



Search:

Go

Not logged in

Reference

<istream>

istream

ignore

register

log in

C++

Information
Tutorials
Reference
Articles
Forum

Reference

Library:
Containers:
Input/Output:
 <fstream>
 <iomanip>
 <ios>
 <iosfwd>
 <iostream>
 <istream>
 <ostream>
 <sstream>
 <streambuf>
Multi-threading:
Other:

<istream>

class templates:
 basic_istream
 basic_istream
classes:
 istream
 istream
 wistream
 wistream
manipulators:
 ws

istream

istream::istream
istream::~istream
member classes:
 istream::sentry
member functions:
 istream::gcount
 istream::get
 istream::getline
 istream::ignore
 istream::operator>>
 istream::peek
 istream::putback
 istream::read
 istream::readsome
 istream::seekg
 istream::sync
 istream::tellg
 istream::unget
non-member overloads:
 operator>> (istream)
protected members:
 istream::operator=
 istream::swap

public member function

std::istream::ignore

<istream> <iostream>

```
istream& ignore (streamsize n = 1, int delim = EOF);
```

Extract and discard characters

Extracts characters from the input sequence and discards them, until either *n* characters have been extracted, or one compares equal to *delim*.

The function also stops extracting characters if the *end-of-file* is reached. If this is reached prematurely (before either extracting *n* characters or finding *delim*), the function sets the *eofbit* flag.

Internally, the function accesses the input sequence by first constructing a *sentry* object (with *noskipws* set to true). Then (if *good*), it extracts characters from its associated *stream buffer* object as if calling its member functions *sbumpc* or *sgetc*, and finally destroys the *sentry* object before returning.

Parameters*n*

Maximum number of characters to extract (and ignore).

If this is exactly `numeric_limits<streamsize>::max()`, there is no limit: As many characters are extracted as needed until *delim* (or the *end-of-file*) is found. *streamsize* is a signed integral type.

delim

Delimiting character: The function stops extracting characters as soon as an extracted character compares equal to this.

Note that the *delimiting character* is extracted, and thus the next input operation will continue on the character that follows it (if any).

If this is the *end-of-file* value (*EOF*), no character will compare equal, and thus exactly *n* characters will be discarded (unless the function fails or the *end-of-file* is reached).

Return Value

The *istream* object (**this*).

Errors are signaled by modifying the *internal state flags*:

flag	error
<i>eofbit</i>	The function stopped extracting characters because the input sequence has no more characters available (<i>end-of-file</i> reached).
<i>failbit</i>	The construction of <i>sentry</i> failed (such as when the <i>stream state</i> was not <i>good</i> before the call).
<i>badbit</i>	Error on stream (such as when this function catches an exception thrown by an internal operation). When set, the integrity of the stream may have been affected.

Multiple flags may be set by a single operation.

If the operation sets an *internal state flag* that was registered with member *exceptions*, the function throws an exception of member type *failure*.

Example

```
1 // istream::ignore example
2 #include <iostream>      // std::cin, std::cout
3
4 int main () {
5     char first, last;
6
7     std::cout << "Please, enter your first name followed by your surname: ";
8
9     first = std::cin.get();    // get one character
10    std::cin.ignore(256, ' '); // ignore until space
11
12    last = std::cin.get();     // get one character
13
14    std::cout << "Your initials are " << first << last << '\n';
15
16    return 0;
17 }
```

Possible output:

```
Please, enter your first name followed by your surname: John Smith
Your initials are JS
```

● **Data races**

Modifies the stream object.
Concurrent access to the same stream object may cause data races, except for the standard stream object `cin` when this is *synchronized with `stdio`* (in this case, no data races are initiated, although no guarantees are given on the order in which extracted characters are attributed to threads).

● **Exception safety**

Basic guarantee: if an exception is thrown, the object is in a valid state.
It throws an exception of member type `failure` if the resulting *error state flag* is not `goodbit` and member `exceptions` was set to throw for that state.
Any exception thrown by an internal operation is caught and handled by the function, setting `badbit`. If `badbit` was set on the last call to `exceptions`, the function rethrows the caught exception.

📌 **See also**

<code>istream::peek</code>	Peek next character (public member function)
<code>istream::get</code>	Get characters (public member function)
<code>istream::getline</code>	Get line (public member function)
<code>istream::read</code>	Read block of data (public member function)
<code>istream::readsome</code>	Read data available in buffer (public member function)

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

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