**C File management**

A File can be used to store a large volume of persistent data. Like many other languages 'C' provides following file management functions,

1. Creation of a file
2. Opening a file
3. Reading a file
4. Writing to a file
5. Closing a file

In C language, we use a structure **pointer of file type** to declare a file.

FILE \*fp;

C provides a number of functions that helps to perform basic file operations. Following are the functions,

|  |  |
| --- | --- |
| **Function** | **description** |
| fopen() | create a new file or open a existing file |
| fclose() | closes a file |
| getc() | reads a character from a file |
| putc() | writes a character to a file |
| fscanf() | reads a set of data from a file |
| fprintf() | writes a set of data to a file |
| getw() | reads a integer from a file |
| putw() | writes a integer to a file |
| fseek() | set the position to desire point |
| ftell() | gives current position in the file |
| rewind() | set the position to the begining point |

|  |  |
| --- | --- |
| **File Mode** | **Description** |
| r | Open a file for reading. If a file is in reading mode, then no data is deleted if a file is already present on a system. |
| w | Open a file for writing. If a file is in writing mode, then a new file is created if a file doesn't exist at all. If a file is already present on a system, then all the data inside the file is truncated, and it is opened for writing purposes. |
| a | Open a file in append mode. If a file is in append mode, then the file is opened. The content within the file doesn't change. |
| r+ | open for reading and writing from beginning |
| w+ | open for reading and writing, overwriting a file |
| a+ | open for reading and writing, appending to file |