Phase 2.5 — Practical Assignments (Week 1)

Week 1 — Filesystem & Links

🎯 Assignment 1: Filesystem Forensics

1. Create a directory structure like
   ```
   |--project/
        |----src/
        |--build/
        |----log/
   ```
2. Create files inside src/
3. Create:
   a. One hard link
   b. One symbolic link

4. Delete the original file.
5. Observe behavior using:
   ls -li
   Stat

Solutions: -

1: Create Directory Structure

In the given tree structure, we can observe all names end with `/`, they represent directories. (no file is created).

- `mkdir` → creates directories
- `-p` → allows creation of parent + child folders together (prevents error)

Command We Will Used:

mkdir -p project/src project/build project/log

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ mkdir -p project/src project/build project/log
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls
project
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ cd project && ls -li
total 12
29755223 drwxrwxr-x 2 garima garima 4096 Feb 22 11:20 build
29755229 drwxrwxr-x 2 garima garima 4096 Feb 22 11:20 log
29755218 drwxrwxr-x 2 garima garima 4096 Feb 22 11:20 src
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project$
```

2: Create Files Inside `src/`

- We will use `touch` command to create an empty file.

- `touch` → creates a empty file if it does /doesn't exist

Command:

touch project/src/main.txt

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project$ cd src && ls
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$ cd .. && cd .. && touch project/src/main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ cd project && cd src && ls
main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$
```

Now `main.txt` exists inside `src`.

3: Create One Hard Link

- We use `ln` (without options).

- A hard link creates an additional filename that refers to the same inode as the original file.
- Both filenames point to the same inode and share the same data on disk.
- Basically it is one file accessible through two different names.

Command:

ln project/src/main.txt project/src/hardlink.txt

.

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ln project/src/main.txt project/src/hardlink.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ cd project
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project$ cd src && echo "----in src---" && ls -lia
----in src---
total 8
29755218 drwxrwxr-x 2 garima garima 4096 Feb 22 11:28 .
29753682 drwxrwxr-x 5 garima garima 4096 Feb 22 11:20 ..
29755221 -rw-rw-r-- 2 garima garima    0 Feb 22 11:23 hardlink.txt
29755221 -rw-rw-r-- 2 garima garima    0 Feb 22 11:23 main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$
```

4: Create One Symbolic Link

- We will use `ln -s`.

Command:

ln -s project/src/main.txt project/src/symlink.txt

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ln -s project/src/main.txt project/src/symlink.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ cd project && cd src && echo "----in src---" && ls -lia
----in src---
total 8
29755218 drwxrwxr-x 2 garima garima 4096 Feb 22 11:35 .
29753682 drwxrwxr-x 5 garima garima 4096 Feb 22 11:20 ..
29755221 -rw-rw-r-- 2 garima garima    0 Feb 22 11:23 hardlink.txt
29755221 -rw-rw-r-- 2 garima garima    0 Feb 22 11:23 main.txt
29755233 lrwxrwxrwx 1 garima garima   20 Feb 22 11:35 symlink.txt -> project/src/main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/projec
```

- A symbolic link is like a shortcut.
- `-s` → creates symbolic link
- It stores the path of the original file.
- Has different inode from original file

5: Delete the Original File

- We remove the original file to observe behavior.
- We will use rm

Command:

rm project/src/main.txt

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$ cd .. && rm src/main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project$ cd src && echo "----in src---" && ls -lia
----in src---
total 8
29755218 drwxrwxr-x 2 garima garima 4096 Feb 22 11:37 .
29753682 drwxrwxr-x 5 garima garima 4096 Feb 22 11:20 ..
29755221 -rw-rw-r-- 1 garima garima    0 Feb 22 11:23 hardlink.txt
29755233 lrwxrwxrwx 1 garima garima   20 Feb 22 11:35 symlink.txt -> project/src/main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$
```

- `rm` → removes file
- File data deletes only when link count becomes 0

Lets now observe behavior.

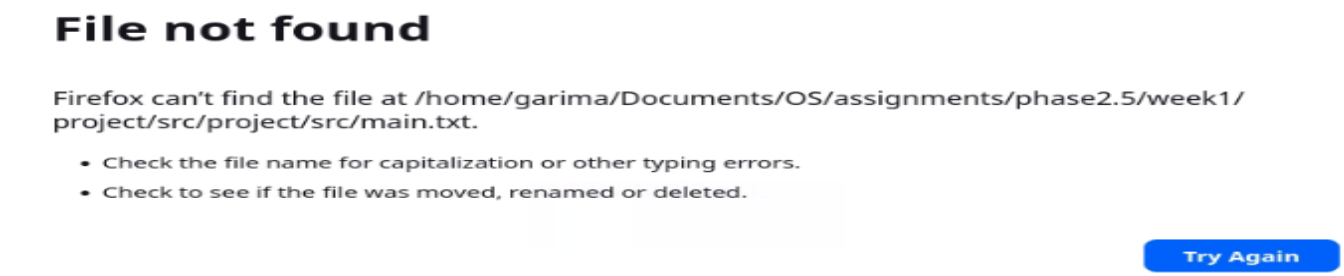
6.a: behavior of `ls -li`

Command:

ls -li project/src

- `l` → detailed listing
- `-i` → shows inode number
- A for hidden file

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$ ls -ali
total 8
29755218 drwxrwxr-x 2 garima garima 4096 Feb 22 11:37 .
29753682 drwxrwxr-x 5 garima garima 4096 Feb 22 11:20 ..
29755221 -rw-rw-r-- 1 garima garima    0 Feb 22 11:23 hardlink.txt
29755233 lrwxrwxrwx 1 garima garima   20 Feb 22 11:35 symlink.txt -> project/src/main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$
```



Observation:

- Hard link still works. It still has the same inode as the original.
- Symbolic link becomes broken (shows in red or error when opened).

6.b: Observe Using `stat`

Command:

stat project/src/hardlink.txt

stat project/src/symlink.txt

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$ stat hardlink.txt
  File: hardlink.txt
  Size: 0               Blocks: 0          IO Block: 4096   regular empty file
Device: 8,2     Inode: 29755221    Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1000/  garima)   Gid: ( 1000/  garima)
Access: 2026-02-22 11:23:55.749835789 +0530
Modify: 2026-02-22 11:23:55.749835789 +0530
Change: 2026-02-22 11:37:15.391563502 +0530
 Birth: 2026-02-22 11:23:55.749835789 +0530
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$ stat symlink.txt
  File: symlink.txt -> project/src/main.txt
  Size: 20              Blocks: 0          IO Block: 4096   symbolic link
Device: 8,2     Inode: 29755233    Links: 1
Access: (0777/lrwxrwxrwx)  Uid: ( 1000/  garima)   Gid: ( 1000/  garima)
Access: 2026-02-22 11:35:29.714288721 +0530
Modify: 2026-02-22 11:35:08.186658974 +0530
Change: 2026-02-22 11:35:08.186658974 +0530
 Birth: 2026-02-22 11:35:08.186658974 +0530
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$
```

- `stat` → shows inode,size, permissions, link count

Observations-

- Hard link survives because it shares inode.
- Symbolic link breaks because it only stores path.

🎯 Assignment 2: Permission Debugging Drill

- You are given:
  `Permission denied`
- Simulate 3 different causes:
  Missing execute on directory
  Wrong ownership
  Incorrect file mode
- Then fix them without using `sudo` unless necessary.

Solution: -

Each digit is a sum of:
4 = read (r)
2 = write (w)
1 = execute (x)

1. Missing "execute" permission on directory

- If a directory does not have execute permission, we cannot enter it.
- We will use chmod command to do so

Command:

chmod 600 project/src
cd project/src

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ chmod 600 project/src
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ cd project/src
bash: cd: project/src: Permission denied
```

`chmod` → changes permission
- `600` → removes execute permission
- `+x` → adds execute permission
- Directory needs `x` to enter

FIX-

chmod +x project/src or chmod 700 project/src

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ chmod +x project/src
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ cd project/src
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1/project/src$ ls
hardlink.txt  symlink.txt
```

2. Wrong ownership

- If file belongs to another user, access may fail.
- We will use chown command:

- `ls -l` → shows owner(detailed listing)
- `chown` → changes ownership

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls -l project/src
total 0
-rw-r--r-- 1 garima garima  0 Feb 22 11:23 hardlink.txt
lrwxrwxrwx 1 garima garima 20 Feb 22 11:35 symlink.txt -> project/src/main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ sudo chown root:root project/src/hardlink.txt
[sudo] password for garima:
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls -l project/src
total 0
-rw-r--r-- 1 root   root    0 Feb 22 11:23 hardlink.txt
lrwxrwxrwx 1 garima garima 20 Feb 22 11:35 symlink.txt -> project/src/main.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$
```

3. Incorrect file mode

- Incorrect file mode can cause problem as we can see here
- We can resolve this using chmod command

File mode be:

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ chmod 000 project/src/hardlink.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ cat project/src/hardlink.txt
cat: project/src/hardlink.txt: Permission denied
```

- 000 → no permission
- 644 → owner read/write, others read

FIX:

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ chmod 644 project/src/hardlink.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ cat project/src/hardlink.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$
```

- Three digits in `chmod XYZ` (e.g., `chmod 754 file`)
- First digit (X) – Owner permissions.
- Second digit (Y) – Group permissions
- Third digit (Z) – Others permissions

🎯 Assignment 3: Safe Backup Workflow

Create a script-less manual workflow that:

- Archives a directory
- Compresses it
- Stores it with timestamp
- Verifies size before and after

Solution: -

Way 1: first archieving and then compressing

- c → create
- v → verbose(show progress)
- f → file name

1. Archives a directory

   - We use `tar.`

   Command:

   tar -cvf backup.tar project/

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ tar -cvf backup.tar project/
project/
project/src/
project/src/hardlink.txt
project/src/symlink.txt
project/log/
project/build/
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls
```

2. **Compresses it**

   - We will use gzip

   Command :

   gzip backup.tar

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls
backup.tar  project
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ gzip backup.tar
```

Way 2 : archieving and compressing at same time

- z→ compress using gzip
- `$(date +%F)` → adds current date

Command :

tar -cvzf backup_$(date +%F).tar.gz project/

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ tar -cvzf backup_$(date +%F).tar.gz project/
project/
project/src/
project/src/hardlink.txt
project/src/symlink.txt
project/log/
project/build/
```

Verfying the size for both :

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls -lai
total 20
29755175 drwxrwxr-x 3 garima garima 4096 Feb 22 13:41 .
29755174 drwxrwxr-x 3 garima garima 4096 Feb 22 11:20 ..
29754908 -rw-rw-r-- 1 garima garima  245 Feb 22 13:41 backup_2026-02-22.tar.gz
29755241 -rw-rw-r-- 1 garima garima  256 Feb 22 13:32 backup.tar.gz
29753682 drwxrwxr-x 5 garima garima 4096 Feb 22 11:20 project
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$
```

The two backups differ in size because one compressed the tar **after creating it**, while the other compressed **during creation**, causing small differences in gzip headers and metadata.

🎯 Assignment 4: Shell Personalization & Alias Behavior

Task:

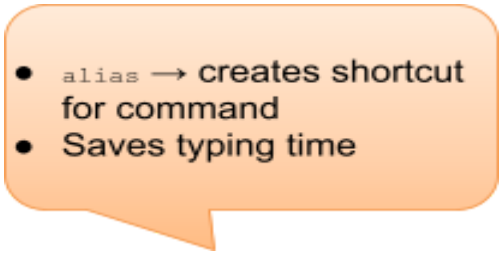
1. Create 5 useful aliases:
    - One for `ls`
    - One for navigation
    - One for search
    - One for git (if installed)
    - One custom productivity shortcut
2. Make them:
    - Work in current session
    - Persist across terminal restarts
3. Verify:
    - `type <alias_name>`
    - `alias`
    - Restart terminal and test again

- Create an alias that breaks something intentionally.
  Example:
  `alias rm='rm -i'`

- Then override it temporarily using:
  `\rm`

Deliverable:

- Explain why `\command` bypasses alias
- Explain why aliases don't work inside non-interactive scripts
- Show difference between alias and function



Solutions:

Creating 5 Aliases:

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ alias ll='ls -lh'
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ alias proj='cd ~/project'
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ alias search='grep -rn'
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ alias gs='git status'
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ alias cls='clear'
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ alias
```

- `alias` → shows all aliases
- `type` → shows if command is alias or built-in

Make them work in terminal

- We use the alias command.:

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ alias
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s
*alert$//'\'')"'
alias cls='clear'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias gs='git status'
alias l='ls -CF'
alias la='ls -A'
alias ll='ls -lh'
alias ls='ls --color=auto'
alias proj='cd ~/project'
alias search='grep -rn'
```

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ll
total 12K
-rw-rw-r-- 1 garima garima  245 Feb 22 13:41 backup_2026-02-22.tar.gz
-rw-rw-r-- 1 garima garima  256 Feb 22 13:32 backup.tar.gz
drwxrwxr-x 5 garima garima 4.0K Feb 22 11:20 project
```

- `.bashrc` → shell configuration file
- `source` → reloads file without restarting terminal

Make Aliases Permanent:

- We need to add all the aliases in the ~/ .bashrc

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ nano ~/.bashrc
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ source ~/.bashrc
```

- Save it and run source ~/.bashrc

Verify:-

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ type ll
ll is aliased to `ls -lh'
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ type gs
gs is aliased to `git status'
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$
```

`-i` → asks before deleting file(interactive mode)

Alias that changes behaviour:

```
garima@garima-HP-Pavilion-Notebook:~$ alias rm='rm -i'
garima@garima-HP-Pavilion-Notebook:~$ alias
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s
*alert$//'\'')"'
alias cls='clear'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias gs='git status'
alias l='ls -CF'
alias la='ls -A'
alias ll='ls -lh'
alias ls='ls --color=auto'
alias proj='cd ~/project'
alias rm='rm -i'
alias search='grep -rn'
```

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ echo "hello world" | touch test.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls
backup_2026-02-22.tar.gz  backup.tar.gz  project  test.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ rm test.txt
rm: remove regular empty file 'test.txt'? y
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls
backup_2026-02-22.tar.gz  backup.tar.gz  project
```

Override Alias Temporarily

```
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ echo "hello world" | touch test.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls
backup_2026-02-22.tar.gz  backup.tar.gz  project  test.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ \rm test.txt
garima@garima-HP-Pavilion-Notebook:~/Documents/OS/assignments/phase2.5/week1$ ls
backup_2026-02-22.tar.gz  backup.tar.gz  project
```

- \ → ignores alias
- Runs original command

Why \command Bypasses Alias?

Because backslash disables alias expansion.

Why Aliases Don't Work in Scripts?

Aliases work only in interactive shell.
Scripts run in non-interactive mode, so aliases are not expanded.

Difference Between Alias and Function

Alias:

- Simple shortcut
- One-line replacement

Function:

- Can take arguments
- Can contain multiple commands
- More powerful