File concept:

- A file is a collection of related information that is stored on secondary storage.
- Information stored in files must be persistent i.e. not affected by power failures & system reboots. Files may be of free from such as text files or may be formatted rigidly.
- Files represent both programs as well as data.
- Part of the OS dealing with the files is known as file system.The important file concepts include:

1. File attributes:

A file has certain attributes which vary from oneoperating system to another.

- ✓ **Name:** Every file has a name by which it is referred.
- ✓ **Identifier:** It is unique number that identifies the file within the file system.
- ✓ **Type:** This information is needed for those systems that support different types of files.
- ✓ **Location:** It is a pointer to a device & to the location of the file on that device
- ✓ **Size:** It is the current size of a file in bytes, words or blocks.
- ✓ **Protection:** It is the access control information that determines who can read, write & execute a file.
- ✓ **Time, date & user identification:** It gives information about time of creation or last modification & last use.

2. File operations:

The operating system can provide system calls to create, read, write, reposition, delete and truncate files.

- ✓ **Creating files:** Two steps are necessary to create a file. First, space must be found for the file in the file system. Secondly, an entry must be made in the directory for the new file.
- ✓ **Reading a file:** Data & read from the file at the current position. The system must keep a read pointer to know the location in the file from where the next read is to take place. Once theread has been taken place, the read pointer isupdated.
- ✓ **Writing a file:** Data are written to the file at the current position. The system must keep a write pointer to know the location in the file where the next write is to take place. The write pointer mustbe updated whenever a write occurs.
- ✓ **Repositioning within a file (seek):** The directory is searched for the appropriate entry & the current file position is set to a given value. After repositioning data can be read from or written into that position.
- ✓ **Deleting a file:** To delete a file, we search the directory for the required file. After deletion, the space is released so that it can be reused by other files.
- ✓ **Truncating a file**: The user may erase the contents of a file but allows all attributes to remain unchanged expect the file length which is rest to _O‘ & the space is released

3. File types:

- The file name is spilt into 2 parts, Name & extension. Usually these two parts are separated by a period. Theuser & the OS can know the type of the file from the extension itself.
- Listed below are some file types along with theirextension:

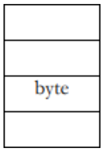
File Type	**Extension**
✓ Executable File	exe, bin, com
✓ Object File	obj, o (compiled)
✓ Source Code file	C, C++, Java, pas
✓ Batch File	bat, sh
✓ Text File	txt, doc
✓ Archive File	arc, zip, tar
✓ Multimedia File	mpeg

4. File structure:

Files can be structured in several ways. Three common possible are:

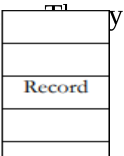
a) Byte sequence:

- The figure shows an unstructured sequence of bytes. The OS doesn‘t care about the content of file. It only sees the bytes.
- This structure provides maximum flexibility.
- Users can write anything into their files & name them according to their convenience.
- Both UNIX & windows use this approach.



b) Record sequence:

- In this structure, a file is a sequence of fixed length records.
- Here the read operation returns one records & thewrite operation overwrites or append or record.



c)**<u>Tree:</u>**

- In this organization, a file consists of a tree ofrecords of varying lengths.
- Each record consists of a key field.
- The tree is stored on the key field to allow firstsearching for a particular key.

5. **Access methods:**

Basically, access method is divided into 2 types:

Sequential access:

- It is the simplest access method. Information in the file is processed in order i.e. one record after another.
- A process can read all the data in a file in order starting from beginning but can't skip & read arbitrarily from any location.
- Sequential files can be rewound.
- It is convenient when storage medium was magnetic tape rather than disk.

Direct access:

- A file is made up of fixed length-logical records that allow programs to read & write records rapidly in no particular O order.
- This method can be used when disk are used for storing files.
- This method is used in many applications e.g. database systems.
- If an airline customer wants to reserve a seat on a particular flight, the reservation program must be ableto access the record for that flight directly without reading the records before it.
- In a direct access file, there is no restriction in the order of reading or writing.
- **For example,** we can read block 14, then read block 50 & then write block 7 etc. Direct access files are very useful for immediate access to large amount of information.

Directory structure:

- The file system of computers can be extensive. Some systems store thousands of file on disk. To manage allthese data, we need to organize them.

The organization is done in 2 steps. The file system is broken into partitions. Each partition contains information about file within it.

Operation on a directory:

- ✓ **Search for a file**: We need to be able to search a directory for a particular file.
- ✓ **Create a file:** New files are created & added to the directory.
- ✓ **Delete a file:** When a file is no longer needed, wemay remove it from the directory.
- ✓ **List a directory:** We should be able to list the files of the directory.
- ✓ **Rename a file:** The name of a file is changed when the contents of the file changes.
- ✓ **Traverse the file system:** It is useful to be ableto access every directory & every file within a directory.

Structure of a directory:

The most common schemes for defining the structure of the directory are:

1. **Single level directory:**

- It is the simplest directory structure.
- All files are present in the same directory.
- So it is easy to manage & understand. Limitation: A single level directory is difficult to manage when the no. of files increases or when there is more than one user.
- Since all files are in same directory, they must have unique names. So, there is confusion of file names between different users.

2. **Two level directories:**

- The solution to the name collision problem in single level directory is to create a separate directory for each user.
- In a two level directory structure, each user has its own user file directory. When a user logs in, then master file directory is searched. It is indexed by user name & each entry points to the UFD of that user.
- **Limitation:** It solves name collision problem. But it isolates one user from another.
- It is an advantage when users are completely independent. But it is a disadvantage when theusers need to access each other's files & co- operate among themselves on a particular task.

3. Tree structured directories:

- It is the most common directory structure.
- A two level directory is a two level tree. So, the generalization is to extend the directory structureto a tree of arbitrary height.
- It allows users to create their own subdirectories &organize their files.
- Every file in the system has a unique path name.
- It is the path from the root through all the sub-directories to a specified file.
- A directory is simply another file but it is treated in a special way. One bit in each directory entry defines the entry as a file (O) or as sub-directories.