

Language **IN**tegrated **Q**uery

تحویلی بزرگ در، ارتباط با داده ها

MohaMad Fathi
www.MiMFa.net



Language Integrated Query

MohaMad Fathi



سرآمد

کد و داده دو عنصر اساسی در نرم افزار می باشند که هر یک دارای جایگاه تعریف شده مختص به خود است. یکی از فعالیت های مهم در زمان پیاده سازی نرم افزار، نوشتن کد مورد نیاز جهت دستیابی به داده است. برای نوشتن کد، پیاده کنندگان می توانند از زبان های برنامه نویسی مختلفی استفاده کنند. انتخاب زبان برنامه نویسی به عوامل متعددی نظیر نوع رفتار برنامه، پیش زمینه و دانش پیاده کنندگان، مهارت های تیم پیاده کننده، نوع سیستم عامل و سیاست های سازمان بستگی دارد.

صرف نظر از این که پیاده کنندگان چه زبانی را برای نوشتن کد انتخاب می نمایند، در زمان پیاده سازی نرم افزار به دفعات مجبور خواهیم بود که با داده سرو کار داشته باشیم. داده می تواند بر روی دیسک، جداولی در بانک اطلاعاتی و یا اسناد XML ذخیره شده باشد.

LINQ

ظهور فناوری LINQ (Language INtegrated Query) تلاشی که در راستای تحول در نحوه دستیابی به منابع داده (صرف نظر از نوع منبع داده)، مطرح گردیده است . پس از مطرح شدن هر گونه فناوری در عرصه نرم افزار، طراحان و پیاده کنندگان با سه پرسش اساسی مواجه می شوند:

- این فناوری چیست ؟
- چرا ما به آن نیاز داریم ؟
- و چه کار می کند ؟

LINQ چیست؟

در کنفرانس توسعه دهندگان حرفه ای مایکروسافت در سال ۲۰۰۵ آقای هلسبرگ یک تکنولوژی جدید که بتوان به وسیله آن با هر نوع منبع داده ای به یک روش یکسان اتصال برقرار کرد، با نام LINQ معرفی نمود. (LINK نیز تلفظ می شود)

LINQ یک راه حل یکسان برای اتصال برقرار کردن و بازیابی اطلاعات از هر شی که رابط Enumerable را پیاده سازی کرده باشد فراهم می کند. به وسیله LINQ می توان با آرایه ها و مجموعه های درون حافظه، پایگاه داده های رابطه ای و حتی اسناد xml را به عنوان منبع داده در نظر گرفت و با آن کار کرد!

بوسیله LINQ می توان اطلاعات را از هر منبع داده ای با گرامری مشابه و خوش شکل بازیابی کرد. گرامری که بسیار شبیه به نوشتار نحوی SQL است، توجه داشته باشید که هدف تیم سازنده LINQ، اضافه کردن یک راه جدید برای بازیابی داده ها نیست، بلکه فراهم کردن یک مجموعه دستورات محلی و جامع برای بازیابی اطلاعات که از هر نوع منبع داده ای پشتیبانی می کند.



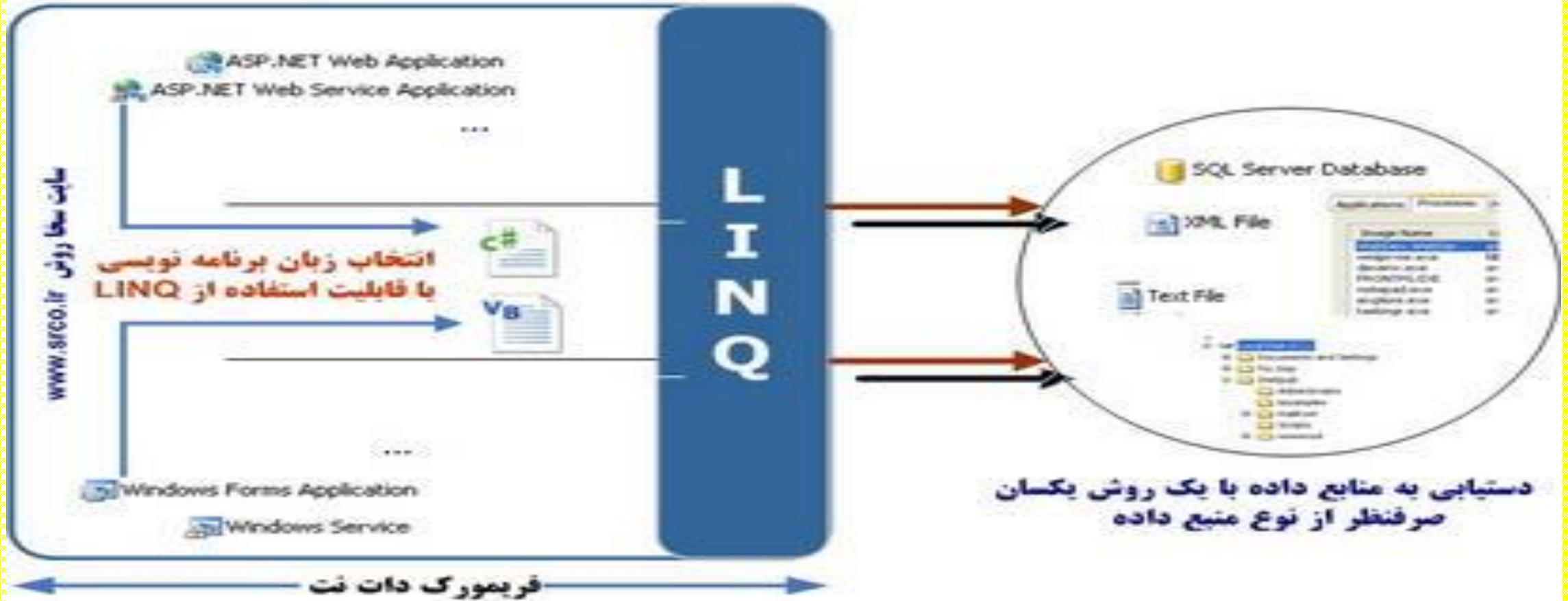
LINQ چیست؟

LINQ یک سری مجموعه دستورات توانمند را ارائه می کند که بوسیله آن ها می توان پرس و جوهای پیاده سازی نمود که از مواردی چون joinها، توابع Aggregation، مرتب سازی، فیلتر و ... پشتیبانی کند. این دستورات را language-level می نامند و دیگر نیازی به کامپایل برای دیدن نتیجه نیست!

LINQ یک ابزار نیست بلکه یک تکنولوژی است که بتوان Provider هایی را پیاده سازی کرد تا به وسیله آن ها با منابع داده ارتباط برقرار کرد؛ به طور مثال Providerهایی مانند LINQ TO SQL و یا LINQ TO XML که توسط تیم توسعه دات نت پیاده سازی شده اند که به آن مدل Provider اطلاق می شود.

ارتباط با LINQ

دستیابی به منابع داده مختلف با استفاده از فناوری LINQ



ساختار LINQ

.NET Language-Integrated Query

C#

VB

Other

.NET Language-Integrated Query (LINQ)

LINQ enabled data sources

ADO.NET LINQ Technologies

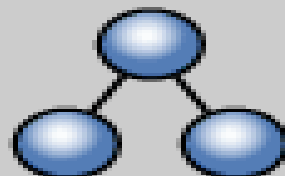
LINQ
to Objects

LINQ
to DataSet

LINQ
to SQL

LINQ
to Entities

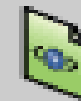
LINQ
to XML



Objects



Relational



XML

ساختار LINQ

بالاترین سطح در تصویر نشان دهنده زبان هایی است که پشتیبانی کاملی از LINQ انجام می دهند.
سطح میانی این نمودار شامل ۵ بخش اصلی پروژه LINQ می باشد:

- LINQ to Objects
- LINQ to DataSet
- LINQ to SQL
- LINQ to Entities
- LINQ to XML

LINQ to Objects

یک API است که متدهایی که نشان دهنده عملگرهای استاندارد پرس و جو می باشند را فراهم می کند. این متد ها برای بازیابی اطلاعات از تمامی اشیائی که رابط Enumerable را پیاده سازی کرده باشند، استفاده می شود.
مانند آرایه ها، لیست ها و به طور کلی مجموعه های عام و غیر عام درون حافظه.

```
int[] values = { 2, 9, 5, 0, 3, 7, 1, 4, 8 };

var filtered =
    from value in values
    where value > 4
    select value;

foreach(int f in filtered)
    Console.WriteLine(f);
```

LINQ to DataSet

این مدل از عملیات پرس و جو بر روی DataTable ها و DataSet های موجود در ADO.NET پشتیبانی می کند.

```
DataSet ds = new DataSet();
ds.Locale = CultureInfo.InvariantCulture;
FillDataSet(ds);

DataTable products = ds.Tables["Product"];

IEnumerable<string> query =
    from product in products.AsEnumerable()
    select product.Field<string>("Name");

Console.WriteLine("Product Names:");
foreach (string productName in query)
{
    Console.WriteLine(productName);
}
}
```

LINQ to SQL

نامی است که برای نوعی API معین شده که به وسیله آن می توان از بانک های رابطه ای مانند SQL Server استفاده کرد. به طور خلاصه باعث تسهیل در استفاده از بانک اطلاعاتی برای بهره بردن از عملیاتی مانند پرس و جو، درج، حذف و ... می شود.

```
// Attach the log to show generated SQL.  
db.Log = Console.Out;  
  
// Query for customers in London.  
IQueryable<Customer> custQuery =  
    from cust in Customers  
    where cust.City == "London"  
    select cust;
```

LINQ to Entities

یک راه حل ارائه شده توسط Microsoft ORM می باشد که توسعه یافته ی LINQ to SQL است.
LINQ to Entities بین پایگاه داده ی فیزیکی و طراحی منطقی و تجاری قرار می گیرد و اجازه استفاده از آن را به صورت موجودیت می دهد.

```
using (AdventureWorksEntities context = new AdventureWorksEntities())
{
    IQueryable<string> productNames =
        from p in context.Products
        select p.Name;

    Console.WriteLine("Product Names:");
    foreach (String productName in productNames)
    {
        Console.WriteLine(productName);
    }
}
```

LINQ to XML

علاوه بر تعمیم عملگرهای استاندارد پرس و جو شامل یک سری خصوصیات ویژه XML برای ایجاد اسناد XML و همچنین پرس و جو بر روی آن ها می باشد.
البته تیم توسعه لinq خصوصیت جدیدی را برای استفاده از اسناد XML طراحی نکرده بلکه استاندارد XML DOM را پشتیبانی کرده است یعنی دیگر نیاز به یادگیری Xpath نیست.

```
XElement root = XElement.Load("PurchaseOrder.xml");
IEnumerable<XElement> address =
    from el in root.Elements("Address")
    where (string)el.Attribute("Type") == "Billing"
    select el;
foreach (XElement el in address)
    Console.WriteLine(el);
```

LINQ در نگاه کلی

مدل رابطه ای دارای مزایایی است که با نگرش دقیق تر، به آن پی می برید:

۱. برنامه نویس می تواند با تصور خود کوئری طراحی کند و آن ها را به صورت بصری ویرایش کند.
۲. با فراهم شدن گزینه اول شرایطی به وجود می آید که برنامه نویس می تواند کوئری خود را به حداکثر کارایی برساند چون کوئری را مشاهده می کند.
۳. برنامه نویس می تواند Provider پی برای منبع داده خود طراحی کند تا دیگران با آن به منبع داده او دسترسی داشته باشند.

LINQ در نگاه کلی

LINQ یک فناوری قدرتمند در زمان کار با داده است که دستاوردهای متعددی را برای پیاده کنندگان به ارمغان آورده است :

- روشی ساده جهت نوشتن query
- تسریع در پیاده سازی نرم افزار با توجه به حذف خطاهای زمان اجراء
- امکان استفاده از امکاناتی نظیر اشکال زدائی و IntelliSense در زمان پیاده سازی
- حذف خلاء موجود بین داده رابطه ای و پیاده سازی شی گراء
- استفاده از یک گرامر یکسان جهت نوشتن query صرف نظر از نوع منبع داده
- افزایش بازدهی و راندمان برنامه نویسی ، چراکه برنامه نویسان از یک رویکرد یکسان برای نوشتن query و بهنگام سازی داده از طریق زبان برنامه نویسی استفاده خواهند کرد .

The End

Collected By: MiMFa



Language Integrated Query

MohaMad Fathi

