

$$\frac{\varepsilon_i^+}{\mu_i + \nu_i + \varepsilon_i^-} < 1, \quad \frac{\varepsilon_i^-}{\tau_i} < 1,$$

а для любых $i = \overline{M+1, N}$ выполнено условие

$$\frac{\varepsilon_i^+}{\mu_i + \varepsilon_i^-} < 1,$$

тогда марковский процесс $x(t)$ эргодичен, а финальное стационарное распределение вероятностей состояний сети имеет следующий вид:

$$p(x) = p_1(x_1) p_2(x_2) \dots p_N(x_N), \quad x \in X,$$

где для $i = \overline{1, M}$

$$p(x_i) = \left[\frac{\varepsilon_i^+}{\mu_i + \nu_i + \varepsilon_i^-} \right]^{n_i} \cdot \left[\frac{\varepsilon_i^-}{\tau_i} \right]^{m_i} p_i(0), \quad p_i(0) = \frac{(\mu_i + \nu_i + \varepsilon_i^- - \varepsilon_i^+) \cdot (\tau_i - \varepsilon_i^-)}{(\mu_i + \nu_i + \varepsilon_i^-) \cdot \tau_i},$$

а для $i = \overline{M+1, N}$

$$p(x_i) = \left[\frac{\varepsilon_i^+}{\mu_i + \varepsilon_i^-} \right]^{n_i} \cdot p_i(0), \quad p_i(0) = \frac{(\mu_i + \varepsilon_i^- - \varepsilon_i^+)}{(\mu_i + \varepsilon_i^-)},$$

а $(\varepsilon_i^+, \varepsilon_i^-, i = \overline{1, N})$ находятся как решение нелинейных уравнений трафика.

Доказательство теоремы проводится стандартным образом, подстановкой стационарного распределения в уравнения равновесия.

Литература

1. Gelenbe, E. Stability of product-form G-networks/ E. Gelenbe, R. Schassberger // Probab. Eng. and Inf. Sci. 1992. V. 6. P.271–276.
2. Данфорд, Н. Линейные операторы. Общая теория. / Н. Данфорд, Дж. Т. Шварц М.:ИЛ, 1962. 522 с.
3. Якубович, О.В. Сети массового обслуживания со случайным временем пребывания в узлах положительных заявок и сигналов / О.В.Якубович // Теория вероятностей, математическая статистика и их приложения. Сборник научных статей. Выпуск 2. Минск: БГУ. 2009. С. 269–274.

УДК 004.7

Н. В. Лысенко, Н. Б. Осипенко

ИСПОЛЬЗОВАНИЕ ПЛАТФОРМЫ .NET ДЛЯ РАЗРАБОТКИ ПРИЛОЖЕНИЯ ЭКСПОРТА-ИМПОРТА ДАННЫХ

В статье описывается работа над созданием приложения, главной функцией которого является экспорт-импорт данных. Основные данные - это база данных, созданная на MS SQL Server. Доступ к данным обеспечивает разработанное приложение при помощи объектно-ориентированной технологии доступа к данным Entity Framework. Базу данных можно изменять как на сервере так и через приложение. Данные таблиц можно экспортировать в XML документ, возможно обновлять базу данных новыми данными, экспортировав данные XML документа в БД.

ADO.NET Entity Framework (EF) – объектно-ориентированная технология доступа к данным, является object-relational mapping (ORM) решением для .NET Framework от Microsoft. Предоставляет возможность взаимодействия с объектами как посредством LINQ в виде LINQ to Entities, так и с использованием Entity SQL. Для облегчения построения web-решений используется как ADO.NET Data Services (Astoria), так и связка из Windows Communication Foundation и Windows Presentation Foundation, позволяющая строить многоуровневые приложения, реализуя один из шаблонов проектирования MVC, MVP или MVVM.

Изначально с самой первой версии Entity Framework поддерживал подход Database First, который позволял по готовой базе данных сгенерировать модель edmx. Затем эта модель использовалась для подключения к базе данных. Позже был добавлен подход Model First. Он позволял создать вручную с помощью визуального редактора модель edmx, и по ней создать базу данных. Начиная с 5.0, предпочтительным подходом становится Code First. Его суть – сначала пишется код модели на с#, а затем по нему генерируется база данных. При этом модель edmx уже не используется.

Для начала работы с моим приложением необходимо выполнить SQL запрос, генерирующий базу данных на сервере. Данный запрос создает все необходимые таблицы, устанавливает связи между ними и заполняет их данными. Такой подход поможет в случае отказа системы быстро восстановить практически без потерь исходную базу данных.

В ходе выполнения работы было разработано клиент-серверное приложение для работы с базой данных аэропорта. Данное приложение позволит сотрудникам аэропорта быстро, легко и удобно получать всю необходимую информацию, через удобное меню.

Все данные хранятся на сервере, доступ к которым осуществляется через данное приложение. Обработка данных происходит также на сервере. Это позволит снизить требования к производительности рабочего компьютера.

Для работы приложения необходим сервер. В данном случае необходим MS SQL Server.

Основные функции и схему общения пользователя с приложением можно увидеть на рисунке 1.

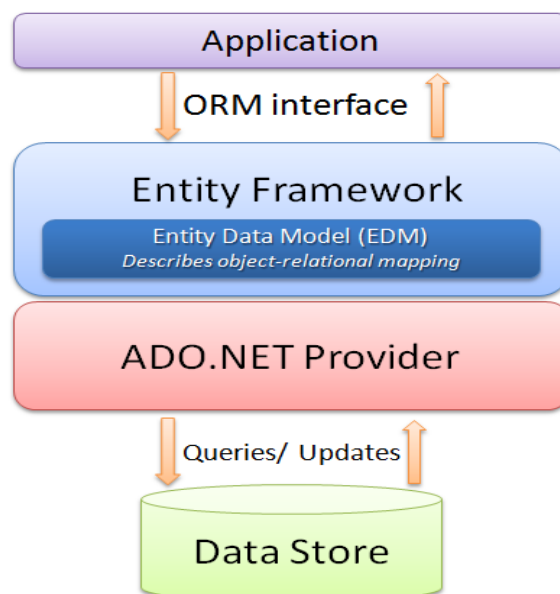


Рисунок 1 – Схема работы приложения

В ходе выполнения данной работы была разработана база данных, предназначенная для использования. Разработанная база данных состоит из 10 таблиц, описание которых в таблице 1.

Таблица 1 – Состав таблиц базы данных приложения

Имя таблицы	Описание
Покупатель	Информация о покупателях
Билет	Список билетов покупателя
Багаж	Список багажа билета
Расписание	Информация о маршрутах полета
Рейсы	Информация о рейсах
Самолеты	Информация о самолетах авиакомпаний
Марка	Список марок самолетов
Авиакомпании	Список авиакомпаний
Сотрудники	Список сотрудников самолёта
Экипаж	Список состава экипажа рейса

Связи между таблицами базы данных отображены на рисунке 2.

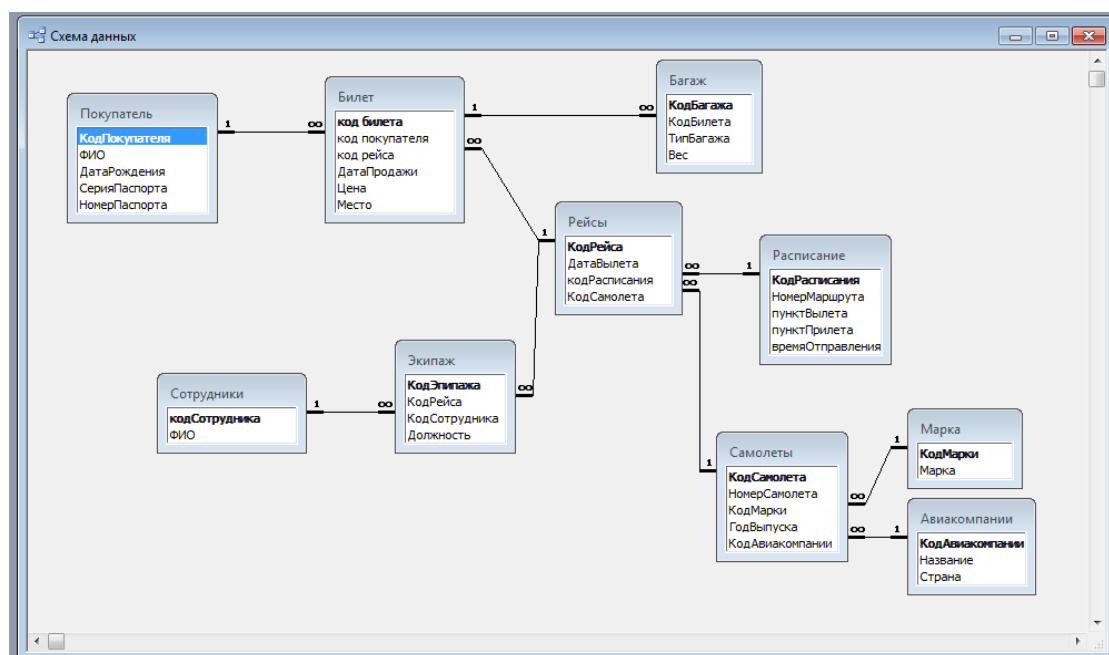


Рисунок 2 – Связи между таблицами

В данном приложении существует 2 вида таблиц:

- таблицы-справочники;
- дочерние таблицы.

Таблицы-справочники содержат в себе основную информацию аэропорта. В данном приложении это информация о покупателях, расписание полетов, экипаж самолета, авиакомпании и марка самолета.

Данные таблицы можно изменять 3-мя разными способами:

1. через SQL запрос к БД;
2. через форму таблицы;
3. через импорт XML документа.

Для открытия XML документа необходимо выбрать пункт меню – открыть файл. После этого в диалоговом окне выбрать нужный XML документ. Выбранный документ сохранится в памяти приложения. После этого от отправки данных из этого документа необходимо выбрать пункт меню отправить. В выпадающем списке будет отображаться только доступная таблица соответствующая формату XML выбранного документа.

Для изменения данных через приложение необходимо выбрать соответствующий пункт меню таблиц (Рисунок 3).

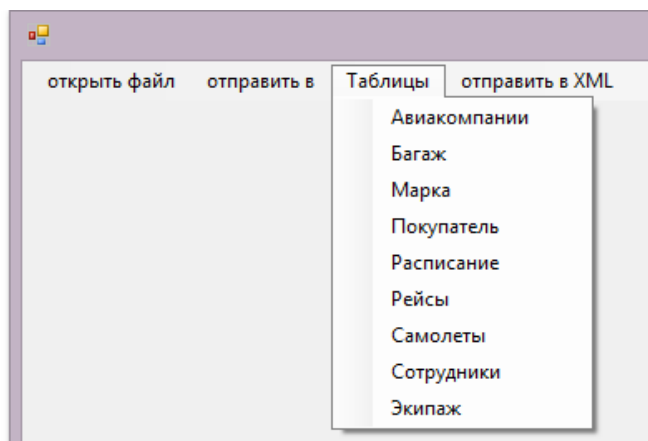


Рисунок 3 – Выбор таблицы-справочника

После выбора нужной таблицы откроется её форма (Рисунок 4).

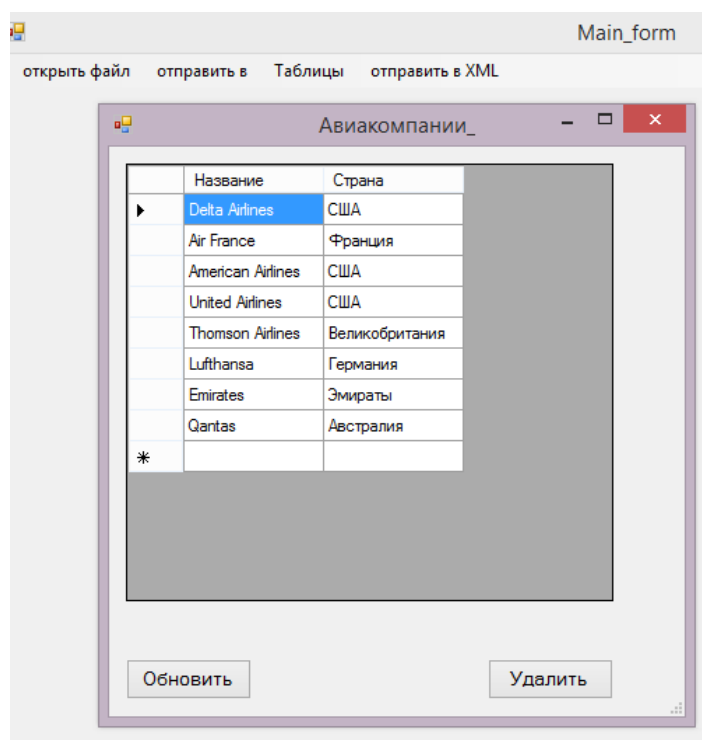


Рисунок 4 – Отображение таблицы на форме

Для добавления записи необходимо ввести данные в нижние поля таблицы после чего нажать на кнопку обновить. На экране отобразится информационное сообщение на рисунке 5.

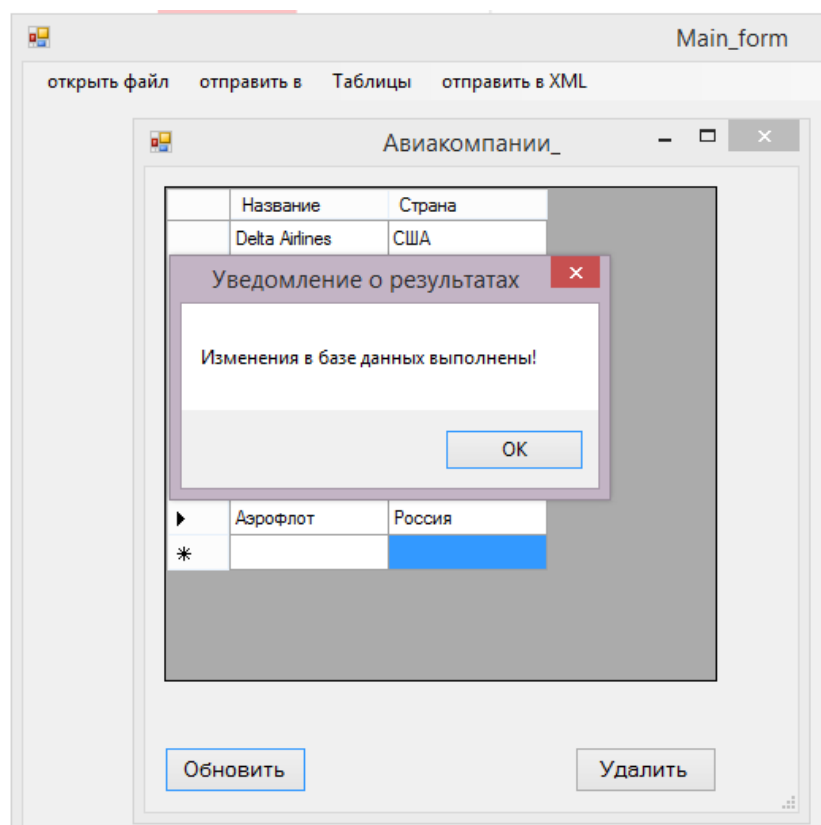


Рисунок 5 – Информационное сообщение

Для отправки данных таблицы в XML документ необходимо открыть нужную таблицу и выбрать пункт меню отправить в XML. После этого отобразится информационное сообщение и в корне программы будет создан XML документ с данными таблицы.

Помимо таблиц-справочников существуют дочерние таблицы. В них содержится дополнительная информация о таблицах-справочниках. Дочерние таблицы хранят в себе данные о билетах покупателя, его багаже, информацию о рейсах, самолетах и экипаже рейса.

Каждая дочерняя таблица изменяется, как и таблицы-справочники. Вместо кодов родительских таблиц используется удобный выпадающий список родительских полей. Изменение, удаление и обновление производится аналогично таблицам-справочникам.

В результате выполнения данной работы было разработано приложение экспорта-импорта базы данных управления аэропорта. Доступ к данным был осуществлен при помощи entity framework. Данные таблиц можно как экспортировать в XML документ, так и импортировать из него в соответствующую этому документу таблицу.

Возможности платформы entity framework значительно упрощают задачи доступа к данным БД. Автоматическая генерация модели данных, автоматическое создание классов и свойств значительно сокращают разработку приложений.

Литература

1 Винкоп, С. Использование Microsoft SQL Server 7.0 : специальное издание / С. Винкоп. – СПб. : Издательский дом «Вильямс», 2001. – 816 с.

2 Хоторн, Р. Разработка баз данных Microsoft SQL Server 2000 на примерах / Р. Хоторн. – М. : Бинوم, 2001. – 464 с.

- 3 Коггзолл, Д., РНР 5. Полное руководство / пер. с англ. – СПб. : Питер, 2010. – 752 с.
- 4 Троелсен, Э. Язык программирования С# 5.0 и платформа .NET 4.5 / Э. Троелсен. – 6-е изд. – СПб. : Питер, 2013. – 1312 с.
- 5 Фримен, А. ASP.NET 4.5 с примерами на С# 5.0 для профессионалов / А. Фримен. – СПб. : Издательский дом «Вильямс», 2014. – 1120 с.
- 6 Албахари, А., Албахари Б. С# 5.0. Справочник. Полное описание языка/ А. Албахари, Б. Албахари. – СПб.: Издательский дом «Вильямс», 2012. – 1008 с.
- 7 Гриффитс, И, Э. Программирование на С# 5.0 / И. Гриффитс – М.:«Эксмо», 2014. – 1136 с.
- 8 Шилдт, Г. С# 4.0. Полное руководство / Г. Шилдт. – СПб. : Издательский дом «Вильямс», 2011. – 1056 с.

УДК 53(077)

В. В. Малиновский

ДИДАКТИЧЕСКИЕ ОСНОВЫ СИСТЕМЫ РАБОТЫ УЧИТЕЛЯ ФИЗИКИ ПО АКТИВИЗАЦИИ ПОЗНАВАТЕЛЬНОЙ ДЕЯТЕЛЬНОСТИ УЧАЩИХСЯ

Статья посвящена дидактическим основам активизации познавательной деятельности и развитию мотивов учения учащихся при изучении физики. Рассмотрены основные методические приемы для мотивации учащихся к обучению, развитию их логического мышления и повышению качества знаний. На основе дидактических приемов мотивов обучения приведены основные цели для закрепления знаний учащихся и развитию их экспериментальных навыков.

Основная цель работы учителя по активизации познавательной деятельности учащихся – развитие их творческих способностей. Достижение этой цели позволяет решить многие задачи обучения, обеспечить прочные и осознанные знания изучаемого материала, подготовить учащихся к активному участию в производственной деятельности, умению самостоятельно пополнять знания, воплощать в жизнь научно-технические решения, осваивать новые специальности, дать высшим учебным заведениям страны хорошо подготовленных абитуриентов, способных творчески овладеть выбранной специальностью.

Развитие творческих познавательных способностей учащихся, цель деятельности учителя, а применение различных приемов активизации является средством достижения этой цели. Понимание этого важно для работы учителя. Заботясь о развитии учащихся, необходимо чаще использовать активные методы обучения. Но одновременно необходимо отдавать себе отчет в том, являются ли используемые приемы и методы оптимальными, отвечающими имеющемуся развитию учащихся и задаче дальнейшего совершенствования их познавательных умений.

Применяя те или иные методы и приемы активизации, необходимо всегда учитывать имеющийся уровень развития познавательных способностей учащихся. Сложные познавательные задачи можно предъявлять лишь ученикам, обладающим высоким уровнем развития познавательных способностей. Задачи, не соотнесенные с уровнем развития познавательных сил учащегося, превышающие возможности ученика, предъявляющие к нему требования, значительно опережающие уровень имеющегося у него развития, не могут сыграть положительную роль в обучении. Они подрывают у учащихся веру в свои силы и способности.

Необходимость соотносить предъявляемые учащимся задания с уровнем их развития вытекает из теории мышления. Все это позволяет заключить, что развитие познавательных способностей учащихся – длительный процесс.