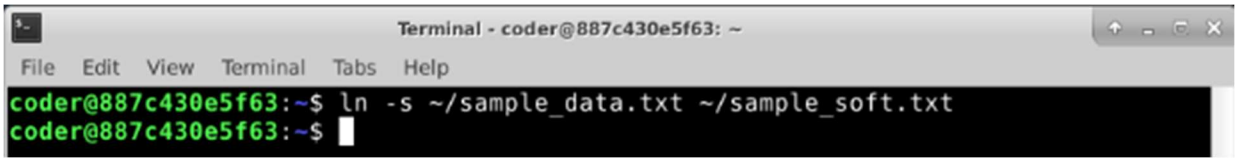


3. Symbolic Link Creation

Create a symbolic (soft) link to sample_data.txt named sample_soft.txt.

ln -s ~/sample_data.txt ~/sample_soft.txt

- -s tells ln to create a symbolic (soft) link instead of a hard link.
- sample_soft.txt will now point to sample_data.txt, acting like a shortcut to that file.

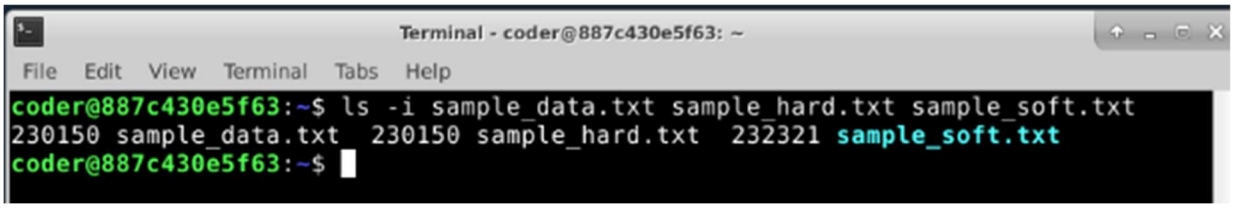


4. Inode Verification

Display the inode numbers of sample_data.txt, sample_hard.txt, and sample_soft.txt.

cd ~

ls -i sample_data.txt sample_hard.txt sample_soft.txt (This command prints the inode number as the first field, followed by each filename, so we will see the inode for all three entries on one line each)



5. Inode Analysis

Identify which files share the same inode number and briefly explain the reason.

sample_data.txt and **sample_hard.txt** will share the same inode number; **sample_soft.txt** will have a different inode.

- A hard link (here **sample_hard.txt**) is just another directory entry pointing to the *same inode* as the original file (**sample_data.txt**), so both names reference identical metadata and data blocks.
- A symbolic link (**sample_soft.txt**) is its own separate inode that simply stores a pathname to the target, so it necessarily has a different inode number.

6. File Metadata Inspection

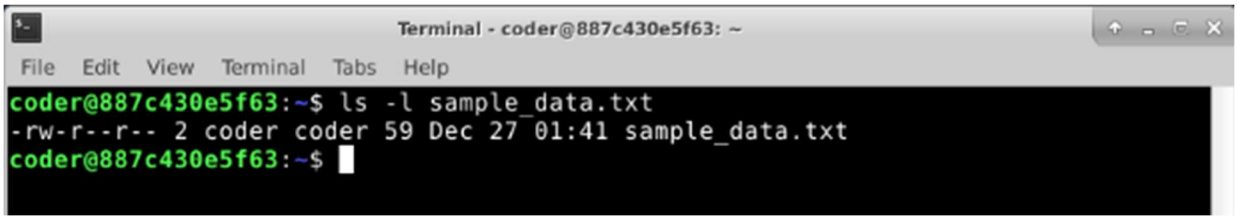
Display detailed file information (permissions, ownership, size, timestamps) of sample_data.txt.

ls -l sample_data.txt

This shows:

- File type and permissions (e.g., -rw-r--r--)
- Number of links
- Owner username and group

- File size in bytes
- Last modification timestamp
- The filename sample_data.txt



7. Disk Usage Check

Display the disk usage of your home directory in a human-readable format.

8. File Size Overview

Display the size of each file present in your home directory in a human-readable format.

du -sh ~

- du reports disk usage for files and directories.
- -s shows only the summary total for the given path.
- -h prints the size in a human-readable form like KB, MB, or GB.

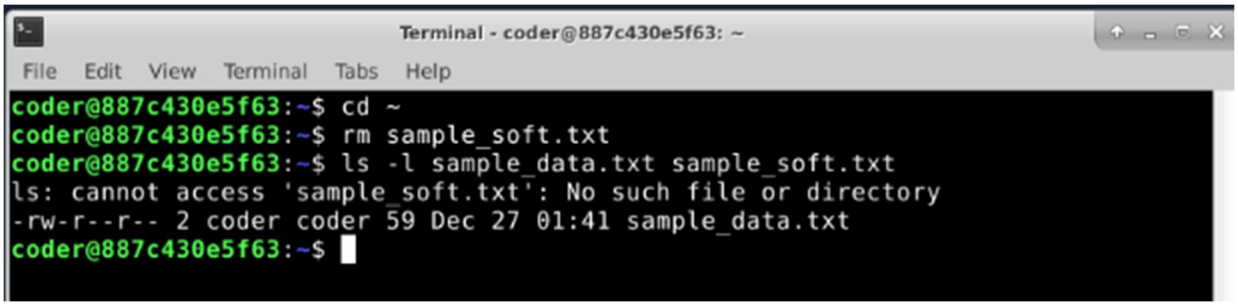
9. Link Deletion Test

Delete the symbolic link sample_soft.txt and verify that the original file sample_data.txt is unaffected.

cd ~

rm sample_soft.txt (This command removes only the symbolic link)

ls -l sample_data.txt sample_soft.txt {This command will still show sample_data.txt present, while sample_soft.txt should now be missing (or give “No such file or directory” if listed alone), confirming the original file is unaffected}

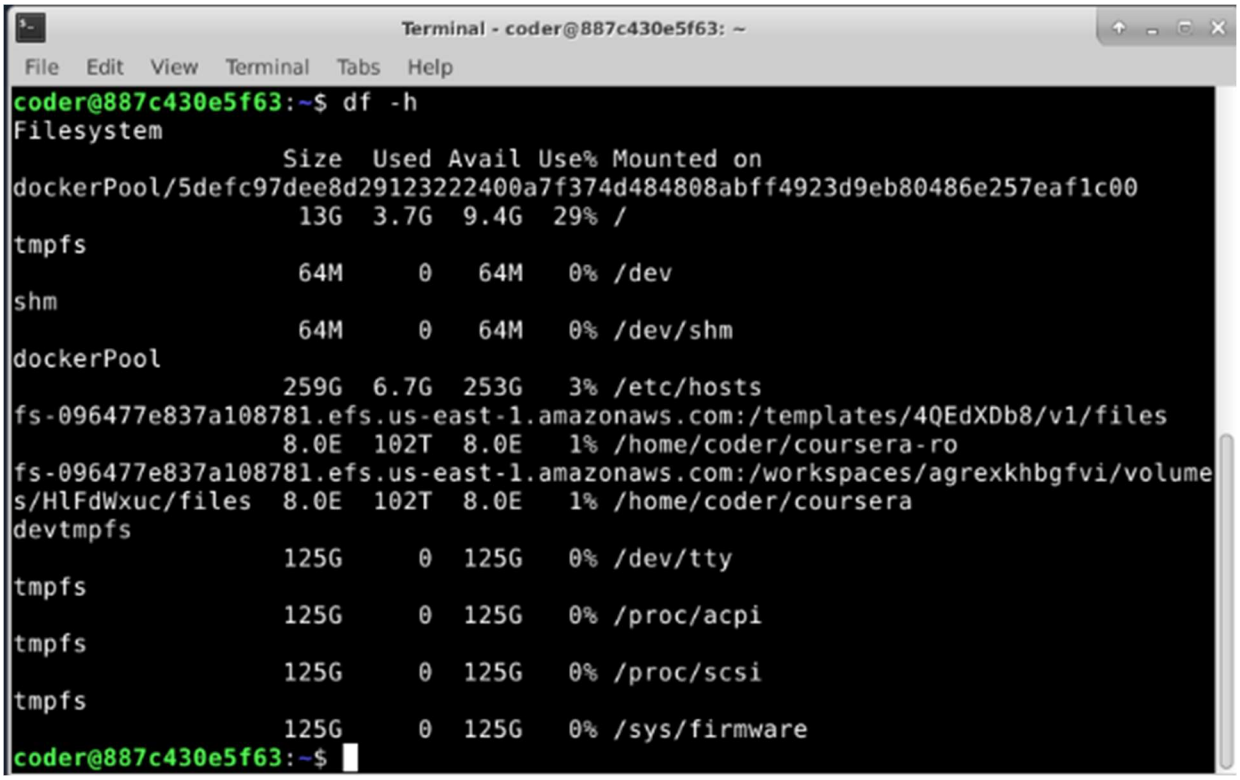


10. Disk Utility Demonstration

Demonstrate the usage of du and df commands using various useful options and briefly explain the output.

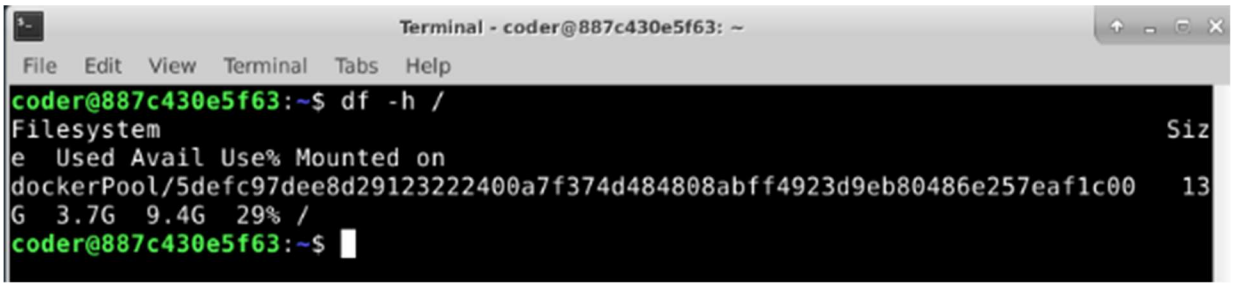
df -h

- df shows disk space usage per mounted filesystem.
- -h prints sizes in a human-readable form (KB/MB/GB), with columns like:
    - Filesystem: device or mount.
    - Size, Used, Avail: total, used, and available space.
    - Use%: percentage used.
    - Mounted on: mount point path.



To focus just on our root filesystem:

df -h /



du -sh ~

- du reports how much space files/directories consume.
- -s gives only a summary total for the specified path.
- -h again makes the size human-readable.

This command tells us how much disk space our home directory uses, complementing the filesystem overview from df.


Terminal - coder@887c430e5f63: ~
File Edit View Terminal Tabs Help
coder@887c430e5f63:~$ du -sh ~
8.9M /home/coder
coder@887c430e5f63:~$