cplusplus.com
Search: Go
Not logged in
Reference <ios> noskipws
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:

<ios>
types:
basic_ios
fpos
ios
ios_base
io_errc C++11
streamoff
streampos
streamsize
wios
wstreampos
manipulators:
boolalpha
dec
defaultfloat C++11
fixed
hex
hexfloat C++11
internal
left
noboolalpha
noshowbase
noshowpoint
noshowpos
noskipws
nounitbuf
nouppercase
oct
right
scientific
showbase
showpoint
showpos
skipws
unitbuf
uppercase
other functions:
iostream_category C++11

function

<ios> <iostream>

std::noskipws

```
ios_base& noskipws (ios_base& str);
```

Do not skip whitespaces

Clears the `skipws` format flag for the *str* stream.

When the `skipws` format flag is not set, all operations on the stream consider initial whitespace characters as valid content to be extracted.

Tab spaces, carriage returns and blank spaces are all considered whitespaces (see `isspace`).

This flag can be set with the `skipws` manipulator. When set, as many initial whitespace characters as necessary are read and discarded from the stream until a non-whitespace character is found. This would apply to *every* formatted input operation performed with `operator>>` on the stream.

Notice that many extraction operations consider the whitespaces themselves as the terminating character, therefore, with the `skipws` flag disabled, some extraction operations may extract no characters at all from the stream.

For standard streams, the `skipws` flag **is set** on initialization.

Parameters

`str`
: Stream object whose *format flag* is affected.
Because this function is a manipulator, it is designed to be used alone with no arguments in conjunction with the *insertion* (`<<`) and *extraction* (`>>`) operations on streams (see example below).

Return Value

Argument *str*.

Example

```
// skipws flag example
#include <iostream>     // std::cout, std::skipws, std::noskipws
#include <sstream>      // std::istringstream

int main () {
  char a, b, c;

  std::istringstream iss ("  123");
  iss >> std::skipws >> a >> b >> c;
  std::cout << a << b << c << '\n';

  iss.seekg(0);
  iss >> std::noskipws >> a >> b >> c;
  std::cout << a << b << c << '\n';
  return 0;
}
```

Output:

```
123
  1
```

Notice that in the first set of extractions, the leading spaces were ignored, while in the second they were extracted as valid characters.

Data races

Modifies *str*. Concurrent access to the same stream object may cause data races.

Exception safety

Basic guarantee: if an exception is thrown, *str* is in a valid state.

See also

skipws	Skip whitespaces (function)
ios_base::flags	Get/set format flags (public member function)

ios_base::setf	Set specific format flags (public member function)
ios_base::unsetf	Clear specific format flags (public member function)