Information
Tutorials
Reference
Articles
Forum

Reference
C library:
Containers:
<array>
<deque>
<forward_list>
<list>
<map>
<queue>
<set>
<stack>
<unordered_map>
<unordered_set>
<vector>
Input/Output:
Multi-threading:
Other:

<queue>
priority_queue
queue

queue
queue::queue
member functions:
queue::back
queue::emplace
queue::empty
queue::front
queue::pop
queue::push
queue::size
queue::swap
non-member overloads:
relational operators (queue)
swap (queue)
non-member specializations:
uses_allocator<queue>

public member function

std::queue::front

<queue>

C++98 | C++11

```
      reference& front();
const_reference& front() const;
```

Access next element

Returns a reference to the *next element* in the `queue`.

The *next element* is the "oldest" element in the `queue` and the same element that is popped out from the `queue` when `queue::pop` is called.

This member function effectively calls member `front` of the *underlying container* object.

Parameters

none

Return value

A reference to the *next element* in the `queue`.

C++98 | C++11

Member types `reference` and `const_reference` are aliases of the *underlying container*'s types with the same name.

Example

```
// queue::front
#include <iostream>       // std::cout
#include <queue>          // std::queue

int main ()
{
  std::queue<int> myqueue;

  myqueue.push(77);
  myqueue.push(16);

  myqueue.front() -= myqueue.back();    // 77-16=61

  std::cout << "myqueue.front() is now " << myqueue.front() << '\n';

  return 0;
}
```

Output:

```
myqueue.front() is now 61
```

Complexity

Constant (calling `front` on the *underlying container*).

Data races

The container is accessed (neither the const nor the non-const versions modify the container).
The reference returned can be used to access or modify the next element.

Exception safety

Provides the same level of guarantees as the operation performed on the container (no-throw guarantee for standard non-empty containers).

See also

queue::pop	Remove next element (public member function)
queue::back	Access last element (public member function)