

## Code Challenge #12 Insertion Sort (Easy)

Difficulty: ■ Category: Insertion Sort Successful Submissions: 39,286+

### Insertion Sort ● ★

Write a function that takes in an array of integers and returns a sorted version of that array. Use the Insertion Sort algorithm to sort the array.

If you're unfamiliar with Insertion Sort, we recommend watching the Conceptual Overview section of this question's video explanation before starting to code.

**Sample Input**

```
array = [8, 5, 2, 9, 5, 6, 3]
```

**Sample Output**

```
[2, 3, 5, 5, 6, 8, 9]
```

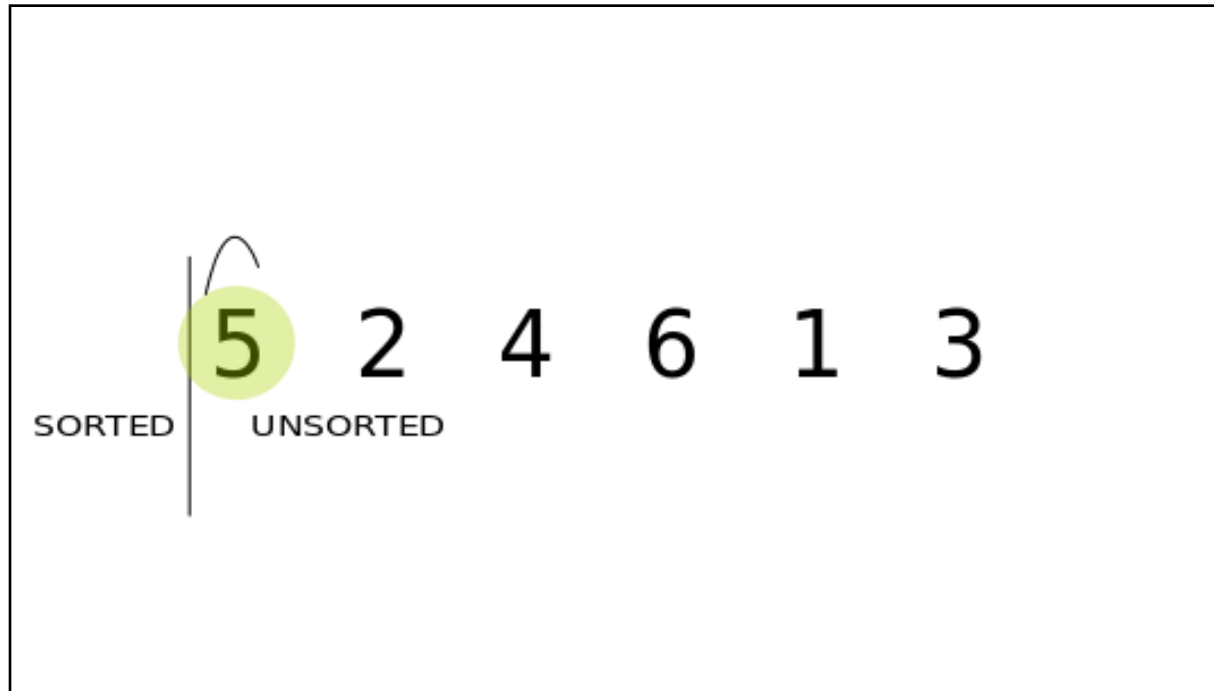
### Solution #1

```
1. function insertionSort(array) {
2.   for (let i = 1; i < array.length; i++) {
3.     let j = i;
4.     while (j > 0 && array[j] < array[j - 1]){
5.       swap(j, j - 1, array);
6.       j -= 1;
7.     }
8.   }
9.   return array;
10. }
11.
12. function swap(i, j, array) {
13.   const temp = array[j];
14.   array[j] = array[i];
15.   array[i] = temp;
16. }
17.
```

### Explanation

Insertion sort is a sorting algorithm where we assume that the first item is a sorted item and compare its value to the next item. If the next item is smaller than it, we

swap places. We then move on to next item and compare it to the previous items one at a time to see if it is smaller than it. We keep repeating this process till we have a sorted array. The largest item would be the last item in the array to the right. See animated gif below.



The solution for this uses two loops. The first loop is a for loop where we use  $i = 1$  as long as  $i < \text{array.length}$  we increment  $i$ . Within the for loop we have another let variable called  $j$  which is equal to  $i$ . We then use a while loop to check values to see if the right value is smaller than the left value. In the while loop we state that as long as  $j > 0$  and  $\text{array}[j] < \text{array}[j - 1]$  we swap values using a helper function called swap. We then decrement  $j$  by one in order to keep checking the values until  $j$  goes to 0 aka the first item in array. When this happens the for loop continues to next item and the while loop keeps running again. Once all the checks are finished, we return the array. The helper function called swap is a simple function that takes in three arguments. The three arguments are  $i$ ,  $j$  and array. Inside the helper function we assign the value of  $\text{array}[j]$  to a const variable called temp. We then assign the value of  $\text{array}[i]$  to  $\text{array}[j]$ . We then assign the value of variable temp to  $\text{array}[i]$ . The Big O is  $O(n^2)$  for worst case scenario and average case. The best case is  $O(n)$  if the array is already sorted.