



Search:

Go

Not logged in

Reference

<ios>

ios_base

setf

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

streamoff

streampos

streamsize

wios

wstreampos

manipulators:

boolalpha

dec

defaultfloat

fixed

hex

hexfloat

internal

left

noboolalpha

noshowbase

noshowpoint

noshowpos

noskipws

nounitbuf

nouppercase

oct

right

scientific

showbase

showpoint

showpos

skipws

unitbuf

uppercase

other functions:

iostream_category

ios_base

ios_base::ios_base

ios_base::~ios_base

member functions:

ios_base::flags

ios_base::getloc

ios_base::imbue

ios_base::iword

ios_base::precision

ios_base::pword

ios_base::register_callback

ios_base::setf

ios_base::sync_with_stdio

ios_base::unsetf

ios_base::width

ios_base::xalloc

member types:

ios_base::event

ios_base::event_callback

public member function

std::ios_base::setf

<ios> <iostream>

set (1) fmtflags setf (fmtflags fmtfl);

mask (2) fmtflags setf (fmtflags fmtfl, fmtflags mask);

Set specific format flags

The first form (1) sets the stream's *format flags* whose bits are set in *fmtfl*, leaving unchanged the rest, as if a call to `flags(fmtfl|flags())`.

The second form (2) sets the stream's *format flags* whose bits are set in both *fmtfl* and *mask*, and clears the *format flags* whose bits are set in *mask* but not in *fmtfl*, as if a call to `flags((fmtfl&mask)|(flags()&~mask))`.

Both return the value of the stream's *format flags* before the call.

The format flags of a stream affect the way data is interpreted in certain input functions and how it is written by certain output functions. See `ios_base::fmtflags` for the possible values of this function's arguments.

The first form of `setf` (1) is generally used to set *independent format flags*: `boolalpha`, `showbase`, `showpoint`, `showpos`, `skipws`, `unitbuf` and `uppercase`, which can also be unset directly with member `unsetf`.

The second form (2) is generally used to set a value for one of the selective flags, using one of the field bitmasks as the mask argument:

<i>fmtfl</i> format flag value	<i>mask</i> field bitmask
left, right or internal	adjustfield
dec, oct or hex	basefield
scientific or fixed	floatfield

The parameterized manipulator `setiosflags` behaves in a similar way as the first form of this member function (1).

Parameters

fmtfl

Format flags to be set. If the second syntax is used, only the bits set in both *fmtfl* and *mask* are set in the stream's format flags; the flags set in *mask* but not in *fmtfl* are cleared.

mask

Mask containing the flags to be modified.

Member type `fmtflags` is a bitmask type (see `ios_base::fmtflags`).

Return Value

The format flags selected in the stream before the call.

Example

```
1 // modifying flags with setf/unsetf
2 #include <iostream> // std::cout, std::ios
3
4 int main () {
5     std::cout.setf ( std::ios::hex, std::ios::basefield ); // set hex as the basefield
6     std::cout.setf ( std::ios::showbase ); // activate showbase
7     std::cout << 100 << '\n';
8     std::cout.unsetf ( std::ios::showbase ); // deactivate showbase
9     std::cout << 100 << '\n';
10    return 0;
11 }
```

Output:

```
0x64
64
```

Data races

Modifies the stream object.

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

Exception safety

Basic guarantee: if an exception is thrown, the stream is in a valid state.

ios_base::failure

ios_base::fmtflags

ios_base::Init

ios_base::iostate

ios_base::openmode

ios_base::seekdir

PatchIT Updating Librar

PatchIT offers fully automated updating libraries for coding

📌 See also	
ios_base::flags	Get/set format flags (public member function)
ios_base::unsetf	Clear specific format flags (public member function)
ios_base::fmtflags	Type for stream format flags (public member type)
setiosflags	Set format flags (function)
resetiosflags	Reset format flags (function)