**6 kyu**

**Grouping Consecutive Numbers**

7297% of1750[WillKirkby](https://www.codewars.com/users/WillKirkby)

C#

* [TRAIN AGAIN](https://www.codewars.com/kata/grouping-consecutive-numbers/train/csharp)
* [NEXT KATA](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/grouping-consecutive-numbers/solutions/csharp)

[Discourse (8)](https://www.codewars.com/kata/grouping-consecutive-numbers/discuss/csharp)

* Add to Collection
* |
* Share this kata:

Given an array of integers, produce a string of the numbers in ascending order, with runs of consecutive numbers compacted into a "start-end" form, as follows:

{1,2,3,5} should produce an output string of "1-3, 5"

Duplicate numbers should be ignored, so:

{1,2,2,3} should produce an output string of "1-3"

Numbers should be sorted in ascending order, so:

{5,3,4,1,2} should produce an output string of "1-5"

<https://www.codewars.com/kata/grouping-consecutive-numbers/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

public static string GenerateGroupings(int[] input)

{

Array.Sort(input);

HashSet<int> hash = new HashSet<int>(input);

input = hash.ToArray();

List<List<int>> lista = new List<List<int>>();

int pre = input[0];

List<int> aux = new List<int>();

aux.Add(pre);

//lista.Add(aux);

for (int i = 1; i < input.Length; i++)

{

if (pre + 1 != input[i])

{

lista.Add(aux);

aux = new List<int>();

pre = input[i];

aux.Add(pre);

}

else

{

aux.Add(input[i]);

pre = input[i];

}

}

if (aux.Count > 0) lista.Add(aux);

List<string> concat = new List<string>();

//string s = "";

foreach(List<int> l in lista)

{

//foreach(int item in l)

//{

// Console.Write(item + " ");

//}

//Console.WriteLine();

if(l.Count > 1)

{

concat.Add( l[0] + "-" + l.Last());

}

else

{

concat.Add( l[0] + "");

}

}

return String.Join(", ", concat);

}

static void Main(string[] args)

{

//int[] arr = { 3,9,1,7,2,8,8,4,5,6 };

int[] arr = { 1, 2, 3, 5 }; //"1-3, 5"

//int[] arr = { 1, 3};

//int[] arr = { 1, 2, 3 };

//int[] arr = { 1 };

//int[] arr = { 0, 1 };

Console.WriteLine(GenerateGroupings(arr));

Console.ReadLine();

}

}

}