SYMLINKS

A symbolic link, or symlink, is a type of file in a file system that acts as a reference or shortcut to another file or directory.

Why use symlinks?

- **Flexibility in File Management**: Symlinks allow you to create shortcuts or references to files or directories located elsewhere in the filesystem. This can simplify file management by providing multiple access points to the same file or directory.
- **Reducing Redundancy**: Instead of duplicating files or directories, you can use symlinks to reference a single copy. This is useful for saving disk space and maintaining consistency.
- **Managing Large Directories**: Symlinks can help organize large directory structures by creating a more accessible hierarchy. For example, you can create a symlink to a deep directory structure in a more convenient location.
- **Simplifying Path Changes**: If you need to move a file or directory, you can update the symlink to point to the new location instead of updating every reference to the file or directory throughout your system.

The two different types of links used in file systems to reference files.

1. **Soft links (symbolic links)**
2. **hard links**
   - A **soft link**, or symbolic link, is a file that points to another file or directory by storing its path.
   - Soft link is the shortcut to a file, Soft link can created for files and directory,
   - It does not access the data available in the original file.
   - Soft link has different inode number, permission and ownership as the original file.
   - If the original file is deleted, soft link becomes invalid.

Softlink Syntax: **ln -s /path/to/target /path/to/symlink**

- A **hard link** is a direct reference to the file's inode (the data structure containing the file's metadata and data) on the file system.
- Hard link is the copy of the original file, Hard link can't be crated for the directory.
- Hard link has same inode number, permission and ownership as the original file.
- If the original file is deleted, hard link file will contain data.
- If the earlier selected file is deleted, the hard link to the file will still contain the data of that file.

Hard link Syntax: **ln /path/to/target /path/to/hardlink**

What is Inode?

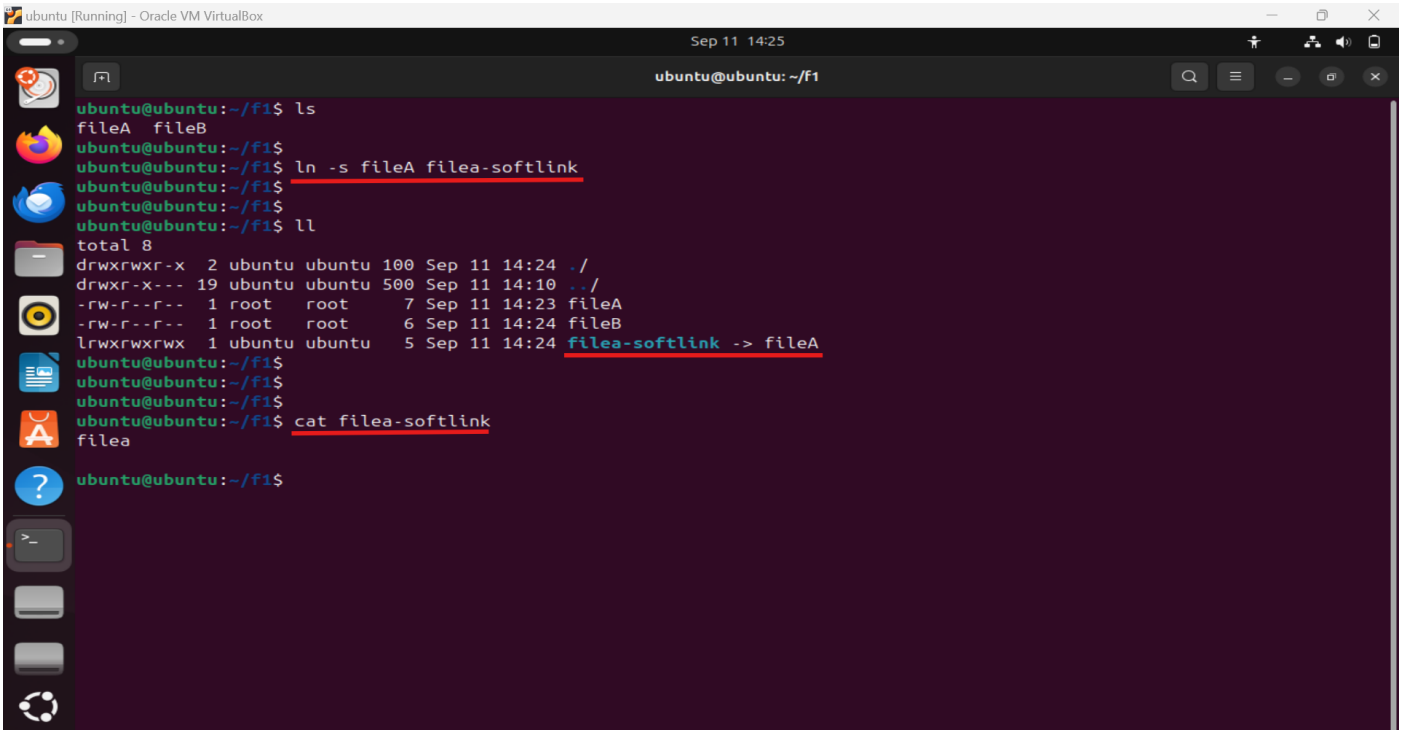
An **inode (Index Node)** is a fundamental data structure used in Unix-like file system to store information about a file or directory. The inode contains metadata that describes the file or directory but does not include the file name or actual data content.

- **File Type, Permissions, Owner(UID), Group(GID) Size, Timestamps** (Includes various timestamps such as Creation Time, Modification Time, Access Time, Change Time), **Link Count, Pointers to Data Blocks**

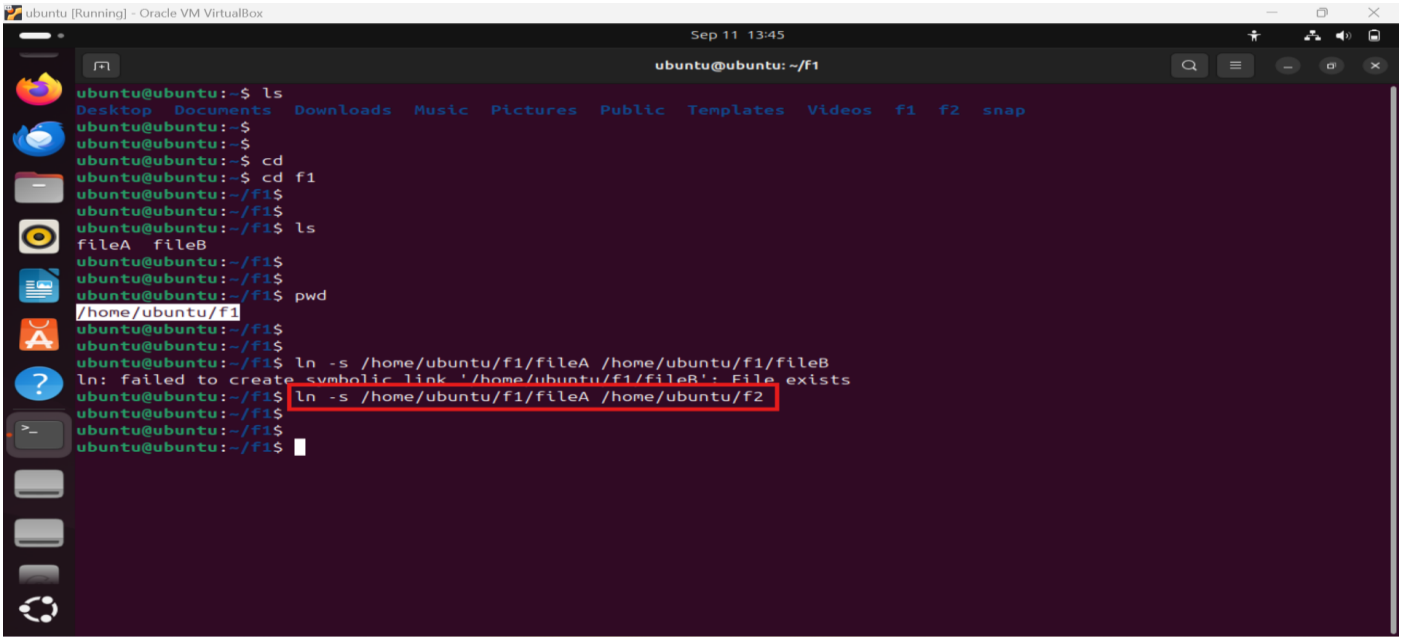
Syntax: **ls -i** [**ls -i "file name"**] (or) **stat file name**

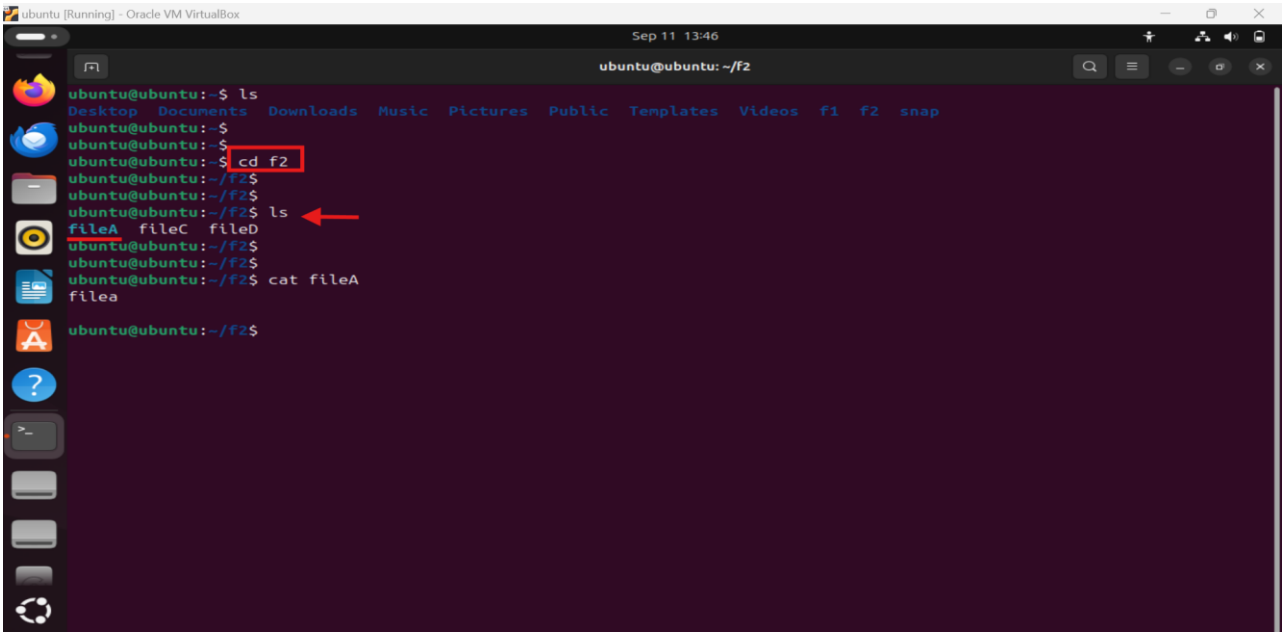
SOFTLINK:

- In same directory

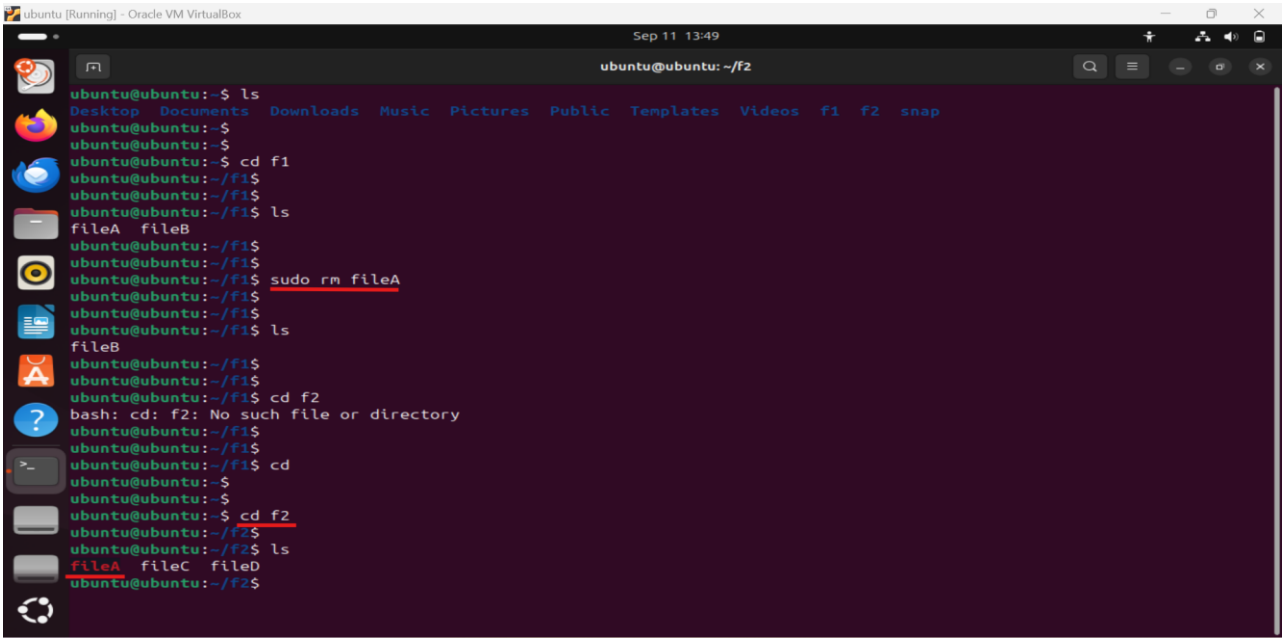


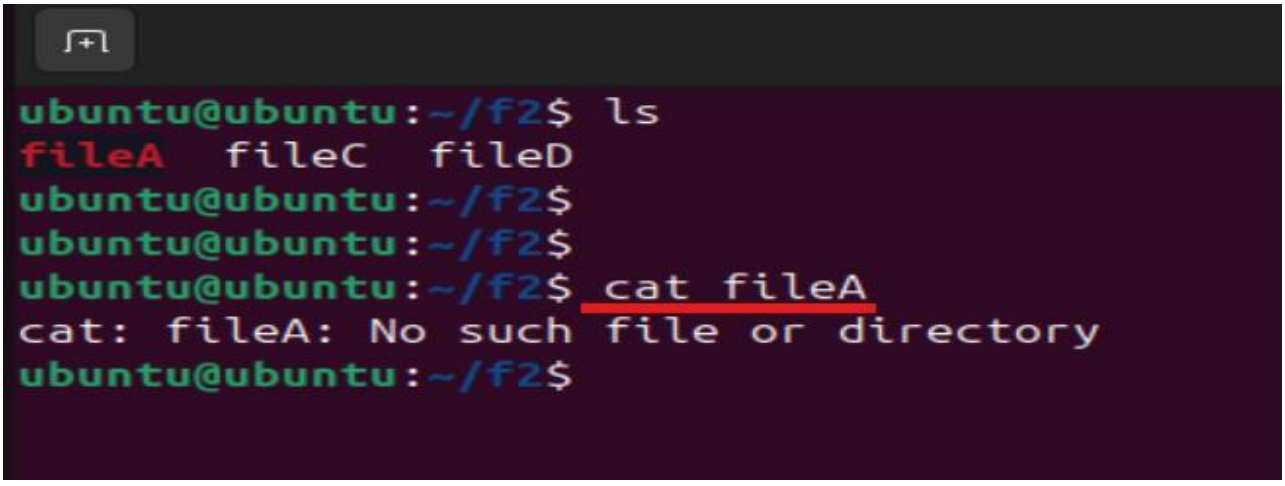
- In different directory.





- After removing the original file soft link is not available.





Ranjith_kumar