

Search:

Go

Not logged in

Reference <ios> **ios_base** precision

register

log in

C++

Information
Tutorials
Reference
Articles
Forum

Reference

C library:

Containers:

Input/Output:

<fstream>

<iomanip>

<ios>

<iosfwd>

<iostream>

<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::precision

<ios> <iostream>

get (1) streamsize precision() const;

set (2) streamsize precision (streamsize prec);

Get/Set floating-point decimal precision

The first form (1) returns the value of the current floating-point precision field for the stream.

The second form (2) also sets it to a new value.

The *floating-point precision* determines the maximum number of digits to be written on insertion operations to express floating-point values. How this is interpreted depends on whether the floatfield **format flag** is set to a specific notation (either *fixed* or *scientific*) or it is unset (using the *default notation*, which is not necessarily equivalent to either *fixed* nor *scientific*).

For the default locale:

- Using the default floating-point notation, the precision field specifies the maximum number of meaningful digits to display in total counting both those before and those after the decimal point. Notice that it is not a minimum, and therefore it does not pad the displayed number with trailing zeros if the number can be displayed with less digits than the *precision*.
- In both the *fixed* and *scientific* notations, the precision field specifies exactly how many digits to display after the decimal point, even if this includes trailing decimal zeros. The digits before the decimal point are not relevant for the *precision* in this case.

This *decimal precision* can also be modified using the parameterized manipulator `setprecision`.**Parameters**

prec

New value for the floating-point precision.

`streamsize` is a signed integral value.**Return Value**The *precision* selected in the stream before the call.**Example**

```
1 // modify precision
2 #include <iostream> // std::cout, std::ios
3
4 int main () {
5     double f = 3.14159;
6     std::cout.unsetf ( std::ios::floatfield ); // floatfield not set
7     std::cout.precision(5);
8     std::cout << f << '\n';
9     std::cout.precision(10);
10    std::cout << f << '\n';
11    std::cout.setf( std::ios::fixed, std::ios::floatfield ); // floatfield set to fixed
12    std::cout << f << '\n';
13    return 0;
14 }
```

Possible output:

```
3.1416
3.14159
3.1415900000
```

Notice how the first number written is just 5 digits long, while the second is 6, but not more, even though the stream's precision is now 10. That is because precision with the default floatfield only specifies the *maximum* number of digits to be displayed, but not the minimum.

The third number printed displays 10 digits after the decimal point because the floatfield format flag is in this case set to fixed.

Data races

Accesses (1) or modifies (2) 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

Answers to C++ Questions

Free Answers to Your Programming Language Questions. Register Now!

🔗 See also	
setprecision	Set decimal precision (function)
ios_base::width	Get/set field width (public member function)
ios_base::setf	Set specific format flags (public member function)