**Bubble Sort**

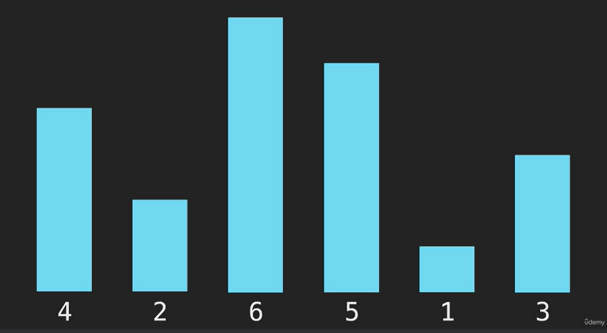
Starting from the first item, we compare that with the next item in the array.

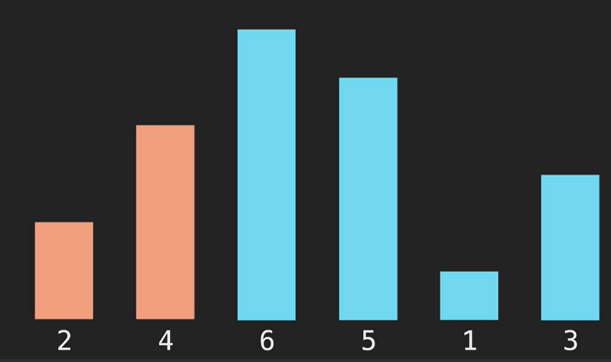
If the next item is bigger, we swap the items.

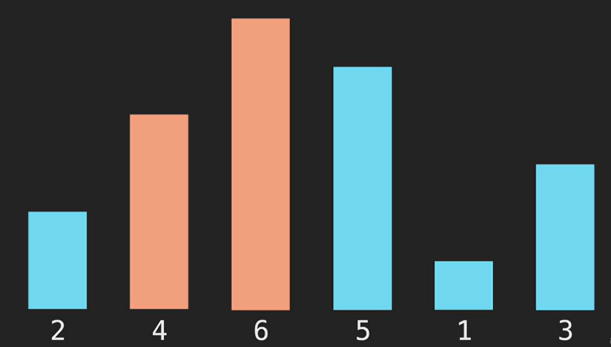
Once we get to the end of the array, the largest item will have been the last item.

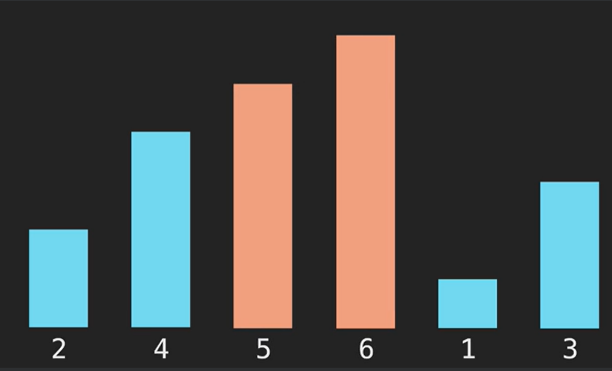
Then we repeat these steps again starting from the beginning of the array.

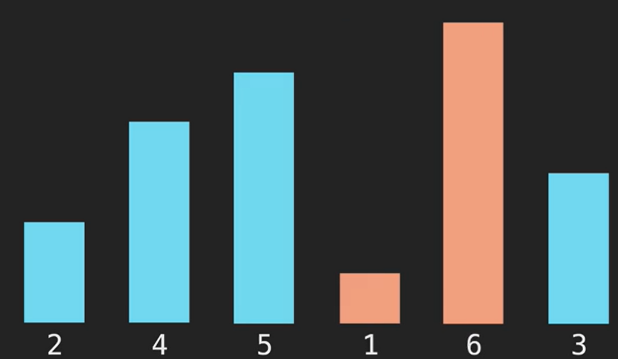
We do this until all the items are sorted.



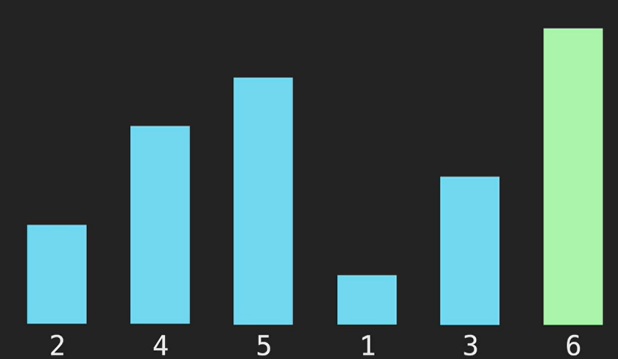




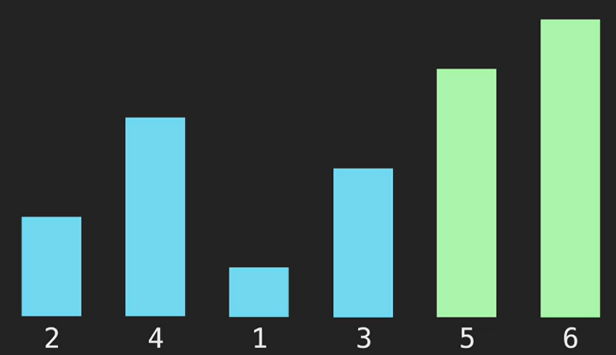


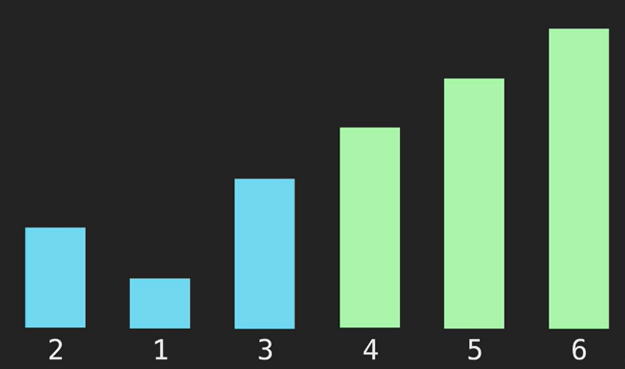


Now 6 is sorted

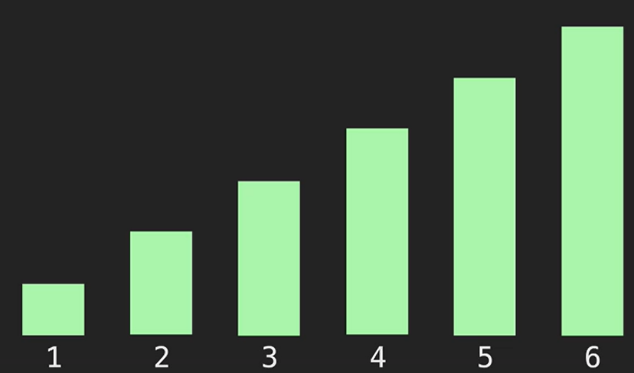


We repeat starting from beginning.





Do this until the array is sorted.



function bubbleSort(array) {

    for (let i = array.length - 1; i > 0; i--) {

        for (let j = 0; j < i; j++) {

            if (array[j] > array[j + 1]) {

                let temp = array[j];

                array[j] = array[j + 1];

                array[j + 1] = temp;

            }

        }

    }

    return array;

}

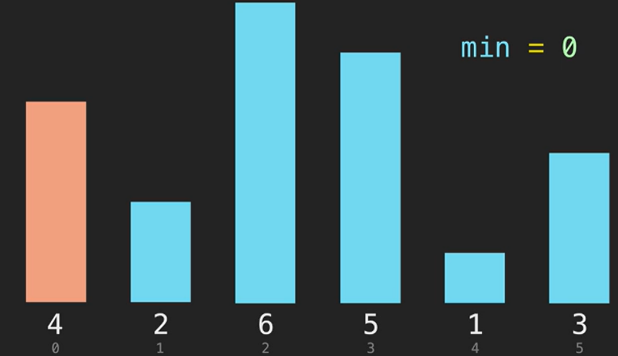
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Selection Sort**

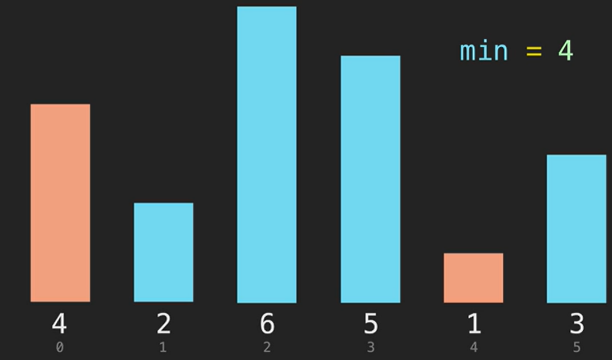
We use the array indices to sort.

Loop through the array and store the index of the lowest value in a variable called min.

Once we have a min value, we swap the value at the indices.

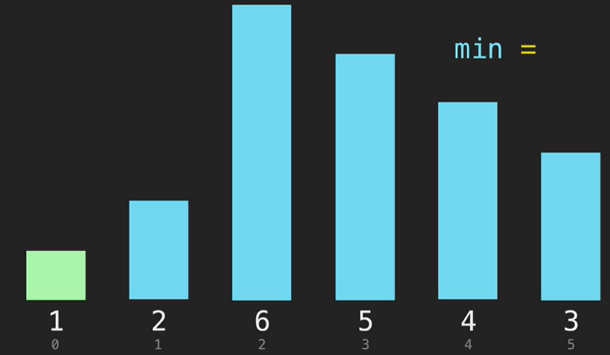


Once we loop through on the first iteration, min is index 4

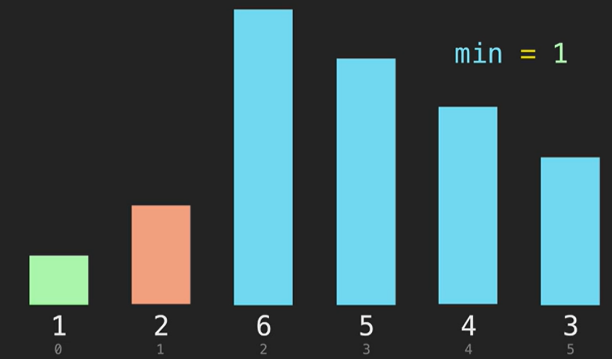


Once we find the min value on the first iteration, we switch the value at index 0 with the value at the min index.

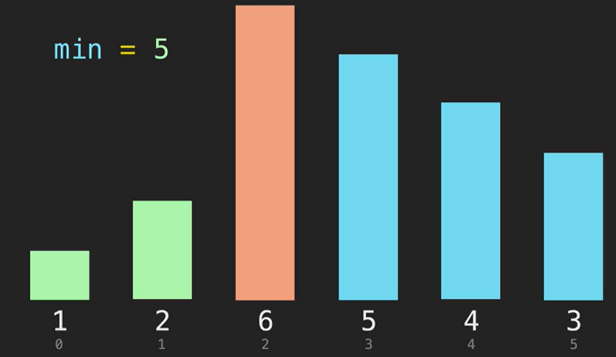
Here, the values at index 0 and index 4 are switched.

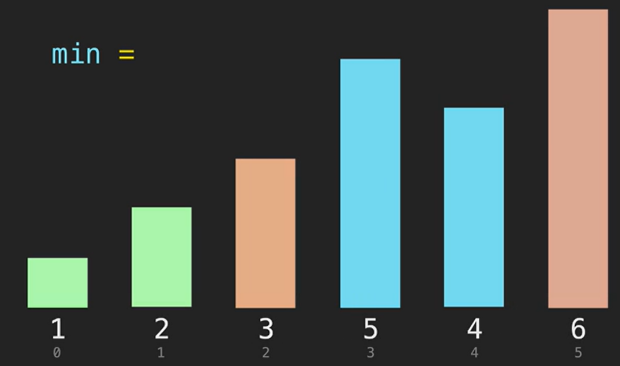


On second iteration, we start at index 1 and set that to the min.

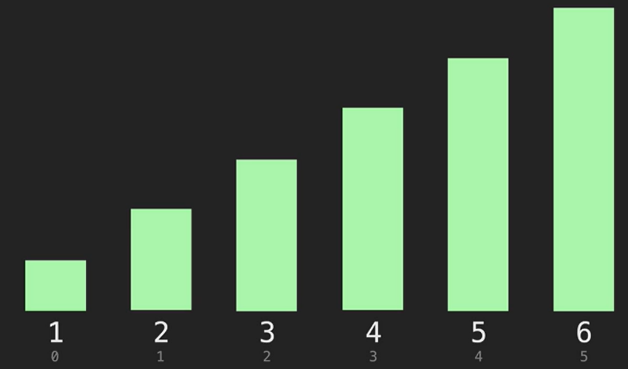


Iterate through the array again and repeat the steps swapping the min value.





Repeat until the array is sorted.



function selectionSort(array) {

    for (let i = 0; i < array.length - 1; i++) {

        let min = i;

        for (let j = i + 1; j < array.length; j++) {

            if (array[j] < array[min]) {

                min = j;

            }

        }

        if (i !== min) { // If the value at i is already the min, do not swap anything.

            let temp = array[i];

            array[i] = array[min];

            array[min] = temp;

        }

    }

    return array;

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Insertion Sort**

We start at the second item in the array.

Compare it to the item before it.

If the second item is lower than the first item, then swap them.

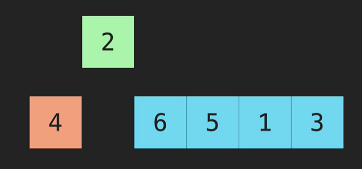
Now we move onto the third item in the array.

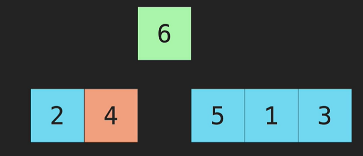
Compare it to the item before it.

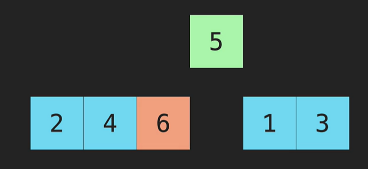
If it is lower, then swap the items.

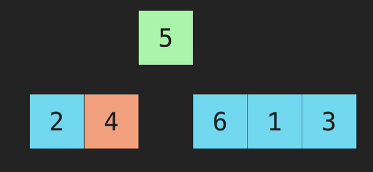
If not, then compare it to the item before that one until you reach the beginning of the array.

If it is lower than any item during the iteration, then swap those items.

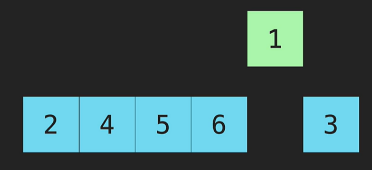


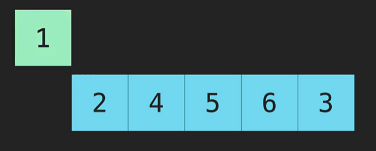


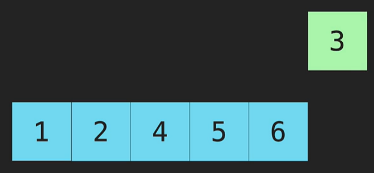


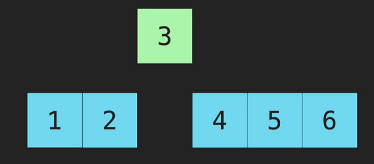














function insertionSort(array) {

    for (let i = 1; i < array.length; i++) { // i=1 because we start at the second item in array

        let temp = array[i];

        for (var j = i - 1; array[j] > temp && j > -1; j--) { // we use var j-1 because we need j to be accessible outside of that forloop. Using ‘let’ will not allow us to access j outside of forloop.

            array[j + 1] = array[j];

        }

        array[j + 1] = temp;

    }

    return array;

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*