**Sectional Array Sort**

12585% of 6334 of254[kodejuice](http://www.codewars.com/users/kodejuice)

Java

* [TRAIN AGAIN](http://www.codewars.com/kata/sectional-array-sort/train/java)
* [NEXT KATA](http://www.codewars.com/trainer/java)

Details

[Solutions](http://www.codewars.com/kata/sectional-array-sort/solutions/java)

[Forks (1)](http://www.codewars.com/kata/sectional-array-sort/forks/java)

[Discourse (18)](http://www.codewars.com/kata/sectional-array-sort/discuss/java)

* Add to Collection
* |
* Share this kata:

In this kata you are given an array to sort but you're expected to start sorting from a specific position of the array (in ascending order) and optionally you're given the number of items to sort.

Examples:

SectionalArray.sort([1, 2, 5, 7, 4, 6, 3, 9, 8], 2) //=> [1, 2, 3, 4, 5, 6, 7, 8, 9]

SectionalArray.sort([9, 7, 4, 2, 5, 3, 1, 8, 6], 2, 5) // => [9, 7, 1, 2, 3, 4, 5, 8, 6]

Documentation:

SectionalArray.sort(array, start, length);

* array - array to sort
* start - position to begin sorting
* length - number of items to sort (optional)

if the **length** argument is not passed or is zero, you sort all items to the right of the start postiton in the array

ALGORITHMS

ARRAYS

SORTING

DATA STRUCTURES

<http://www.codewars.com/kata/sectional-array-sort/java>

* [OlegDemura](http://www.codewars.com/users/OlegDemura)
* **import java.util.Arrays;**
* **class SectionalArray {**
* **static int[] sort(final int[] array, final int start, final int... length) {**
* **if (length.length==0 || length[0]>array.length-start){**
* **Arrays.sort(array, start, array.length);**
* **}else {**
* **Arrays.sort(array, start, start+length[0]);**
* **}**
* **return array;**
* **}**
* **}**
  + - Best Practices0
    - Clever2
  + 0
  + [Fork](http://www.codewars.com/kumite/new?group_id=594e001b4ccb1576d20000a0&review_id=5911e63f152c8c15c4000a45)
  + [Link](http://www.codewars.com/kata/reviews/5911e63f152c8c15c4000a45/groups/594e001b4ccb1576d20000a0)
* [efvilloldo](http://www.codewars.com/users/efvilloldo)
* **import java.util.\*;**
* **class SectionalArray {**
* **static int[] sort(final int[] array, final int start, final int... length) {**
* **Integer j = length.length > 0 ? length[0] : array.length;**
* **if (j+start<array.length) j=j+start;**
* **else j=array.length;**
* **int[] b = Arrays.copyOf(array, array.length);**
* **Arrays.sort(b, start, j);**
* **return b;**
* **}**
* **}**
  + - Best Practices0
    - Clever0
  + 0
  + [Fork](http://www.codewars.com/kumite/new?group_id=594bf68df3ad3179b0000162&review_id=5911e63f152c8c15c4000a45)
  + [Link](http://www.codewars.com/kata/reviews/5911e63f152c8c15c4000a45/groups/594bf68df3ad3179b0000162)
* [squla](http://www.codewars.com/users/squla)
* **import java.util.Arrays;**
* **class SectionalArray {**
* **static int[] sort(final int[] array, final int start, final int... length) {**
* **int l = array.length;**
* **if(length.length > 0 && start + length[0] <= array.length) l = start + length[0];**
* **Arrays.sort(array, start, l);**
* **return array;**
* **}**

**}**