1. Что такое LINQ?

**LINQ** (Language-Integrated Query) представляет простой и удобный язык запросов к источнику данных. В качестве источника данных может выступать объект, реализующий интерфейс IEnumerable (например, стандартные коллекции, массивы), набор данных DataSet, документ XML. Но вне зависимости от типа источника LINQ позволяет применить ко всем один и тот же подход для выборки данных.

2. В чем разница между отложенными операциями и не отложенными операциями LINQ to Object?

Есть два способа выполнения запроса LINQ: **отложенное** (deferred) и **немедленное** (immediate) выполнение.

При отложенном выполнении LINQ-выражение не выполняется, пока не будет произведена итерация или перебор по выборке, например, в цикле foreach. Обычно подобные операции возвращают объект IEnumerable<T> или IOrderedEnumerable<T>.

* **Select**
* **Skip**
* **SkipWhile**
* **Take**
* **TakeWhile**
* **Union**
* **Where**

С помощью ряда методов мы можем применить немедленное выполнение запроса. Это методы, которые возвращают одно атомарное значение или один элемент или данные типов Array, List и Dictionary.

* **All**
* **Any**
* **Average**
* **Contains**
* **Count**

3. Что такое лямбда-выражения?

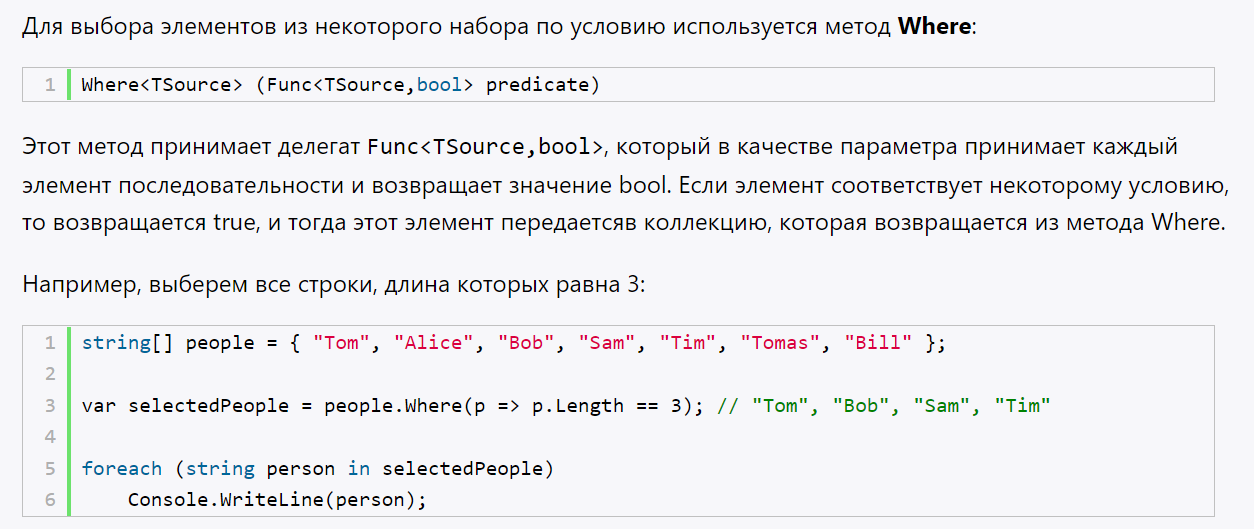
Лямбда-выражения представляют упрощенную запись анонимных методов. Лямбда-выражения позволяют создать емкие лаконичные методы, которые могут возвращать некоторое значение и которые можно передать в качестве параметров в другие методы.

4. Какие есть группы операции в LINQ to Object? Перечислите

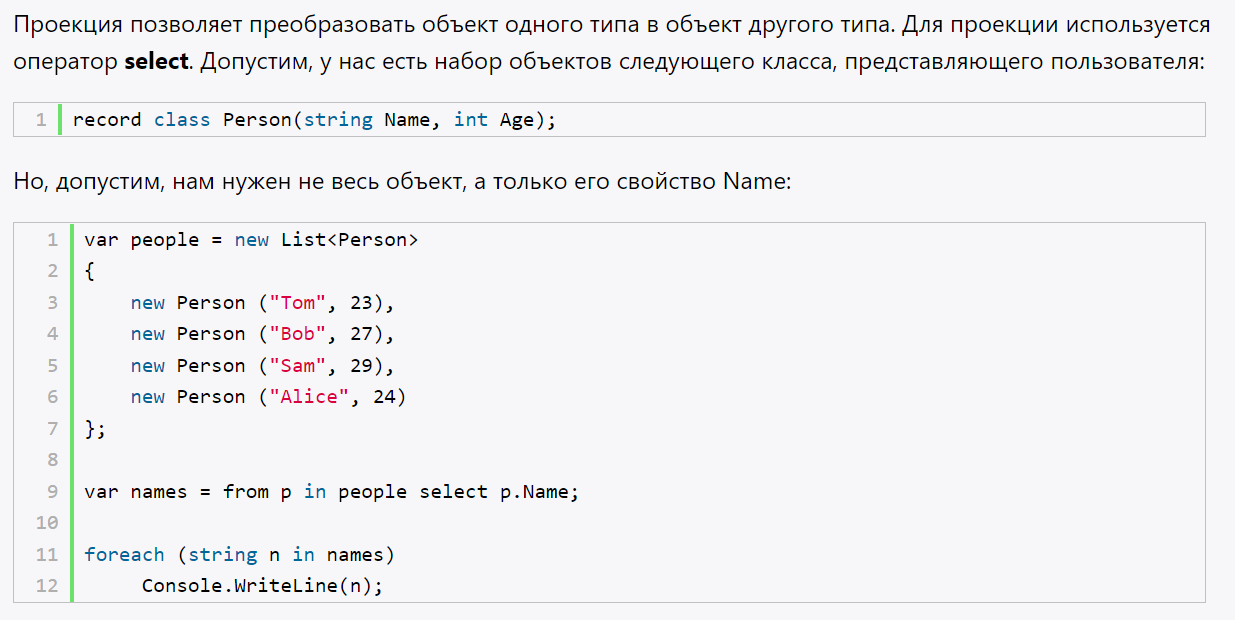
**\*LINQ to Objects**: применяется для работы с массивами и коллекциями

Группы операций: условия, проекций, упорядочивания, группировки, агрегирования, кванторов и разбиения.

5.Как используется операция Where в LINQ to Object?

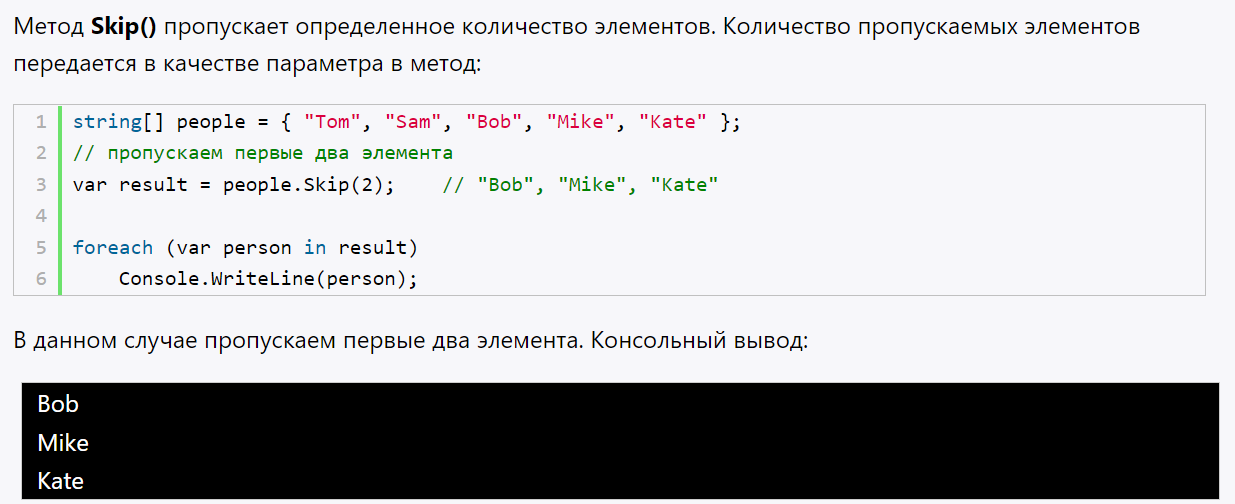


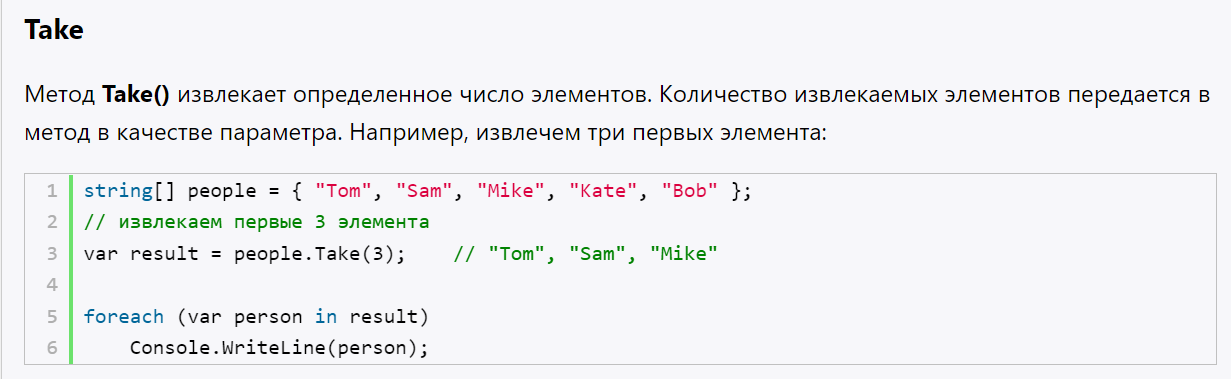
6. Как используется операция Select ? \*Проекция данных



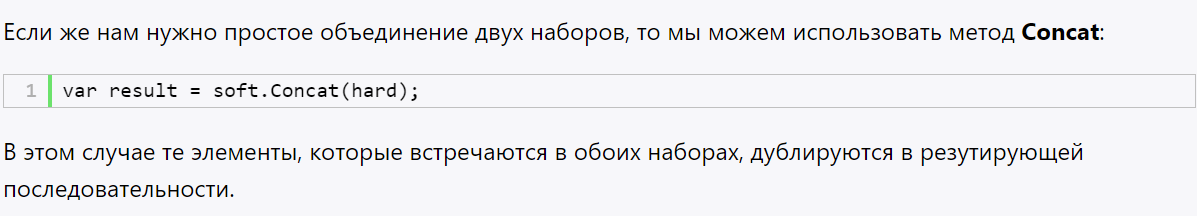
7. Как используются операции Take, Skip? \*разбиение

Ряд методов в LINQ позволяют получить часть коллекции, в частности, такие методы как Skip, Take, SkipWhile, TakeWhile.





8. Как используется операция Concat ? \*работа с множествами



9. Как используется операция OrderBy? \*сортировка (упорядочивание)

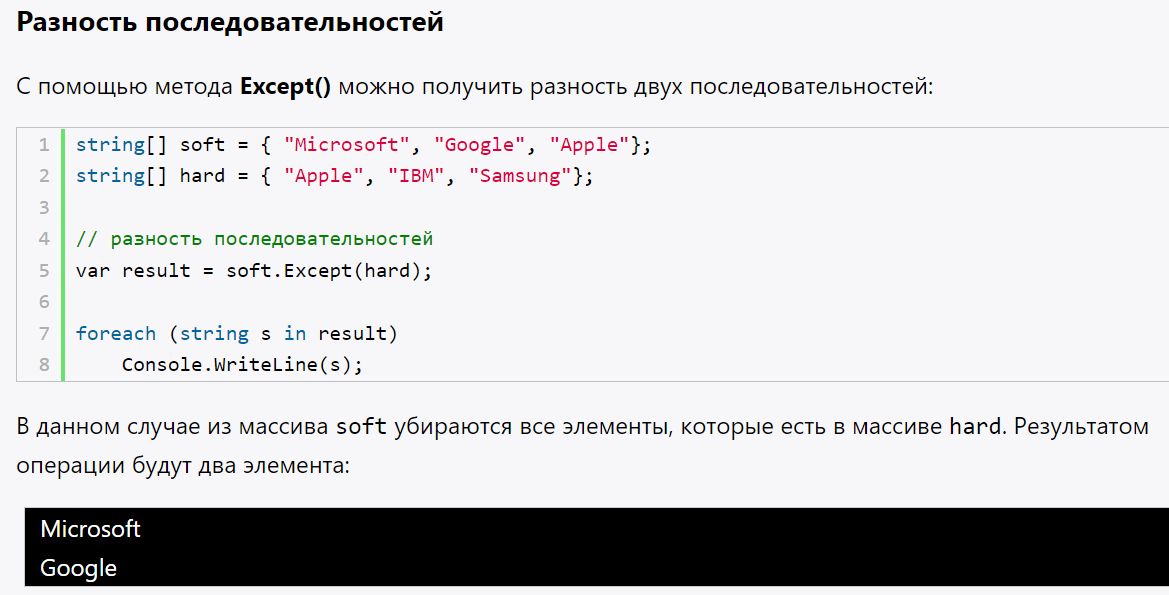
Оператор ORDER BY выполняет сортировку выходных значений. Оператор можно применять как к числовым столбцам, так и к строковым. В последнем случае, сортировка будет происходить по алфавиту.

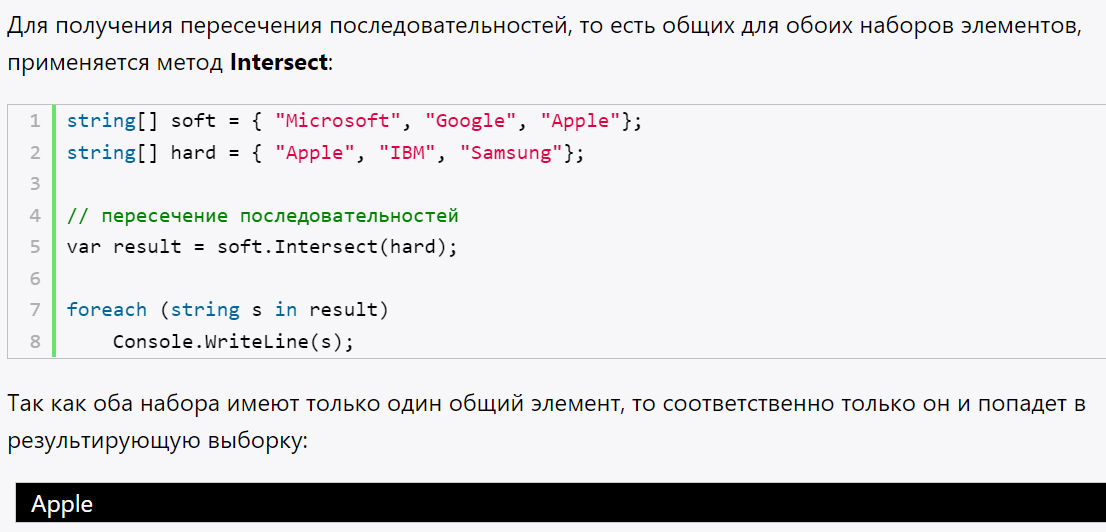
10. Как используется операция Join?

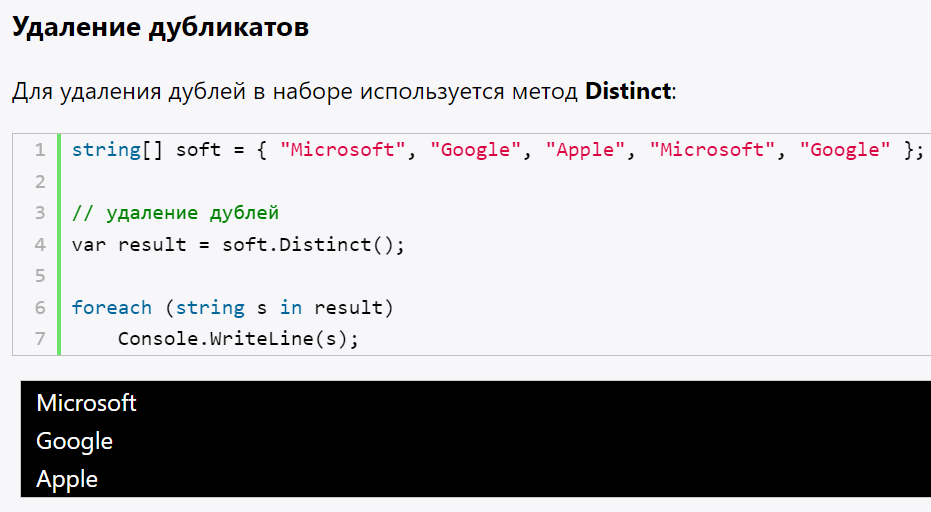
В LINQ операторы Join используются для интеграции двух источников данных в один источник данных, который имеет некоторые общие атрибуты.

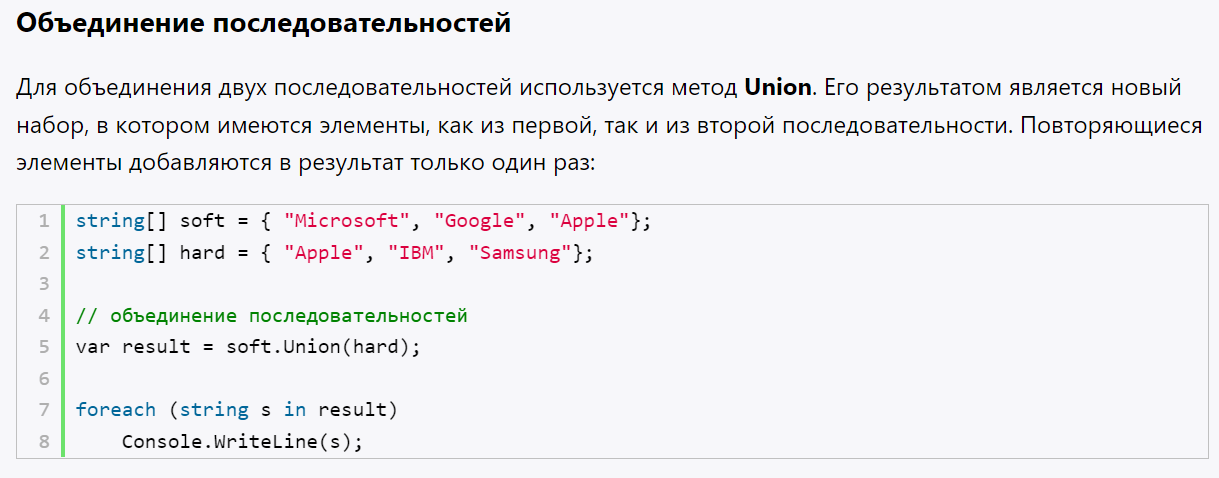


11. Как используются операции Distinct, Union, Except и Intersect? \*работа с множествами

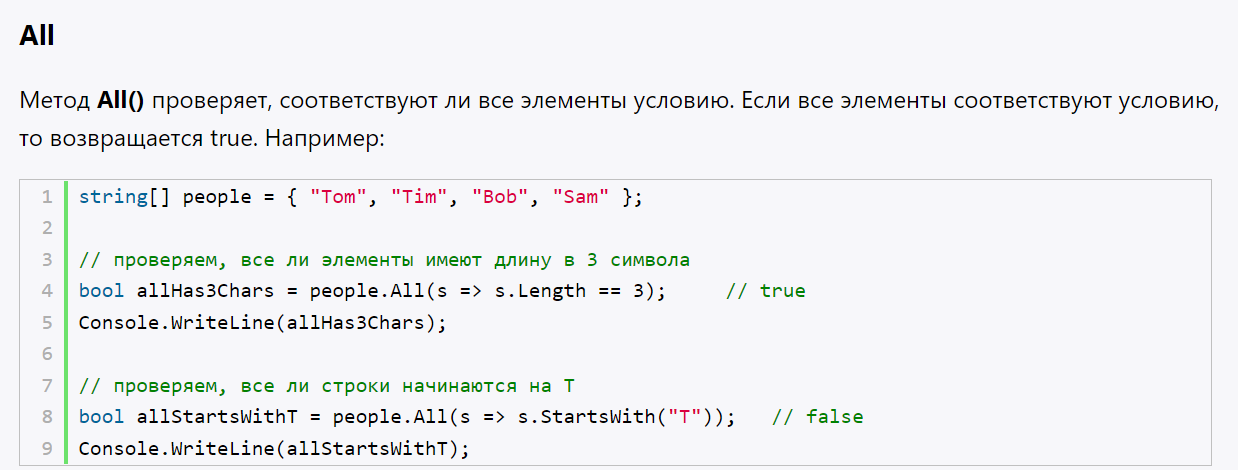








12. Как используются операции First, Last, Any, All и Contains? \*кванторы



Метод **Any()** действует подобным образом, только возвращает true, если хотя бы один элемент коллекции определенному условию.

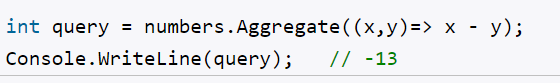
Метод **Contains()** возвращает true, если коллекция содержит определенный элемент.

Метод **First()** возвращает первый элемент последовательности.

Метод **Last()** аналогичен по работе методу First, только возвращает последний элемент. Если коллекция не содержит элемент, который соответствуют условию, или вообще пуста, то метод генерирует исключение.

13. Как используются операции Count, Sum, Min и Max, Average?

\*Агрегатные операции



Count – количество

Sum – сумма

Min, Max – максимальное, минимальное значение

Average – среднее значение

14. Что выведет на экран данный код?

class Test

{

public static void Main()

{

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

list.AddRange(new int[] { 3, 1, 4, 8, 10, 4 });

List<int> some = list.FindAll(i => (i>=9));

foreach (int x in some)

Console.Write(x);

}

}

Ничего не выведет, т.к. в коллекции нету значений, которые удовлетворяют условию.