Практика №2

Создание LINQ-запросов в C# для выборки данных из коллекции.  
Технология LINQ to Object.

|  |  |
| --- | --- |
| **Дисциплина** | Технологии создания программного обеспечения |
| **Институт** | Перспективных технологий и индустриального программирования |
| **Кафедра** | Индустриального программирования |
| **Вид учебного материала** | Практика |
| **Преподаватель** | Евдошенко Олег Игоревич |
| **Семестр** | 1 семестр, 2023-2024 |

Задачи для самостоятельного выполнения:

Внимание! Данные необходимо хранить в списках (коллекции).

1. Даны строковые последовательности *A* и *B*; все строки в каждой последовательности различны, имеют ненулевую длину и содержат только цифры и заглавные буквы латинского алфавита. Получить последовательность всевозможных комбинаций вида «*Ea=Ев*», где *Еа -* некоторый элемент из *A*, *Ев -* некоторый элемент из *B*, причем оба элемента оканчиваются цифрой (например, «AF3=D78»). Упорядочить полученную последовательность в лексикографическом порядке по возрастанию элементов *EA*, а при одинаковых элементах *EA* - в лексикографическом порядке по убыванию элементов *EB* (для перебора комбинаций использовать методы SelectMany и Select).
2. Даны последовательности положительных целых чисел *A* и *B*; все числа в последовательности *A* различны. Получить последовательность строк вида «*S*:*E*», где *S* обозначает среднее арифметическое тех чисел из *B*, которые оканчиваются на ту же цифру, что и число *E* - один из элементов последовательности *A* (например, «74:23»); если для числа *E* не найдено ни одного подходящего числа из последовательности *B*, то в качестве *S* указать 0. Расположить элементы полученной последовательности по возрастанию значений найденных сумм, а при равных суммах - по убыванию значений элементов *A*.
3. В организации имеется 3 отдела. В каждом отделе имеется от 3 до 5 сотрудников. Используя группировку по отделу, вывести список сотрудников и средний оклад по каждому отделу. Определите долю суммы окладов всех сотрудников одного отдела в общей сумме окладов по всему предприятию.
4. Дано целое число *K -* код одного из клиентов фитнес-центра. Исходная последовательность содержит сведения о клиентах этого фитнес-центра. Каждый элемент последовательности включает следующие целочисленные поля:

*<Код клиента> <Год> <Номер месяца>  
<Продолжительность занятий (в часах)>*

Для каждого года, в котором клиент с кодом *K* посещал центр, определить месяц, в котором продолжительность занятий данного клиента была наименьшей для данного года (если таких месяцев несколько, то выбирать первый из этих месяцев в исходном наборе; месяцы с нулевой продолжительностью занятий не учитывать). Сведения о каждом годе выводить на новой строке в следующем порядке: наименьшая продолжительность занятий, год, номер месяца. Упорядочивать сведения по возрастанию продолжительности занятий, а при равной продолжительности - по возрастанию номера года. Если данные о клиенте с кодом *K* отсутствуют, то записать в результирующий файл строку «Нет данных».

**Указание.** Для отбора данных, связанных с клиентом K, использовать метод Where. Затем выполнить группировку по полю «год» и для каждой полученной последовательности выбрать требуемый месяц с помощью сортировки по набору ключей «продолжительность занятий, номер месяца». Обработку особой ситуации, связанной с отсутствием требуемых данных, выполнять с использованием метода DefaultIfEmpty с параметром «Нет данных».

1. Исходная последовательность содержит сведения об абитуриентах. Каждый элемент последовательности включает следующие поля:

*<Номер школы> <Год поступления> <Фамилия>*

Для каждого года, присутствующего в исходных данных, вывести число различных школ, которые окончили абитуриенты, поступившие в этом году (вначале указывать число школ, затем год). Сведения о каждом годе выводить на новой строке и упорядочивать по возрастанию числа школ, а для совпадающих чисел — по возрастанию номера года.

1. Из последовательности (см. п.5) определить, в какие годы общее число абитуриентов для всех школ было наибольшим и наименьшим, и вывести это число, а также годы, в которые оно было достигнуто (годы упорядочивать по возрастанию, каждое число выводить на новой строке).
2. Исходная последовательность содержит сведения о задолжниках по оплате коммунальных услуг, живущих в 144-квартирном 9-этажном доме. Каждый элемент последовательности включает следующие поля:

*<Задолженность> <Фамилия> <Номер квартиры>*

Задолженность указывается в виде дробного числа (целая часть — рубли, дробная часть — копейки). В каждом подъезде на каждом этаже располагаются по 4 квартиры. Для каждого из 4 подъездов дома найти трех жильцов с наибольшей задолженностью и вывести сведения о них: задолженность (выводится с двумя дробными знаками), номер подъезда, номер квартиры, фамилия жильца. Считать, что в наборе исходных данных все задолженности имеют различные значения. Сведения о каждом задолжнике выводить на отдельной строке и упорядочивать по убыванию размера задолженности (номер подъезда при сортировке не учитывать). Если в каком-либо подъезде число задолжников меньше трех, то включить в полученный набор всех задолжников этого подъезда.

1. Даны последовательности A и B, включающие следующие поля: *A: категория, артикул товара, страна производитель; B: артикул товара, цена, название магазина.* Для каждой категории товаров определить количество магазинов, предлагающих товары данной категории, а также количество стран, в которых произведены товары данной категории, представленные в магазинах (вначале выводится количество магазинов, затем название категории, затем количество стран). Если для некоторой категории не найдено ни одного товара, представленного в каком- либо магазине, то информация о данной категории не выводится. Сведения о каждой категории выводить на новой строке и упорядочивать по убыванию количества магазинов, а в случае одинакового количества — по названиям категорий в алфавитном порядке.
2. Даны последовательности A, B и C, включающие следующие поля: *A: улица, код потребителя, год рождения; B: страна производителя, категория, артикул товара; C: артикул товара, код потребителя, название магазина.* Для каждого года рождения из A определить страну, в которой было произведено максимальное количество товаров, приобретенных потребителями этого года рождения (вначале выводится год, затем название страны, затем максимальное количество покупок). Если для некоторой пары «год-страна» отсутствует информация о проданных товарах, то эта пара не обрабатывается (в частности, если потребители некоторого года рождения не сделали ни одной покупки, то информация об этом годе не выводится). Если для какого-либо года рождения имеется несколько стран с наибольшим числом приобретенных товаров, то выводятся данные о первой из таких стран (в алфавитном порядке). Сведения о каждом годе выводить на новой строке и упорядочивать по убыванию номера года.