**Euclidean distance in n dimensions**

4091% *of* 2818 *of* 84[kingcobra](https://www.codewars.com/users/kingcobra)

JavaScript

* [Train Again](https://www.codewars.com/kata/euclidean-distance-in-n-dimensions/train/javascript)
* [Next Kata](https://www.codewars.com/trainer/javascript)

Details

[Solutions](https://www.codewars.com/kata/euclidean-distance-in-n-dimensions/solutions/javascript)

[Discourse (3)](https://www.codewars.com/kata/euclidean-distance-in-n-dimensions/discuss/javascript)

* Add to Collection
* |
* Share this kata:

Implement a function to calculate the distance between two points in n-dimensional space. The two points will be passed to your function as arrays of the same length (tuples in Python).

Round your answers to two decimal places.

<https://www.codewars.com/kata/euclidean-distance-in-n-dimensions/javascript>

<html>

    <head>

        <title>problemas codewars</title>

        <script type="text/javascript">

function euclideanDistance(point1, point2){

var distance = 0;

for (var i = 0; i < point1.length; i++) {

distance += Math.pow(point1[i] - point2[i], 2);

}

return parseFloat(Math.pow(distance, .5).toFixed(2));

}

[p1,p2] = [[-1,2],[2,4]]

//[p1,p2] = [[-1,2,5],[2,4,9]]

document.write(euclideanDistance(p1,p2));

        </script>

    </head>

    <body>

    </body>

</html>