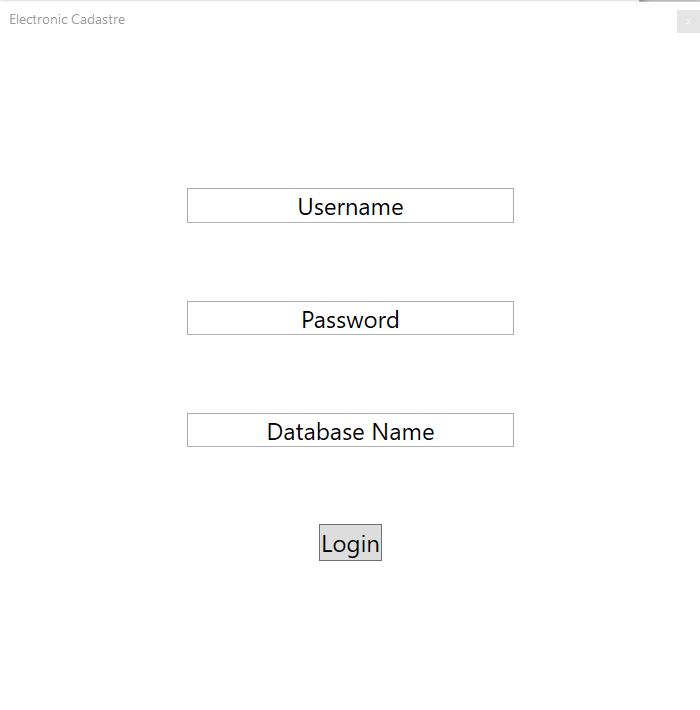
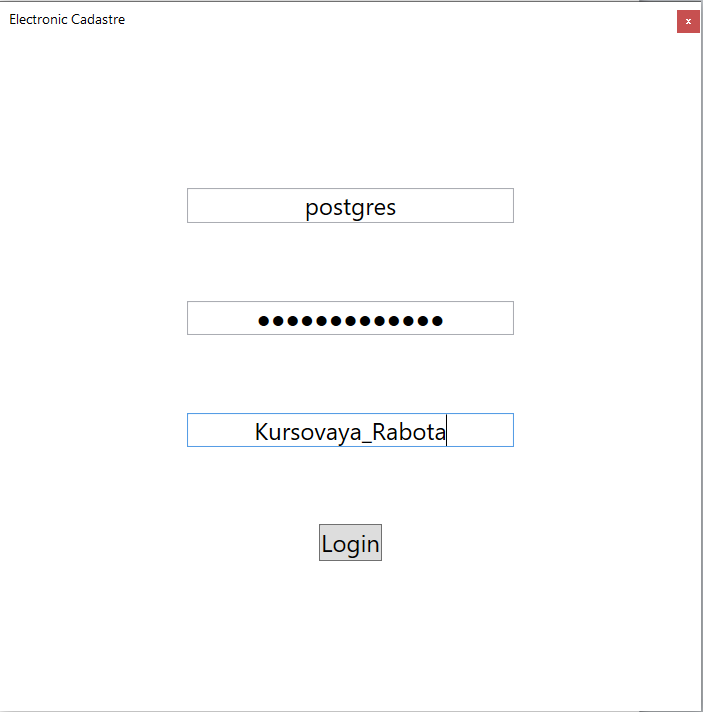
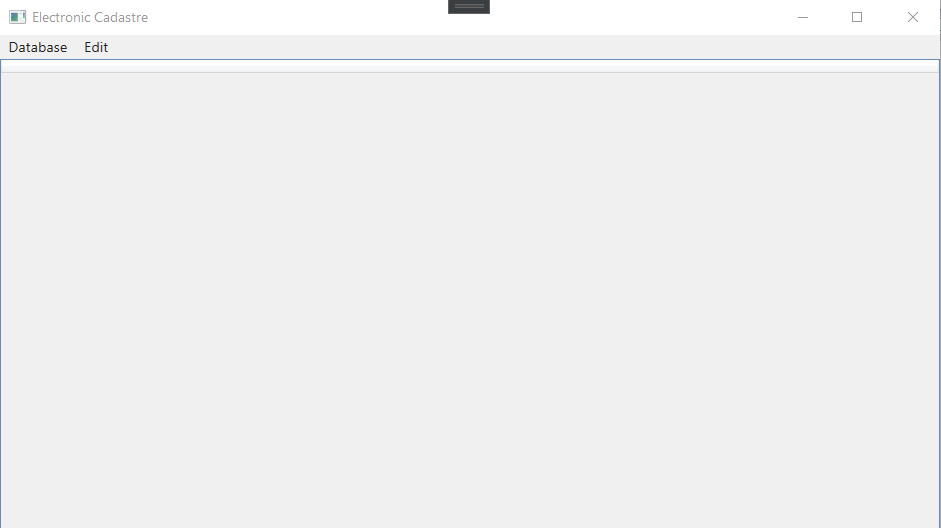
**GUI**

В окошке логина необходимо ввести логин, пароль и название базы данных:

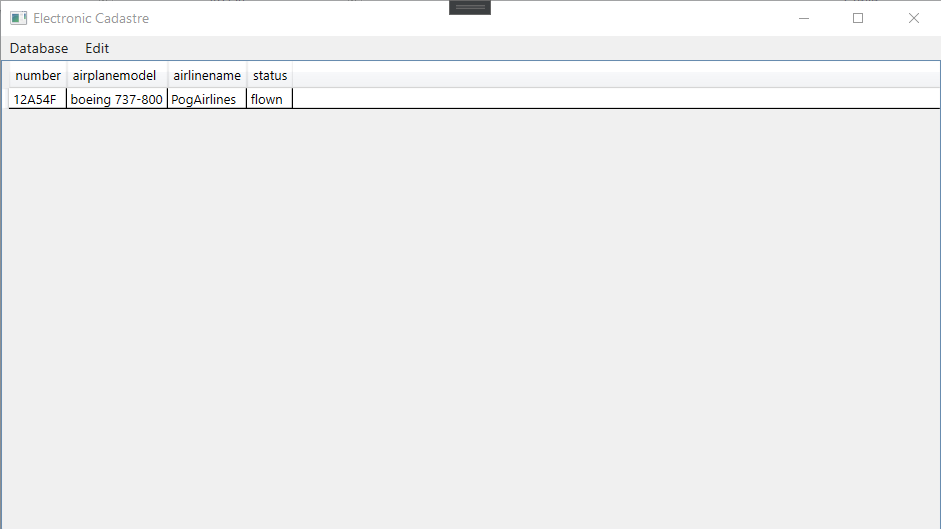




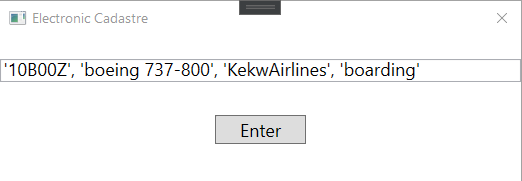
После логина откроется подобное окно:



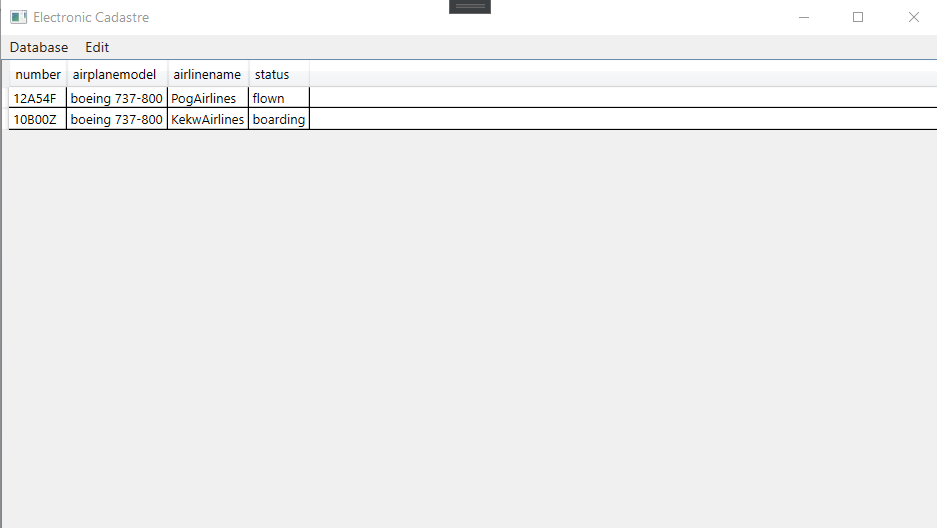
Чтобы взаимодействовать с таблицами, базы данных необходимо нажать Database -> Open, затем выбрать нужную таблицу и нажать Select.



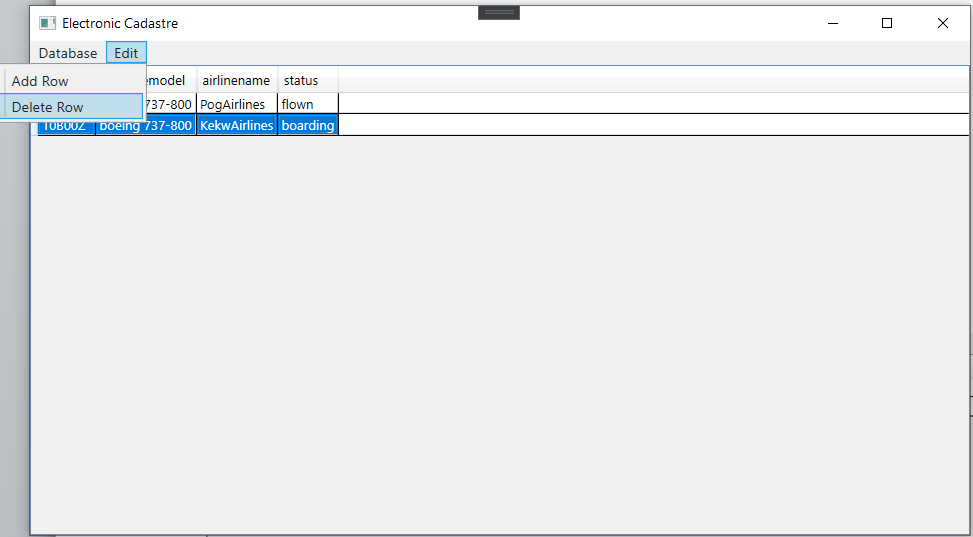
Чтобы добавить строку в таблицу, необходимо нажать Edit -> Add Row, ввести данные в строку подобным образом:



нажать Enter и строка добавлена:



Чтобы удалить строку, необходимо её выделить и нажать Edit -> Delete Row:



Чтобы закрыть таблицу, необходимо нажать Database -> Close.

Чтобы войти в другого пользователя, необходимо нажать Database -> Relog.

Чтобы выйти из приложения, необходимо нажать Database -> Exit.

**Code**

*DatabaseController.cs:* Подключение к базе данных создаётся в конструкторе *public DatabaseController* и открывается в методе *public string ConnectToDatabase*, который возвращает результат в виде строки. Также в этом классе присутствуют статический метод *public static DataTable ExecuteQuery* (который отвечает за выполнение SQL запросов) и *public void Disconnect* (который отвечает за отключение от базы данных).

*LoginWindow.xaml.cs:* Этот класс содержит в себе в основном косметические методов. При нажатии кнопки Login создаётся новый экземпляр класса *DatabaseController*, выполняется подключение через этот экземпляр и, при успешном подключении, открывается окно *DatabaseWindow*.

*DatabaseWindow.xaml.cs:* Этот класс также содержит кучу косметических методов, но уже гораздо больше логики, чем *LoginWindow*. В этом классе обрабатывается добавление, удаление и изменение таблиц. Метод *public bool SetDataGrid* изменяет отображение таблицы в окне. То есть он нужен для того, чтобы при изменении данных в базе данных, они изменялись и на экране. Метод возвращает true при успешном выполнении и false при ошибке. Метод *private string GenerateStringToIdentifyTheRow* отвечает за преобразование строки из *DataRowView* в *string* причем такой, что его можно сразу вставлять в запрос SQL. Классы *private void OpenTableMenuItem\_Click* и *private void AddRowMenuItem\_Click* открывают окна для выбора таблицы и для добавления в неё строк.

*RowAddingWindow.xaml.cs:* Этот класс нужен для добавления новой строки в таблицу в базе данных и обновления таблицы в окне *DatabaseWindow*.

*TablesSelectionWindow.xaml.cs:* Этот класс нужен для открытия таблицы в окне *DatabaseWindow*.