**6 kyu**

**String Letter Counting**

1095% of2066[MinimumEntropy](https://www.codewars.com/users/MinimumEntropy)

C#

* [TRAIN AGAIN](https://www.codewars.com/kata/string-letter-counting/train/csharp)
* [NEXT KATA](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/string-letter-counting/solutions/csharp)

[Discourse (5)](https://www.codewars.com/kata/string-letter-counting/discuss/csharp)

* Add to Collection
* |
* Share this kata:

Take a string str and return a string that is made up of the number of occurances of each english letter in str, followed by that letter. The string shouldn't contain zeros; leave them out.

An empty string, or one with no letters, should return an empty string.

Ignore all case.

str will never be null.

For Example:

Kata.StringLetterCount("This is a test sentence.") == "1a1c4e1h2i2n4s4t"

Kata.StringLetterCount("") == ""

Kata.StringLetterCount("555") == ""

<https://www.codewars.com/kata/string-letter-counting/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

public static string StringLetterCount(string str)

{

str = str.ToLower();

Dictionary<char, int> diccio = new Dictionary<char, int>();

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

{

if(diccio.ContainsKey(str[i])) diccio[str[i]]++;

else diccio[str[i]] = 1;

}

string res = "";

for(char ch ='a'; ch<='z'; ch++)

{

if(diccio.ContainsKey(ch)) if(diccio[ch] > 0 ) res += diccio[ch] + "" + ch;

}

return res;

}

static void Main(string[] args)

{

//Console.WriteLine(StringLetterCount("The quick brown fox jumps over the lazy dog."));

Console.WriteLine(StringLetterCount("aaaa"));

Console.WriteLine(StringLetterCount(""));

Console.ReadLine();

}

}

}