**НИТУ «МИСиС»**

Институт ИТАСУ

**Кафедра инженерной кибернетики**

Направление подготовки: 01.03.04 «прикладная математика»

Квалификация (степень): бакалавр

**КУРСОВАЯ РАБОТА**

**учебная дисциплина**

**«Специальные главы по базам данных»**

**VII семестр 2019 – 2020 у. г.**

**Студент: Новицкий Дмитрий**

**Группа: БПМ-16-2**

**Преподаватель: доц., к.т.н. А. С. Кожаринов**

**Оценка:**

**Дата:**

**Москва 2019**

Оглавление

[Список используемых основных сокращений 3](#_Toc28566101)

[Введение 4](#_Toc28566102)

[Постановка задачи 5](#_Toc28566103)

[Цель работы 5](#_Toc28566104)

[Описание работы приложения 5](#_Toc28566105)

[Использованные средства разработки и системные требования 7](#_Toc28566106)

[Краткое описание языковых и инструментальных средств, использованных для создания приложения 7](#_Toc28566107)

[Модель данных 8](#_Toc28566108)

[Описание таблиц базы данных 9](#_Toc28566109)

[Описание структуры таблиц 10](#_Toc28566110)

[Описание клиентского приложения 17](#_Toc28566111)

[Общие сведения 17](#_Toc28566112)

[Список классов 17](#_Toc28566113)

[Организация ролевого доступа к данным 23](#_Toc28566114)

[Тематические запросы к данным 25](#_Toc28566115)

[Выводы 27](#_Toc28566116)

[Список используемых источников 28](#_Toc28566117)

# Список используемых основных сокращений

1. ИС – информационная система.
2. АИС – автоматизированная информационная система.
3. БД – база данных.
4. СУБД – система управления базами данных.
5. ВПП – взлётно-посадочная полоса.
6. SQL – structured query language («язык структурированных запросов»)

# Введение

В настоящее время главное место в экономике организаций занимает развитие информационных технологий. Успех практически любого предприятия состоит в том, чтобы организация работала как слаженный механизм.

С появлением информационных технологий все большее количество процессов прошло автоматизацию. С каждым годом сильнее ощущается потребность в них – без хорошей, отлаженной техники, программного обеспечения, обученного к работе с ней персонала невозможно конкурировать на рынке. Одной из наиболее важных характеристик эффективности деятельности организации является наличие хорошей информационной технологии, позволяющей работать всем подразделениям предприятия достаточно быстро, правильно и без сбоев.

Инструментом, позволяющим получить информацию необходимого объема и качества, являются АИС различного назначения.

# Постановка задачи

Разрабатываемое приложение предназначено для предметной области «Аэропорт».

Рассматриваемая предметная область представляет собой ИС аэропорта, направленную на сбор и обработку информации для предоставления услуг авиаперевозок. ИС аэропорта связывает поставщика услуг – аэропорт и их потребителей – пассажиров. Для оптимального взаимодействия этих сторон необходимо автоматизировать информационные процессы авиаоператора, это приведет к улучшению качества и повышению быстродействия работы аэропорта. БД должна решать довольно узкий круг задач, таких как:

* Обслуживание клиентов, то есть:
  + Возможность покупки билетов на рейс
  + Возможность бронирования мест на рейс
  + Возможность отслеживания билетов и броней
  + Разработка комфортного для пользователя интерфейса
* Доступ к заданным данным и возможность их редактирования сотрудниками и администратором аэропорта.
* Разработка функционала для возможности регулирования деятельности аэропорта в зависимости от внешних факторов (погодных условий, человеческих факторов и другие).
* Изменение необходимой информации по мере её поступления в систему аэропорта.

## Цель работы

Цель работы – создать оптимальную модель предметной области, в качестве которой выступает система отношений между аэропортом, сотрудниками и клиентами аэропорта.

## Описание работы приложения

При запуске приложения запускается стартовое окно, которое позволит пользователю либо зарегистрироваться, либо войти в уже созданный аккаунт. После выполнения аутентификации пользователя запускается окно пользователя, в котором клиент сможет провести покупку/бронь билета на рейс, сможет отследить уже купленные билеты, узнать основную информацию о приложении и, при необходимости, изменить интерфейс клиентского приложения по своему желанию.

# Использованные средства разработки и системные требования

Исходя из того, что большинство сотрудников аэропорта работает за персональным компьютером или ноутбуком, было принято решение о создании клиентского приложения на персональный компьютер.

Были проанализированы данные, необходимые для грамотного функционирования приложения такие как, информация о рейсах, информация о пассажирах, информация о билетах, возможности взаимосвязи данных, в результате чего был сделан вывод о необходимости и удобстве систематизации данных, что привело к выбору реляционной СУБД.

## Краткое описание языковых и инструментальных средств, использованных для создания приложения

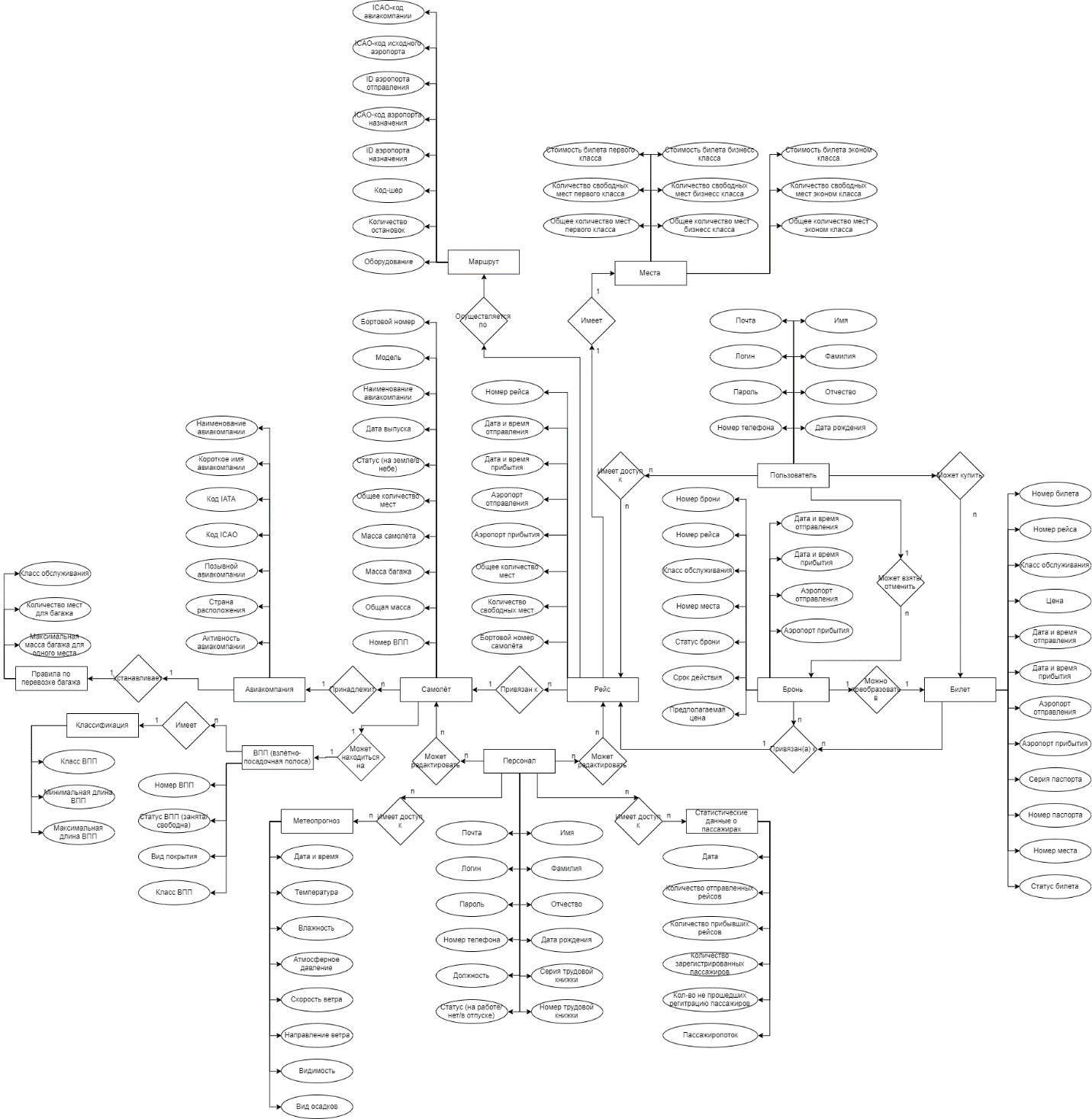
В качестве языка программирования базы данных был выбран объектно-ориентированный язык программирования C#. Среда программирования – Visual Studio 2019. База данных реализована с помощью использования элементов Visual C# Windows Forms.

Как было замечено ранее, в качестве СУБД была выбрана реляционная модель данных. По степени распределённости данная СУБД относится к классу локальной, по способу доступа к БД данная СУБД относится к классу клиент-серверная.

# Модель данных

Первым этапом реализации базы данных – это создание ER-модели базы данных. На рис. 1 изображена данная модель. Рассмотрим её более подробно. Для начала введём следующие обозначения:

* Множества сущностей изображаются в виде прямоугольников.
* Множества отношений изображаются в виде ромбов.
* Если сущности участвуют в отношениях, они будут связаны линией.
* Атрибуты изображаются в виде овалов и связываются линией с одной сущностью.



*Рис. 1. ER-модель базы данных.*

Всего на схеме изображено:

* 14 сущностей
* 16 отношений
* 109 атрибутов

Ключевыми сущностями в данной схеме являются «Пользователь», «Билет» и «Бронь», поскольку с помощью этих сущностей будет осуществляться взаимодействие клиента с системой.

## Описание таблиц базы данных

Каждая из представленных на рис. 1 сущностей – это отдельная таблица в базе данных. Рассмотрим каждую из них поподробнее в виде таблицы.

|  |  |  |
| --- | --- | --- |
| № | Название таблицы | Назначение |
| 1 | Авиакомпания | Необходима для хранения данных об авиакомпании. |
| 2 | Билет | Одна из ключевых таблиц в БД. Необходима для хранения данных о билетах, купленных пользователями |
| 3 | Бронь | Одна из ключевых таблиц в БД. Необходима для хранения данных о бронях, созданных пользователями |
| 4 | ВПП | Необходима для хранения информации о ВПП |
| 5 | Классификация ВПП | Необходима для хранения данных о классах ВПП |
| 6 | Маршрут | Необходима для хранения данных о маршрутах. |
| 7 | Места | Необходима для просмотра пользователем информации о наличии мест определённого класса на выбранный рейс и о стоимости мест |
| 8 | Метеопрогноз | Необходима для сохранения информации о погодных условиях. Данная таблица будет необходима для принятия решений об изменении функционирования работы аэропорта |
| 9 | Персонал | Необходима для хранения данных о персонале. В дальнейшем таблицы «Персонал» и «Пользователь» будут объединены |
| 10 | Пользователь | Одна из ключевых таблиц в БД. Необходима для хранения данных о пользователе |
| 11 | Правила по перевозке багажа | Необходима для хранения дополнительных данных о возможностях перевозки багажа пассажиром |
| 12 | Рейс | Необходима для хранения данных о рейсах |
| 13 | Самолёт | Необходима для хранения данных о технических характеристиках самолёта |
| 14 | Статистические данные о пассажирах | Необходима для сбора и систематизации данных о пассажирах для дальнейшего принятия решения системой об изменениях в системе работы аэропорта |

*Таблица 1. Общий список таблиц в БД.*

## Описание структуры таблиц

Рассмотрим краткое описание структуры каждой таблицы.

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | ID | INT | PK |
| 2 | Наименование | NVARCHAR | – |
| 3 | Короткое имя | NVARCHAR | – |
| 4 | Код IATA | NVARCHAR | – |
| 5 | Код ICAO | NVARCHAR | – |
| 6 | Позывной авиакомпании | NVARCHAR | – |
| 7 | Страна расположения | NVARCHAR | – |
| 8 | Активность авиакомпании | INT | – |
| 9 | ID правила по перевозке багажа | INT | FK |

*Таблица 2. Структура таблицы «Авиакомпания»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | Номер билета | INT | PK |
| 2 | Номер рейса | INT | – |
| 3 | Класс обслуживания | NVARCHAR | – |
| 4 | Цена | FLOAT | – |
| 5 | Дата и время отправления | DATETIME | – |
| 6 | Дата и время прибытия | DATETIME | – |
| 7 | Аэропорт отправления | NVARCHAR | – |
| 8 | Аэропорт прибытия | NVARCHAR | – |
| 9 | Серия паспорта | INT | – |
| 10 | Номер паспорта | INT | – |
| 11 | Номер места | INT | – |
| 12 | Статус билета | INT | FK |

*Таблица 3. Структура таблицы «Билет»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | Номер брони | INT | PK |
| 2 | Номер рейса | INT | – |
| 3 | Класс обслуживания | NVARCHAR | – |
| 4 | Цена | FLOAT | – |
| 5 | Дата и время отправления | DATETIME | – |
| 6 | Дата и время прибытия | DATETIME | – |
| 7 | Аэропорт отправления | NVARCHAR | – |
| 8 | Аэропорт прибытия | NVARCHAR | – |
| 9 | Номер места | INT | – |
| 10 | Статус брони | INT | FK |
| 11 | Срок действия | DATETIME | – |

*Таблица 4. Структура таблицы «Бронь»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | Номер | INT | PK |
| 2 | Статус | INT | – |
| 3 | Вид покрытия | NVARCHAR | – |
| 4 | Класс | INT | – |

*Таблица 5. Структура таблицы «ВПП»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | Класс | INT | PK |
| 2 | Минимальная длина | INT | – |
| 3 | Максимальная длина | INT | – |

*Таблица 6. Структура таблицы «Классификация ВПП»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | ID | INT | PK |
| 2 | ICAO-код авиакомпании | NVARCHAR | – |
| 3 | ICAO-код исходного аэропорта | NVARCHAR | – |
| 4 | ID аэропорта отправления | INT | – |
| 5 | ICAO-код аэропорта назначения | NVARCHAR | – |
| 6 | ID аэропорта назначения | INT | – |
| 7 | Код-шер | NVARCHAR | – |
| 8 | Количество остановок | INT | – |
| 9 | Оборудование | NVARCHAR | – |

*Таблица 7. Структура таблицы «Маршрут»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | ID | INT | PK |
| 2 | Общее количество мест первого класса | INT | – |
| 3 | Количество свободных мест первого класса | INT | – |
| 4 | Стоимость билета первого класса | FLOAT | – |
| 5 | Общее количество мест бизнес класса | INT | – |
| 6 | Количество свободных мест бизнес класса | INT | – |
| 7 | Стоимость билета бизнес класса | FLOAT | – |
| 8 | Общее количество мест эконом класса | INT | – |
| 9 | Количество свободных мест эконом класса | INT | – |
| 10 | Стоимость билета эконом класса | FLOAT | – |

*Таблица 8. Структура таблицы «Места»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | ID | INT | PK |
| 2 | Дата и время | DATETIME | – |
| 3 | Температура | FLOAT | – |
| 4 | Влажность | INT | – |
| 5 | Атмосферное давление | INT | – |
| 6 | Скорость ветра | FLOAT | – |
| 7 | Направление ветра | NVARCHAR | – |
| 8 | Видимость | INT | – |
| 9 | Вид осадков | NVARCHAR | – |

*Таблица 9. Структура таблицы «Метеопрогноз»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | ID | INT | PK |
| 2 | Почта | NVARCHAR | – |
| 3 | Логин | NVARCHAR | – |
| 4 | Пароль | NVARCHAR | – |
| 5 | Имя | NVARCHAR | – |
| 6 | Фамилия | NVARCHAR | – |
| 7 | Отчество | NVARCHAR | – |
| 8 | Дата рождения | DATE | – |
| 9 | Номер телефона | NVARCHAR | – |
| 10 | Роль | NVARCHAR | – |

*Таблица 10. Структура таблицы «Пользователь»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | ID | INT | PK |
| 2 | Класс обслуживания | NVARCHAR | – |
| 3 | Количество мест для багажа | INT | – |
| 4 | Максимальная масса багажа для одного места | INT | – |

*Таблица 11. Структура таблицы «Правила по перевозке багажа»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | Номер рейса | INT | PK |
| 2 | Дата и время отправления | DATETIME | – |
| 3 | Дата и время прибытия | DATETIME | – |
| 4 | Аэропорт отправления | NVARCHAR | – |
| 5 | Аэропорт прибытия | NVARCHAR | – |
| 6 | Общее количество мест | INT | – |
| 7 | Количество свободных мест | INT | – |
| 8 | ID самолёта | INT | FK |
| 9 | ID маршрута | INT | FK |
| 10 | ID таблицы мест | INT | FK |

*Таблица 12. Структура таблицы «Рейс»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | ID | INT | PK |
| 2 | Бортовой номер | NVARCHAR | – |
| 3 | Модель | NVARCHAR | – |
| 4 | ID авиакомпании | INT | – |
| 5 | Дата выпуска | DATE | – |
| 6 | Статус самолёта | INT | – |
| 7 | Общее количество мест | INT | – |
| 8 | Масса самолёта | FLOAT | – |
| 9 | Масса багажа | FLOAT | – |
| 10 | Общая масса | FLOAT | – |
| 11 | Номер ВПП | INT | – |

*Таблица 13. Структура таблицы «Самолёт»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название поля | Тип данных | Ключ (PK/FK) |
| 1 | ID | INT | PK |
| 2 | Дата | DATE | – |
| 3 | Количество отправленных рейсов | INT | – |
| 4 | Количество прибывших рейсов | INT | – |
| 5 | Количество зарегистрированных пассажиров | INT | – |
| 6 | Количество не прошедших регистрацию пассажиров | INT | – |
| 7 | Пассажиропоток | INT | – |

*Таблица 14. Структура таблицы «Статистические данные о пассажирах»*

# Описание клиентского приложения

## Общие сведения

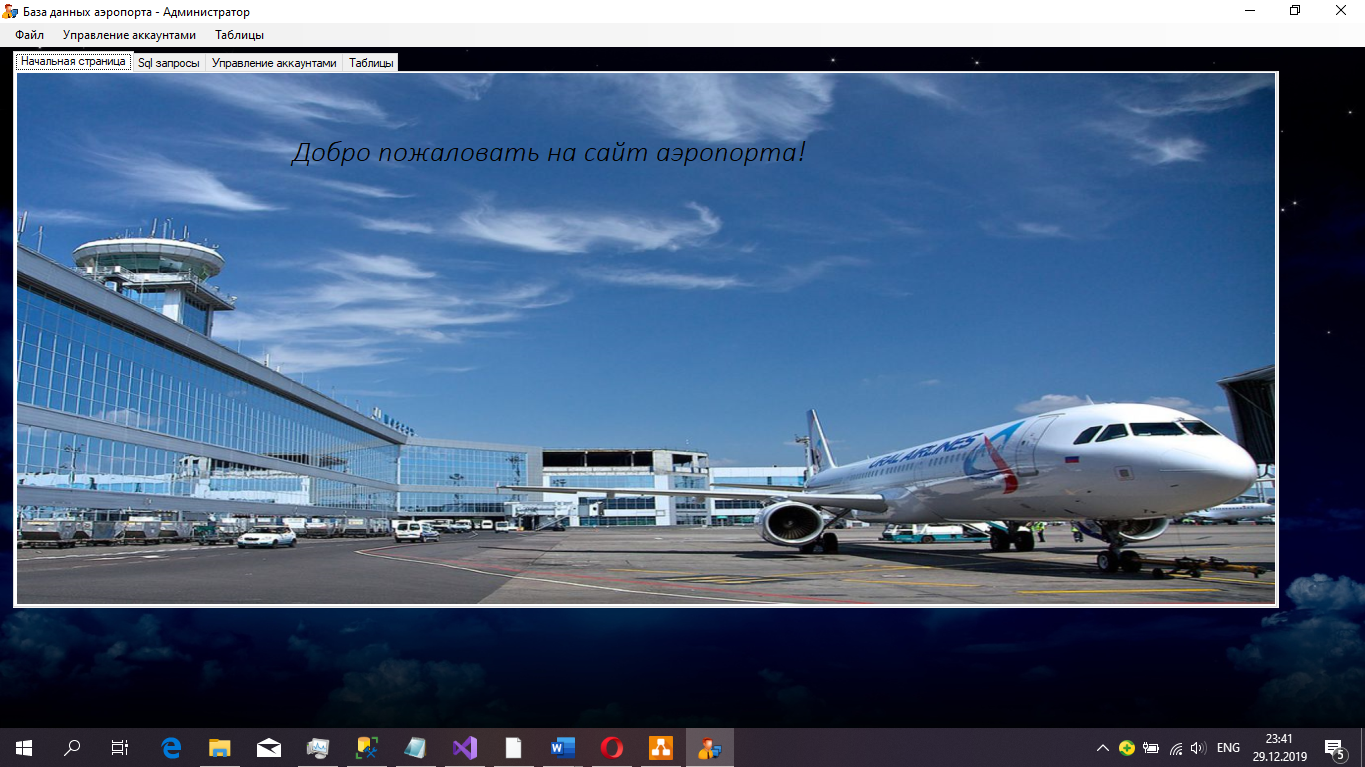
Данное клиентское приложение направлено на клиентов (пользователей), сотрудников и администраторов аэропорта.

Со стороны клиента приложение реализует минимальный функционал для продажи билетов на рейсы, а также предоставляет возможность бронирования мест на выбранный рейс. Сотрудники аэропорта имеют ограниченный доступ к данным, которые могут непосредственно влиять на работу аэропорта. Данные предоставляются для сотрудников, чтобы последние имели возможность своевременно реагировать на непредвиденные обстоятельства, либо на обстоятельства, которые могут негативным образом отразиться на работе аэропорта. Администраторы имеют возможность вносить изменения в любые таблицы, а также имеют возможность добавлять в базу данных пользователей с любыми правами («пользователь», «сотрудник», «администратор»).

## Список классов

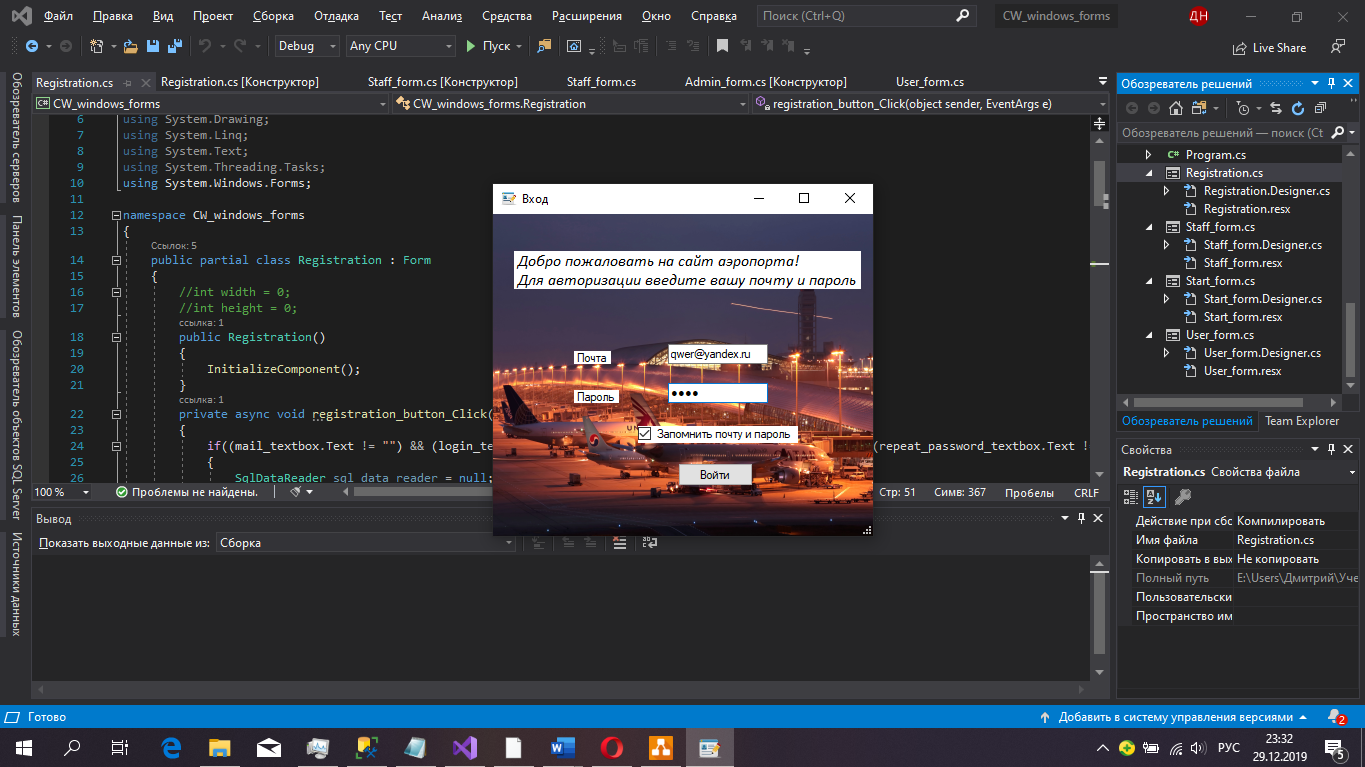
Каждый из представленных ниже классов представляет отдельную форму (окно) со своим функционалом и предназначением. Рассмотрим подробнее каждый из классов.

* **Admin form** (форма администратора). Данная форма активируется, если зайти в клиентское приложение от роли администратора. Данная форма предоставляет возможности для работы с каждой из таблиц в БД как с помощью SQL запросов, так и с помощью элемента управления DataGridView, позволяющего отображать данные в табличном формате, в настраиваемой сетке.



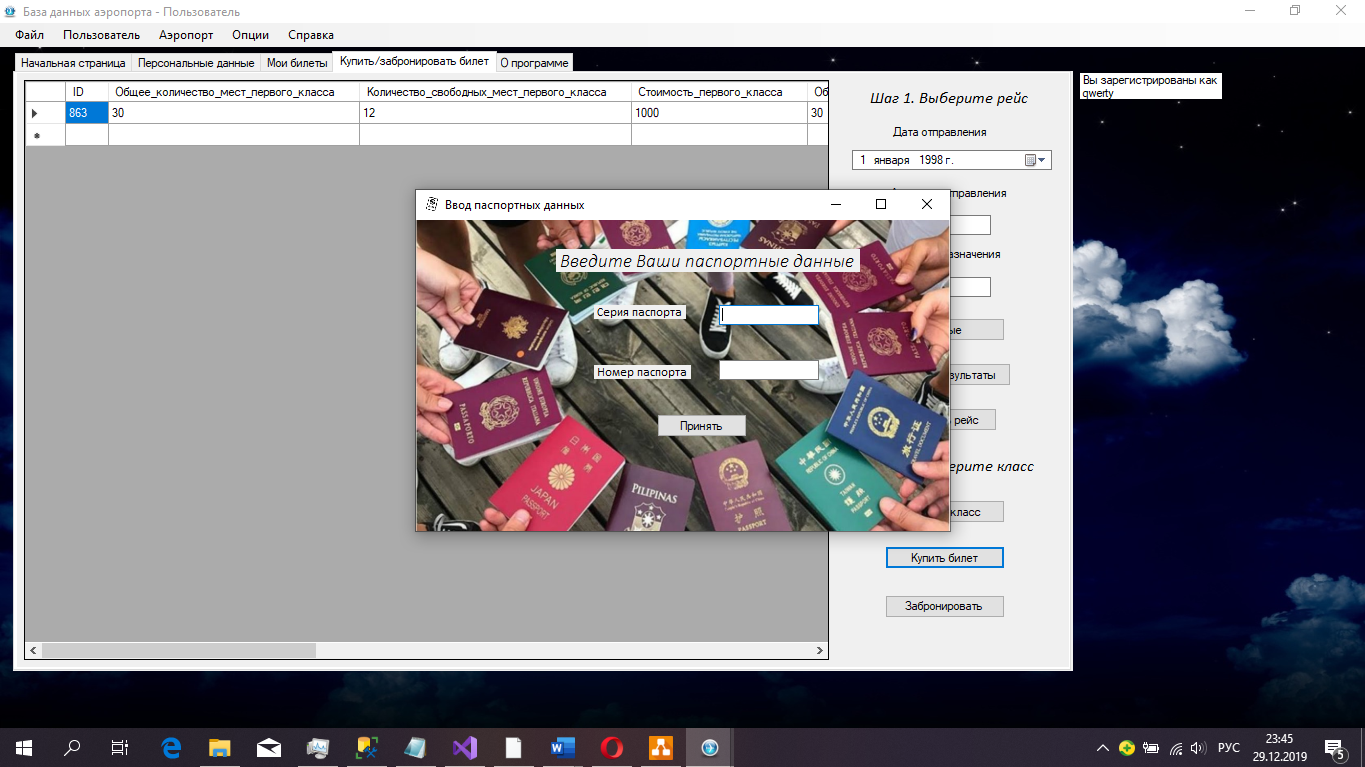
*Рис 2. Форма администратора.*

* **Authorization form** (форма авторизации). Данная форма активируется, если пользователь выберет пункт «Войти» на стартовой форме. Данная форма предоставляет необходимый функционал для авторизации пользователя в системе.



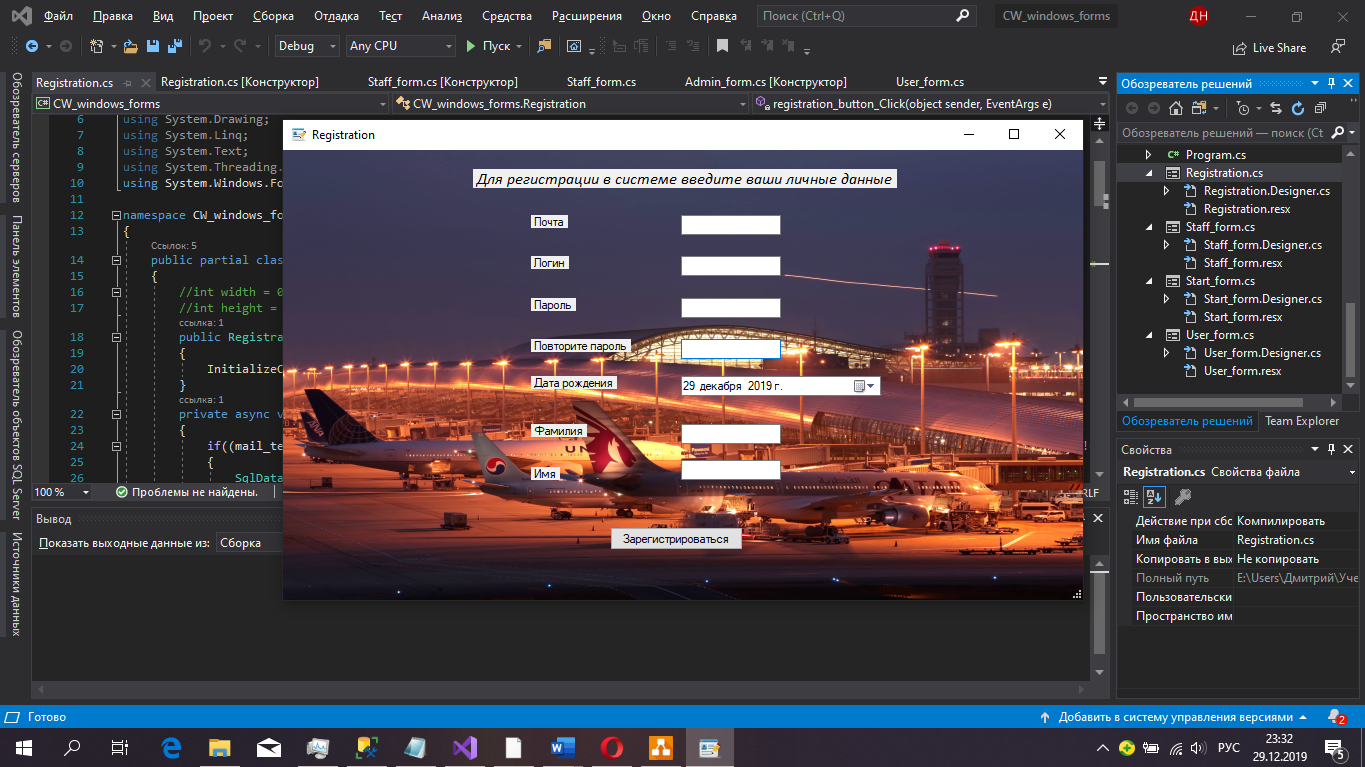
*Рис 3. Форма авторизации.*

* **Passport data form** (форма паспортных данных). Данная форма активируется, если пользователь в пользовательской форме выберет опцию покупки билета. В таком случае пользователю будет предложено ввести паспортные данные. Данная форма предоставляет пользователю возможность ввода паспортных данных для завершения процедуры покупки билета.



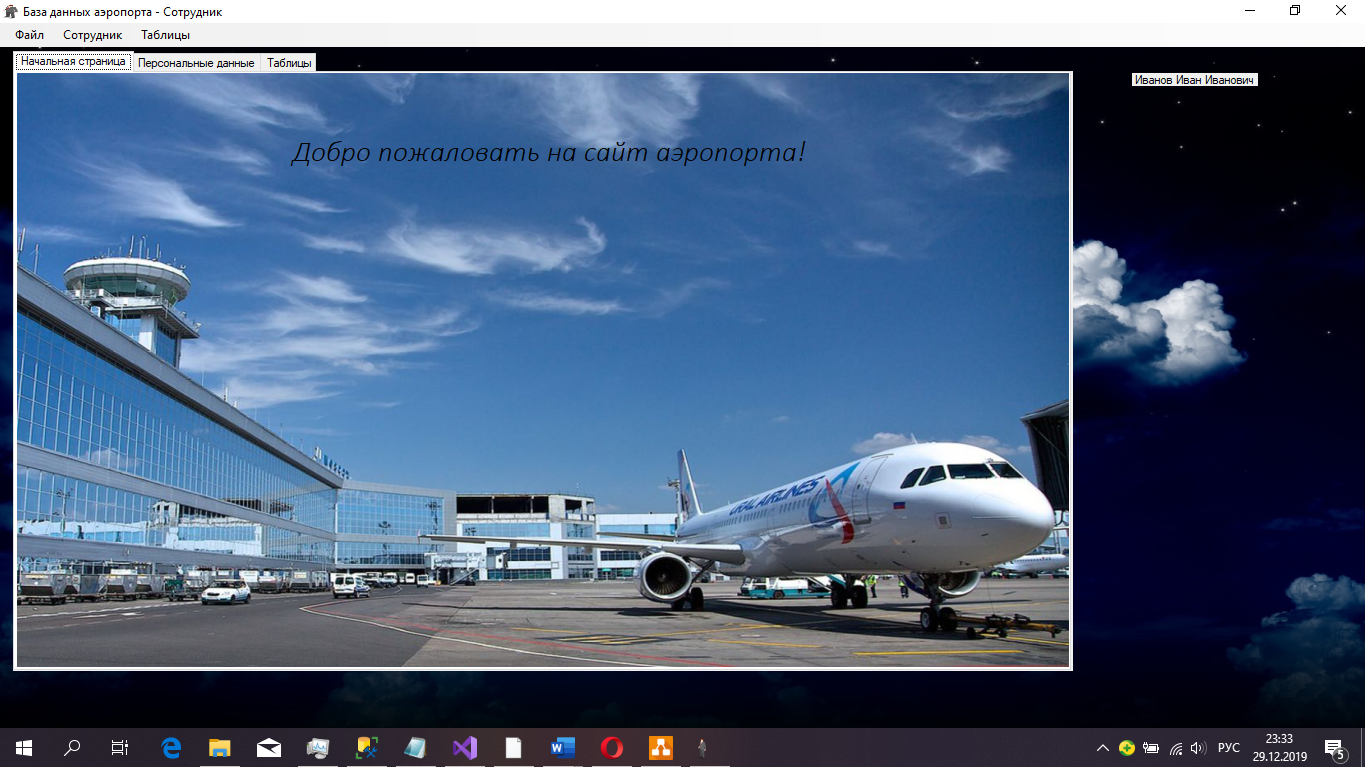
*Рис 4. Форма паспортных данных.*

* **Registration form** (форма регистрации). Данная форма активируется, если пользователь выберет пункт «Зарегистрироваться» на стартовой форме. Данная форма предоставляет необходимый функционал для регистрации пользователя. В данной форме можно зарегистрировать только пользователя с ролью «Пользователь».



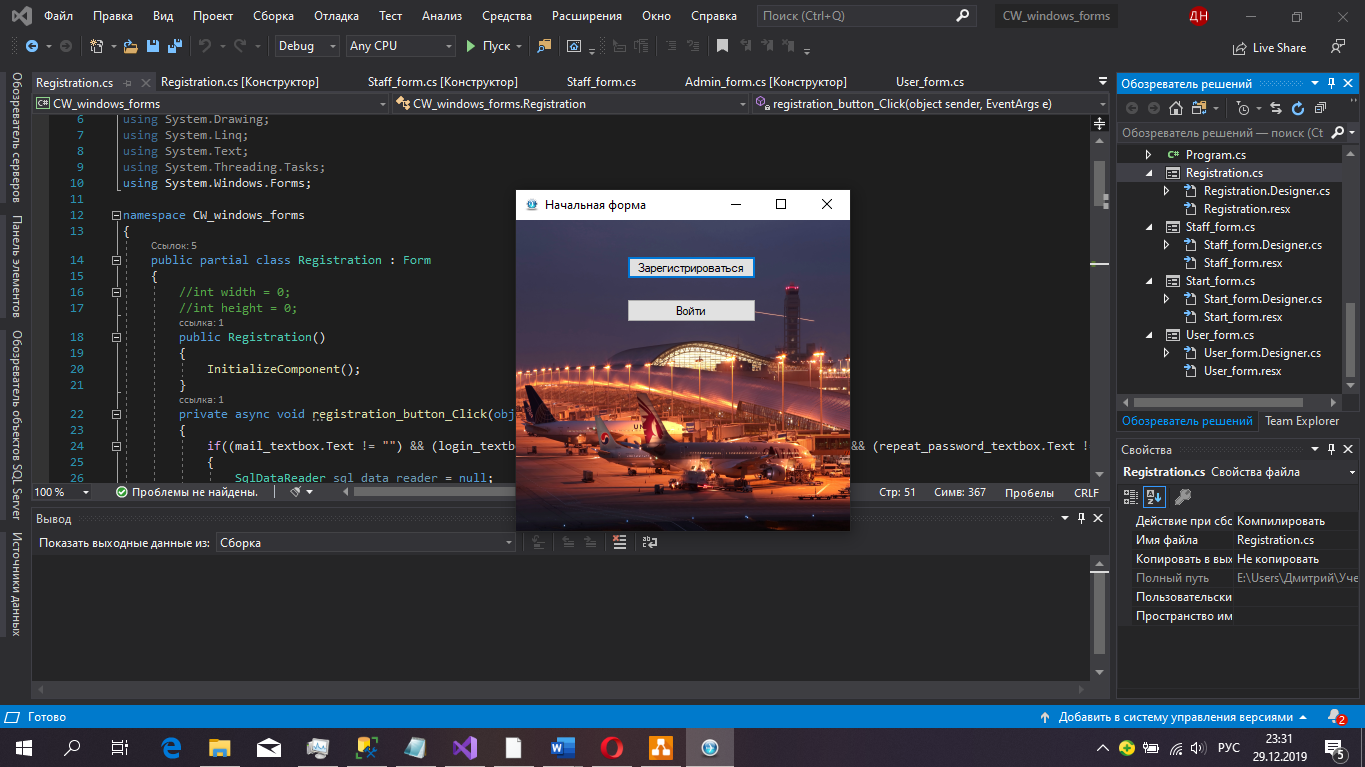
*Рис. 5. Форма регистрации.*

* **Staff form** (форма для персонала). Данная форма активируется, если зайти в клиентское приложение от роли персонала. Данная форма предоставляет возможности для работы персонала с ограниченным количеством таблиц для поддержания работы аэропорта в стабильном состоянии.



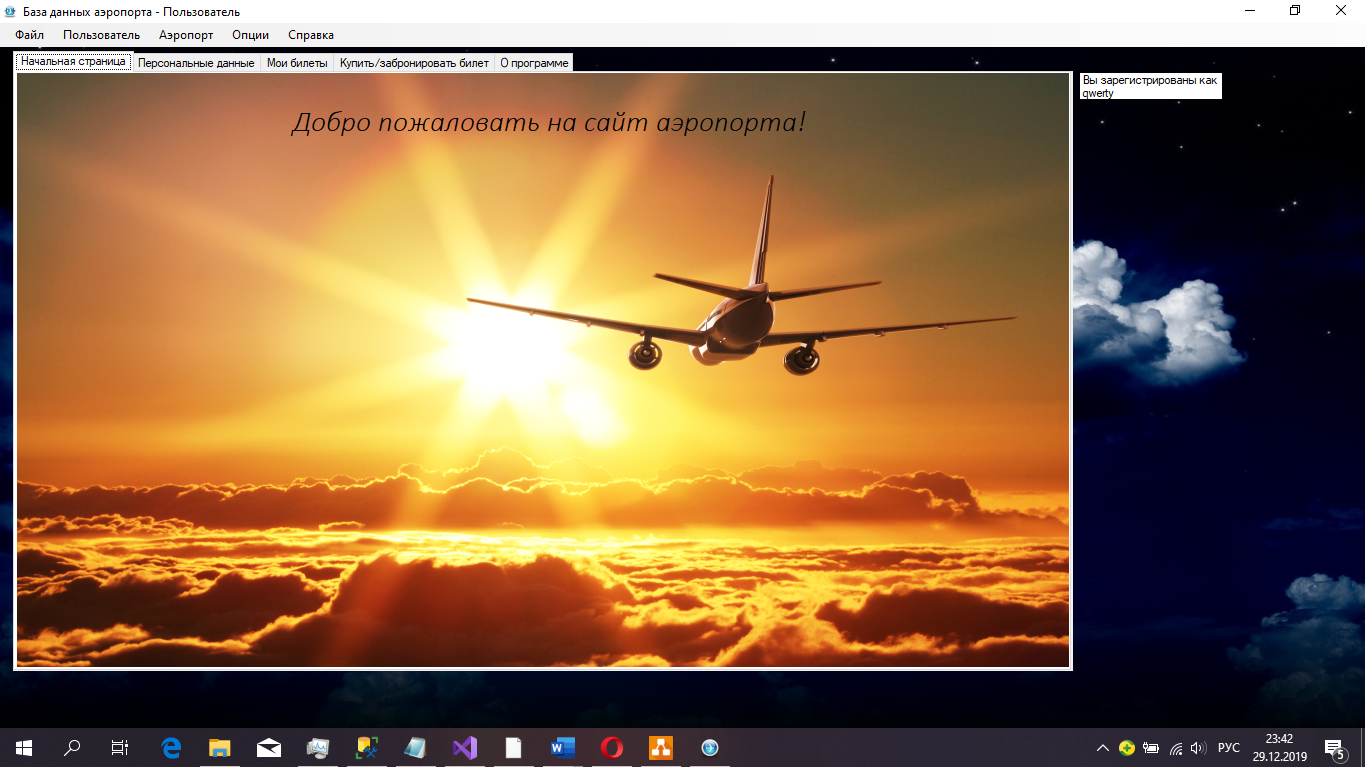
*Рис. 6. Форма для персонала.*

* **Start form** (начальная форма). Данная форма активируется при запуске приложения. Данная форма предоставляет пользователю возможность либо пройти авторизацию, либо пройти процесс регистрации.



*Рис. 7. Начальная форма.*

* **User form** (форма пользователя). Данная форма активируется, если зайти в клиентское приложение от роли пользователя. Данная форма предоставляет пользователю возможности для покупки и бронирования билета, просмотра уже имеющихся билетов и броней, а также возможности для изменения интерфейса приложения в соответствии с потребностями пользователя.



*Рис. 8. Форма пользователя.*

# Организация ролевого доступа к данным

В клиентском приложении предусмотрено выделение следующих трёх ролей:

* Пользователь
* Сотрудник
* Администратор

Рассмотрим назначение и систему привилегий, закреплённой за ролью более подробно.

Пользователь, после входа в свой аккаунт имеет возможность просматривать только свои билеты и брони из соответствующих таблиц. Кроме того, пользователь имеет возможность изменить некоторые данных своего аккаунта (фамилия, имя, отчество, логин, пароль, номер телефона), однако, для изменения пароля будет необходимо ввести старый пароль, чтобы увеличить уровень безопасности данных клиента. Также пользователь имеет возможность купить билет на рейс или забронировать место в соответствии с наличием свободных мест, то есть, если свободные места на рейс отсутствуют, то пользователь не сможет забронировать место или купить билет.

Сотрудник, после входа в свой аккаунт имеет возможность просматривать (но не изменять) следующие таблицы:

* Авиакомпания
* Активность авиакомпании
* ВПП
* Классификация ВПП
* Маршрут
* Места
* Метеопрогноз
* Правила по перевозке багажа
* Рейс
* Самолёт
* Статистические данные о пассажирах
* Статус билета
* Статус брони
* Статус самолёта

Отсутствие возможности редактировать данные обусловлено тем предположением, что база данных должна автоматически собирать необходимые данные (например, автоматически обновляться данные в таблице «Метеопрогноз» через подключение через интернет к одной из крупных баз данных погоды), а сотрудник должен анализировать данные и, только в случае необходимости направить администратору просьбу о необходимости отредактировать те или иные данные.

Администратор, после входа в свой аккаунт имеет возможность просматривать и редактировать данные в любой из таблиц БД. Администратор может работать с табличными данными как с помощью sql команд, так и с помощью упрощённого графического интерфейса для работы с данными.

# Тематические запросы к данным

Рассмотрим основные тематические запросы к данным, которые в большей степени отражают процесс функционирования БД.

Данный запрос необходим для вывода на экран билетов, которые закреплены за пользователем:

SELECT Билет.Номер\_билета, Билет.Номер\_рейса, Билет.Класс\_обслуживания, Билет.Цена, Билет.Дата\_время\_отправления, Билет.Дата\_время\_прибытия, Билет.Аэропорт\_отправления, Билет.Аэропорт\_прибытия, Билет.Номер\_места, Билет.Статус\_билета

FROM Билет, Билет\_Пользователь

WHERE Билет\_Пользователь.ID\_пользователя = 1 and Билет.Номер\_билета = Билет\_Пользователь.Номер\_билета

Следующий sql запрос реализуется для выбора подходящих для пользователя рейсов:

SELECT Номер\_рейса, Дата\_время\_отправления, Дата\_время\_прибытия, Аэропорт\_отправления, Аэропорт\_прибытия, Общее\_количество\_мест, Количество\_свободных\_мест

FROM Рейс

WHERE (Рейс.Аэропорт\_отправления = 'Москва') and (Рейс.Аэропорт\_прибытия = 'Париж') and

(DAY(Рейс.Дата\_время\_отправления) = 1) and (MONTH(Рейс.Дата\_время\_отправления) = 1) and (YEAR(Рейс.Дата\_время\_отправления) = 1998)

Нижеприведённая транзакция реализуется при покупке пользователем билета:

BEGIN TRANSACTION

INSERT INTO Билет (Номер\_рейса, Класс\_обслуживания, Цена, Дата\_время\_отправления, Дата\_время\_прибытия, Аэропорт\_отправления, Аэропорт\_прибытия, Серия\_паспорта, Номер\_паспорта, Номер\_места, Статус\_билета)

VALUES (1, 1, '', '', '', '', '', '', '', '', '')

-- Уменьшаем количество свободных мест в таблице с рейсами

UPDATE Рейс

SET Количество\_свободных\_мест = Количество\_свободных\_мест - 1

WHERE Номер\_рейса = 1

-- Уменьшаем количество свободных мест по классу в таблице с местами

UPDATE Места

SET Количество\_свободных\_мест\_первого\_класса = Количество\_свободных\_мест\_первого\_класса - 1

WHERE ID = (SELECT ID\_таблицы\_мест FROM Рейс WHERE Номер\_рейса = 1)

DECLARE @my\_ticket INT

SET @my\_ticket = (SELECT Билет.Номер\_билета FROM Билет WHERE (Номер\_рейса = 1 and Класс\_обслуживания = 'FIRST' and Номер\_места = 1))

INSERT INTO Билет\_Пользователь (Номер\_билета, ID\_пользователя)

VALUES (@my\_ticket, '')

COMMIT

# Выводы

В рамках данной курсовой работы был выполнен системный анализ предметной области и её изучение. Была разработана даталогическая и ER-модель базы данных. Исходя из данных моделей была разработана база данных, отражающая наиболее значимые факторы предметной области. Полученная база данных полностью удовлетворяет поставленным целям данной курсовой работы. Работа по реализации базы данных полностью выполнена.

# Список используемых источников

1. <https://otherreferats.allbest.ru/transport/00335587_0.html> – описание предметной области аэропорта.
2. <https://www.draw.io> – удобное приложение для разработки ER-диаграммы.
3. <https://icon-icons.com> – сайт, на котором можно найти тысячи разных иконок.

# Приложения

## Скрипт базы данных

USE [master]

GO

/\*\*\*\*\*\* Object: Database [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] Script Date: 30.12.2019 2:11:53 \*\*\*\*\*\*/

CREATE DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'AirportDB\_2', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS\MSSQL\DATA\Copy\_DB\AirportDB\_2.mdf' , SIZE = 8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )

LOG ON

( NAME = N'AirportDB\_2\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS\MSSQL\DATA\Copy\_DB\AirportDB\_2\_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET COMPATIBILITY\_LEVEL = 140

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF].[dbo].[sp\_fulltext\_database] @action = 'enable'

end

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET ANSI\_NULL\_DEFAULT OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET ANSI\_NULLS OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET ANSI\_PADDING OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET ANSI\_WARNINGS OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET ARITHABORT OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET AUTO\_CLOSE ON

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET AUTO\_SHRINK OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET AUTO\_UPDATE\_STATISTICS ON

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET CURSOR\_CLOSE\_ON\_COMMIT OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET CURSOR\_DEFAULT GLOBAL

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET CONCAT\_NULL\_YIELDS\_NULL OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET NUMERIC\_ROUNDABORT OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET QUOTED\_IDENTIFIER OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET RECURSIVE\_TRIGGERS OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET DISABLE\_BROKER

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET DATE\_CORRELATION\_OPTIMIZATION OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET ALLOW\_SNAPSHOT\_ISOLATION OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET READ\_COMMITTED\_SNAPSHOT OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET HONOR\_BROKER\_PRIORITY OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET RECOVERY SIMPLE

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET MULTI\_USER

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET PAGE\_VERIFY CHECKSUM

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET DB\_CHAINING OFF

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET TARGET\_RECOVERY\_TIME = 60 SECONDS

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET DELAYED\_DURABILITY = DISABLED

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET QUERY\_STORE = OFF

GO

USE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Авиакомпания] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Авиакомпания](

[ID] [int] NOT NULL,

[Наименование] [nvarchar](max) NOT NULL,

[Короткое\_имя] [nvarchar](max) NOT NULL,

[Код\_IATA] [nvarchar](max) NOT NULL,

[Код\_ICAO] [nvarchar](max) NOT NULL,

[Позывной] [nvarchar](max) NOT NULL,

[Страна\_расположения] [nvarchar](max) NOT NULL,

[Активность] [int] NOT NULL,

[ID\_правила\_по\_перевозке\_багажа] [int] NOT NULL,

CONSTRAINT [PK\_Авиакомпания] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Активность\_авиакомпании] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Активность\_авиакомпании](

[Активность] [int] NOT NULL,

[Значение] [nvarchar](max) NOT NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Билет] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Билет](

[Номер\_билета] [int] IDENTITY(1,1) NOT NULL,

[Номер\_рейса] [int] NOT NULL,

[Класс\_обслуживания] [nvarchar](max) NOT NULL,

[Цена] [float] NOT NULL,

[Дата\_время\_отправления] [datetime] NOT NULL,

[Дата\_время\_прибытия] [datetime] NOT NULL,

[Аэропорт\_отправления] [nvarchar](max) NOT NULL,

[Аэропорт\_прибытия] [nvarchar](max) NOT NULL,

[Серия\_паспорта] [int] NOT NULL,

[Номер\_паспорта] [int] NOT NULL,

[Номер\_места] [int] NOT NULL,

[Статус\_билета] [int] NOT NULL,

CONSTRAINT [PK\_Билет] PRIMARY KEY CLUSTERED

(

[Номер\_билета] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Билет\_Пользователь] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Билет\_Пользователь](

[Номер\_билета] [int] NOT NULL,

[ID\_пользователя] [int] NOT NULL,

CONSTRAINT [PK\_Билет\_Пользователь] PRIMARY KEY CLUSTERED

(

[Номер\_билета] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Бронь] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Бронь](

[Номер\_брони] [int] IDENTITY(1,1) NOT NULL,

[Номер\_рейса] [int] NOT NULL,

[Класс\_обслуживания] [nvarchar](max) NOT NULL,

[Цена] [float] NOT NULL,

[Дата\_время\_отправления] [datetime] NOT NULL,

[Дата\_время\_прибытия] [datetime] NOT NULL,

[Аэропорт\_отправления] [nvarchar](max) NOT NULL,

[Аэропорт\_прибытия] [nvarchar](max) NOT NULL,

[Номер\_места] [int] NOT NULL,

[Статус\_брони] [int] NOT NULL,

[Срок\_действия] [datetime] NULL,

CONSTRAINT [PK\_Бронь] PRIMARY KEY CLUSTERED

(

[Номер\_брони] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Бронь\_Пользователь] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Бронь\_Пользователь](

[Номер\_брони] [int] NOT NULL,

[ID\_пользователя] [int] NOT NULL,

CONSTRAINT [PK\_Бронь\_Пользователь] PRIMARY KEY CLUSTERED

(

[Номер\_брони] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[ВПП] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ВПП](

[Номер] [int] NOT NULL,

[Статус] [int] NOT NULL,

[Вид\_покрытия] [nvarchar](max) NOT NULL,

[Класс] [int] NOT NULL,

CONSTRAINT [PK\_ВПП] PRIMARY KEY CLUSTERED

(

[Номер] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Классификация\_ВПП] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Классификация\_ВПП](

[Класс] [int] NOT NULL,

[Минимальная\_длина] [int] NOT NULL,

[Максимальная\_длина] [int] NOT NULL,

CONSTRAINT [PK\_Классификация] PRIMARY KEY CLUSTERED

(

[Класс] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Маршрут] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Маршрут](

[ID] [int] NOT NULL,

[ICAO-код\_авиакомпании] [nvarchar](max) NOT NULL,

[ICAO-код\_исходного\_аэропорта] [nvarchar](max) NOT NULL,

[ID\_аэропорта\_отправления] [int] NOT NULL,

[ICAO-код\_аэропорта\_назначения] [nvarchar](max) NOT NULL,

[ID\_аэропорта\_назначения] [int] NOT NULL,

[Код-шер] [nvarchar](max) NOT NULL,

[Количество\_остановок] [int] NOT NULL,

[Оборудование] [nvarchar](max) NOT NULL,

CONSTRAINT [PK\_Маршрут] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Места] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Места](

[ID] [int] NOT NULL,

[Общее\_количество\_мест\_первого\_класса] [int] NOT NULL,

[Количество\_свободных\_мест\_первого\_класса] [int] NOT NULL,

[Стоимость\_первого\_класса] [float] NOT NULL,

[Общее\_количество\_мест\_бизнесс\_класса] [int] NOT NULL,

[Количество\_свободных\_мест\_бизнесс\_класса] [int] NOT NULL,

[Стоимость\_бизнесс\_класса] [float] NOT NULL,

[Общее\_количество\_мест\_эконом\_класса] [int] NOT NULL,

[Количество\_свободных\_мест\_эконом\_класса] [int] NOT NULL,

[Стоимость\_эконом\_класса] [float] NOT NULL,

CONSTRAINT [PK\_Места] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Метеопрогноз] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Метеопрогноз](

[ID] [int] NOT NULL,

[Дата\_время] [datetime] NOT NULL,

[Температура] [float] NOT NULL,

[Влажность] [int] NOT NULL,

[Атмосферное\_давление] [int] NOT NULL,

[Скорость\_ветра] [float] NOT NULL,

[Направление\_ветра] [nvarchar](max) NOT NULL,

[Видимость] [int] NOT NULL,

[Вид\_осадков] [nvarchar](max) NOT NULL,

CONSTRAINT [PK\_Метеопрогноз] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Пользователь] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Пользователь](

[ID] [int] IDENTITY(1,1) NOT NULL,

[Почта] [nvarchar](max) NOT NULL,

[Логин] [nvarchar](max) NOT NULL,

[Пароль] [nvarchar](max) NOT NULL,

[Имя] [nvarchar](max) NOT NULL,

[Фамилия] [nvarchar](max) NOT NULL,

[Отчество] [nvarchar](max) NULL,

[Дата\_рождения] [date] NOT NULL,

[Номер\_телефона] [nvarchar](max) NULL,

[Роль] [nvarchar](max) NOT NULL,

CONSTRAINT [PK\_Пользователь] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Правила\_по\_перевозке\_багажа] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Правила\_по\_перевозке\_багажа](

[ID] [int] NOT NULL,

[Класс\_обслуживания] [nvarchar](max) NOT NULL,

[Количество\_мест\_для\_багажа] [int] NOT NULL,

[Максимальная\_масса\_багажа\_для\_одного\_места] [int] NOT NULL,

CONSTRAINT [PK\_Правила\_по\_перевозке\_багажа] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Рейс] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Рейс](

[Номер\_рейса] [int] NOT NULL,

[Дата\_время\_отправления] [datetime] NOT NULL,

[Дата\_время\_прибытия] [datetime] NOT NULL,

[Аэропорт\_отправления] [nvarchar](max) NOT NULL,

[Аэропорт\_прибытия] [nvarchar](max) NOT NULL,

[Общее\_количество\_мест] [int] NOT NULL,

[Количество\_свободных\_мест] [int] NOT NULL,

[ID\_самолёта] [int] NOT NULL,

[ID\_маршрута] [int] NOT NULL,

[ID\_таблицы\_мест] [int] NOT NULL,

CONSTRAINT [PK\_Рейс] PRIMARY KEY CLUSTERED

(

[Номер\_рейса] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Самолёт] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Самолёт](

[ID] [int] NOT NULL,

[Бортовой\_номер] [nvarchar](max) NOT NULL,

[Модель] [nvarchar](max) NOT NULL,

[ID\_авиакомпании] [int] NOT NULL,

[Дата\_выпуска] [date] NOT NULL,

[Статус\_самолёта] [int] NOT NULL,

[Общее\_количество\_мест] [int] NOT NULL,

[Масса\_самолёта] [float] NOT NULL,

[Масса\_багажа] [float] NOT NULL,

[Общая\_масса] [float] NOT NULL,

[Номер\_ВПП] [int] NOT NULL,

CONSTRAINT [PK\_Самолёт] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Статистические\_данные\_о\_пассажирах] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Статистические\_данные\_о\_пассажирах](

[ID] [int] NOT NULL,

[Дата] [date] NOT NULL,

[Количество\_отправленных\_рейсов] [int] NOT NULL,

[Количество\_прибывших\_рейсов] [int] NOT NULL,

[Количество\_зарегистрированных\_пассажиров] [int] NOT NULL,

[Количество\_не\_прошедших\_регистрацию\_пассажиров] [int] NOT NULL,

[Пассажиропоток] [int] NOT NULL,

CONSTRAINT [PK\_Статистические\_данные\_о\_пассажирах] PRIMARY KEY CLUSTERED

(

[ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Статус\_билета] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Статус\_билета](

[Статус\_билета] [int] NOT NULL,

[Значение] [nvarchar](max) NOT NULL,

CONSTRAINT [PK\_Статус\_билета] PRIMARY KEY CLUSTERED

(

[Статус\_билета] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Статус\_брони] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Статус\_брони](

[Статус\_брони] [int] NOT NULL,

[Значение] [nvarchar](max) NOT NULL,

CONSTRAINT [PK\_Статус\_брони] PRIMARY KEY CLUSTERED

(

[Статус\_брони] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Статус\_самолёта] Script Date: 30.12.2019 2:11:54 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Статус\_самолёта](

[Статус\_самолёта] [int] NOT NULL,

[Значение] [nvarchar](max) NOT NULL,

CONSTRAINT [PK\_Статус\_самолёта] PRIMARY KEY CLUSTERED

(

[Статус\_самолёта] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[Билет] WITH CHECK ADD CONSTRAINT [FK\_Билет\_Билет\_Пользователь] FOREIGN KEY([Номер\_билета])

REFERENCES [dbo].[Билет\_Пользователь] ([Номер\_билета])

GO

ALTER TABLE [dbo].[Билет] CHECK CONSTRAINT [FK\_Билет\_Билет\_Пользователь]

GO

ALTER TABLE [dbo].[Билет] WITH CHECK ADD CONSTRAINT [FK\_Билет\_Рейс] FOREIGN KEY([Номер\_рейса])

REFERENCES [dbo].[Рейс] ([Номер\_рейса])

GO

ALTER TABLE [dbo].[Билет] CHECK CONSTRAINT [FK\_Билет\_Рейс]

GO

ALTER TABLE [dbo].[Билет] WITH CHECK ADD CONSTRAINT [FK\_Билет\_Статус\_билета] FOREIGN KEY([Статус\_билета])

REFERENCES [dbo].[Статус\_билета] ([Статус\_билета])

GO

ALTER TABLE [dbo].[Билет] CHECK CONSTRAINT [FK\_Билет\_Статус\_билета]

GO

ALTER TABLE [dbo].[Билет\_Пользователь] WITH CHECK ADD CONSTRAINT [FK\_Билет\_Пользователь\_Пользователь] FOREIGN KEY([ID\_пользователя])

REFERENCES [dbo].[Пользователь] ([ID])

GO

ALTER TABLE [dbo].[Билет\_Пользователь] CHECK CONSTRAINT [FK\_Билет\_Пользователь\_Пользователь]

GO

ALTER TABLE [dbo].[Бронь] WITH CHECK ADD CONSTRAINT [FK\_Бронь\_Бронь\_Пользователь] FOREIGN KEY([Номер\_брони])

REFERENCES [dbo].[Бронь\_Пользователь] ([Номер\_брони])

GO

ALTER TABLE [dbo].[Бронь] CHECK CONSTRAINT [FK\_Бронь\_Бронь\_Пользователь]

GO

ALTER TABLE [dbo].[Бронь] WITH CHECK ADD CONSTRAINT [FK\_Бронь\_Рейс] FOREIGN KEY([Номер\_рейса])

REFERENCES [dbo].[Рейс] ([Номер\_рейса])

GO

ALTER TABLE [dbo].[Бронь] CHECK CONSTRAINT [FK\_Бронь\_Рейс]

GO

ALTER TABLE [dbo].[Бронь] WITH CHECK ADD CONSTRAINT [FK\_Бронь\_Статус\_брони] FOREIGN KEY([Статус\_брони])

REFERENCES [dbo].[Статус\_брони] ([Статус\_брони])

GO

ALTER TABLE [dbo].[Бронь] CHECK CONSTRAINT [FK\_Бронь\_Статус\_брони]

GO

ALTER TABLE [dbo].[Бронь\_Пользователь] WITH CHECK ADD CONSTRAINT [FK\_Бронь\_Пользователь\_Пользователь] FOREIGN KEY([ID\_пользователя])

REFERENCES [dbo].[Пользователь] ([ID])

GO

ALTER TABLE [dbo].[Бронь\_Пользователь] CHECK CONSTRAINT [FK\_Бронь\_Пользователь\_Пользователь]

GO

USE [master]

GO

ALTER DATABASE [C:\PROGRAM FILES\MICROSOFT SQL SERVER\MSSQL14.SQLEXPRESS\MSSQL\DATA\COPY\_DB\AIRPORTDB\_2.MDF] SET READ\_WRITE

GO

## Пример программного кода клиентского приложения

В качестве примера программного кода клиентского приложения приведена функция, реализующая первую стадию покупки билета на рейс пользователем.

private void buy\_ticket\_button\_Click(object sender, EventArgs e)

{

try

{

if (value\_table != "Места")

{

MessageBox.Show("Сначала выберите рейс", "Уведомление", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else if (choose\_class\_button.Text == "Выберите класс")

{

MessageBox.Show("Сначала выберите класс", "Предупреждение", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

int selected\_row\_index = airport\_dataGridView.CurrentCell.RowIndex;

if (((choose\_class\_button.Text == "Первый класс") && (Convert.ToInt32(airport\_dataGridView[2, selected\_row\_index].Value) == 0)) ||

((choose\_class\_button.Text == "Бизнесс класс") && (Convert.ToInt32(airport\_dataGridView[5, selected\_row\_index].Value) == 0)) ||

((choose\_class\_button.Text == "Эконом класс") && (Convert.ToInt32(airport\_dataGridView[8, selected\_row\_index].Value) == 0)))

{

MessageBox.Show("Свободных мест нет", "Уведомление", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

{

My\_values.passport\_form = new Passport\_data\_form();

My\_values.passport\_form.Show();

My\_values.passport\_form.Activate();

// Поставить дальнейшее выполнение программы на ожидание

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString(), ex.Source.ToString(), MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}