

Рубежный контроль №2

Как реализовать inner join в LINQ?

Использование inner join является часто используемым средством объединения списков в LINQ.

Пусть существует класс Data с полями id, grp, value, класс связи между списками DataLink, а также списки d1 и d2, код которых выглядит следующим образом:

```
static List<Data> d1 = new List<Data>()
{
    new Data(1, "group1", "11"),
    new Data(2, "group1", "12"),
    new Data(3, "group2", "13"),
    new Data(5, "group2", "15")
};
static List<Data> d2 = new List<Data>()
{
    new Data(1, "group2", "21"),
    new Data(2, "group3", "221"),
    new Data(2, "group3", "222"),
    new Data(4, "group3", "24")
};
```

Тогда inner join можно реализовать с использованием условия where следующим образом:

```
...
Console.WriteLine("Inner Join с использованием Where");
var q = from x in d1
        from y in d2
        where x.id == y.id (*)
        select new {v1 = x.value, v2 = y.value};
foreach (var x in q) Console.WriteLine(x);
...
```

В этом примере накладывается условие в строке (*), что $x.id = y.id$, то есть происходит соединение данных двух списков x и y по полям id. Это поле id при этом является как первичным, так и вторичным ключом. Значения, которые выведет программа:

```
Inner Join с использованием Where
{ v1 = 11, v2 = 21 }
{ v1 = 12, v2 = 221 }
{ v1 = 12, v2 = 222 }
```

Inner join можно реализовать таким образом:

```
...
Console.WriteLine("Cross Join (Inner Join) с использованием Join");
var q = from x in d1
        join y in d2 on x.id equals y.id
        select new {v1 = x.value, v2 = y.value};
foreach (var x in q) Console.WriteLine(x);
...
```

Особенность данного синтаксиса – ключевое слово `equals`, которое позволяет генерировать корректное сравнение данных при SQL запросе.

Inner join также может осуществлять работу с целыми объектами, а не отдельными полями:

```
...
Console.WriteLine("Cross Join (сохранение объекта)");
var q = from x in d1
        join y in d2 on x.id equals y.id
        select new {v1 = x.value, d2Group = y};
foreach (var x in q) Console.WriteLine(x);
...
```