Урок 6

[Взаимодействие с базой данных.Обзор технологии ADO.Net. Обзор технологии LINQ to SQL. 1](#_Toc487203984)

[1. Взаимодействие с базой данных 1](#_Toc487203985)

[2. Обзор технологии ADO.NET 1](#_Toc487203986)

[2.1. Основные компоненты ADO.NET 1](#_Toc487203987)

[3. Обзор технологии LINQ to SQL 3](#_Toc487203988)

[4. ДЗ 3](#_Toc487203989)

# Взаимодействие с базой данных.Обзор технологии ADO.Net. Обзор технологии LINQ to SQL.

1. Взаимодействие с базой данных
2. Обзор технологии ADO.NET

ADO.NET – это семейство технологий, которые позволяют разработчикам .NET приложений взаимодействовать с данными используя стандартные и структурированные подходы.

Библиотека ADO.Net подключается к проекту с помощью добавления ссылки на пространство имен System.Data.

ADO.NET осуществляет управление как внутренними данными, т.е. созданными в памяти компьютера и использующимися внутри приложения, так и внешними данными, т.е находящимися вне приложения, например в базе данных или текстовых файлах.

Вне зависимости от источника данных, ADO.NET представляет данные в коде приложения в табличном виде, в виде строк и столбцов.

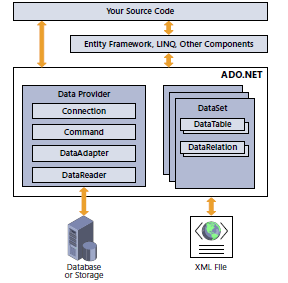
До выпуска компанией Microsoft .NET фреймворка, одной из основных технологий доступа к данным, используемой в приложениях, было ADO, ActiveX Data Object.

После выхода .NET фреймворка ADO.NET стала преемницей ADO.

При взаимодействии с внешними источниками данных ADO.NET может использовать технологию работы с отсоединенными данными. При использовании более ранних технологий, разработчики обычно создавали постоянное подключение к БД и использовали различные способы блокировки записей для выполнения безопасных и корректных изменений данных. Но с приходом эры Интернет, стратегия поддержания открытых соединений для каждого из множества одновременных HTTP запросов к веб приложению, показала свою нежизнеспособность. Для ADO.NET предпочтительной стратегией является открытие подключение до запроса к БД и его моментальное закрытие после выполнения запроса.

* 1. Основные компоненты ADO.NET

Пространство имен System.Data включает в себя множество отдельных классов ADO.NET, которые работают вместе и обеспечивают доступ к табличным данным. Библиотека ADO.NET включает в себя две группы классов: для работы с данными внутри приложения и для работы с внешними данными.



Центральным объектом библиотеки является DataTable. Схожий по назначению с таблицами в БД, DataTable управляет актуальными данными, с которыми вы работаете.

Каждый объект DataTable содержит 0 или более строк данных.

Таблицы содержат описание элементов DatаColumn, каждый из которых описывает значения, хранящиеся в строках таблиц. DataColumn содержит описание типа данных, хранящейся в колонке информации.

Каждой строке в таблице соответствует объект DataRow. ADO.NET содержит методы для добавления, удаления, изменения и получения отдельной строки DataTable. Для таблиц, связанных с внешними источниками данных, любые изменения могут быть повторены во внешних источниках.

Имеется возможность установления связей между DataTable с использованием объектов DataRelation.

Произвольные ограничения могут быть наложены на таблицы и, составляющие их данные, с помощью объектов Constraint.

Объект DataView реализует представление в отдельной DataTable.

Таблицы могут быть объединены в DataSet. Однако, если планируется использовать в работе только одну таблицу БД, то эффективнее ограничиться использованием DataTable.

Для соединения с внешними данными, находящимися в БД, ADO.NET содержит множество провайдеров данных. Для соединения с БД, не имеющих собственного провайдера, используются общие провайдеры ODBC или OLEDB, которые тоже входяи в состав ADO.NET. Некоторые провайдеры данных распространяются самими производителями БД. Так обстоит ситуация с провайдером Oracle.

Все взаимодействия с внешними источниками данных осуществляются с помощью объекта Connection. ADO.NET использует технологию пула подключений для повышения скорости соединения с БД.

SQL запросы помещаются в объекты Command для отправки источнику данных. Объект Command может содержать необязательные объекты Parameter, которые позволяют вызывать хранимые процедуры или параметризованные запросы.

Объект DataAdapter содержит в себе стандартные запросы для взаимодействия с БД, тем самым исключая необходимость написания однообразных запросов, требующихся для чтения или записи отдельных строк.

Объект DataReader поддерживает быстрое чтение данных без возможности внесения изменений.

1. Обзор технологии LINQ to SQL
2. ДЗ

Измените WPF приложение для ведения списка сотрудников компании, из урока №5, **используя связывание данных, ListView, ObservableCollection и INotifyPropertyChanged**.  
1. Создайте сущности Employee и Department и заполните списки сущностей начальными данными.  
2. Для списка сотрудников и списка департаментов предусмотрите визуализацию (отображение). Это можно сделать, например, с использованием ComboBox или ListView.  
3. Предусмотрите возможность редактирования сотрудников и департаментов. Должна быть возможность изменить департамент у сотрудника. Список департаментов для выбора, можно выводить в ComboBox, это все можно выводить на дополнительной форме.  
4. Предусмотрите возможность создания новых сотрудников и департаментов. Реализуйте данную возможность либо на форме редактирования, либо сделайте новую форму.