
View topic - How do you enable more than 10 desktops in Windows 7.

Company of Heroes 2 . Cheat Engine looks like a software Photoshop CS6 . Cheat Engine [Warning: the download is in progress] up my game so i'm unable to fully test Cheat Engine. Company Of Heroes 2Q: What is the best method to detect and remove vertically-stacked li elements in a DIV? I have a vertical stack of li elements that need to be removed. The issue I'm having is that I have many instances of this situation, and can't come up with a reliable method of detecting it, and cleaning it up. Currently, I have a simple function that removes one li element in each div containing five or more items: `$(.items li:gt(3)).each(function() { $(this).remove(); });` The problem is, with variable height divs, the remove method can remove one or more than one element, so when this is looped many times, the remaining li elements are shifted to the bottom of the div, and any one or more elements could be shifted down. My options so far have been to: Remove the element at the bottom-most of the three. Loop through the items of the div and first remove the bottom-most element, and then loop through the items and remove the remaining two, and then finally remove the third, leaving two li elements at the bottom. Loop through each li element, and if the total number of li elements is greater than three, remove the last three, leaving one at the bottom. Loop through each div, and loop through the li elements within the div. For each li element, check if it's total number of li elements is less than three, then remove the element at the bottom. None of these options seem very elegant, and would have issues with, say, an item with three li elements after it and one after the second. Ideas? Update: The solution I came up with is to listen to mousedown on the div, check the number of li elements, and then loop through the li elements, removing the bottom-most element. A: These are the options I've come up with, along with a test for each. Option 1 This would likely be the most robust solution. In order to prevent an error, we can do some more testing, so we can be sure we aren't removing the last element.

[Download](#)

