class template

std::basic_string

<string>

```
template < class charT,
           class traits = char_traits<charT>,    // basic_string::traits_type
           class Alloc = allocator<charT>        // basic_string::allocator_type
           > class basic_string;
```

Generic string class

The `basic_string` is the generalization of class string for any character type (see string for a description).

Template parameters

`charT`
Character type.
The string is formed by a sequence of characters of this type.
This shall be a non-array POD type.

`traits`
Character traits class that defines essential properties of the characters used by basic_string objects (see char_traits).
`traits::char_type` shall be the same as `charT`.
Aliased as member type `basic_string::traits_type`.

`Alloc`
Type of the allocator object used to define the storage allocation model. By default, the allocator class template is used, which defines the simplest memory allocation model and is value-independent.
Aliased as member type `basic_string::allocator_type`.

Note: Because the first template parameter is not aliased as any member type, `charT` is used throughout this reference to refer to this type.

Template instantiations

string	String class (class)
wstring	Wide string (class)
u16string	String of 16-bit characters (class)
u32string	String of 32-bit characters (class)

Member types

C++98 C++11

member type	definition	notes
`traits_type`	The second template parameter (`traits`)	defaults to: `char_traits<charT>`
`allocator_type`	The third template parameter (`Alloc`)	defaults to: `allocator<charT>`
`value_type`	`traits_type::char_type`	shall be the same as `charT`
`reference`	`value_type&`	
`const_reference`	`const value_type&`	
`pointer`	`allocator_traits<allocator_type>::pointer`	for the default allocator: `charT*`
`const_pointer`	`allocator_traits<allocator_type>::const_pointer`	for the default allocator: `const charT*`
`iterator`	a random access iterator to `charT`	convertible to `const_iterator`
`const_iterator`	a random access iterator to `const charT`	
`reverse_iterator`	`reverse_iterator<iterator>`	
`const_reverse_iterator`	`reverse_iterator<const_iterator>`	
`difference_type`	`allocator_traits<allocator_type>::difference_type`	usually the same as ptrdiff_t
`size_type`	`allocator_traits<allocator_type>::size_type`	usually the same as size_t

Member functions

(constructor)	Construct basic_string object (public member function)
(destructor)	String destructor (public member function)
operator=	String assignment (public member function)

Iterators:

begin	Return iterator to beginning (public member function)
end	Return iterator to end (public member function)
rbegin	Return reverse iterator to reverse beginning (public member function)
rend	Return reverse iterator to reverse end (public member function)
cbegin	Return const_iterator to beginning (public member function)
cend	Return const_iterator to end (public member function)

crbegin	Return const_reverse_iterator to reverse beginning (public member function)
crend	Return const_reverse_iterator to reverse end (public member function)

Capacity:

size	Return size (public member function)
length	Return length of string (public member function)
max_size	Return maximum size (public member function)
resize	Resize string (public member function)
capacity	Return size of allocated storage (public member function)
reserve	Request a change in capacity (public member function)
clear	Clear string (public member function)
empty	Test whether string is empty (public member function)
shrink_to_fit	Shrink to fit (public member function)

Element access:

operator[]	Get character of string (public member function)
at	Get character of string (public member function)
back	Access last character (public member function)
front	Access first character (public member function)

Modifiers:

operator+=	Append to string (public member function)
append	Append to string (public member function)
push_back	Append character to string (public member function)
assign	Assign content to string (public member function)
insert	Insert into string (public member function)
erase	Erase characters from string (public member function)
replace	Replace portion of string (public member function)
swap	Swap string values (public member function)
pop_back	Delete last character (public member function)

String operations:

c_str	Get C-string equivalent
data	Get string data (public member function)
get_allocator	Get allocator (public member function)
copy	Copy sequence of characters from string (public member function)
find	Find first occurrence in string (public member function)
rfind	Find last occurrence in string (public member function)
find_first_of	Find character in string (public member function)
find_last_of	Find character in string from the end (public member function)
find_first_not_of	Find non-matching character in string (public member function)
find_last_not_of	Find non-matching character in string from the end (public member function)
substr	Generate substring (public member function)
compare	Compare strings (public member function)

Non-member function overloads

operator+	Concatenate strings (function template)
relational operators	Relational operators for basic_string (function template)
swap	Exchanges the values of two strings (function template)
operator>>	Extract string from stream (function template)
operator<<	Insert string into stream (function template)
getline	Get line from stream into string (function template)

Member constants

npos	Maximum value of size_type (public static member constant)