

Контест 13

11 янв 2022, 18:48:55

старт: 11 июн 2021, 12:30:00

финиш: 20 июн 2021, 23:59:59

длительность: 9д. 11ч.

начало: 11 июн 2021, 12:30:00

конец: 20 июн 2021, 23:59:59

В. Новый API

| | |
|---------------------|-------------------|
| Ограничение времени | 10 секунд |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод |
| Вывод | стандартный вывод |

Вася успешно прошел курс программирования в университете и решил пройти стажировку в крупной компании на позиции бэкенд-разработчика. У компании есть филиалы по всей России, поэтому при запросе городов, где эти филиалы расположены, сервер возвращает ответ в виде набора `id` федеральных округов и городов. Первое задание Васи – это поменять формат API и вместо набора `id` возвращать строковые значения сущностей из базы данных. Так как Вася работает только первый день, у него совершенно не получается решить эту задачу, поэтому он просит помощи у вас.

Формат ввода

Имеются два текстовых файла: `areas.txt` – список федеральных округов и `cities.txt` – список городов. В этих файлах записаны данные в формате "`<id> <name>`".
Сначала программа принимает на вход целое положительное число `n` – количество федеральных округов, в которых расположены филиалы компании. Далее, `n` раз, программа принимает на вход следующие данные: с новой строки, `id` федерального округа, а затем, с новой строки, `id` городов, разделенные пробелом.

Формат вывода

Необходимо вывести федеральные округа и города, каждый с новой строки, в формате: "`<название федерального округа>: <название города 1>; <название города 2>; ... <название города k>`";

Пример

| | |
|------------------------|--|
| Ввод | Вывод |
| 2 | Московская область: г. Москва - город федерального значения; Городской округ Химки; Зеленоградский городской |
| 69 | Ленинградская область: г. Санкт-Петербург - город федерального значения; Выборгский муниципальный район; Лен |
| 2511 417 153 2175 2045 | |
| 28 | |
| 2513 1916 609 | |

Примечания

В систему необходимо сдавать **один** файл `Program.ConverterAndLoader.cs`.
Для чтения и записи используйте `"StreamReader"/"StreamWriter"`.

Набрать здесь

Отправить файл

```
1 using System;
2 using System.IO;
3 using System.Collections.Generic;
4 using System.Linq;
5
6 public partial class Program
7 {
8     private static Dictionary<string, List<string>> Convert(KeyPredicate keyPredicate, ValuePredicate valuePredicate, Dictionary<int, List<string>> data)
9     {
10         Dictionary<string, List<string>> result = new Dictionary<string, List<string>>();
11
12         foreach (var el in data.Keys)
13         {
14             var idList = (from id in data[el] select valuePredicate(id)).ToList();
15             result.Add(keyPredicate(el), idList);
16         }
17
18         return result;
19     }
20
21     private static void LoadAreas()
22     {
23         using (var reader = new StreamReader(new FileStream("areas.txt", FileMode.Open)))
24         {
25             while (!reader.EndOfStream)
26             {
27                 int count = 0;
28
29                 string input = reader.ReadLine();
30
31                 for (int i = 0; i < input.Length; i++)
32                 {
33                     if ((input[i] < '0' || input[i] > '9') && input[i] != ' ')
34                         break;
35                     else
36                         count++;
37                 }
38             }
39         }
```

[Отправить](#)[Предыдущая](#)[Следующая](#)