



Search:

[Reference](#)
[<sstream>](#)
[istream](#)
[rdbuf](#)

[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:

<sstream>

class templates:

basic\_istream

basic\_ostream

basic\_stringbuf

basic\_stringstream

classes:

istream

ostream

stringbuf

stringstream

wstringstream

wostream

wstringbuf

wstringstream

istream

istream::istream

public member functions:

istream::operator=

istream::rdbuf

istream::str

istream::swap

non-member overloads:

swap (istream)

PatchIT Updating Libraries

PatchIT offers fully automated updating libraries for coding

public member function

std::istream::rdbuf

<sstream>

stringbuf\* rdbuf() const;

Get stream buffer

Returns a pointer to the internal `stringbuf` object, with which the object was associated on construction.

Notice however, that this is not necessarily the same as the currently *associated stream buffer* (returned by `ios::rdbuf`).

Parameters

none

Return Value

A pointer to the internal `stringbuf` object.

Example

```

1 // istream::rdbuf
2 #include <string> // std::string
3 #include <iostream> // std::cout
4 #include <sstream> // std::istream, std::stringbuf
5
6 int main () {
7     std::istream iss;
8     std::stringbuf *pbuf = iss.rdbuf();
9
10    // using stringbuf directly:
11    pbuf->str("Example string");
12
13    int size = pbuf->in_avail();
14
15    while (pbuf->in_avail()>0)
16        std::cout << static_cast<char>(pbuf->sgetc());
17
18    return 0;
19 }
```

Example string

Data races

Accesses the stream object.

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

Exception safety

Strong guarantee: if an exception is thrown, there are no changes in the *stream buffer*.

See also

stringbuf

String stream buffer (class )

ios::rdbuf

Get/set stream buffer (public member function )

Home page | Privacy policy

© cplusplus.com, 2000-2015 - All rights reserved - v3.1

Spotted an error? contact us