

5.LINQ 是拉取数据

在之前我们实现了如下代码:

```
private void Start()
{
    var testNumbers = new List<int> {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

    testNumbers.ListWhere(testNumber => testNumber % 2 == 0)
        .ListSelect(number => number / 2)
//        .ToList() 省略了, 本身就是 List 类型。
        .ListForEach(resultNumber => { Debug.Log(resultNumber); });
}
```

ListWhere 和 ListSelect 还有 ListForEach 加起来, 我们进行了三次的遍历。

而如果换成 LINQ 则不会这样。

```
/* *****
 * http://sikiedu.com liangxie
 * ***** */

using System.Collections.Generic;
using System.Linq;
using UnityEngine;

namespace UniRxLesson
{
    public class LINQExample : MonoBehaviour
    {
        private void Start()
        {
            var testNumbers = new List<int> {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

            testNumbers.Where(testNumber => testNumber % 2 == 0)
```

```

        .Select(number => number / 2)
        .ToList()
        .ForEach(resultNumber =>
        {
            Debug.Log(resultNumber);
        });
    }
}
}

```

其中只有 Where 和 ForEach 进行了遍历，也就是两次遍历。中间的 Select 会和 Where 合并成一个 SelectWhere 操作符。

而且，与老师实现的不同。LINQ 是一个拉取数据的操作。Where 和 Select 有对应的 WhereIterator 和 SelectIterator。这些 Iterator 是可以组装的。到最后的 ForEach 阶段，可能只有一个 Root Iterator，其余的都会作为 Root Iterator 的子 Iterator。当进行遍历的时候，会对 Root Iterator 进行一个 MoveNext 操作，而 RootIterator 的 MoveNext 又有可能会调用子 Iterator 的 MoveNext 操作。就这样不断地拉取数据。而不是像老师的实现方式一样，每一个操作符都进行一个遍历操作。

相比老师的实现 LINQ 的性能会更好，所浪费的计算量会更少。

所以 LINQ 实际上是 Pull 数据。

到此，我们只要记住这个结论就足够了。

今天的内容就这些。