

**مقدمه**

کد و داده دو عنصر اساسی در نرم افزار بوده که هر یک دارای جایگاه تعریف شده مختص به خود میباشند. 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. صرفنظر 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. در برخی موارد لازم است که با تمامی نوع های اشاره شده در یک برنامه سرو کار داشته باشیم. فراموش نکنیم که مدیریت داده برای هر پروژه نرم افزاری یک "باید" است.

پیاده کنندگان نرم افزار در زمان نوشتن کد و تولید امکانات مورد نیاز در هر یک از لایه های متفاوت (با توجه به نوع معماری نرم افزار) ، همواره این پرسش را با خود مطرح می نمایند که محیط پیاده سازی و یا فریمورک هانی نظیر دات نت چه امکاناتی را در اختیار آنها خصوصا در زمان کار با داده قرار می دهند. به عبارت دیگر ، یک سیستم عامل چگونه می تواند نیاز پیاده کنندگان نرم افزار را پوشش دهد و چه امکانات و پتانسیل هانی را در اختیار علاقه مندان به طراحی و پیاده سازی نرم افزار قرار می دهد. دات نت به عنوان یکی از فریمورک های پیاده سازی نرم افزار در سالیان اخیر ، امکانات متعددی را جهت کار با داده در اختیار پیاده کنندگان نرم افزار قرار می دهد. LINQ ، تلاشی است جهت یکپارچگی عمیق تر و بیش تر بین زبان های برنامه نویسی و داده. در واقع LINQ مشتمل بر مجموعه ای از عملگرهای query استاندارد است که معماری لازم جهت حرکت ، فیلترینگ و اجرای عملیات بر روی "تقریبا" هر نوع منبع داده نظیر XML ، بانک های اطلاعاتی رابطه ای و اشیاء درون حافظه را فراهم می نماید.

**LINQ (پرس و جوی یکپارچه شده با زبان)**

با این که فناوری LINQ در پانیز سال 2005 مطرح گردید ولی فرآیند پیاده سازی آن از سال 2003 آغاز شده بود. هدف از ابداع فناوری فوق ، ارائه تسهیلات لازم برای پیاده کنندگان جهت کار با داده در بانک های اطلاعاتی SQL و XML عنوان شده است. در واقع ، وجود یک حلقه گمشده بین داده رابطه ای (بانک های اطلاعاتی) ، اسناد XML با زبان های برنامه نویسی ، ضرورت ابداع فناوری فوق را توجیه کرده بود. پیاده کنندگان مجبور بودند برای کار با هر یک از منابع داده ، از روش های مختلفی استفاده نمایند. LINQ را می توان به منزله حلقه ای گمشده بین دنیای داده و زبان های برنامه نویسی

همه منظوره تصور کرد. با استفاده از LINQ امکان دستیابی به داده صرفنظر از نوع داده، با روشی مشابه و یکسان فراهم می گردد. ما برای نوشتن کدها از یک زبان برنامه نویسی نظیر VB.NET و یا C# و برای گفتگو با سیستم بانک اطلاعاتی از یک زبان دیگر نظیر SQL استفاده می کنیم.

## Language Integrated Query = LINQ

- در مواردی که فریمورک دات نت را به عنوان پلت فرم انتخاب کرده باشیم، دات نت قادر به درک کدهای SQL نخواهد بود.
- در مواردی که از زبان های برنامه نویسی حمایت شده در دات نت جهت نوشتن کد در محیط ویژوال استودیو استفاده می کنیم، همواره یک ابزار قدرتمند در کنار ما است تا در صورت اشتباه در گرامر دستورات توصیه های لازم را ارائه نماید.
- امکان بررسی نوع ها در زمان ترجمه وجود ندارد. این بدان معنی است که پیاده کنندگان تا زمانی که برنامه اجرا نگردد، نمی توانند مشکلات احتمالی را مشاهده و قبل از زمان اجرا با آنها برخورد نمایند.
- شرکت مایکروسافت امکانات متعددی را در فریمورک دات نت جهت کار با اسناد XML ارائه کرده است. ارائه System.Xml، System.Xml.XPath و System.Xml.Schema نمونه هایی در این زمینه می باشند. پیاده کنندگان نرم افزار در زمان کار با اسناد XML مجبور بودند که از فناوری های متعددی نظیر DOM (برگرفته شده از Mode Document Object) و XQuery استفاده نمایند. همین موضوع باعث شده بود که کار با اسناد XML نظیر خواندن و یا نوشتن داده در آنها برای بسیاری از پیاده کنندگان مشکل باشد.

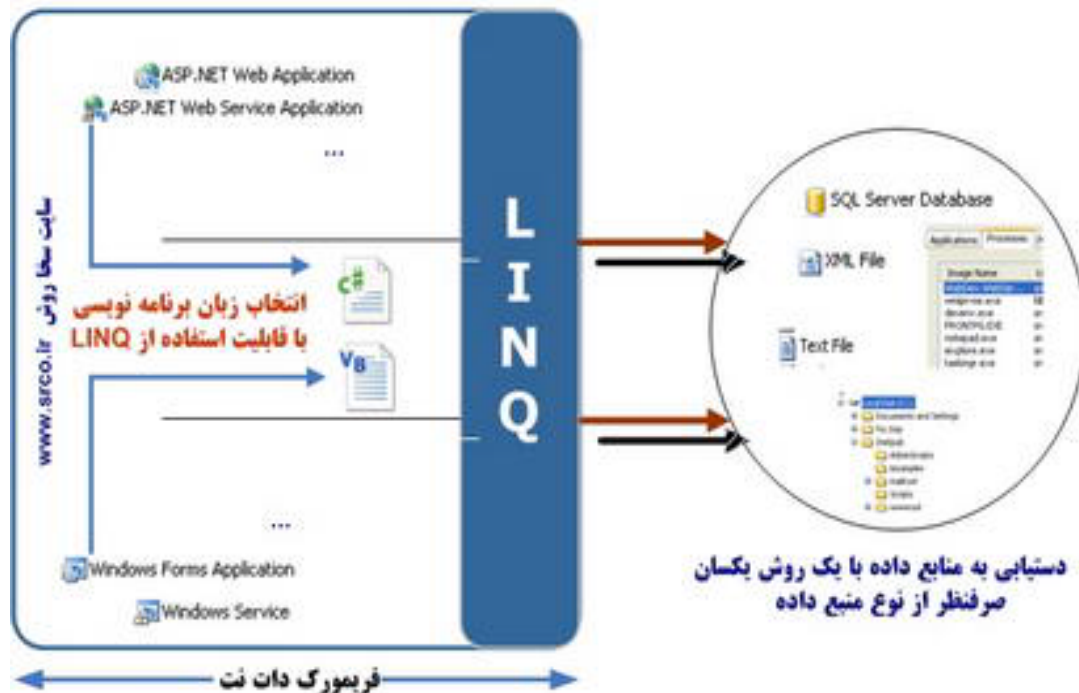
شرکت مایکروسافت برای حل مشکلات فوق، بررسی دو راهکار را در دستور کار خود قرار داد:

- **ایجاد بناسیل های مورد نیاز مختص XML و با داده رابطه ای در هر یک از زبان های برنامه نویسی و زمان اجراء.** (راهکار فوق نه تنها مشکل اصلی را حل نمی کرد بلکه در مواردی مشکلات را خصوصا در زمینه نگهداری افزایش می داد)
- **اضافه کردن قابلیت های همه منظوره نوشتن query در فریمورک دات نت.** (به عبارت دیگر یک فریمورک همه منظوره با قابلیت نوشتن query درون فریمورک دات نت که زبان های VB.NET و C# بتوانند به سادگی از مزایای آن استفاده نمایند)

خوشبختانه، شرکت مایکروسافت گزینه دوم را انتخاب نمود. ایجاد یک زیرساخت جهت نوشتن query در بین اشیاء، اسناد XML، داده رابطه ای و ... هم اینک پیاده کنندگان می توانند از مزایای یک الگوی تعریفی در هر یک از زبان های دات نت جهت کار با داده استفاده نمایند. ارائه مجموعه اپراتورهای استاندارد جهت نوشتن query یکی از دستاوردهای مهم LINQ محسوب می گردد. بدین ترتیب، پیاده کنندگان می توانند با بکارگیری یک مجموعه یکسان از اپراتورهای نوشتن query در هر یک از زبان های برنامه نویسی حمایت شده در دات نت، از داده ذخیره شده در منابع داده مختلف صرفنظر از نوع منبع داده استفاده نمایند. همچنین، امکان استفاده از پتانسیل های نظیر IntelliSense و بررسی نوع ها در زمان ترجمه نیز وجود خواهد داشت.

شکل زیر نحوه عملکرد LINQ را نشان می دهد:

## دستیابی به منابع داده مختلف با استفاده از فناوری LINQ



### خلاصه :

1-LINQ یک ارتباط بین داده های ارتباطی و اسناد XML با زبان های برنامه نویسی است با استفاده از LINQ امکان دستیابی به داده صرف نظر از نوع داده وجود دارد .

2-LINQ استاندارد است که یک سری اپراتورهای استاندارد جهت نوشتن Query را به ما می دهند.

3-وقتی در visual studio کار می کنیم یک Intelesince برای ما هست که در صورت اشتباه قابلیت اصلاح را به ما می دهد ولی برای کار در Database این امکان برای ما وجود ندارد که LINQ این خلا را از بین می برد .

4-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 زدایی 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 صرفنظر 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 .

LINQ-5 از دو بخش که مکمل یکدیگر می باشند تشکیل شده است :

مجموعه ای از ابزار ها جهت کار با اشیاء ، اسناد XML ، بانک های اطلاعاتی رابطه ای و سایر نوع های داده و مجموعه ای از ضمائم برای زبان های برنامه نویسی نظیر VB و C# .

LINQ-6 دسترسی به داده ها را خیلی راحت کرده و این داده ها می توانند داده های ذخیره شده در حافظه مثل مجموعه و آرایه های تعریف شده در برنامه باشند، یا داده های درون یک دیتابیس و داده های XML باشند.

LINQ-7 بطور کلی یک مدل Attribute-Base است، با قرار دادن صفات می توان با استفاده از کلاس DataContext به داده های خود دسترسی پیدا کرد.

معماری LINQ شامل موارد زیر است که عبارتند از:

select/select more  
sum/min/max/average/aggregate  
where  
tak/take while  
skip/skip while  
oftype  
concat  
by orderby/then