

File Handling in C

- **File Handling in C**

- In programming, we may require some specific input data to be generated several numbers of times.
- Sometimes, it is not enough to only display the data on the console. The data to be displayed may be very large, and only a limited amount of data can be displayed on the console, and since the memory is volatile, it is impossible to recover the programmatically generated data again and again.

- However, if we need to do so, we may store it onto the local file system which is volatile and can be accessed every time.
- Here, comes the need of file handling in C.
- File handling in C enables us to create, update, read, and delete the files stored on the local file system through our C program.
- The following operations can be performed on a file.

- Creation of the new file
- Opening an existing file
- Reading from the file
- Writing to the file
- Deleting the file
- **Functions for file handling**
- There are many functions in the C library to open, read, write, search and close the file. A list of file functions are given below:

No.	Function	Description
1	fopen()	opens new or existing file
2	fprintf()	write data into the file
3	fscanf()	reads data from the file
4	fputc()	writes a character into the file
5	fgetc()	reads a character from file

6	<code>fclose()</code>	closes the file
7	<code>fseek()</code>	sets the file pointer to given position
8	<code>fputw()</code>	writes an integer to file
9	<code>fgetw()</code>	reads an integer from file
10	<code>ftell()</code>	returns current position
11	<code>rewind()</code>	sets the file pointer to the beginning of the file

- **Opening File: fopen()**

- We must open a file before it can be read, write, or update. The fopen() function is used to open a file. The syntax of the fopen() is given below.
- `FILE *fopen(const char * filename, const char * mode);`
- The fopen() function accepts two parameters:
- The file name (string). If the file is stored at some specific location, then we must mention the path at which the file is stored. For example, a file name can be like "c://some_folder/some_file.ext".

- The mode in which the file is to be opened. It is a string.
- We can use one of the following modes in the `fopen()` function.

Mode	Description
r	opens a text file in read mode
w	opens a text file in write mode
a	opens a text file in append mode
r+	opens a text file in read and write mode

w+	opens a text file in read and write mode
a+	opens a text file in read and write mode
rb	opens a binary file in read mode
wb	opens a binary file in write mode
ab	opens a binary file in append mode
rb+	opens a binary file in read and write mode
wb+	opens a binary file in read and write mode
ab+	opens a binary file in read and write mode

- **The fopen function works in the following way.**
- Firstly, It searches the file to be opened.
- Then, it loads the file from the disk and place it into the buffer. The buffer is used to provide efficiency for the read operations.
- It sets up a character pointer which points to the first character of the file.
- Consider the following example which opens a file in write mode.

```
#include<stdio.h>
void main( )
{
FILE *fp ;
fp = fopen("file_handle.txt","w");
if(fp==NULL)
{
    printf("File doesn't exist");
}
```

```
else
{
    printf("File is opened");
    fclose(fp);
}
fclose (fp ) ;
}
```

Output:

File is opened

Closing File: fclose()

The fclose() function is used to close a file. The file must be closed after performing all the operations on it. The syntax of fclose() function is given below:

```
int fclose( FILE *fp );
```

- **Writing File : fputc() function**

- The fputc() function is used to write a single character into file. It outputs a character to a stream.

- **Syntax:**

- `int fputc(int c, FILE *stream)`

```
#include<stdio.h>
void main( )
{
FILE *fp ;
Char name[50] = "CSE";
Int length = strlen(name);

fp = fopen("file_handle.txt","w");
if(fp==NULL)
{
    printf("File doesn't exist");
}
```

- else
- {
- printf("File is opened");
- fclose(fp);
- }
- fclose (fp) ;
- }

- **C fputs() and fgets()**

- The fputs() and fgets() in C programming are used to write and read string from stream. Let's see examples of writing and reading file using fgets() and fgets() functions.

- **Writing File : fputs() function**

- The fputs() function writes a line of characters into file. It outputs string to a stream.
- **Syntax:**
- `int fputs(const char *s, FILE *stream)`

```
#include<stdio.h>
void main( )
{
FILE *fp ;
char name[50];

fp = fopen("file_handle.txt","w");
if(fp==NULL)
{
    printf("File doesn't exist");
}
```

```
else
{
    printf("File is opened\n");
    printf("Enter name:");
    gets(name);
    fputs(name,fp);
    printf("File is written successfully");
    fclose(fp);
}
fclose (fp ) ;
}
```

