

一.LinQ相关

Action

Action的定义如下：

```
public delegate void Action<in T>(T obj)
```

本质其实就是一个委托，一个没有返回值的委托

Func

Func的定义和action类型，就是func是带返回参数的，func中最后一个参数一定是返回参数。

为了代码的简洁性要多用Action和Func，尽量不用delegate。

Action和Func

二：Expression表达式目录树

Expression表达式目录树定义：是一种语法树，或者说是一种数据结构，可以被我们解析

注意：表达式目录树只能一行，不能有大括号

Expression的再代码里面的本质是如下的密封类

```
public sealed class Expression<TDelegate> : LambdaExpression
```

可以通俗的把表达式目录树理解为一种存储数据的树结构。把变量和操作符等都存储起来，然后组合起来操作。

至于如何通过Expression来操作SQL的话，其实就是框架内置的方法把对应的lambda表达式全部都解析成对应的SQL。

ExpressionVisitor是用递归的方式得到Expression表达式目录树里面所有的树结构，如何按照自己的要求来操作对应的元素

```
/// <summary>
```

```
/// ExpressionVisitor:肯定得递归解析表达式目录树，因为不知道深度的一棵树
```

```
/// 只有一个入口叫Visit
```

```
/// 首先检查是个什么类型的表达式，然后调用对应的protected virtual visit方法
```

```
/// 得到结果继续去 检查类型--调用对应的visit方法--再检测--再调用。。。 
```

```
/// </summary>
```