

СОЕДИНЕНИЕ КОЛЛЕКЦИЙ ПРИ ПОМОЩИ МЕТОДОВ LINQ

НА ЯЗЫКЕ C#

Баймухаметов А. Д.

Студент 3 курса

Направление «Прикладная информатика»

Факультет физики и математики

г. Бирск ФГБОУ ВО Бирский филиал УУНиТ

Аннотация: В статье рассматриваются основные методы объединения коллекций на языке программирования C# используя библиотеку LINQ.

Ключевые слова: Join, GroupJoin, Zip.

Основными методами объединения коллекций из библиотеки LINQ являются

- Метод `Join()`- объединяет две последовательности на основании функции селектора ключа и извлекает два значения;
- Метод `GroupJoin()` – объединяет две последовательности на основании функции селектора ключа с последующей группировкой для каждого элемента, полученных при сопоставлении данных;
- Метод `Zip()` – поочерёдно объединяет элементы из текущей последовательности с элементами из второй последовательности. То есть первый элемент из первой коллекции объединяется с первым элементом из второй коллекции и так далее.

Теоретические материалы для определений по рассматриваемому вопросу позаимствованы из источников литературы [1].

Join()

Метод `Join()` позволяет объединить две разные коллекции по определенному признаку (ключу) и имеет следующий синтаксис:

```
Join(IEnumerable<TInner> inner,  
  
    Func<TOuter,TKey> outerKeySelector,  
  
    Func<TInner,TKey> innerKeySelector,  
  
    Func<TOuter,TInner,TResult> resultSelector);
```

Метод имеет четыре основных параметра

- **inner** - Первым параметром данного метода является вторая последовательность, которая соединяется с текущей.
- **outerKeySelector** - Второй параметр - это соединяющий делегат, который определяет свойство объекта из первого списка.
- **innerKeySelector** - Третий параметр - это соединяющий делегат, который определяет свойство объекта из второго списка.
- **resultSelector** - Третий параметр - это делегат, определяющий финальную последовательность.

На выходе из метода получается новая последовательность.

GroupJoin()

Метод GroupJoin помимо объединения двух коллекций также производит её группировку.

GroupJoin работает очень похоже на Join, за исключением того, что Join передает один элемент внешней последовательности с одним соответствующим элементом внутренней последовательности методу resultSelector. Это значит, что множество элементов внутренней последовательности, соответствующих одному элементу внешней последовательности, приведут в результате к множеству вызовов resultSelector для этого элемента внешней последовательности.

Данная операция имеет следующий синтаксис:

```
GroupJoin(IEnumerable<TInner> inner,  
  
    Func<TOuter, TKey> outerKeySelector,  
  
    Func<TInner, TKey> innerKeySelector,  
  
    Func<TOuter, IEnumerable<TInner>, TResult> resultSelector);
```

И имеет также четыре основных параметра:

- Первый параметр - это вторая последовательность, соединяемая с текущей последовательностью.
- Второй параметр – это делегат, который определяет свойство объекта из текущей коллекции, по которому идет соединение и будет идти группировка.
- Третий параметр - это делегат, определяющий свойство объекта из второй коллекции, по которому идет соединение
- Четвёртый параметр - делегат, определяющий новый объект в результате объединения. Этот делегат получает группу - объект текущей коллекции, по которому шла группировка, и набор объектов из второй коллекции, которые составляют группу.

Данная информация взята из ресурса[4]

На выходе получаем новую последовательность.

Метод Zip()

Данный метод в отличие от предыдущих имеет лишь один параметр - последовательность, которая будет соединяться с текущей. Метод объединяет элементы последовательности в соответствии с их индексом. Метод не проводит проверок на равенство ключей в отличие от предыдущих методов, а просто механически

соединяет две разные коллекции.

Метод имеет следующую структуру:

```
Zip(IEnumerable<TInner>)
```

На выходе мы получаем новую преобразованную последовательность.

Из выше перечисленных фактов можно сделать вывод, что LINQ удобная и полезная библиотека, способная облегчить жизнь программистам при работе с последовательностями.

Литература

1) Операции Join (C#) [электронный ресурс] URL: <https://learn.microsoft.com/ru-ru/dotnet/csharp/programming-guide/concepts/linq/join-operations>
дата обращения (14.02.2023)

2) Операции Join и GroupJoin [электронный ресурс] URL: https://professorweb.ru/my/LINQ/base/level2/2_7.php Д

ата обращения (14.02.2023)

3) Enumerable.Zip Метод [электронный ресурс] URL <https://learn.microsoft.com/ru-ru/dotnet/api/system.linq.enumerable.zip?view=netcore-3.1>

дата обращения (14.02.2003)

4) Операции Join (C#) [электронный ресурс] URL <https://learn.microsoft.com/ru-ru/dotnet/csharp/programming-guide/concepts/linq/join-operations>

дата обращения (14.02.2003)