

LINQ - язык интегрированных запросов

Ниже представлены разделы, которые входят в данную тему:

🔗 [LINQ to Objects \(linq_index.php\)](#)

LINQ to Objects - название, данное API-интерфейсу `IEnumerable<T>` для *стандартных операций запросов (Standard Query Operators)*. Именно LINQ to Objects позволяет выполнять запросы к массивам и находящимся в памяти коллекциям данных. Стандартные операции запросов - это статические методы класса `System.Linq.Enumerable`, которые используются для создания запросов LINQ to Objects.

🔗 [LINQ to XML \(../linq_xml/level5/linq_to_xml_index.php\)](#)

LINQ to XML — название, назначенное API-интерфейсу LINQ, который ориентирован на работу с XML. В Microsoft не только добавили необходимые библиотеки XML для работы с LINQ, но также восполнили недостатки стандартной модели XML DOM, существенно облегчив работу с XML. Прошли времена, когда нужно было создавать `XmlDocument` только для того, чтобы поработать с небольшим фрагментом XML-кода. Чтобы воспользоваться преимуществами LINQ to XML, в проект понадобится добавить ссылку на сборку `System.Xml.Linq.dll` и директиву `using System.Xml.Linq`.

🔗 [LINQ to DataSet и SQL \(../linq_sql/level8/linq_to_dataset_and_sql_index.php\)](#)

LINQ to DataSet — название, данное API-интерфейсу LINQ, который предназначен для работы с DataSet. У многих разработчиков есть масса кода, полагающегося на DataSet. Те, кто не хотят отставать от новых веяний, но и не готовы переписывать свой код, благодаря этому интерфейсу могут воспользоваться всей мощью LINQ.

LINQ to SQL — наименование, присвоенное API-интерфейсу `IQueryable<T>`, который позволяет запросам LINQ работать с базой данных Microsoft SQL Server. Чтобы воспользоваться преимуществами LINQ to SQL в проект понадобится добавить ссылку на сборку `System.Data.Linq.dll`, а также директиву `using System.Data.Linq`.

🔗 [LINQ to Entities \(../linq_entities/level14/linq_to_entities_index.php\)](#)

LINQ to Entities — альтернативный API-интерфейс LINQ, используемый для обращения к базе данных. Он отделяет сущностную объектную модель от физической базы данных, вводя логическое отображение между ними двумя. С таким отделением возрастает мощь и гибкость, но также растет и сложность. Если нужна более высокая производительность, чем обеспечивается LINQ to SQL, имеет смысл рассмотреть эту альтернативу.

В частности, сущностная объектная модель и базой данных, если сущностные объекты конструируются из нескольких таблиц или требуется большая гибкость в моделировании сущностных объектов, то в

этом случае LINQ to Entities может стать оптимальным выбором.

🔗 [Parallel LINQ \(../PLINQ/level16/plinq_index.php\)](http://Parallel LINQ (../PLINQ/level16/plinq_index.php))

Формально отдельного продукта LINQ, который нужно было бы получать отдельно, не существует. LINQ полностью интегрирован в .NET Framework, начиная с версии 3.5 и Visual Studio 2008. В NET 4.0 и Visual Studio 2010 добавлена поддержка средств Parallel LINQ, которые рассматриваются в данном разделе.



Лучший чат для C# программистов (<https://t.me/professorweb>)

Professor Web (/)

Наш любимый хостинг (/)

Пройди тесты

C# тест (легкий) (<https://professorweb.ru/test/c-sharp-test.html>)

.NET тест (средний) (<https://professorweb.ru/test/asp-test.html>)