**Лабораторная работа №1**

**Тема**: Создание клиентского приложения, взаимодействующего с SQLite.

**Цель работы:** получение практических навыков при работе с ограничениями в SQLite и создании клиентского приложения.

**Краткие сведения из теории:**

В СУБД, которые представляют собой клиент-серверное приложение, есть выделенное приложение – сервер, которое принимает запросы от клиентов и выполняет их, оно непосредственно занимается записью/чтением данных на диск и предоставляет различные функции.

SQLite не использует парадигму *клиент-сервер.* Встраиваемая СУБД не имеет выделенного сервера, ее движок включается в приложение в виде библиотеки и предоставляет доступ к базе через специализированный API.

SQLite хранит всю базу данных (включая определения, таблицы, индексы и данные) в единственном стандартном файле на том компьютере, на котором исполняется *программа*.

Исходный код библиотеки SQLite общедоступен.

На сегодняшний день SQLite является одной из самых популярных решений такого типа. Она используется :

1. в музыкальных плеерах с функцией органайзера;
2. в мобильных телефонах и играх;
3. браузер Google Chrome использует SQLite для хранения Cookies;
4. в Skype(хранение истории сообщений);
5. в системах Airbus A350.

SQLite обладает феноменальной надежностью. Несмотря на то, что она является открытой, разработчики практически не принимают сообщения об ошибках.

Значение внутри БД может принадлежать к одному из следующих типов хранения (*storage class*):

1. NULL - пустое значение
2. INTEGER - целое число
3. REAL - дробное число
4. TEXT - строка
5. BLOB - для изображений и бинарных файлов

В SQLite по умолчанию отключена поддержка внешних ключей.

PRAGMA foreign\_keys=on;(включение поддержки внешних ключей )

**Задание 1.** Создать приложение, которое будет представлять собой форму для работы с базой данных, содержащей имена писателей и названия их произведений. Внешний вид представлен на рисунке ниже.

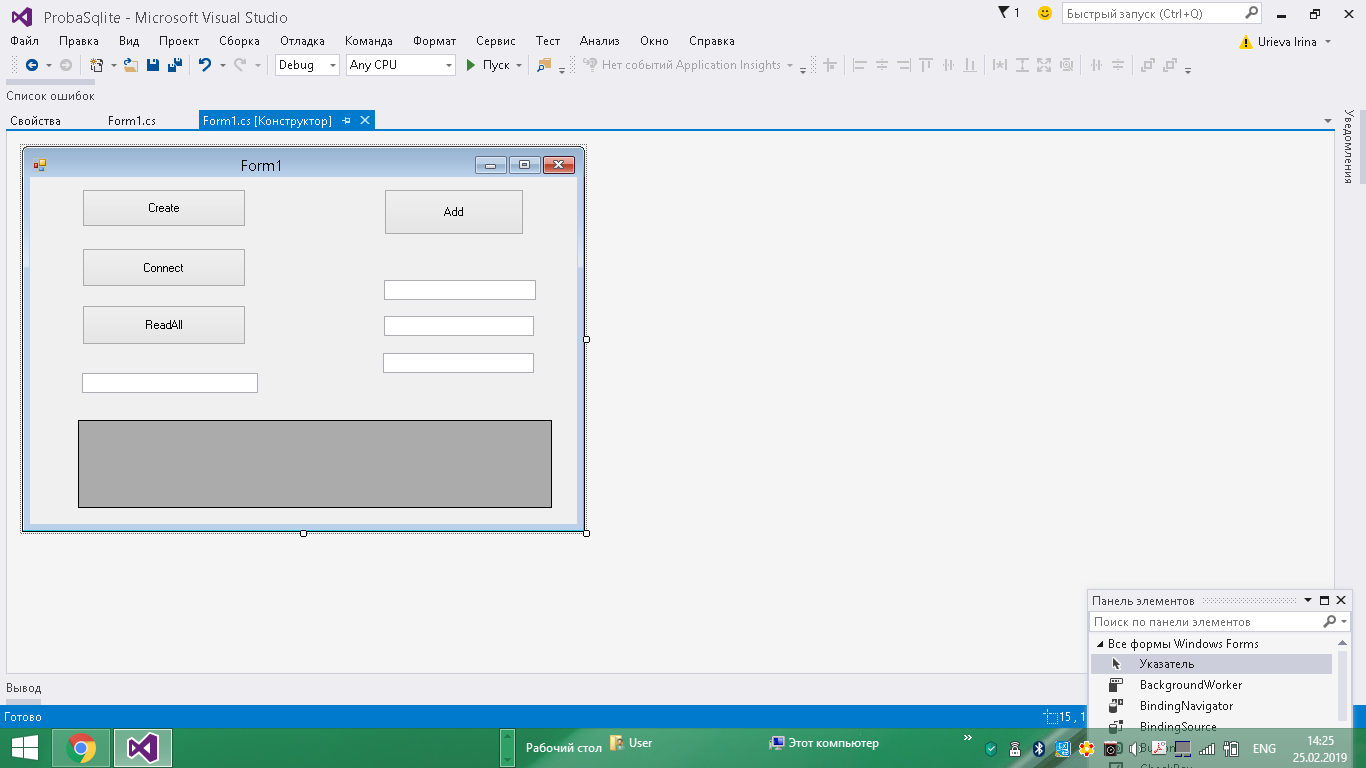


Рисунок 1 – Интерфейс приложения

На форме имеются несколько кнопок, поля ввода и dataGrid:

1. Create – создание БД и таблицы, если они ещё пока не созданы;
2. Connect – подключение к БД;
3. ReadAll – считать все записи из таблицы;
4. Add – добавить запись.

В отдельном поле отображается статус:

* Disconnected – нет связи с БД;
* Connected – есть связь с БД.

**Выполнение работы:**

1. Предварительно скачаем необходимые библиотеки и подключим в нашем проекте пространство имен System.Data.SQLite.Core или System.Data.SQLite(выбор библиотеки зависит от шаблона проекта), а также добавим переменные для связи с базой данных.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

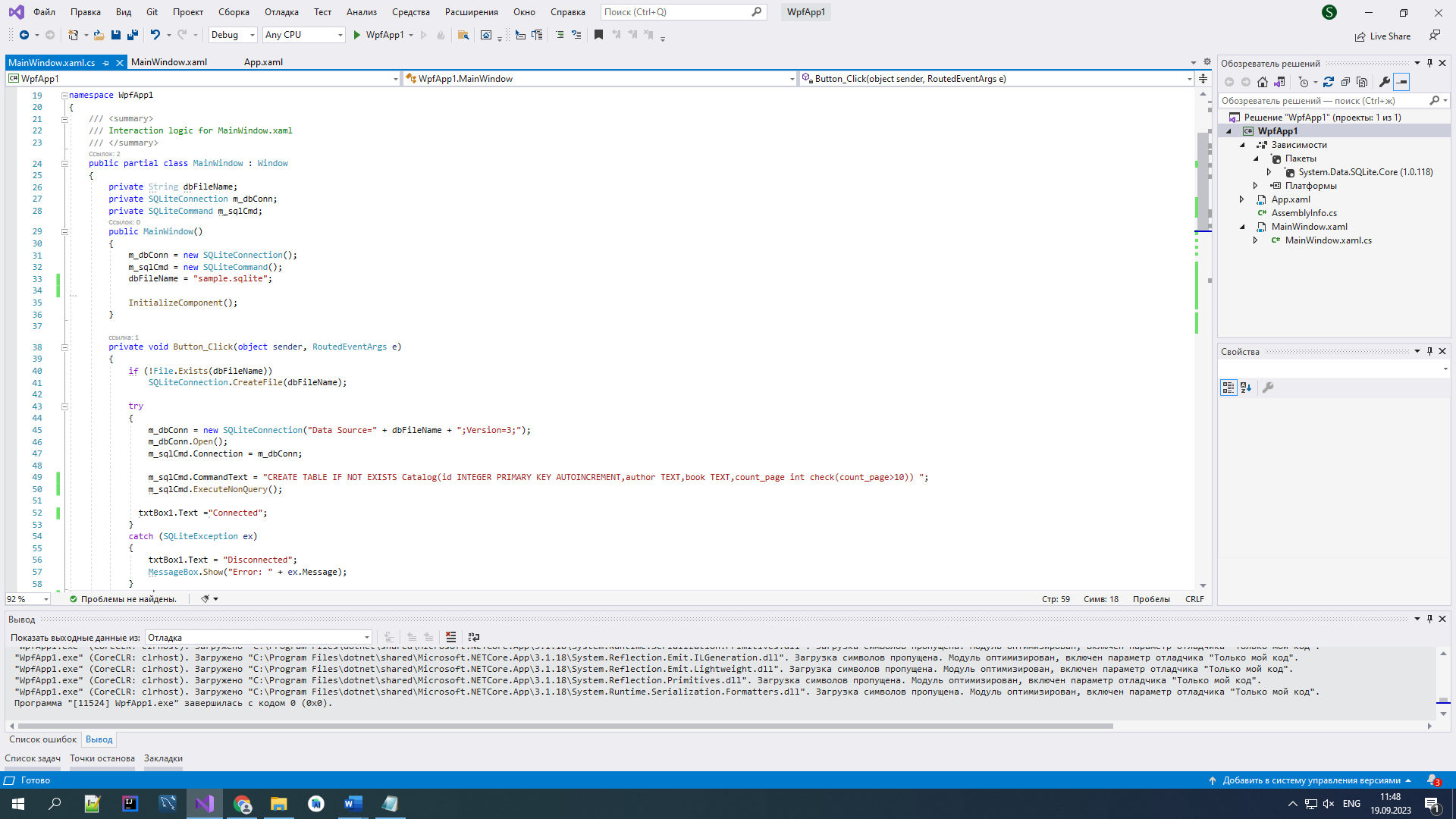
Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

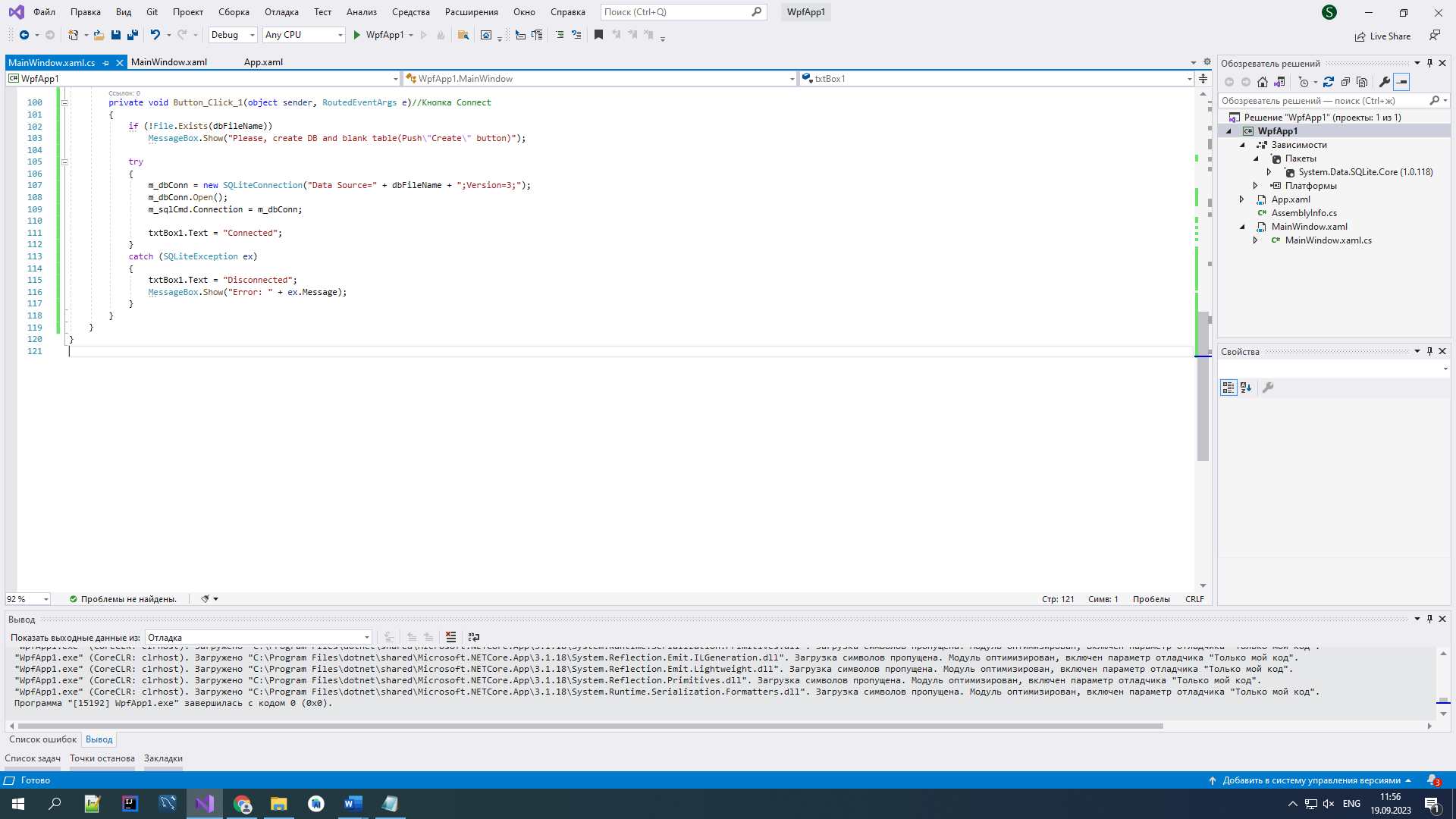
Рисунок 2 – Скачивание пакета

1. Создание БД и таблицы, если они ещё не созданы, осуществляется при нажатии на кнопку **Create.**

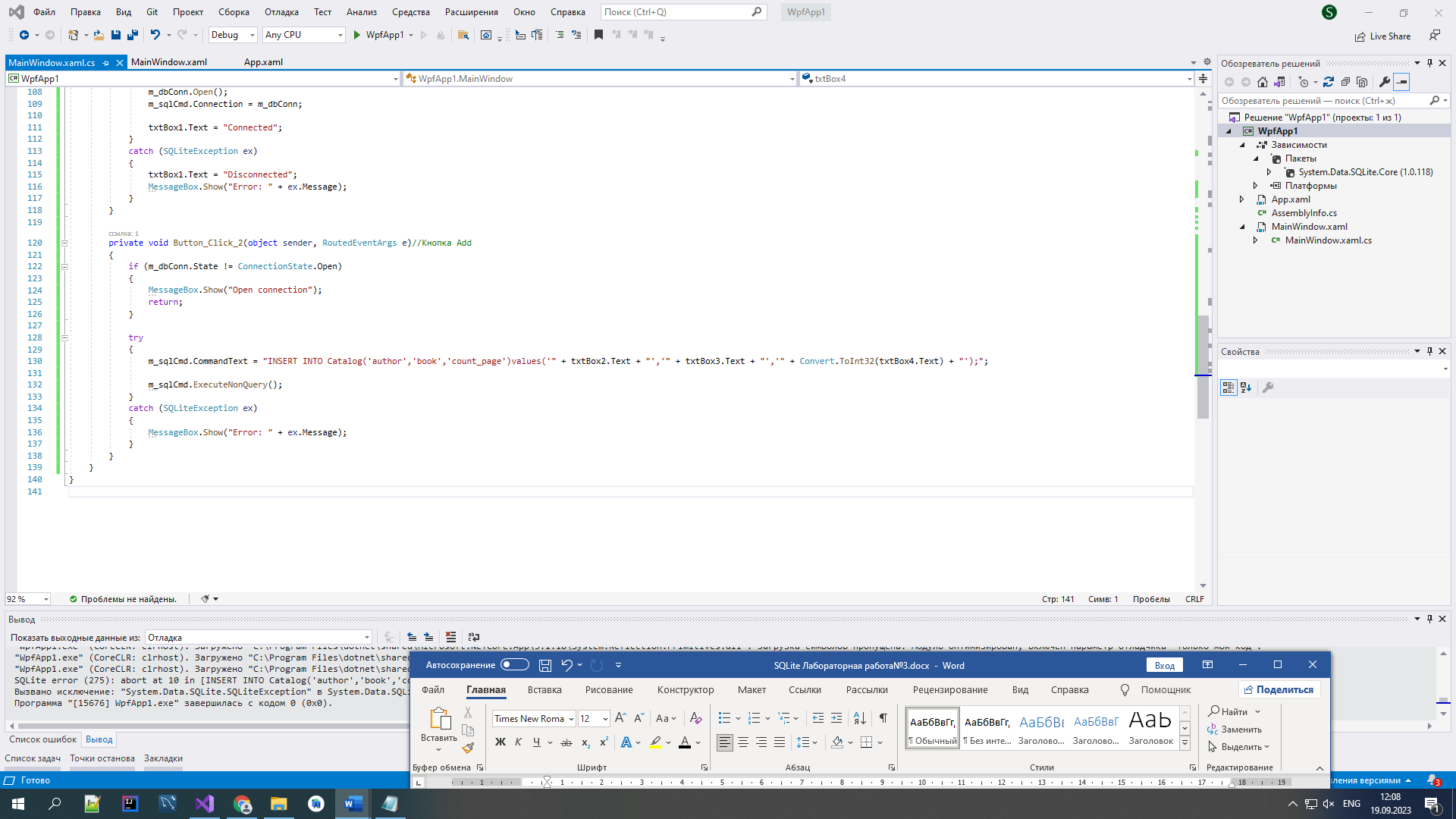


В данном обработчике, сначала проверяем, существует ли файл с именем “sample.sqlite” (переменная dbFileName), если его нет, то создаем файл. После этого подключаемся к созданной БД и создаем таблицу с необходимыми полями.

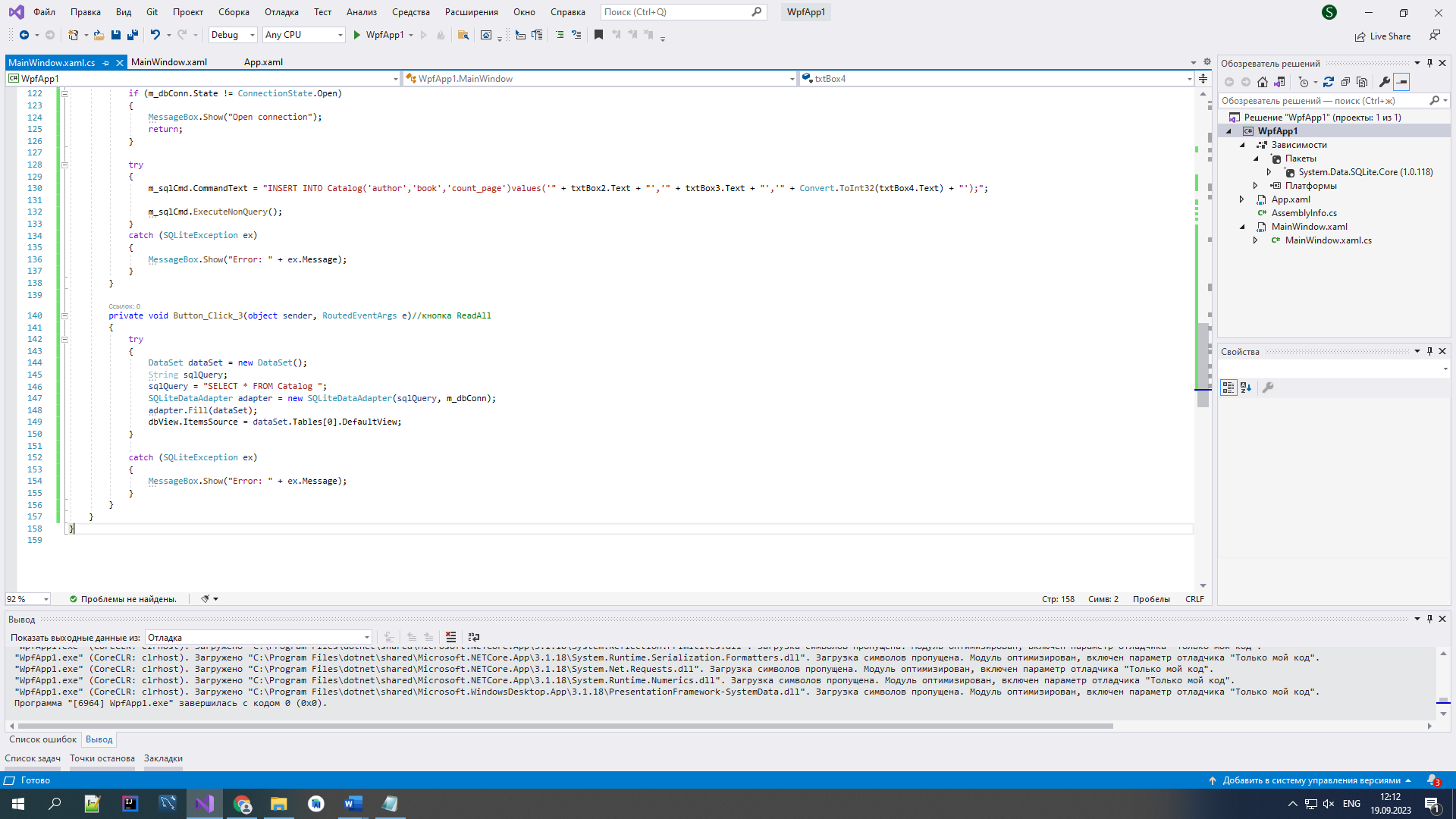
1. Подключение к уже существующей таблице осуществляется при нажатии на кнопку **Connect.**



1. Добавление данных в БД осуществляется при нажатии на кнопку **Add.**



1. Чтение данных осуществляется при нажатии на кнопку ReadAll, при этом из существующей БД считываются все данные и выводятся в таблицу.



1. Проверить корректную работу обработчиков событий(**Последовательность нажатий кнопок на форме:Create→ Connect→Add→ Read all).**

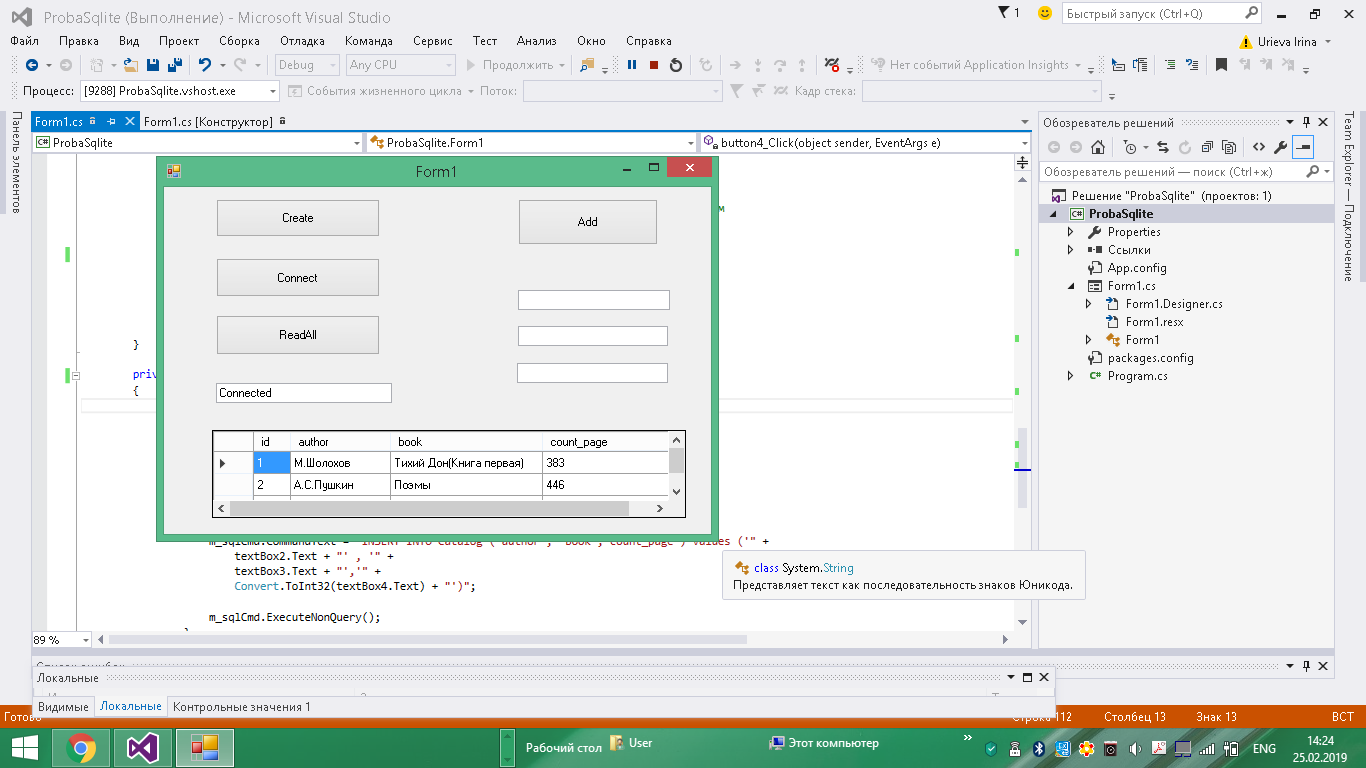


Рисунок 3 – Результат работы программы

**Задание 2.** Создать клиентское приложение для взаимодействия с SQLite в соответствии с выданным вариантом.

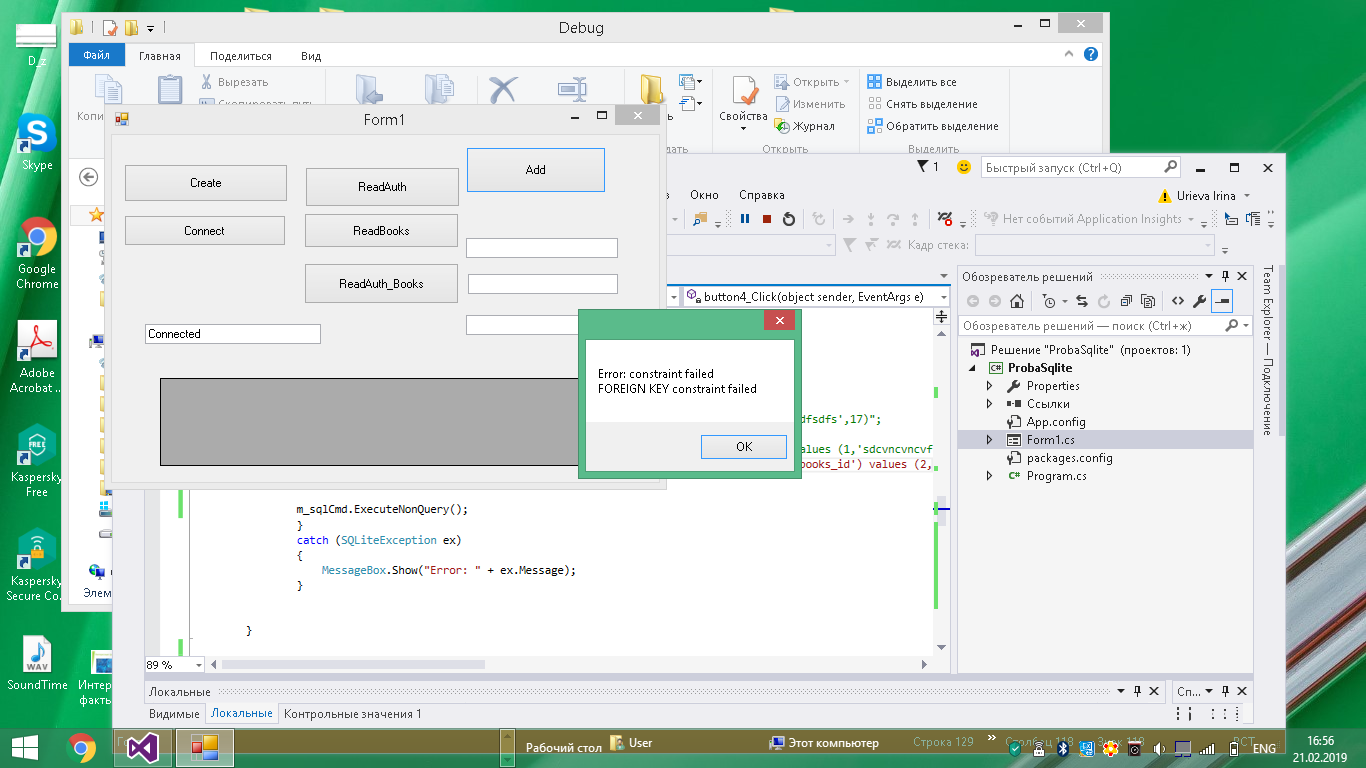
* 1. Создать и связать любые три таблицы из индивидуального задания.
  2. Показать, как работают следующие ограничения:

1) PRIMARY KEY

2) FOREIGN KEY

3) CHECK

Отразить это в отчете, т.е. зафиксировать вывод сообщения об ошибке.



Вывести информацию в отчет в следующем формате(представить соответствующие скриншоты):

1. Данные в таблице до добавления;
2. Добавленные данные;
3. Сообщение об ошибке.

**Задание 3.** Дополнительное задание на усмотрение преподавателя.