**Odd-Even String Sort**

5296% *of* 756 *of* 358[Joz](https://www.codewars.com/users/Joz)

C#

* [Train Again](https://www.codewars.com/kata/odd-even-string-sort/train/csharp)
* [Next Kata](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/odd-even-string-sort/solutions/csharp)

[Forks (3)](https://www.codewars.com/kata/odd-even-string-sort/forks/csharp)

[Discourse (10)](https://www.codewars.com/kata/odd-even-string-sort/discuss/csharp)

* Add to Collection
* |
* Share this kata:

Given a string *S.* You have to return another string such that even-indexed and odd-indexed characters of *S* are grouped and groups are space-separated (see sample below)

Note:

0 is considered to be an even index.

All input strings are valid with no spaces

input: **'CodeWars'** output **'CdWr oeas'**

S[0] = 'C'

S[1] = 'o'

S[2] = 'd'

S[3] = 'e'

S[4] = 'W'

S[5] = 'a'

S[6] = 'r'

S[7] = 's'

Even indices 0, 2, 4, 6, so we have **'CdWr'** as the first group  
odd ones are 1, 3, 5, 7, so the second group is **'oeas'**  
And the final string to return is **'Cdwr oeas'**

<https://www.codewars.com/kata/odd-even-string-sort/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

public static string SortMyString(string s)

{

string pares = "", impares = "";

for (int i = 0; i < s.Length; i++)

{

if (i % 2 == 0)

{

pares += s[i];

}

else

{

impares += s[i];

}

}

return pares + " " + impares;

}

static void Main(string[] args)

{

string input= "CodeWars" ;

Console.WriteLine(SortMyString(input));

Console.ReadLine();

}

}

}