public member function

std::ofstream::open

<fstream>

C++98 C++11

```
void open (const char* filename,  ios_base::openmode mode = ios_base::out);
```

Open file

Opens the file identified by argument *filename*, associating it with the stream object, so that input/output operations are performed on its content. Argument *mode* specifies the *opening mode*.

If the stream is already associated with a file (i.e., it is already *open*), calling this function fails.

The file association of a stream is kept by its *internal stream buffer*:
Internally, the function calls `rdbuf()->open(filename,mode|ios_base::out)`

C++98 C++11

The function sets `failbit` in case of failure.

Parameters

`filename`

String with the name of the file to open.
Specifics about its format and validity depend on the library implementation and running environment.

`mode`

Flags describing the requested input/output mode for the file.
This is an object of the bitmask member type `openmode` that consists of a combination of the following member constants:

member constant	stands for	access
`in`	**in**put	File open for reading: the *internal stream buffer* supports input operations.
`out` *	**out**put	File open for writing: the *internal stream buffer* supports output operations.
`binary`	**binary**	Operations are performed in binary mode rather than text.
`ate`	**at e**nd	The *output position* starts at the end of the file.
`app`	**app**end	All output operations happen at the end of the file, appending to its existing contents.
`trunc`	**trunc**ate	Any contents that existed in the file before it is open are discarded.

These flags can be combined with the bitwise OR operator (`|`).
* `out` is always set for `ofstream` objects (even if explicitly not set in argument *mode*).
Note that even though `ofstream` is an output stream, its internal `filebuf` object may be set to also support input operations.

C++98 C++11

If the mode has both `trunc` and `app` set, the opening operation fails. It also fails if both `app` and `in` are set simultaneously.

Return Value

none

If the function fails to open a file, the `failbit` *state flag* is set for the stream (which may throw `ios_base::failure` if that *state flag* was registered using member `exceptions`).

Example

```
1  // ofstream::open / ofstream::close
2  #include <fstream>      // std::ofstream
3
4  int main () {
5
6    std::ofstream ofs;
7    ofs.open ("test.txt", std::ofstream::out | std::ofstream::app);
8
9    ofs << " more lorem ipsum";
10
11   ofs.close();
12
13   return 0;
14 }
```

Data races

Modifies the `ofstream` object.
Concurrent access to the same *stream* object introduce data races.

Exception safety

Basic guarantee: if an exception is thrown, the *stream* is in a valid state.
It throws an exception of member type `failure` if the function fails (setting the `failbit` state flag) and member `exceptions` was set to throw for that state.

See also

ofstream::is_open	Check if file is open (public member function)
ofstream::close	Close file (public member function)
filebuf::open	Open file (public member function)