|  |  |
| --- | --- |
| Департамент образования Ярославской области  Государственное профессиональное образовательное автономное учреждение  Ярославской области  «ЯРОСЛАВСКИЙ ПРОМЫШЛЕННО-ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ  им. Н.П. Пастухова» | |
| **Отчет по УЧЕБНОЙ практике**  **ПМ.02 Осуществление интеграции программных модулей**  УП.02.09.02.07.УП02.10 | |
|  | Студент  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Д.А.Клюева  « 30 » апреля 2021 г. |
| Руководитель практики  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ю.В.Маянцева  « 30 » апреля 2021 г. |
| 2021 | |

Содержание

[Практическая работа №1 2](#_Toc70682022)

[Практическая работа №2 10](#_Toc70682023)

[Практическая работа №3 16](#_Toc70682024)

[Практическая работа №4 22](#_Toc70682025)

# 

# **Практическая работа №1**

**Тема**

Подключение базы данных. Авторизация пользователя.

**Цель работы**

По заданной схеме данных разработать базу данных, создать программное приложение с возможностью авторизации в системе.

После запуска Microsoft SQL необходимо подключиться и в разделе диаграммы баз данных создать новую пустую диаграмму без таблиц. Для этого необходимо нажать на «Диаграммы баз данных» правой кнопкой мыши и выбрать «Создать диаграмму баз данных».

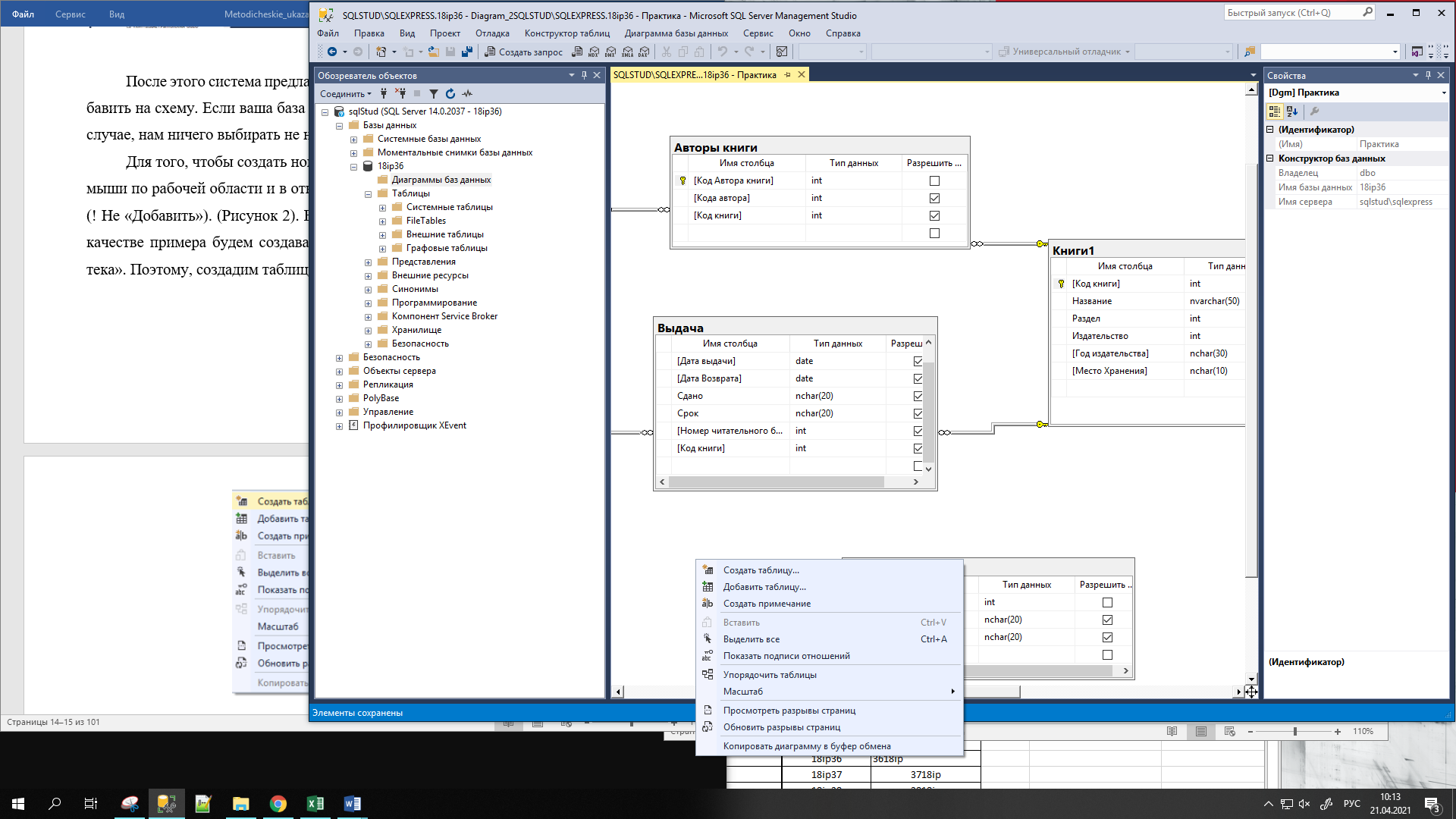


Рисунок 1. Создание таблицы

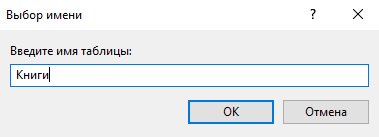


Рисунок 2. Название таблицы

Таким образом создаем все необходимые таблицы. В каждой таблице должен быть задан первичный ключ, для того чтобы его задать необходимо выбрать поле, которое будет уникальный и не иметь повторений и присвоить ему тип данных int. После того как тип данных присвоен выделяем полностью строку и выбираем «задать первичный ключ». Таким образом поступаем со всеми таблицами, каждое поле таблицы имеет свой тип данных. После того таблицы построены и присвоены первичные ключи необходимо задать связи между таблицами. Связь можно задать если перетянуть ячейку к ячейке другой таблицы. Когда таблицы построены и соединены между собой уже выстраивается сама диаграмма базы. В дальнейшем необходимо перейти во вкладку таблицы и выбрать интересующую нас таблицу, нажать правой кнопкой мыши и выбрать изменить первые 200 строк. Откроется наша пустая таблица, в которую мы вводим данных в соответствии с названиями столбцов. Ключевое поле остается неизменным так как там стоит счётчик.

Для подключения базы данных к проекту Visual Studio необходимо добавить элемент \*\*\*\*\*\* и создаем его.

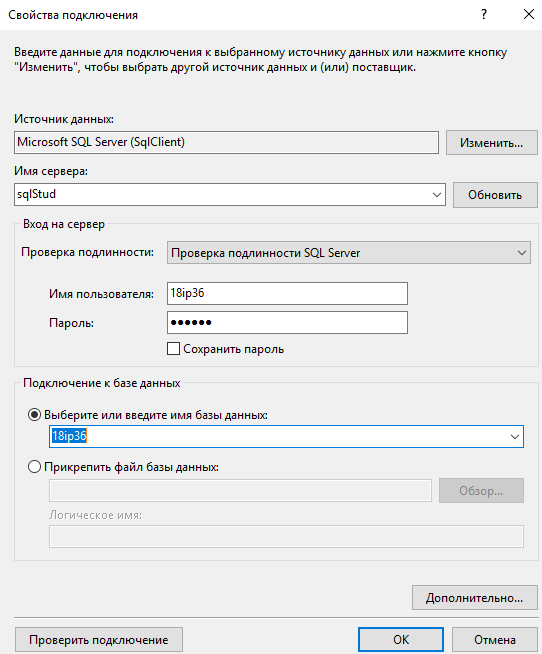


Рисунок 3. Присвоение соединения

Для подключения базы указываем имя сервера и в поле проверка подлинности устанавливаем «Проверка подлинности SQL Server». Дальше необходимо указать имя пользователя и пароль к записи в SQL Server. Имя базы соответствует имени пользователя. После заполнения данных нажимаем на кнопку «Проверка подключения» и проверяем всё указали мы правильно. Проверка покажет подключится база или нет.

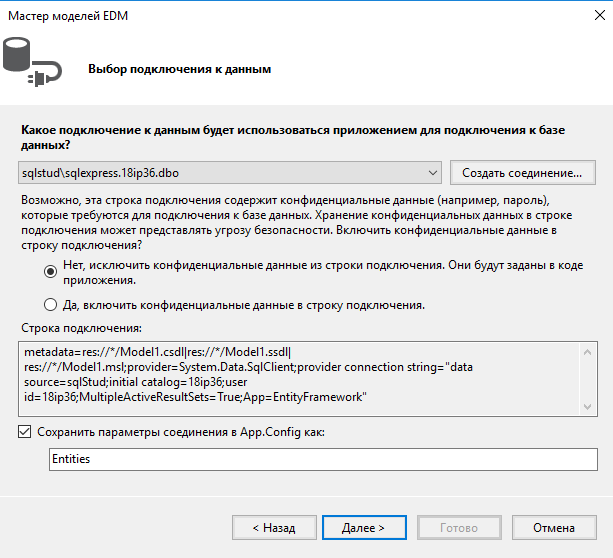


Рисунок 4. Выбираем подключение к данным

Когда мы создали соединение выбираем пункт «Да, включить конфиденциальные данные в строку подключения». Если выбрать поле с «Нет» тогда буду возникать ошибки при переносе данных из базы данных в проект. Далее выбираем интересующие нас данные и выполняем подключение.

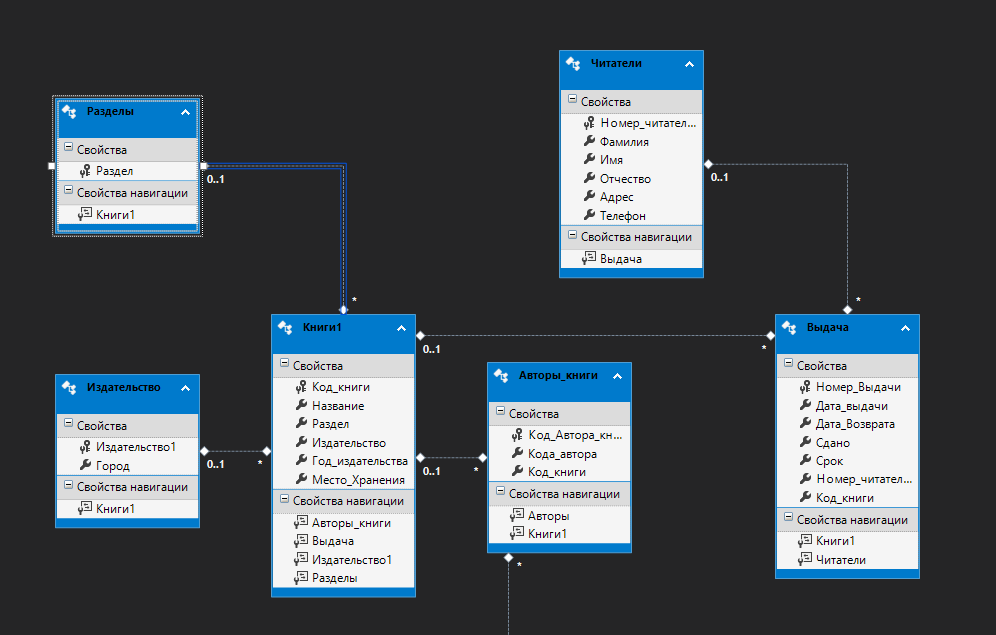


Рисунок 5. Подключенная база

Подключение базы выполнено успешно, дальше накидываем простой интерфейс авторизации. Класс в программе объявляется для того чтобы база подключилась к программе и данные могли переноситься на формы.

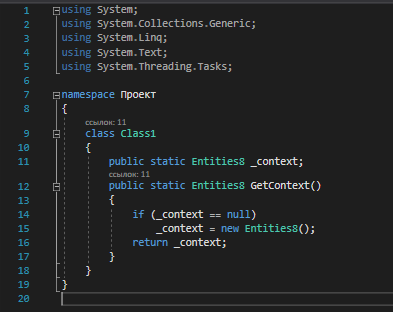


Рисунок 6 Установка класса

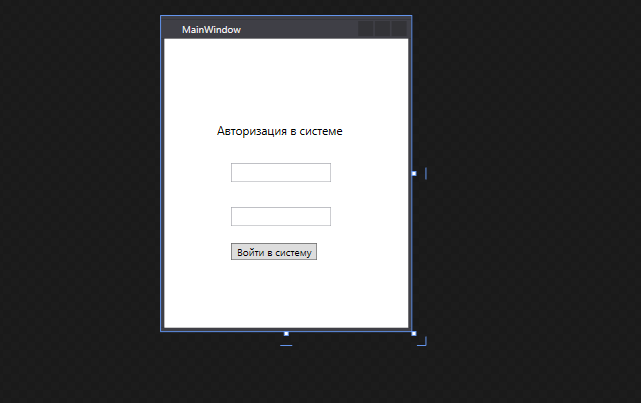


Рисунок 7. Форма авторизации

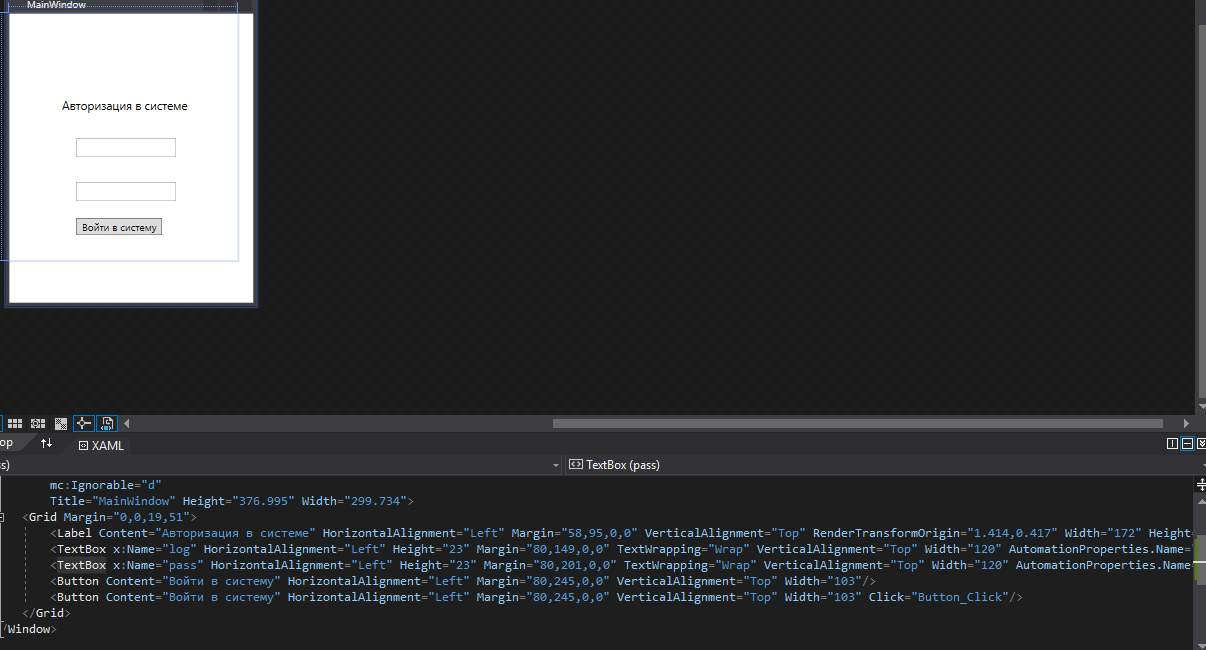


Рисунок 8. Присваиваем имена элементам

Для каждого label, button, textbox присваиваем имя с помощью команды x: Name, в свойствах выбираем поле Text и устанавливаем необходимые названия элементов.

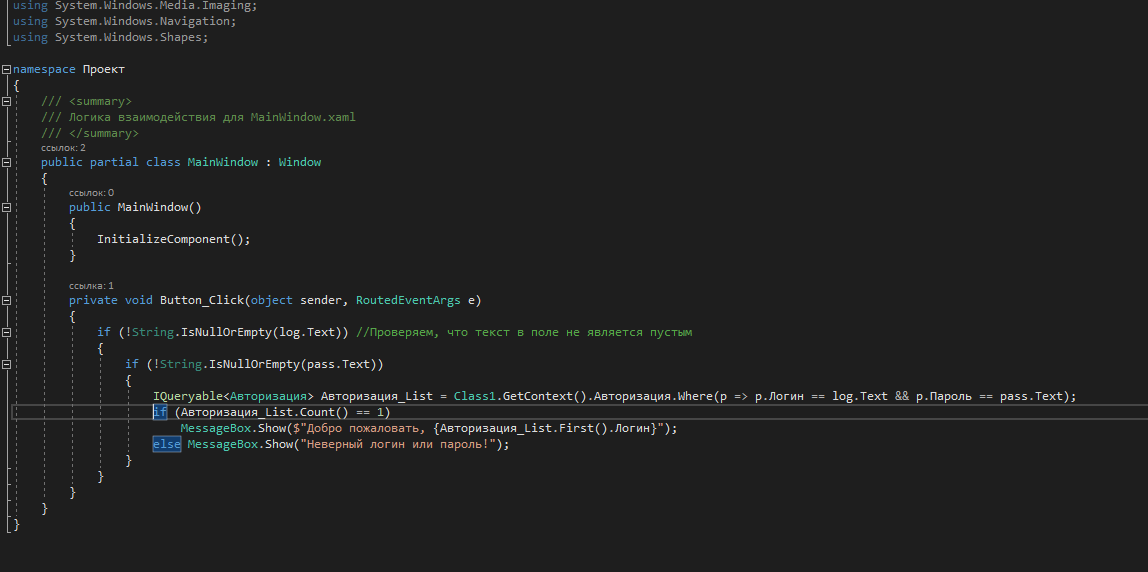


Рисунок 9. Установка соединения с базой

После подключения базы и разработки формы авторизации в программном коде прописываем проверка на пустоту textbox. Это делается для того чтобы программа не авторизировала пользователя с не заполненным логином и паролем.

Используя команду IQueryble устанавливаем соединение таблицы Авторизация с формой. Из таблицы берем такие данные как: логин и пароль. После прописываем условие: если введённые данные соответствуют данным из БД, то пользовать авторизуется и появляется надпись «Добро пожаловать, имя пользователя», если пользователь ввёл некорректные данные то появляется сообщение об ошибке «Неверный логин или пароль» На рисунке 9 показана успешная авторизация пользователя.

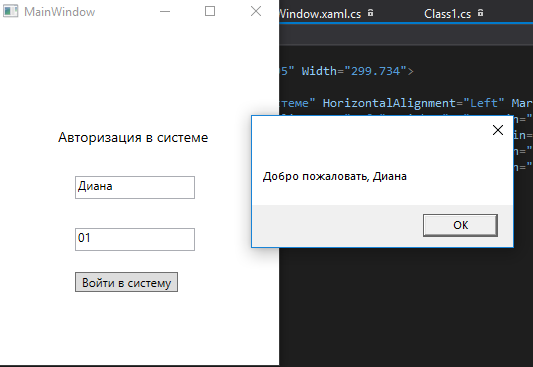


Рисунок 10. Авторизация пользователя

**Контрольные вопросы**

1. **Что такое «база данных»?**

База данных - набор сведений, хранящихся некоторым упорядоченным способом.

1. **Из каких этапов состоит проектирование базы данных?**

* Концептуальное проектирование
* Логическое проектирование
* Физическое проектирование

1. **Что подразумевается под таблицей и полем в SQL?**

Таблица – организованный набор данных в виде строк и столбцов

Поле – пересечение строки и столбца

1. **Что такое первичный ключ (Primary key)?**

Первичный ключ (Primary key) - столбец, значения которого во всех строках различны.

1. **Что такое внешний ключ (Foreign key)?**

Внешний ключ – обеспечивает однозначную логическую связь, между таблицами одной БД.

1. **Какие связи существуют в реляционных базах данных?**

* Один к оному
* Один ко многим
* Много к одному
* Многие ко многим

1. **Что подразумевается под целостностью данных?**

Правильность данных в любой момент времени

1. **Что такое нормализация и каковы ее преимущества?**

Нормализация – это процесс, направленный на уменьшение избыточности информации в реляционной БД

1. **Как создать подключение к базе данных?**
2. Для этого на панели проекта необходимо нажать правой кнопкой мыши по нашему проекту и выберем пункт «Добавить» -> «Создать новый элемент».
3. Из списка элементов выбрать элемент «Модель ADO.NET EDM». Задать данному элементу название.
4. Откроется мастер создания моделей EDM. Далее необходимо выбрать «Конструктор EF из базы данных» и нажать кнопку «Далее».
5. В следующем окне необходимо нажать на кнопку «Создать соединение»
6. В окне «Свойства подключения» указать актуальные данные подключения (рисунок 18). Поменять «Проверка подлинности Windows» на «Проверка подлинности SQL»
7. Выбор версии необходимо оставить без изменений
8. В следующем окне указать необходимые таблицы. Пространство имен может указать свое название, либо оставить без изменений.
9. После завершения работы с мастером, система покажет схему данных вашей базы данных.

Вывод: Для выполнения данной практической работы были изучены методические указания и предложенный курс. В данной практической работе мы приобрели навыки построения и подключения базы данных, а также работе с WPF.

**Практическая работа №2**

**Тема**

Отображение данных на форме.

**Цель работы**

Реализовать отображение данных базы данных на форме.

Для выполнения данной практической работы необходимо создать новую форму, в которую мы добавляем lable, button, image, datagrid. Каждому объекту необходимо присвоить имя с помощью команды x:Name и Binding для дальнейшего добавления в запросы для получения данных в таблицу.

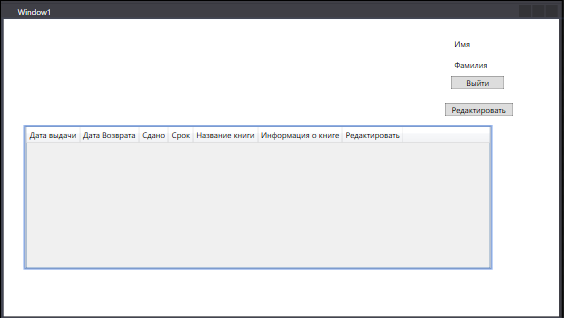


Рисунок 11. Форма с выводом данных из базы

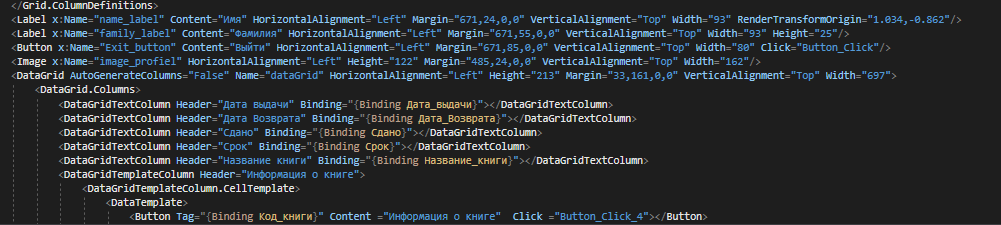


Рисунок 12. Установка имен

Для вывода в окно имени и фамилии необходимо добавить в базу данных в таблицу авторизация эти поля и заполнить таблицу данными. Устанавливаем соединение с таблицей из базы данных «Авторизация». Перечисляем данные которые переносятся в lable.

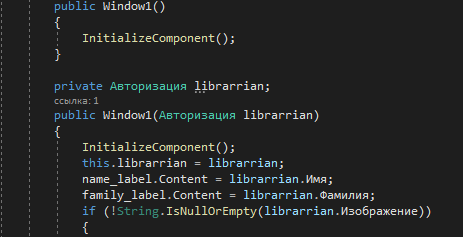


Рисунок 13. Перенос данных из базы

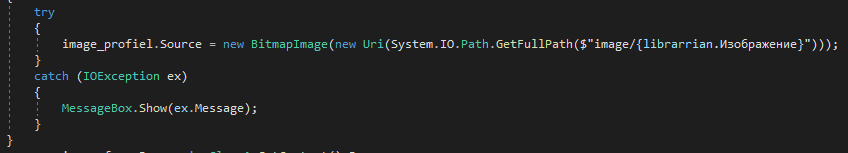


Рисунок 14. Добавление изображения

Устанавливаем подключение с базой librarrian. Так же указываем путь для переноса изображения в image. Для этого необходимо предварительно добавить его в базу данных. Для этого в таблицу Авторизация добавляем столбец с названием «Изображение» туда вставляем полное название изображением и тип формата. Чтобы фотография попала в элемент image необходимо открыть папку программы, перейти в папку bin, debug и создать папку image в которую поместим фотографии для переноса в программу.

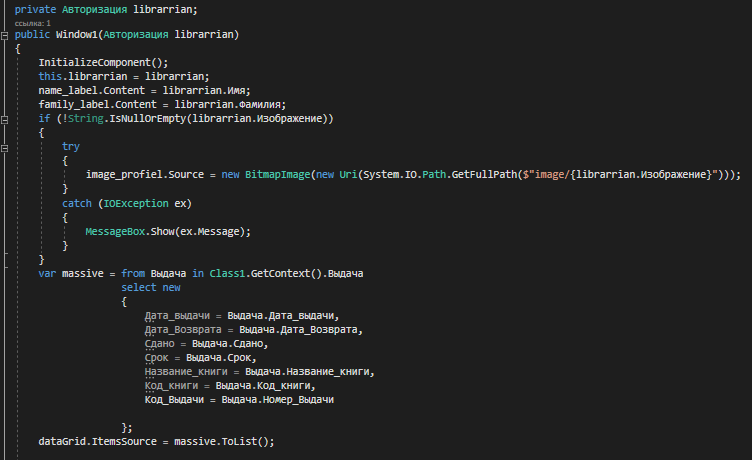


Рисунок 15. Настройка пути для переноса данных

Создаем массив и устанавливаем связь с таблицей «выдача». Необходимо вывести поля: дата выдачи, дата возврата, срок, название книги. В результате выполнения программного кода мы имеем результат как на рисунке 16.

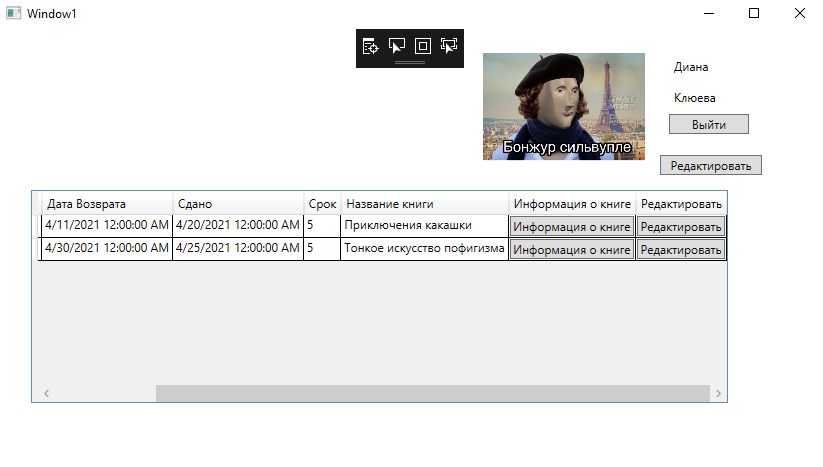


Рисунок 16. Готовое окно с выводом данных

Необходимо создать новое окно с подробной информацией о книге. Для этого переносим на форму lable и textbox. Каждому объекту присваиваем имя.

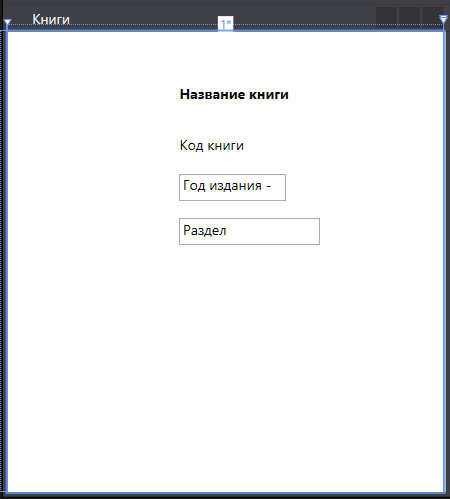


Рисунок 17. Форма для описания книги

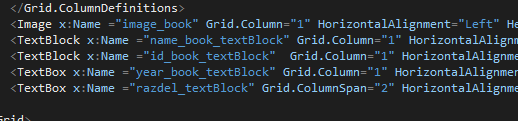


Рисунок 18. Присвоение имен

Создаем в DataGrid кнопку button для перехода в новое окно «информация о книге». Для этого создаем событие click.



Рисунок 19. Создаем кнопку перехода

В событии click прописываем переход на новую форму.

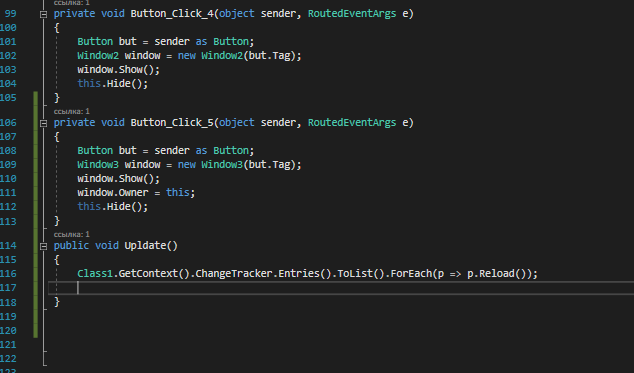


Рисунок 20. Кнопка перехода

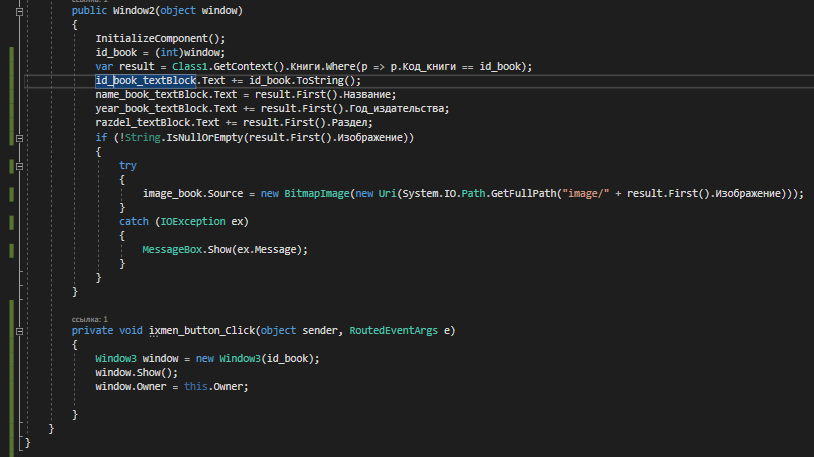


Рисунок 21. Вывод данных

Устанавливаем соединение с таблицей Книги и выводим такие поля как название, код книги, год издания, раздел, изображение.

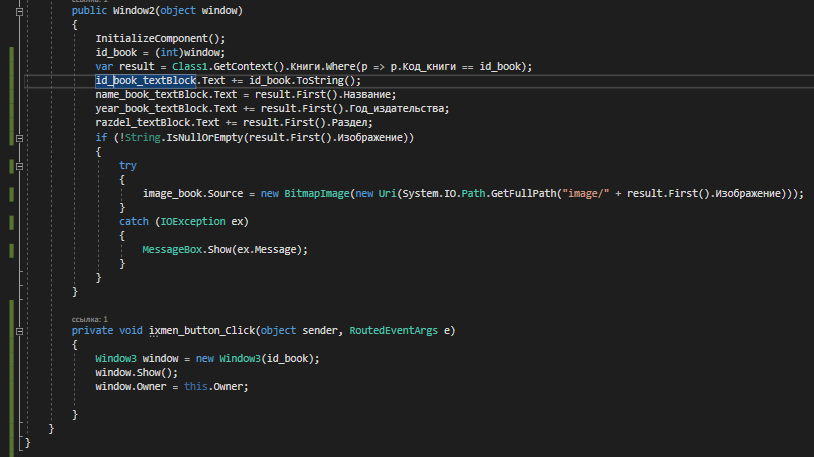


Рисунок 22. Вывод изображения

В базу данных добавляем поле «изображение» в которые заносим название изображения и его расширение. Так же, как и ранее заносим данных изображения в папку и прописываем путь для вывода изображения в image. Результат выполнения операций представлен на рисунке 21.

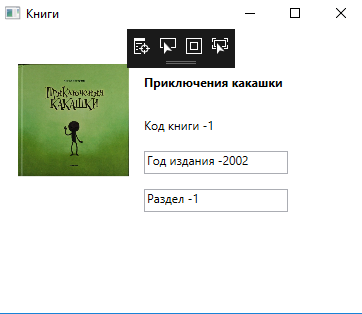


Рисунок 23. Информация о книге

**Контрольные вопросы**

1. **Что такое Windows Presentation Foundation? В чем преимущество перед WinForms?**

Windows Presentation Foundation (WPF) — аналог WinForms, система для построения клиентских приложений Windows с визуально привлекательными возможностями взаимодействия с пользователем, графическая (презентационная) подсистема в составе .NET Framework (начиная с версии 3.0), использующая язык XAML.

Преимущества WPF:

* Использование традиционных языков .NET-платформы - C# и VB.NET для создания логики приложения;
* Возможность декларативного определения графического интерфейса с помощью специального языка разметки XAML
* Независимость от разрешения экрана
* Новые возможности, которых сложно было достичь в WinForms, например, создание трехмерных моделей, привязка данных, использование таких элементов, как стили, шаблоны, темы и др.;
* Хорошее взаимодействие с WinForms
* Богатые возможности по созданию различных приложений: это и мультимедиа, и двухмерная и трехмерная графика, и богатый набор встроенных элементов управления
* Аппаратное ускорение
* Создание приложений под множество ОС семейства Windows - от Windows XP до Windows 10;

1. **Что такое XAML? Для чего он используется в WPF?**

XAML (eXtensible Application Markup Language) - язык разметки, используемый для инициализации объектов в технологиях на платформе .NET. Применительно к WPF (а также к Silverlight) данный язык используется прежде всего для создания пользовательского интерфейса декларативным путем.

1. **Что такое LINQ? Как работает технология LINQ?**

Аббревиатура LINQ обозначает целый набор технологий, создающих и использующих возможности интеграции запросов непосредственно в язык C#. Традиционно запросы к данным выражаются в виде простых строк без проверки типов при компиляции или поддержки IntelliSense. Кроме того, разработчику приходится изучать различные языки запросов для каждого типа источников данных: баз данных SQL, XML-документов, различных веб-служб и т. д. Технологии LINQ превращают запросы в удобную языковую конструкцию, которая применяется аналогично классам, методам и событиям.

Вывод: Для выполнения данной практической работы были изучены методические указания и предложенный курс. В данной практической работе мы приобрели навыки вывода информации из базы данных в программу.

**Практическая работа №3**

**Тема**

Редактирование данных в БД с помощью приложения.

**Цель работы**

Реализовать редактирование данных из базы данных с помощью программного приложения.

Для выполнения данной практик создадим новое окно «редактировать аренду», в которое добавим такие поля как label, datapicter, conbobox. В полях datapicter устанавливается дата выдачи книги, дата возврата книги и когда она сдана. Поле conbobox выводит список всех книг из базы и номера читательских билетов.

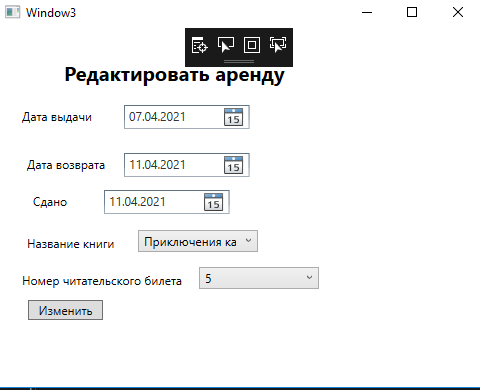


Рисунок 24. Окно редактирования аренды

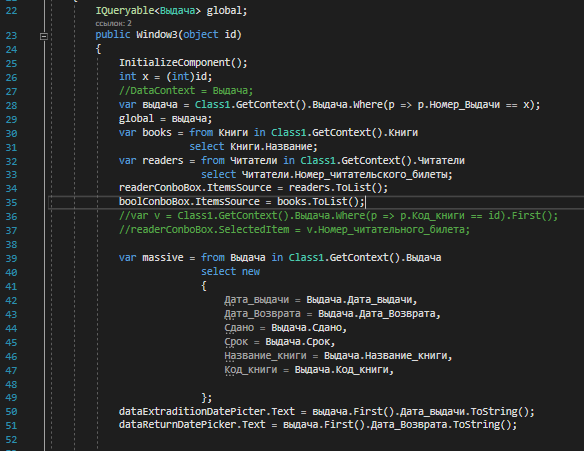


Рисунок 25. Соединение таблицы

Создаем новую переменную x и устанавливаем соединение с таблицей «выдача». Из таблицы необходимо вывести поля название и номер читательского билета.

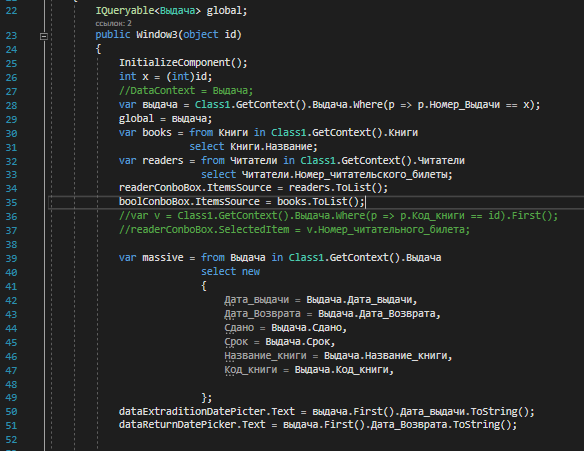


Рисунок 26. Выборка полей таблицы

После мы создаем массив и прописываем необходимые поля для соединения таблицы и дальнейшего внесения изменений.

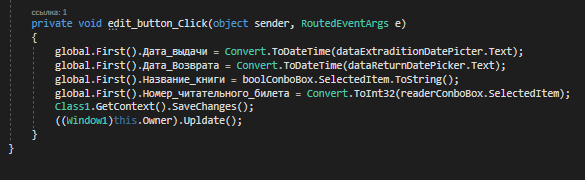


Рисунок 27. Изменение данных

На рисунке 27 представлен код, в котором мы прописываем в какие поля будут вносится изменения и в дальнейшем переноситься в базу данных.

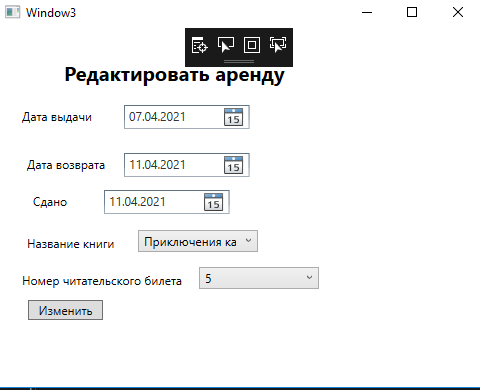


Рисунок 28. Окно редактирования

Вводим данные для редактирования аренды книги. Каждое поле заполняем, указывается название книги и номер читательского билета. Дальше устанавливается дата выдачи книги и возврата. В поле сдано указывается дата, с которой книга возвращается в библиотеку.

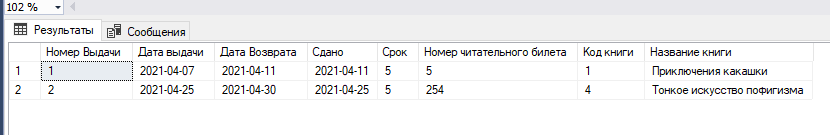


Рисунок 29. Запрос в БД

Создадим запрос в базе данных на вывод информации из таблицы. Наши изменения из окна перенеслись в базу данных. Дальше необходимо сделать так чтобы пользователь мог менять изображение. Для этого переходим в код и прописываем событие click для кнопки «редактировать». Дальше прописываем путь для изображения так же, как и ранее. Благодаря этому мы можем выбрать любое изображение и вставить его

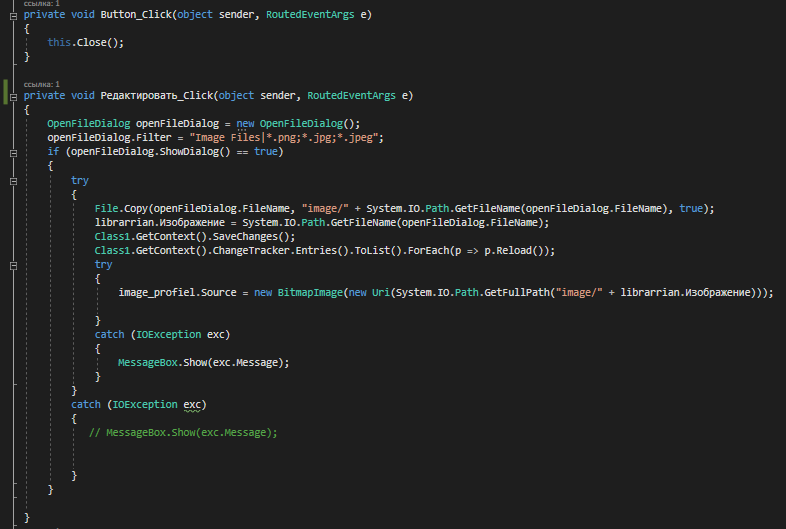


Рисунок 30. Установка пути для смены изображения

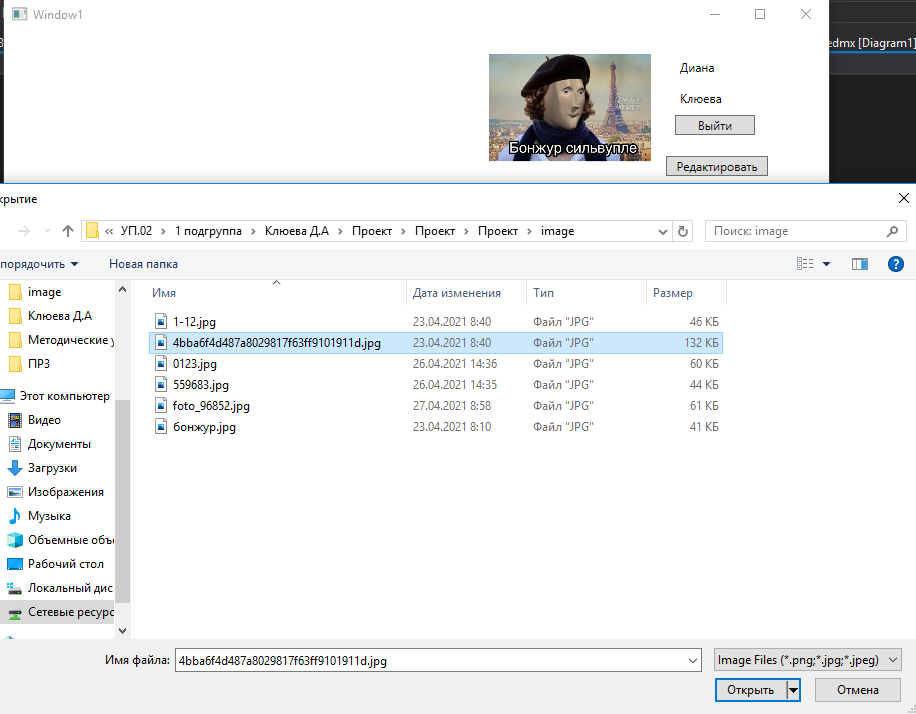


Рисунок 31. Выбор изображения

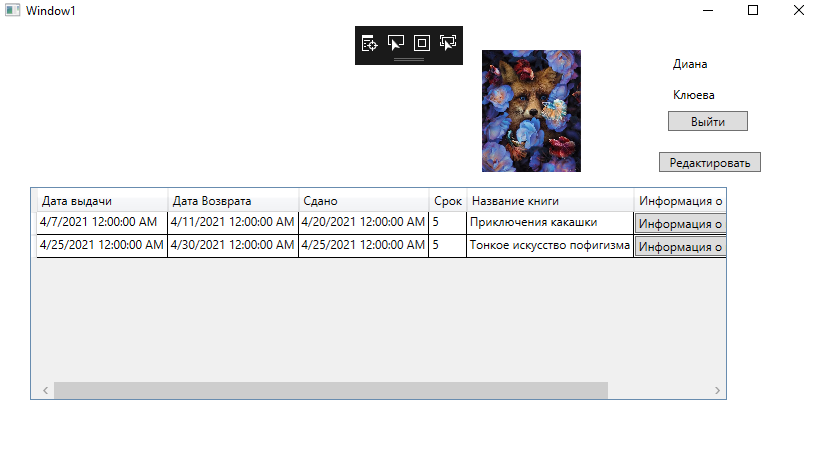


Рисунок 32. Новое изображение

**Контрольные вопросы**

1. **Что такое «реляционная алгебра»? Для чего она используется?**

Реляционная алгебра базируется на теории множеств и является основой логики работы баз данных.

Реляционная алгебра состоит из операций над отношениями и их составляющими (атрибутами и кортежами). Результат любой операции реляционной алгебры - новое отношение. Такие системы операций называются замкнутыми. Рассмотрим семь основных операций (рис. 1) реляционной алгебры. Они разделены на две группы. В первую входят операции, совершаемые над любыми множествами:

* объединение,
* пересечение,
* разность
* декартово произведение.

Во вторую группу входят операции, применимые только к отношениям:

* выборка,
* проекция,
* соединение.

1. **Перечислите основные операции реляционной алгебры. Расскажите подробно об одной из них.**

**Объединение** - операция над двумя отношениями, в результате которой получается новое отношение, состоящее из всех кортежей исходных отношений. Общие для исходных отношений кортежи в новом отношении встречются только по одному разу.

**Пересечение** - операция над двумя отношениями, в результате которой получается новое отношение, состоящее из кортежей, принадлежащих обоим исходным отношениям.

**Разность** - операция над двумя отношениями, в результате которой получается новое отношение, состоящее из кортежей, принадлежащих первому отношению и не принадлежащих второму.

**Декартово произведение** - операция над двумя отношениями, в результате которой получается новое отношение, состоящее из всех возможных кортежей, являющихся попарными сочетаниями кортежей исходных отношений.

**Выборка -** производится над кортежами одного отношения. Результат выборки - новое отношение, состоящее из котежей исходного отношения, удовлетворяющих заданному условию.

**Проекция** также производится над кортежами одного отношения. Результат проекции - новое отношение содержащее только заданные атрибуты исходного отношения.

**Соединение** - операция над двумя отношениями, имеющими общие атрибуты, в результате которой получается новое отношение, состоящее из всех атрибутов исходных отношений и объединяющее только те кортежи исходных отношений, в которых значения общих атрибутов совпадают.

1. **Какую операцию реляционной алгебры вы использовали в данной работе и зачем?**

Не использовала

Вывод: Для выполнения данной практической работы были изучены методические указания и предложенный курс. В данной практической работе мы приобрели навыки внесений составления формы изменений. Через эту форму мы изменяем записи в базу данных.

# **Практическая работа №4**

**Тема:** Добавление и удаление данных из БД с помощью приложения.

**Цель работы:** Реализовать добавление и удаление данных из базы данных с помощью программного приложения.

Для выполнения данной практической работы необходимо создать новое окно и вынести на него такие элементы как lable, DataPicker, comboBox, button. Необходимо выполнить добавление записи аренды, для этого нам необходимо вывести такие поля как дата возврата дата выдачи, сдано, название книги, номер читательского билета.

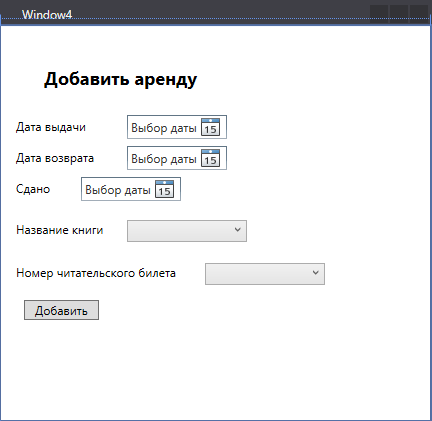


Рисунок 33. Окно добавления аренды

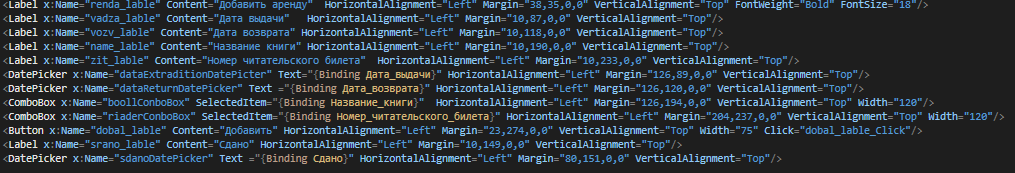


Рисунок 34. Присвоение имен и свойств

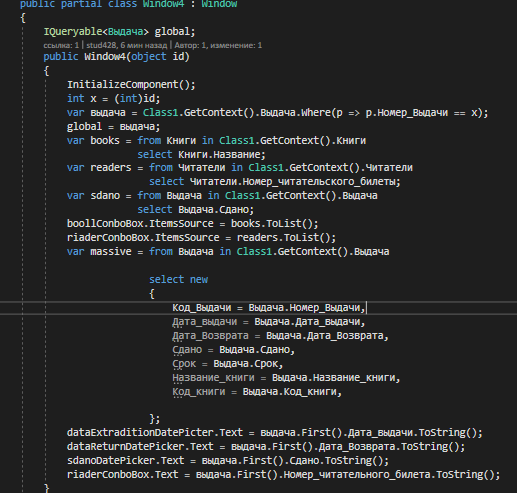


Рисунок 35. Установка соединения и выборка полей

На рисунке 35 указан код используемый для определения соединения с таблицей выдача и установка интересующих нас полей.

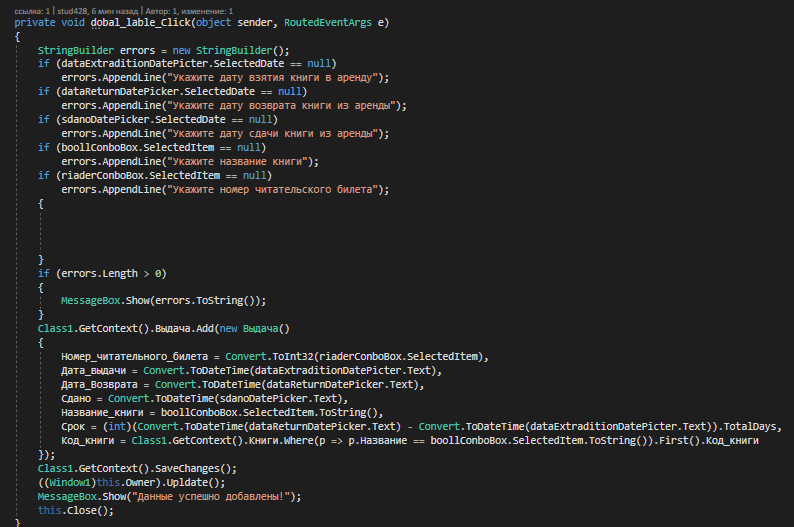


Рисунок 36. Сообщение пользователю

Указываем вывод информации для пользователя чтобы все поля для добавления были заполнены.

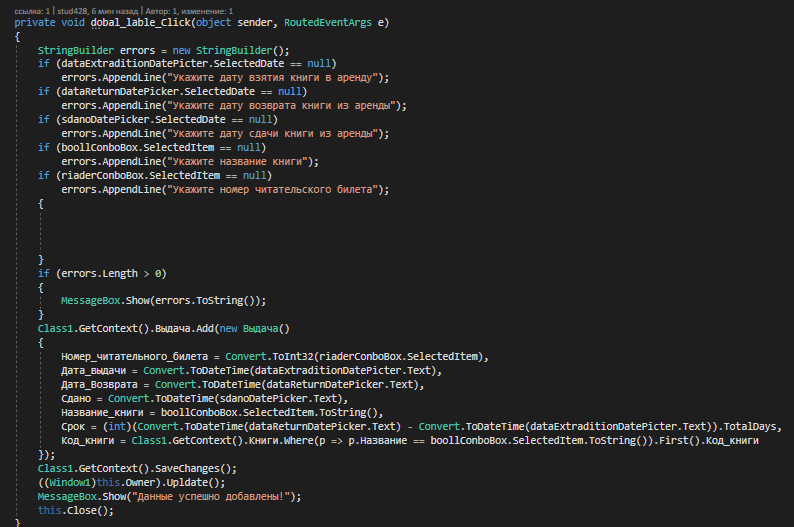


Рисунок 37. Конвертирование данных

Указываем выбранные поля и конвертируем данных в необходимый тип. После того как все поля заполнены и проверены нажимаем кнопку «добавить» и появляется сообщение пользователю «Данные успешно добавлены». Добавленная запись тут же отражается в программе и базе данных.

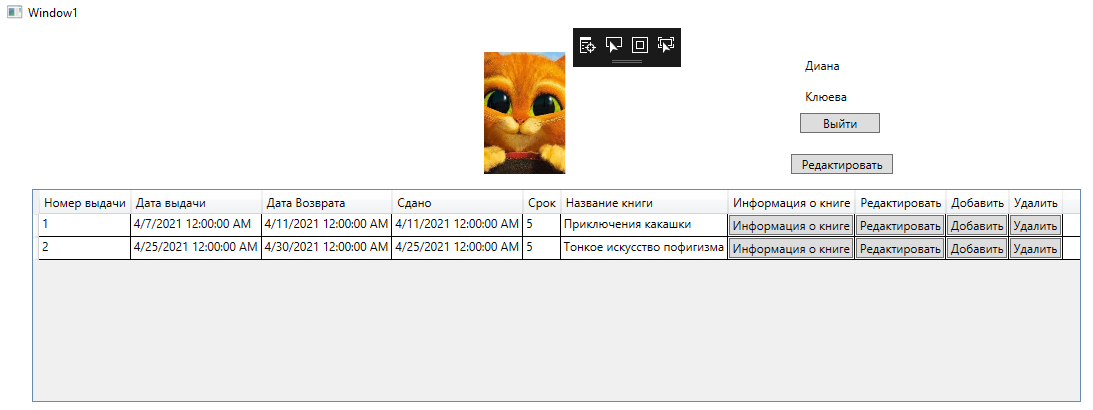


Рисунок 38. Окно с выводом данных

Запущенная программа и выводимые из неё данных о аренде книги.

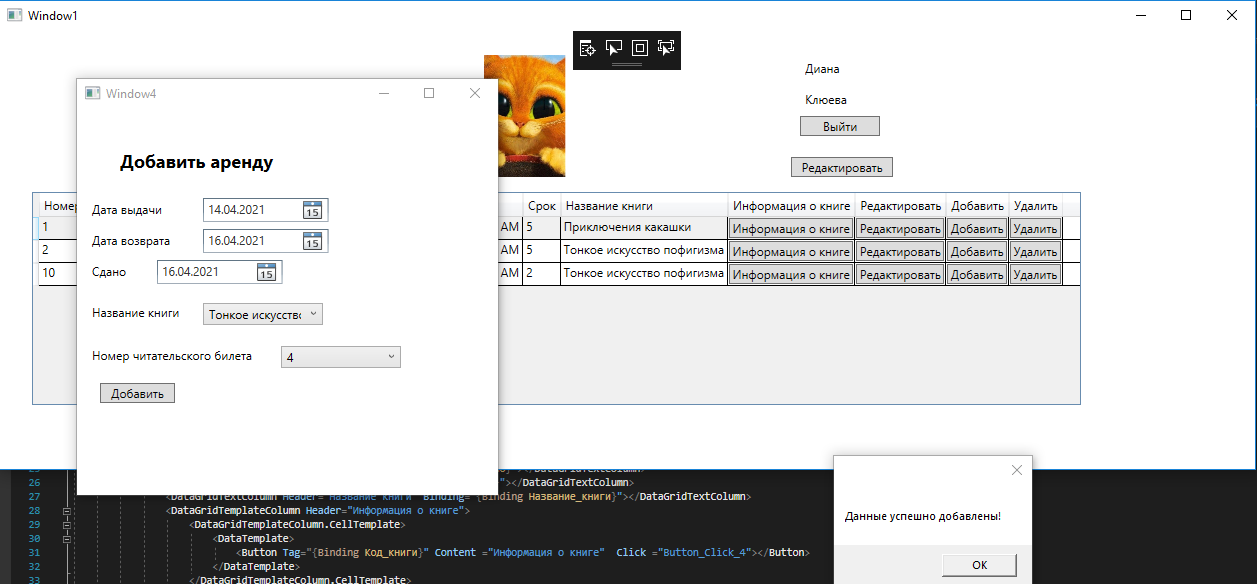


Рисунок 39. Добавление записи

Для добавления заполняем все поля и получаем сообщение об добавлении записи. На рисунке 39 можно видеть, что запись добавлена в базу.

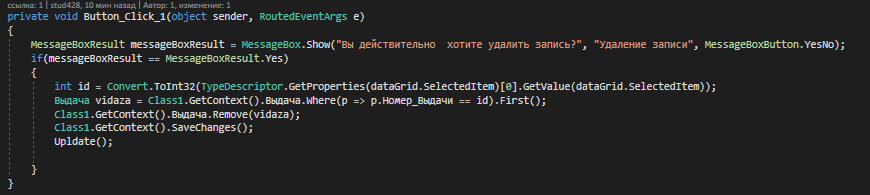


Рисунок 40. Удаление данных

Создаем событие Click в котором прописываем условие для удаления записи из базы данных. При удаление данных выводится сообщение пользователю «Вы действительно хотите удалить запись? и снова выводится сообщение для подтверждения удаления записи. После того как пользователь одобрит удаление записи она автоматом удаляется из программы и базы.

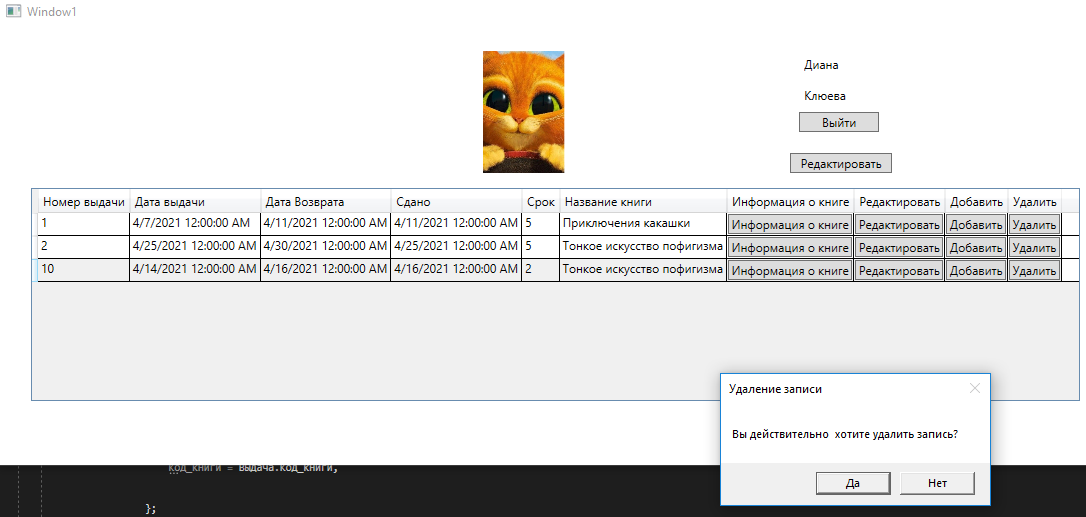


Рисунок 41. Удаление записи

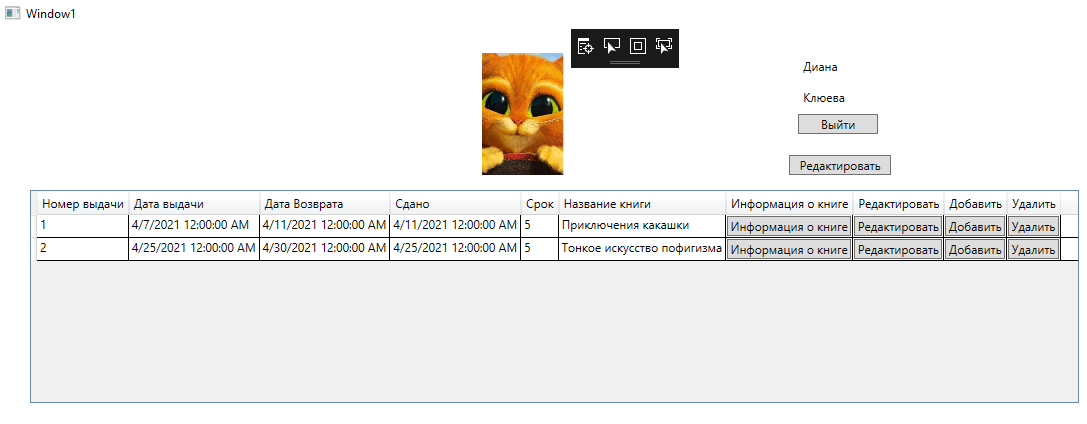


Рисунок 42. Запись удалена

Для того чтобы залить наш проект в гит

**Контрольные вопросы**

1. Опишите структуру команды "INSERT INTO"

Команда "INSERT INTO" (можно перевести с английского как "ВСТАВИТЬ В") вставляет данные в таблицу, название которой идёт после неё.

Insert into ‘name’ set name = ‘name1’

В этом случае это "name". Затем идёт слово "SET" (переводится как "ЗАДАТЬ"), после которого через запятую перечисляются названия столбцов в таблице и их значения, который надо вставить.

1. Опишите структуру команды "DELETE"

Команда DELETE – удаляет строки из таблицы

DELETE FROM name <имя таблицы>

Вывод: Для выполнения данной практической работы были изучены методические указания и предложенный курс. В данной практической работе мы приобрели навыки добавления данных в базу данных.