

LINQ について

作成日: 2017/01/16

作成者: 西村

はじめに

LINQ(System.Linq)は、リストなどに対して SQL のような SELECT, WHERE, ORDER BY などの処理を手軽に行える.NET の標準機能 (メソッド) です。

ただ、LINQ は慣れていないと何をしているかわかりづらい部分があるのでこの資料で簡単に説明します。

使う準備

LINQ を使用するには、使用するファイル(.cs)内で using System.Linq; が必要です。

```
using System;  
(その他もろもろの using。略)  
using System.Linq;
```

LINQ の例

LINQ の例を簡単に紹介します。

元のリスト (FROM にあたる)

```
// 元のリスト  
var list = new List<int>() { 1, 2, 3, 4, 5, 1 };
```

SELECT

```
// それぞれの値を 100 倍にしたリストを作成。  
// Select(値 => 新しい値) で変換をかけています。  
// この「値 => 新しい値」というのは関数を短く書いたものです。(後述)  
var bigList = list.Select(x => x * 100).ToList(); // 100, 200, 300, 400, 500, 100
```

WHERE

```
// 偶数だけの値に絞ったリストを作成。  
// Where(値 => 条件) で絞り込んでいます。  
var evenList = list.Where(x => x % 2 == 0).ToList(); // 2, 4
```

ORDER BY

```
// 並び替えたリストを作成。  
// Where(値 => 並び替え対象の値) で並び替えています。  
var ascList = list.OrderBy(x => x).ToList(); // 1, 1, 2, 3, 4, 5  
var descList = list.OrderByDescending(x => x).ToList(); // 5, 4, 3, 2, 1, 1
```

組み合わせ

```
// Where, Select, OrderBy などは組み合わせて使えます。  
var bigEvenList = list.Where(x => x % 2 == 0).Select(x => x * 100).ToList(); // 200, 400
```

「x => x * 100」などの関数について

LINQ の中に出てくる「x => x * 100」などの短いコードは、関数を表しています。(「ラムダ」と呼ばれています)

var newList = list.Select(x => x * 100).ToList() 中の「x => x * 100」は、「list から 1 つ 1 つ x に値を受け取って、x * 100 をして返す」という意味です。

通常のコードだと下記のような感じになります。

```
var newList = new List<int>();  
foreach (var x in list) { // list から 1 つ 1 つ x に値を受け取る  
    newList.Add(x * 100); // 変換をかけて新しいリストに追加  
}
```

Select()や Where()の中に使うラムダは、短い書き方の他に下記のように少し長い書き方もあります。意味は同じですが、込み入った処理や変数定義が必要なときは長い書き方を 사용합니다。

短い

```
list.Select(x => x * 100);
```

長い

```
list.Select(x => {  
    return x * 100; // この中括弧の中に変数定義などができる  
});
```

その他

- 前の例のように「ToList()」などで変換しなくても、LINQ の結果を foreach などを利用することが出来ます。
 - LINQ は ToList() や foreach などを使うまでは実際の処理を実行しない(「遅延評価」と呼ばれます)ようになっていて、内部でループ回数などが抑えられています。
- LINQ には「メソッド形式」と「クエリ形式」という 2 つの書き方があります。
「クエリ形式」は本当に SQL に近い書き方をするものですが、込み入った処理が書きづらいのと補完が効かないので使いづらいです。

- メソッド形式

```
var evenList = list.Where(x => x % 2 == 0);
```

- クエリ形式

```
var evenList = from x in list where x % 2 == 0 select x;
```