



Search:

Reference <ios> dec

Not logged 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

Start Free Download  
Begin w/ Reading Fanatic for  
Free Access to Unlimited eBooks!



Run IT support the DevOps way  
A webinar presented by ITSM expert John Custy. Yup, that's him.



function  
**std::dec**  
`ios_base& dec (ios_base& str);`  
**Use decimal base**  
Sets the basefield format flag for the *str* stream to dec.  
  
When basefield is set to dec, integer values inserted into the stream are expressed in decimal base (i.e., radix 10). For input streams, extracted values are also expected to be expressed in decimal base when this flag is set.  
  
The basefield format flag can take any of the following values (each with its own manipulator):

flag value	effect when set
dec	read/write integer values using decimal base format.
hex	read/write integer values using hexadecimal base format.
oct	read/write integer values using octal base format.

  
For standard streams, the basefield flag is set to dec on initialization.

**Parameters**  
  
**str**  
Stream object whose basefield *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**

```
1 // modify basefield
2 #include <iostream>    // std::cout, std::dec, std::hex, std::oct
3
4 int main () {
5     int n = 70;
6     std::cout << std::dec << n << '\n';
7     std::cout << std::hex << n << '\n';
8     std::cout << std::oct << n << '\n';
9     return 0;
10 }
```

  
Output:  
70  
46  
106

**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**

<b>hex</b>	Use hexadecimal base ( <a href="#">function</a> )
<b>oct</b>	Use octal base ( <a href="#">function</a> )
<b>ios_base::flags</b>	Get/set format flags ( <a href="#">public member function</a> )
<b>ios_base::setf</b>	Set specific format flags ( <a href="#">public member function</a> )

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