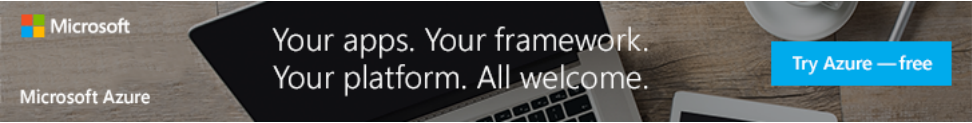


Join the Stack Overflow Community

Stack Overflow is a community of 7.3 million programmers, just like you, helping each other. Join them; it only takes a minute:

Sign up

Remove all elements in vector C++ [duplicate]



Possible Duplicate:  
[Delete all items from a c++ std::vector](#)

I think using begin in an iterator is messing this up since it's only iterating 26 times and skipping every 2 elements. How else can I do it?

```
void clearVector() {
    for (int i = 0; i < 52; i++) {
        vector.erase(vector.begin() + i);
    }
}
```

c++   vector   erase

edited May 23 at 12:02  
 Community  
1 1

asked Nov 30 '12 at 6:38  
 thisiscrazy4  
764 6 20 41

marked as duplicate by [jogojapan](#), [Tom Wijsman](#), [juanchopanza](#), [WhozCraig](#), [ВЈовић](#) Nov 30 '12 at 7:34  
This question has been asked before and already has an answer. If those answers do not fully address your question, please [ask a new question](#).

- 4   `vector.clear()` ? – [GManNickG](#) Nov 30 '12 at 6:39
- 1   Even if you didn't find out about `clear` on your own, reading the documentation for `erase` would have been a start. -1. – [DevSolar](#) Nov 30 '12 at 6:40
- @jogojapan: That question isn't an exact duplicate. – [Marcelo Cantos](#) Nov 30 '12 at 7:02
- @MarceloCantos The OP wants to remove *all* elements from a vector, suggests a method that doesn't work and asks "how else can I do it". How is this not equivalent to asking "how to delete all items from a vector"? – [jogojapan](#) Nov 30 '12 at 7:05
- @jogojapan: Having equivalent titles doesn't make the questions identical. The other question exhibits a different type of misunderstanding, and has an additional aspect — wanting to retain the objects after removal. Those factors warrant a different response to what this question needs. – [Marcelo Cantos](#) Nov 30 '12 at 7:08

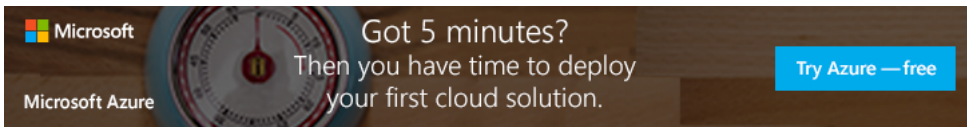
4 Answers

You call the `std::vector::clear()` method:

```
myVector.clear();
```

where I have changed the instance name from `vector` to `myVector`. It is not a good idea to use the name of a class for an instance.

answered Nov 30 '12 at 6:39  
 juanchopanza  
170k 17 241 363



`std::vector` has a `clear` member, you know:

```
void clearVector() {
    vector.clear();
}
```

Anyway, since you're erasing from the left the size will be shrinking also. This works the way you intend, although it is inefficient because it will have to do  $O(N)$  copies towards the front each iteration.

```
void clearVector() {
    for (int i = 0; i < 52; i++) {
        vector.erase(vector.begin());
    }
}
```

edited Nov 30 '12 at 6:50

answered Nov 30 '12 at 6:39



Pubby

36.2k 8 90 152

You can use `std::vector::clear()` to clear elements or swap with an empty container is much faster.

```
vec.clear();
```

or

```
std::vector<DataType>().swap(vec);
```

*Note: your variable `vector` is not good variable name, better change it to something else.*

answered Nov 30 '12 at 6:45



billz

33.9k 4 48 82

In C++11, `std::vector` has a method `shrink_to_fit` that makes the "swap-with-copy" (or "swap-with-empty") trick obsolete. You can now write `vec.clear(); vec.shrink_to_fit();` to erase all elements of the vector and decrease its capacity as well. – [Luc Touraille](#) Nov 30 '12 at 10:05

```
vec.erase(vec.begin() /* first you want delete */,
          vec.begin() + vec.size() /* 1 beyond the last you want to delete
*/);
// or if you have to erase all elements:
vec.clear();
```

so you don't need to iterate.

answered Nov 30 '12 at 6:44



ShinTakezou

6,697 14 26

1 Why not `vec.end()` ? – [GManNickG](#) Nov 30 '12 at 7:03

just to stress it deletes a range, if `last = vec.end()`, better to use directly `vec.clear()`. I've just chosen a "special" range, i.e. all the elements, maybe I should write just `N` instead – [ShinTakezou](#) Nov 30 '12 at 7:14