### core::sample\_allocator

template <class T> class sample\_allocator;

#### Member types

|  |  |  |
| --- | --- | --- |
| **member type** | **definition** | **represents** |
| value\_type | T | Element type |
| pointer | T\* | Pointer to element |
| const\_pointer | const T\* | Pointer to constant element |
| reference | T& | Reference to element |
| const\_reference | const T& | Reference to constant element |
| size\_type | size\_t | Quantities of elements |
| difference\_type | ptrdiff\_t | Difference between two pointers |
| rebind<U> | member class | Its member type other is the equivalent allocator type to allocate elements of type U |
| propagate\_on\_container\_move\_assignment | true\_type | Indicates that allocator shall propagate when the container is move-assigned |
| is\_always\_equal | true\_type |  |

#### Member functions

|  |  |
| --- | --- |
| (constructor) | Construct allocator object (public member function) |
| (destructor) | Allocator destructor (public member function) |
| address | Return address (public member function) |
| allocate | Allocate block of storage (public member function) |
| deallocate | Release block of storage (public member function) |
| max\_size | Maximum size possible to allocate (public member function) |
| construct | Construct an object (public member function) |
| destroy | Destroy an object (public member function) |

#### Template specializations

template <> class sample\_allocator<void>;

##### Member types

|  |  |  |
| --- | --- | --- |
| **member type** | **definition** | **represents** |
| value\_type | void | Element type |
| pointer | void\* | Pointer to element |
| const\_pointer | const void\* | Pointer to constant element |
| size\_type | size\_t | Quantities of elements |
| difference\_type | ptrdiff\_t | Difference between two pointers |
| rebind<U> | member class | Its member type other is the equivalent allocator type to allocate elements of type U |