File Structure

File structure refers to the logical organization of data within a file system, determining how files and directories are arranged and accessed on a storage device. This includes the hierarchical directory structure (like a tree), where directories contain files and other subdirectories, and the internal organization of data within individual files (e.g., contiguous, linked, or indexed allocation).

File Operations

File operations are the fundamental actions that can be performed on files and directories within a file system. These operations are typically provided by the operating system and include:

- **Creation:** Creating new files or directories.
- **Deletion:** Removing existing files or directories.
- **Opening:** Preparing a file for reading or writing.
- **Closing:** Releasing resources associated with an open file.
- **Reading:** Retrieving data from a file.
- **Writing:** Storing data into a file.
- **Seeking:** Changing the current position within a file for reading or writing.
- **Renaming:** Changing the name of a file or directory.
- **Moving:** Relocating a file or directory to a different location.
- **Setting Permissions:** Controlling access rights to files and directories.

Inode Structure

An inode (index node) is a data structure in Unix-like file systems that stores metadata about a file system object (file, directory, or other). Each inode has a unique number within its file system and contains crucial information such as:

- **File type:** Regular file, directory, symbolic link, etc.
- **Permissions:** Access rights for owner, group, and others.
- **Owner and group IDs:** User and group associated with the file.
- **Timestamps:** Time of last access, modification, and inode change.
- **Size:** Size of the file in bytes.

- **Block pointers:** Pointers to the actual data blocks on the disk where the file's content is stored. These can include direct pointers, single, double, and triple indirect pointers for handling files of various sizes.
- **Link count:** Number of hard links pointing to the inode.
  Directories essentially map filenames to inode numbers, allowing the operating system to locate the corresponding inode and access the file's metadata and data.