**Find the middle element**

46496% of 4971,037 of2,221[myTerminal](http://www.codewars.com/users/myTerminal)

JavaScript

* [TRAIN AGAIN](http://www.codewars.com/kata/find-the-middle-element/train/javascript)
* [NEXT KATA](http://www.codewars.com/trainer/javascript)

Details

[Solutions](http://www.codewars.com/kata/find-the-middle-element/solutions/javascript)

[Discourse (106)](http://www.codewars.com/kata/find-the-middle-element/discuss/javascript)

* Add to Collection
* |
* Share this kata:

As a part of this Kata, you need to create a function that when provided with a triplet, returns the index of the numerical element that lies between the other two elements.

The input to the function will be an array of three distinct numbers (Haskell: a tuple).

For example:

gimme([2, 3, 1]) => 0

*2* is the number that fits between *1* and *3* and the index of *2* in the input array is *0*.

Another example (just to make sure it is clear):

gimme([5, 10, 14]) => 1

<http://www.codewars.com/kata/find-the-middle-element/javascript>

<html>

    <head>

        <title>Hola mundo</title>

        <script type="text/javascript">

        var gimme = function (inputArray) {

            // Implement this function

            //var copia = inputArray;

            var copia = inputArray.slice();

            inputArray.sort(function(a, b){return a-b});

            /\*for(var i =0; i<inputArray.length; i++) {

                document.write(inputArray[i]+ " ");

            }

            document.write("<br>");\*/

            return copia.indexOf(inputArray[1]);

        };

     document.write(gimme([5, 10, 14]));

                       //5 10 14

        </script>

    </head>

    <body>

    </body>

</html>

**function gimme(a) {**

**return a.indexOf(a.concat().sort(function(a, b) { return a - b })[1])**

**}**