**Tổng Quan về LINQ**

**Giới thiệu về LINQ**

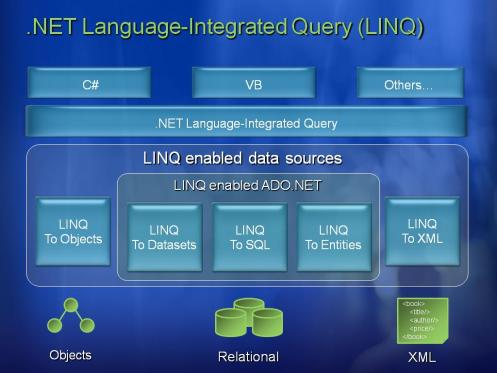
Xử lý thông tin hay dữ liệu là nhiệm vụ quan trọng nhất của bất kỳ phần mềm nào và một trong những trở ngại chính mà các nhà phát triển hiện nay phải đối mặt là khác biệt giữa ngôn ngữ lập trình hướng đối tượng và ngôn ngữ truy vấn dữ liệu, vấn đề càng phức tạp hơn với sự xuất hiện của XML (eXtensible Markup Language – ngôn ngữ đánh dấu mở rộng).

**LINQ là gì ?**

Để giảm gánh nặng thao tác trên nhiều ngôn ngữ khác nhau và cải thiện năng suất lập trình, Microsoft đã phát triển giải pháp tích hợp dữ liệu cho .NET Framework có tên gọi là LINQ (Language Integrated Query), đây là thư viện mở rộng cho các ngôn ngữ lập trình C# và Visual Basic.NET (có thể mở rộng cho các ngôn ngữ khác) cung cấp khả năng truy vấn trực tiếp dữ liệu Object, cơ sở dữ liệu và XML.

Điểm mạnh của LINQ là “viết truy vấn cho rất nhiều các đối tượng dữ liệu”. Từ  cơ sở dữ liệu, XML, Data Object … thậm chí là viết truy vấn cho một biến mảng đã tạo ra trước đó. Vì thế ta có các khái niệm như là LinQ to SQL, LinQ to XML,….

**Các thành phần của LINQ**

Hình 1. Kiến trúc của LINQ trong .NET Framework 3.5

**LINQ to Objects**

“LINQ to Objects” ở đây có nghĩa là nói đến cách sử dụn LINQ đối với các đối tượng Collection mà đã được thực thi giao diện IEnumerable hoặc IEnumerable<T> tức những collection có thể “liệt kê” ra được. Đây là trường hợp sử dụng đơn giản nhất của LINQ khi làm việc với dữ liệu.

**LINQ to SQL**

LINQ to SQL là một phiên bản hiện thực hóa của O/RM (object relational mapping) có bên trong .NET Framework  3.5, nó cho phép bạn mô hình hóa một cơ sở dữ liệu dùng các lớp .NET. Sau đó bạn có thể truy vấn cơ sở dữ liệu dùng LINQ, cũng như cập nhật/thêm/xóa dữ liệu từ đó.

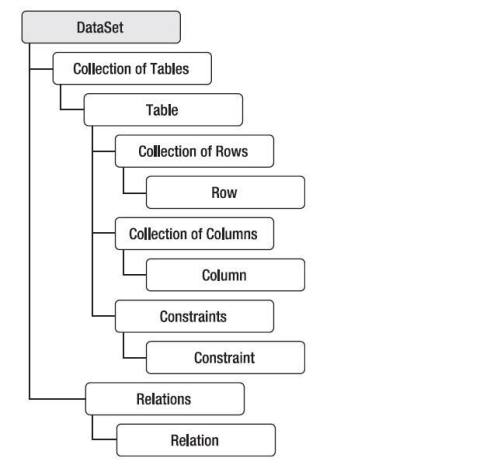
LINQ to SQL hỗ trợ đầy đủ transaction, view và các stored procedure (SP). Nó cũng cung cấp một cách dễ dàng để thêm khả năng kiểm tra tính hợp lệ của dữ liệu và các quy tắc vào trong mô hình dữ liệu của bạn.

LINQ to XML

Sử dụng LINQ với mục đích truy vấn file XML và tiện lợi truy vấn hơn nhiều so với việc dùng XmlDocument, Xpath và Xquery như trước kia.

LINQ to Datasets

Giới thiệu sơ  về DataSet  
DataSet trong ADO.NET là một bước phát triển lớn trong việc phát triển ứng dụng cơ sở dữ liệu đa hệ. Khi lấy và chỉnh sửa dữ liệu, duy trì liên tục kết nối tới Data Source trong khi chờ user yêu cầu thì rõ ràng là tốn tài nguyên máy rất nhiều.  
DataSet giúp ích ở đây rất lớn. Vì DataSet cho phép lưu trữ dữ liệu và chỉnh sửa tại ‘local cache’, hay gọi là offline mode. Có thể xem xét và xử lý thông tin trong khi ngắt kết nối. Sau khi chỉnh sửa và xem xong thì tạo một kết nối và update dữ liệu từ local vào Data Source.  
Dữ liệu trong DataSet được lưu trữ dưới dạng một Collection các Tables và bạn cần phải xử lý thông qua các lớp DataTable (DataRow và DataColumn).

[](https://khangtran.files.wordpress.com/2010/10/dataset.jpg)Kiến trúc DataSet.

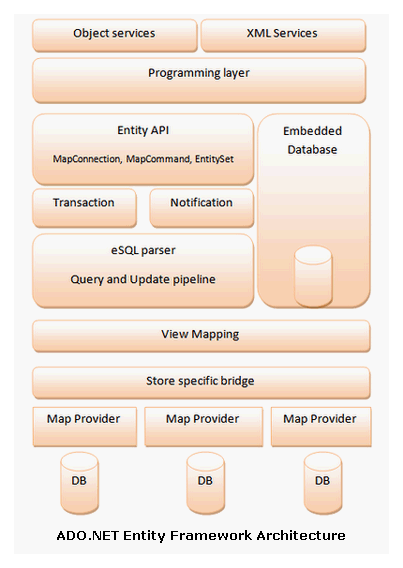
LINQ to DataSet cho phép người lập trình sử dụng DataSets như một nguồn dữ liệu bình thường bằng các cú pháp truy vấn căn bản của LINQ

**LINQ to Entities**

Khái Niệm về Entity Framework

Entity Framework như giản đồ của các dữ liệu được lưu trữ trong cơ sở dữ liệu và trình bày giản đồ khái niệm ứng dụng của nó. Giản đồ này được ánh xạ từ cơ sỡ dữ liệu.

Entity Framework là cơ sở dữ liệu độc lập và xây dựng dựa trên một mô hình chuẩn ADO.NET cho phép truy cập vào cơ sở dữ liệu của bên thứ ba thông qua việc sử dụng ADO.NET.

[](https://khangtran.files.wordpress.com/2010/10/entity.gif)

Kiến trúc của Entity Framework

LINQ to Entities cho phép chúng ta có thể truy vấn các thực thể bên trong Entity Framework.

Các Operator chuẩn trong LINQ

1.Select / SelectMany                                            2.Where

3.Sum / Min / Max / Average / Aggregate              4.Join / GroupJoin

5.Take / TakeWhile                                                6.Skip / SkipWhile

7.OfType                                                                 8.Concat

9.OrderBy / ThenBy                                                10.Reverse

11.GroupBy                                                            12.Distinct

13.Union / Intersect / Except                                14.EqualAll

15.First / FirstOrDefault / Last / LastOrDefault       16.Single

17.ElementAt                                                          18.Any / All / Contains

19.Count

**2. LINQ vs ADO.NET**

Nhều người vẫn hay thắc mắc về mối quan hệ giữa LINQ vs ADO.NET, liệu chúng có đối lập nhau hay hỗ trợ lẫn nhau ?

Nôm na, LINQ là tập mở rộng cho phép viết các câu truy vấn ngay trong các ngôn ngữ lập trình. Nó cho phép bạn làm việc với các kiểu tập hợp dữ liệu, như XML, collection, array,… và cả CSDL.

ADO.NET là công nghệ cho phép các ứng dụng có thể kết nối và làm việc với các loại CSDL khác nhau (truy vấn, cập nhật, thêm, xóa, gọi thủ tục…).

Bản thân LINQ không phải là một công nghệ được tạo ra để thay thế ADO.NET, bạn có thể làm việc với LINQ mà không dính gì đến cơ sở dữ liệu. Tuy nhiên, LINQ to SQL, là một phần mở rộng của LINQ, cho phép bạn có thể làm việc được với CSDL SQL Server, trong trường hợp này thì khi viết bạn có thể bỏ qua các câu lệnhADO.NET mà chỉ quan tâm tới cú pháp mà LINQ cung cấp.

Nhớ rằng dù bạn không hề dùng đến ADO.NET khi viết chương trình sử dụng LINQ to SQL, nhưng đằng sau nó, ADO.NET vẫn được dùng để thực hiện kết nối, gửi các câu lệnh, các lời gọi thủ tục…

**3. DEMO về truy xuất dữ liệu với LINQ**

Trong demo này mình sẽ

Add database vào project để sử dụng với VS

Tạo LINQ to SQL class

Thực hiện các thao tác

-Hiển thị  
-Sửa đổi

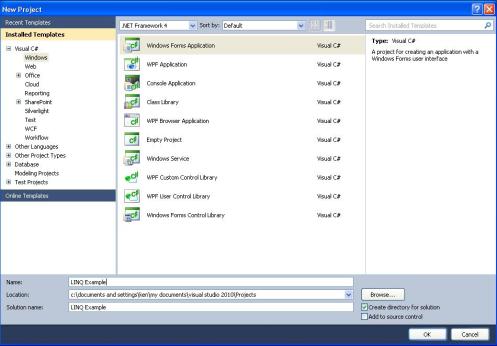
Trong demo mình sẽ sử dụng Visual Studio 2010 phiên bản Ultimate

Link download phần mềm <http://www.microsoft.com/visualstudio/en-us/products/2010-editions/ultimate>

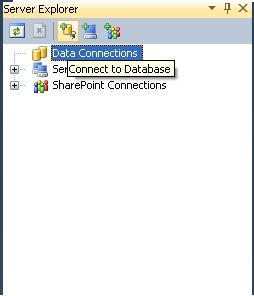
Northwind and pubs Sample Databases for SQL Server 2000

Link download <http://www.microsoft.com/downloads/en/details.aspx?FamilyID=06616212-0356-46a0-8da2-eebc53a68034&displaylang=en>

Trước tiên bạn hãy tao 1 project Win Form bằng cách  chọn File ->New -> Project. Trong tab Installed template bạn chọn Windows-> Windows Form Application. Sửa tên Project là LINQ Example ->OK

[](https://khangtran.files.wordpress.com/2010/10/new-project.jpg)

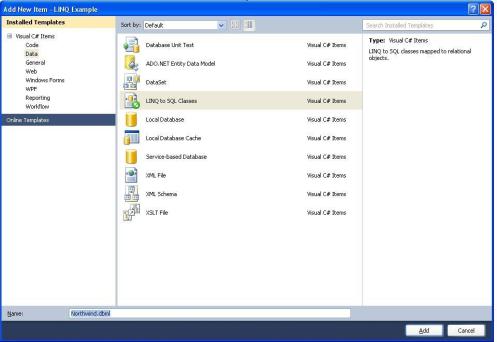
Add Database

Sau đó bạn click View -> Server Explorer để hiển thị cửa sổ Server Explorer. Kế đó bạn chọn vào button Connect to Database ( theo hình )[](https://khangtran.files.wordpress.com/2010/10/add-data.jpg)

Trong phần datasource bạn chon Change. Sau đó bạn chọn Microsoft SQL Server Database file -> OK. Sau đó bạn browse đến chỗ cài đặt SQL Server 2000 Sample Databases mà mình đã download ở trên  ( mặc định là ổ C ). Chọn fileNORTHWND.MDF -> OK.

Hiện giờ thì trong cửa sổ Server Explorer đã có NORTHWND.MDF

Tạo file LINQ to SQL classes

Chuột phải vào project LINQ Example, chọn Add -> New Item -> Chọn LINQ to SQL Classes. Sửa tên lại thành Norwind.dbml[](https://khangtran.files.wordpress.com/2010/10/linqtosql.jpg)

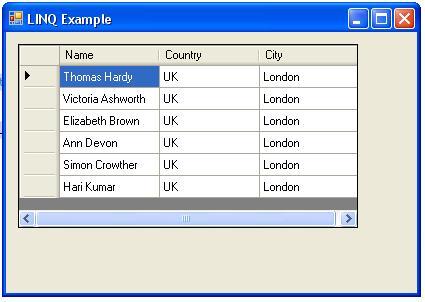
Kế đến bạn bấm vào file Norwind.dbml, kéo bất kỳ 1 bảng nào trong Norwind.mdf (ở đây mình chọn bảng Customer) Thì lập tức bảng Customer sẽ được ánh xạ (mapping) vào datacontext của mình. Bây giờ bảng Customer sẽ được coi như một lớp trong project của mình

Bây giờ mình trở về winform, add Datagirdview control vào form của mình. Datagirdview sẽ được dùng để hiển thị dữ liệu. Sau đó bạn click đúp vào form để vào event page\_load.

Trong sự kiện page\_load, bạn code :

NorthwindDataContext db = new NorthwindDataContext(); // khoi tao doi tuong NorthwindDataContext  
var query = from c in db.Customers  
where c.City == “London”  
select new { Name = c.ContactName, Country = c.Country, City = c.City };  
dataGridView1.DataSource = query;

Đoạn code này sẽ hiển thị tên, quớc gia, thành phố của các khách hàng ở London.

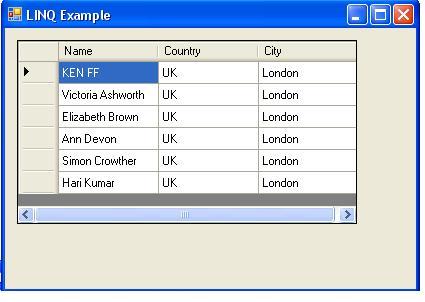
Sau đó bạn ctrl + F5 để chạy chương trình[](https://khangtran.files.wordpress.com/2010/10/kq.jpg)

Tiếp theo mình sẽ sửa đổi lại dữ liệu. Mình sẽ chọn  ra khác hàng có tên là Thomas Hardy và sau đó sửa đổi giá trị ContactName thành KEN FF

Bạn thêm dòng code này vào sau NorthwindDataContext db = new NorthwindDataContext()

NorthwindDataContext db = new NorthwindDataContext();

Customer cus = db.Customers.Single(p => p.ContactName == “Thomas Hardy”);  
cus.ContactName = “KEN FF”;  
db.SubmitChanges();

Sau đó bạn ctrl + F5 để chạy chương trình[](https://khangtran.files.wordpress.com/2010/10/kq2.jpg)

Giá trị của ContactName đựơc sửa lại .

Đây là những thao tác cơ bản trong việc truy vấn dữ liệu dùng LINQ. Bài viết sau mình sẽ giới thiệu về lớp SqlDataAdapter, SqlDataReader … để truy xuất dữ liệu khi làm việc với SQL Server.