

## Лабораторная работа №4

### Группировка и объединение элементов коллекции с помощью LINQ-запросов в C#

В данном примере используем стандартный шаблон LINQ-запроса для группировки элементов списка. Программа формирует список продуктов питания. Следует организовать такие **LINQ-запросы**, которые разделят искомый список на две группы по критерию цены (больше или меньше 90 руб. за единицу) и вычислят среднюю цену по каждой группе продуктов.

Для решения этой задачи запустим **Visual Studio** и выберем проект шаблона **Windows Forms Application**, укажем любое имя. Далее, попав в конструктор формы, из панели элементов **Toolbox** перетащим текстовое поле **TextBox** для вывода в него групп списка. В свойствах текстового поля разрешим ввод множества строк, для этого свойство **Multiline** переведем в состояние true. Затем на вкладке программного кода введем текст, следуя шагам:

1. Объявить класс *Продукт* с полями: наименование и цена

2. Создадим список *Продукты* и заполним его данными

```
List<тип_данных> Продукты = new List<тип_данных> {  
    new Продукт {Наименование="Кефир", Цена=27.00}  
    . . . };
```

3. Создадим **LINQ-запрос** для группировки и отбора продуктов с ценой больше 90

```
var Запрос1 =  
    from П in Продукты  
    group П by new  
        {  
            Критерий = условие, (П.Цена>90)  
        }  
    into g  
    select g;
```

4. Создадим второй **LINQ-запрос** для расчета средней цена по группе

```
var Запрос2 =  
    from p in Продукты
```

```

        group p by условие into g
        select new
        {
            g.Key,
            СредЦенаПоГруппе = g.Average(p => p.Цена)
        };

Single СредЦенаПоГруппе1 = Запрос2.ElementAt(0).СредЦенаПоГруппе;
Single СредЦенаПоГруппе2 = Запрос2.ElementAt(1).СредЦенаПоГруппе;

```

## 5. Вывести результаты обоих запросов в текстовое поле

```

foreach (var Группа in Запрос1)
{
    if (Группа.Критерий == false)
        textBox1.Text += "\r\nЦены 90 руб или меньше:";
    else
        textBox1.Text += "\r\nЦены больше 90 руб:";
    foreach (var Прод in Группа)
    {
        //Вывод в TextBox
    }
    if (Группа.Критерий == false)
        textBox1.Text += String.Format(
            "\r\nСредняя цена по данной группе = {0} руб.\r\n",
            СредЦенаПоГруппе2);
    else
        textBox1.Text += String.Format(
            "\r\nСредняя цена по данной группе = {0} руб.\r\n",
            СредЦенаПоГруппе1);
}

```

В начале программы формируем список продуктов питания. Для этого объявляем новый класс *Продукт*, в котором для каждого продукта задаем два поля: наименование продукта и его цену. После заполнения списка продуктов, задаем первый **LINQ-запрос**, обеспечивающий деление списка на две группы. Вторым **LINQ-запрос** вычисляет среднюю цену продукта по каждой группе. Вывод результатов обоих запросов организуем с помощью двух вложенных циклов **foreach**.

## Задачи для самостоятельного выполнения

1. Имеются сведения о студентах (коллекция): ФИО, группа, размер стипендии. Сгруппировать студентов по группам и вывести список студентов по каждой группе и средний размер стипендии.

2. 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.
3. Даны строковые последовательности  $A$  и  $B$ ; все строки в каждой последовательности различны и имеют ненулевую длину. Получить последовательность строк вида « $E:N$ », где  $E$  обозначает один из элементов последовательности  $A$ , а  $N$  — количество элементов из  $B$ , начинающихся с того же символа, что и элемент  $E$  (например, «abc:4»); количество  $N$  может быть равно 0. Порядок элементов полученной последовательности должен определяться исходным порядком элементов последовательности  $A$ .

*Указание.* Использовать метод GroupJoin.

4. Дана целочисленная последовательность  $A$ . Сгруппировать элементы последовательности  $A$ , оканчивающиеся одной и той же цифрой, и на основе этой группировки получить последовательность строк вида « $D:S$ », где  $D$  — ключ группировки (т. е. некоторая цифра, которой оканчивается хотя бы одно из чисел последовательности  $A$ ), а  $S$  — сумма всех чисел из  $A$ , которые оканчиваются цифрой  $D$ . Полученную последовательность упорядочить по возрастанию ключей.

*Указание.* Использовать метод GroupBy.

5. Дана последовательность непустых строк  $A$ , содержащих только заглавные буквы латинского алфавита. Для всех строк, начинающихся с одной и той же буквы, определить их суммарную длину и получить последовательность строк вида « $S-C$ », где  $S$  — суммарная длина всех строк из  $A$ , которые начинаются с буквы  $C$ . Полученную последовательность упорядочить по убыванию числовых значений сумм, а при равных значениях сумм — по возрастанию кодов символов  $C$ .

