

# File Handling in C++

## Introduction

Files are used to store data in a storage device permanently. File handling provides a mechanism to store the output of a program in a file and to perform various operations on it.

A stream is an abstraction that represents a device on which operations of input and output are performed. A stream can be represented as a source or destination of characters of indefinite length depending on its usage.

In C++ we have a set of file handling methods. These include `ifstream`, `ofstream`, and `fstream`. These classes are derived from `fstreambase` and from the corresponding `istream` class. These classes, designed to manage the disk files, are declared in `fstream` and therefore we must include `fstream` and therefore we must include this file in any program that uses files.

## The `fstream` Library:

The `fstream` library provides C++ programs with three classes for working with files. These classes include:

- `fstream`— This class generally represents a file stream. It comes with `ofstream`/`ifstream` capabilities. This means it's capable of creating files, writing to files, reading from data files.
- `ifstream`— This class represents an input stream. It's used for reading information from data files.
- `ofstream`— This class represents an output stream. It's used for creating files and writing information to files.

All the above three classes are derived from **`fstreambase`** and from the corresponding **`istream`** class and they are designed specifically to manage disk files.

## Operations in File Handling:

- Creating a file
- Reading data from file
- Writing new data into file
- Closing a file

## Creating / Opening a File:

We can open a file using any one of the following methods:

1. Opening file using constructor
2. Opening file using the `open()` function.

### 1) Opening file using constructor

We know that a constructor of class initializes an object of its class when it (the object) is being created. Same way, the constructors of stream classes (`ifstream`, `ofstream`, or `fstream`) are used to initialize file stream objects with the filenames passed to them.

#### Example

**`ofstream fout("myfile.txt");` → This creates object `fout` of `ofstream` class for output only.**

**`ifstream fin("myfile.txt");` → This creates object `fin` of `ifstream` class for input only.**

**`fstream finout("myfile.txt");` → This creates object `finout` of `fstream` class input and output.**

## File State Slags

There are some member functions that are used to check the state of the file. All these functions return a Boolean value.

Function	Description
eof()	Returns true if the end of file is reached while reading the file.
fail()	Returns true when read/write operation fails or format error occurs
bad()	Returns true if reading from or writing to a file fail.
good()	Returns false in the same cases in which calling any of the above functions would return true.

### General Function used for file handling

1. open(): To create a file
2. close(): To close an existing file
3. get(): to read a single character from the file
4. put(): to write a single character in the file
5. read(): to read data from a file
6. write(): to write data into a file

### Closing a file in C++

Closing a file is a good practice, and it is must to close the file. Whenever the C++ program comes to an end, it clears the allocated memory, and it closes the file. We can perform the task with the help of close() function.

Syntax:

```
FileName.close();
```

Example Program: Write program to create a file and read text from file.

```
#include<iostream>
#include<fstream>
using namespace std;
int main()
{
    char c,fname[10];
    cout<<"Enter file name:";
    cin>>fname;
    ifstream file(fname);
    if(!file)
    {
        cout<<"Error! File Does not Exist";
    }
    else
    {
        while(file.eof()==0)
        {
            file.get(c);
            cout<<c;
        }
    }
}
```

Example Program: Write program to create a file and write text into file.

```
#include<iostream>
#include<fstream>
using namespace std;
int main()
{
    char ch,fname[10];
```

```

cout<<"Enter file name:";
cin>>fname;
ofstream out(fname);
if(!out)
{
    cout<<"Error! File Does not Exist";
}
else
{
    out<<"Welcome to world of C++";
}
}

```

## 2) Opening file using open() function

To read or enter data to a file, we need to open it first. This can be performed with the help of 'ifstream' for reading and 'fstream' or 'ofstream' for writing or appending to the file. All these three objects have open() function pre-built in them.

Syntax

```
Filepointer.open(filename, mode);
```

**FileName** – It denotes the name of file which has to be opened.

**Mode** – There different mode to open a file and it explained in this article.

Mode	Description
ios::in	File opened in reading mode
ios::out	File opened in write mode
ios::app	File opened in append mode
ios::ate	File opened in append mode but read and write performed at the end of the file.
ios::binary	File opened in binary mode
ios::trunc	File opened in truncate mode. i.e. Delete Content if exist.
ios::nocreate	The file opens only if it exists
ios::noreplace	The file opens only if it doesn't exist

### ifstream class contains following functions:

- 1) **get():** use to read a single character.
- 2) **getline():** use to read single line.
- 3) **read():** use to read single record.
- 4) **seekg():** Moves get pointer to a specified location.
- 5) **tellg():** Tell the current position of get pointer.

### ofstream class contains following functions:

- 1) **put():** use to write a single character.
- 2) **write():** use to write a single record.
- 3) **seekp():** Moves put pointer to a specified location.
- 4) **tellp():** Tell the current position of put pointer.

Seek Function	Action
Fout.seekg (0,ios::beg) or Fout.seekg (0)	Go to begining of the file
Fout.seekg (m,ios::cur)	Move m+1th byte in the file
Fout.seekg (0,ios::end)	Go to end of file

**Write a program to open file if exist and print its content.**

```
#include<iostream>
#include<fstream>
using namespace std;
int main()
{
    char fname[20], ch;
    cout<<"Enter filename:";
    cin>>fname;
    ifstream file;
    file.open(fname);
    if(!file)
    {
        cout<<"File Opening Error....!!";
    }
    else
    {
        while(file.eof()==0)
        {
            file.get(ch);
            cout<<ch;
        }
    }
}
```

**Write a program to create file and write content in it character by character.**

```
#include<iostream>
#include<fstream>
using namespace std;
int main()
{
    char fname[20], text[80],ch;
    cout<<"Enter filename:";
    cin>>fname;
    ofstream file;
    file.open(fname);
    if(!file)
    {
        cout<<"File Creation Error....!!";
    }
    else
```

```
{
    cout<<endl<<"Enter text:";
    cin>>ch;
    while(ch!='$')
    {
        file.put(ch);
        cin>>ch;
    }
    file.close();
}

}
```