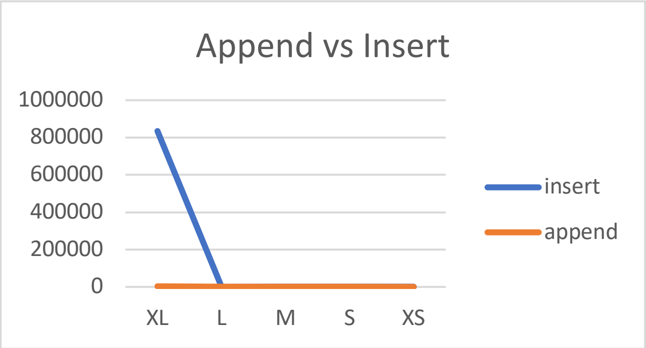
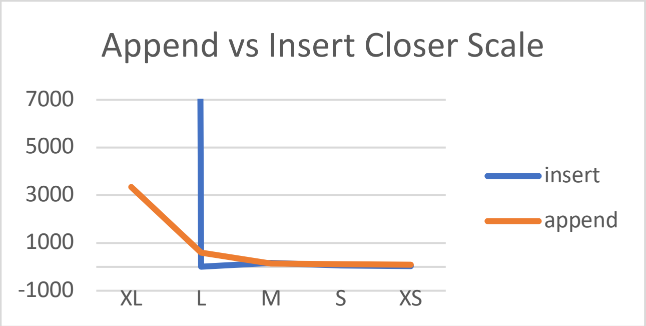
doublerAppend vs doublerInsert Results

	insert	append
XL	834926	3345.73
L	6.58043	604.407
M	163.689	148.826
S	52.428	106.726
XS	41.165	89.776





Max Y-axis set to 7000 to better view append line

The doublerInsert function scales at O(n^2) while the doublerAppend function scales at O(n).

The Append function is faster due to the .push() method used; .push() has a constant time complexity because it is simply adding an element and assigning the new element the next index in the array.

On the other hand, .unshift() has linear time complexity because the nature of this method adding an element of the beginning of the array means the rest of the array must shift in index. Therefore, .unshift() has a linear time complexity of o(n).

In both functions, looping through the array has a time complexity of O(n) by default as n is the length of the array. Since .push() has a time complexity of O(1), this constant would get dropped because O(n) is much more significant. The .unshift() method with time complexity o(n) on top of the for loop causes the insert method to have O(n^2).