

Search:

Go

Not logged in

Reference

<fstream>

ifstream

open

register

log in

C++

Information

Tutorials

Reference

Articles

Forum

Reference

C library:

Containers:

Input/Output:

<fstream>

<iomanip>

<ios>

<iosfwd>

<iostream>

<istream>

<ostream>

<sstream>

<streambuf>

Multi-threading:

Other:

<fstream>

class templates:

basic_filebuf

basic_fstream

basic_ifstream

basic_ofstream

classes:

filebuf

fstream

ifstream

ofstream

wfilebuf

wfstream

wifstream

wofstream

ifstream

ifstream::ifstream

public member functions:

ifstream::close

ifstream::is_open

ifstream::open

ifstream::operator=

C++11

ifstream::rddbuf

ifstream::swap

C++11

non-member overloads:

swap (ifstream)

C++11

Networking 4 Game Dev
64 Network DO's and DON'Ts for Game Developers

● ○

public member function

std::ifstream::open

<fstream>

C++98

C++11



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

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 `rddbuf() -> open(filename, mode)`

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 i/o 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 *	input	File open for reading: the <i>internal stream buffer</i> supports input operations.
out	output	File open for writing: the <i>internal stream buffer</i> supports output operations.
binary	binary	Operations are performed in binary mode rather than text.
ate	at end	The <i>output position</i> starts at the end of the file.
app	append	All output operations happen at the end of the file, appending to its existing contents.
trunc	truncate	Any contents that existed in the file before it is open are discarded.

These flags can be combined with the bitwise OR operator (`|`).

* in is always set for *ifstream* objects (even if explicitly not set in argument *mode*).

Note that even though *ifstream* is an input stream, its internal *filebuf* object may be set to also support output operations.

C++98

C++11



If the mode has *app* set, the opening operation fails. It also fails if *trunc* is set but *out* is not.

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 // print the content of a text file.
2 #include <iostream>      // std::cout
3 #include <fstream>      // std::ifstream
4
5 int main () {
6     std::ifstream ifs;
7
8     ifs.open ("test.txt", std::ifstream::in);
9
10    char c = ifs.get();
11
12    while (ifs.good()) {
13        std::cout << c;
14        c = ifs.get();
15    }
```

```
16
17     ifs.close();
18
19     return 0;
20 }
```

● **Data races**

Modifies the `ifstream` 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**

<code>ifstream::is_open</code>	Check if a file is open (public member function)
<code>ifstream::close</code>	Close file (public member function)
<code>filebuf::open</code>	Open file (public member function)

[Home page](#) | [Privacy policy](#)

© cplusplus.com, 2000-2015 - All rights reserved - v3.1
[Spotted an error? contact us](#)