

# iostream

The most frequently used read/write stream is iostream. We'll look at the implementation shortly.

The stream classes `std::istream` and `std::ostream` are often used for the reading and writing of data. Use of `std::istream` classes requires the `<istream>` header; use of `std::ostream` classes requires the `<ostream>` header. We can have both with the header `<iostream>`. `std::istream` is a typedef for the class `basic_istream` and the character type `char` whereas `std::ostream` is a typedef for the class `basic_ostream`:

```
typedef basic_istream<char> istream;
typedef basic_ostream<char> ostream;
```



C++ has four predefined stream objects for convenience when dealing with the keyboard and the monitor.

Stream object	C pendant	Device	Buffered
<code>std::cin</code>	<code>stdin</code>	keyboard	yes
<code>std::cout</code>	<code>stdout</code>	monitor	yes
<code>std::cerr</code>	<code>stderr</code>	monitor	no
<code>std::clog</code>		monitor	yes

## The four predefined stream objects

**i** The stream objects are also available for `wchar_t`

The four stream objects for `wchar_t` `std::wcin`, `std::wcout`, `std::wcerr`,

and `std::wclog` are by far not as heavily used as their `char` pendants. Therefore, we treat them with only marginal concern.

The stream objects are sufficient to write a program that reads from the command line and returns the sum.

```
#include <iostream>

int main(){

    std::cout << std::endl;
    std::cout << "Type in your numbers(Quit with an arbitrary character): " << std::endl;

    int sum{0};
    int val;

    while ( std::cin >> val ) sum += val;

    std::cout << "Sum: " << sum << std::endl << std::endl;
}
```



>\_



The stream objects

The small program above uses the stream operators `<<` and `>>` and the stream manipulator `std::endl`.

The insert operator `<<` pushes characters onto the output stream `std::cout`; the extract operator `>>` pulls the characters from the input stream `std::cin`. We can build chains of insert or extract operators because both operators return a reference to themselves.

`std::endl` is a stream manipulator because it puts a `'\n'` character onto `std::cout` and flushes the output buffer.

Here are the most frequently used stream manipulators.

Manipulator	Stream type	Description
<code>std::endl</code>	output	Inserts a new-line character and flushes the stream

<code>std::flush</code>	output	Flushes the stream.
<code>std::ws</code>	input	Discards leading whitespace.

### The most frequently used stream manipulators

---

In the next lesson, we'll discuss other functions we can use to perform input/output operations other than `cin` and `cout`.