|  |  |  |
| --- | --- | --- |
| Департамент образования Ярославской области  Государственное профессиональное образовательное автономное учреждение  Ярославской области  «ЯРОСЛАВСКИЙ ПРОМЫШЛЕННО-ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ  им. Н.П. ПАСТУХОВА» | | |
| **Отчет По учебной практике**  **по профессиональному модулю ПМ.07**  **Соадминистрирование и автоматизация баз данных и серверов** | | |
| ОУП ПМ.05.09.02.07.18ИП1.13 | | |
|  | | |
|  | | |
|  | | |
|  | | |
|  | Студент  \_\_\_\_\_\_\_\_\_\_\_\_\_ А.Э. Кязимова  «\_\_\_\_» сентябрь 2021 г. |
|  | Преподаватель  \_\_\_\_\_\_\_\_\_\_\_\_\_ Ю.В. Маянцева  «\_\_\_\_» сентябрь 2021 г. |
| 2021 | | |

Содержание

[**ТЕХНИКА БЕЗОПАСНОСТИ** 3](#_Toc87874773)

[**Практическая работа №1** 6](#_Toc87874774)

[**Практическая работа №2** 12](#_Toc87874775)

[**Практическая работа №3** 17](#_Toc87874776)

[**Практическая работа №4** 25](#_Toc87874777)

[**Практическая работа №5** 30](#_Toc87874778)

**ТЕХНИКА БЕЗОПАСНОСТИ**

**Общие правила безопасности**

Сотрудникам и студенты, которым разрешена работа в ЛВС, выдается имя и пароль для работы в системе, а также, если это необходимо, для работы с дополнительными сервисами. Регистрация производится по письменному представлению заинтересованных подразделений администратором ЛВС колледжа.

При регистрации пользователи обязаны ознакомиться с данными правилами и расписаться в журнале по технике безопасности.

Зарегистрированным пользователям выделяется дисковое пространство и доступ к необходимому программному обеспечению в соответствии с текущими квотами.

При нарушении нормальной работы сети пользователи обязаны поставить в известность обслуживающий персонал.

**Пользователям запрещается:**

Самостоятельно производить установку, настройку, модификацию и тестирование сетевого аппаратного или программного обеспечения.

Использовать переносные носители информации. Работа с внешними носителями разрешается только преподавателю или сотруднику ЦИТ.

Изменять системные и сетевые настройки категорически запрещается.

Передавать по сети информацию, оскорбляющую честь и достоинство других абонентов сети, содержащую призывы к насилию, свержению существующего строя, разжиганию межнациональной розни, информацию в зашифрованном виде.

Использовать ресурсы сети для осуществления любого рода коммерческой деятельности.

Предпринимать какие-либо действия прямо или косвенно направленные на нарушение нормальной работы сетевого оборудования и разрушение общих информационных ресурсов.

Передавать кому-либо свой пароль, работать под чужим регистрационным именем, а также осуществлять любые действия, связанные с получением оных.

**Правила работы в компьютерных классах во время проведения занятий**

Пользователи обязаны придерживаться правил поведения в общественных местах. Количество пользователей не должно превышать количество рабочих мест за исключением специальных случаев (например, сдача зачетов и проч.).

В случае обнаружения неисправности компьютерного оборудования немедленно сообщить об этом преподавателю или сотруднику ЦИТ.

По окончанию работы корректно выйти из системы и сдать рабочее место преподавателю.

При самостоятельной работе по первому требованию преподавателя или сотрудника ЦИТ освободить компьютер.

Пользователям запрещается самостоятельно производить включение, выключение и перезагрузку компьютеров класса, подключение и отключение периферийных устройств.

Не допускается касание монитора руками, ручкой и др. предметами.

Пользователям запрещается находиться в классах в верхней одежде, располагать на рабочих столах шапки, сумки и пр.

Категорически запрещается играть в компьютерные игры, прослушивать музыку, просматривать фильмы, если это не предусмотрено планом урока.

Во время проведения занятий на преподавателей возлагается ответственность за обеспечение порядка в классе; при самостоятельных работах ответственность возлагается на преподавателя или сотрудника ЦИТ.

**Обязанности преподавателей**

Перед проведением практических занятий в компьютерном классе, преподаватель должен согласовать с Учебной частью и руководителем ЦИТ необходимый кабинет. При этом, во избежание срыва занятий, необходимо заранее определить с руководителем ЦИТ список программного обеспечения, которое потребуется для работы.

При необходимости выхода студентов в Интернет, следует подать заявку на внутреннем сайте [www.yxmt.local/staff](http://www.yxmt.local/staff)

При проведении занятий с использованием ресурсов Интернет, преподаватель должен отслеживать посещаемые Интернет-ресурсы студентами и пресекать попытки просмотра материала, содержание которого несовместимо с задачами воспитания и образования.

Во время занятий преподаватель должен находиться в компьютерном классе. В случае необходимости на время покинуть класс, преподаватель имеет право попросить техников или лаборантов ЦИТ присутствовать в классе вместо него.

Преподаватель, во время занятий, несет ответственность за сохранность оборудования и порядок в классе.

**Обязанности техников и лаборантов во время проведения занятий**

Обеспечение проведения занятий и самостоятельной работы пользователей в классах ЦИТ (включение ПК, присутствие в ЦИТ во время занятий для устранения возникающих проблем).

Установка и конфигурирование программного обеспечения, необходимого для проведения занятий.

Восстановление поврежденного системного и прикладного программного обеспечения, а также операционной системы.

Очистка компьютеров и ковриков для мышек от пыли, протирка поверхности мышек и клавиатуры чистящими салфетками, очистка поверхности монитора с помощью предназначенных для этого моющих средств с периодичностью раз в месяц или ранее по заявке преподавателей.

Получение со склада (или приобретение) расходных материалов, необходимых для эксплуатации компьютеров.

Данные правила являются обязательными для выполнения всеми пользователями. При нарушении правил, пользователь может быть лишен доступа к работе в ЛВС. При выполнении действий, влекущих за собой материальный ущерб пользователь несет ответственность, установленную администрацией.

**Практическая работа №1**

**Тема**

Создание базы данных.

**Цель работы**

По разработанной на УП.05 ERD-диаграмме реализовать базу данных, создать скрипт и разместить его в репозитории.

**Ход работы**

По предметной области «Грузоперевозки» была создана ERD-диаграмма (Рис.1).

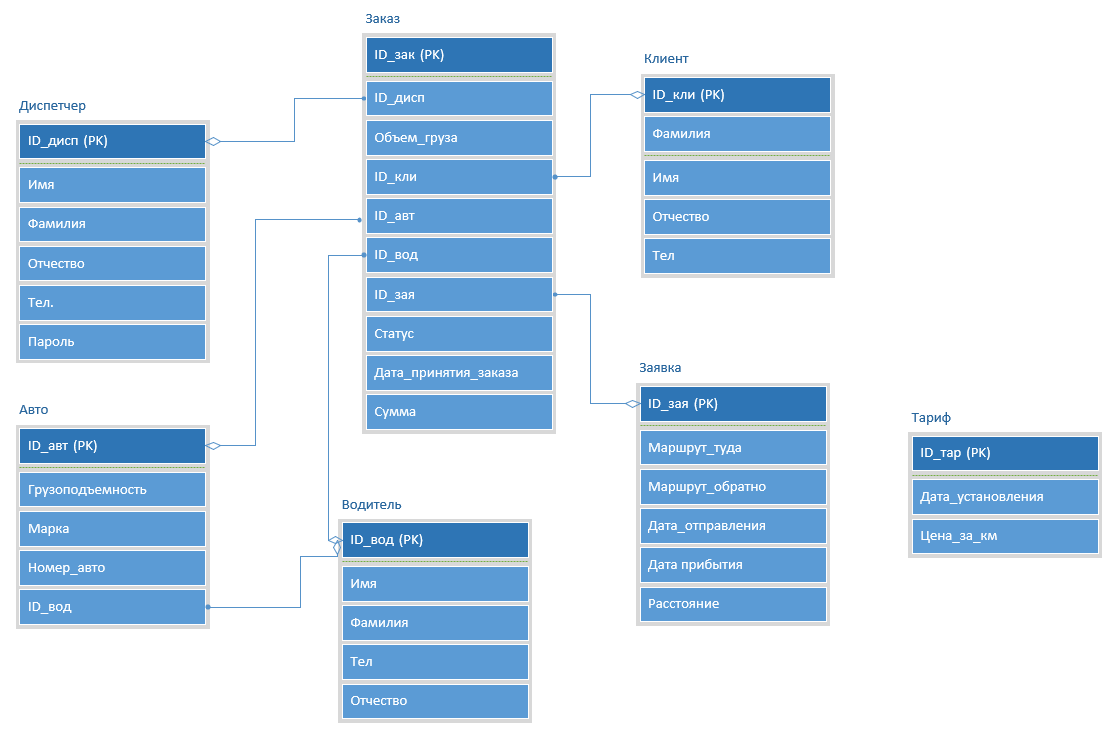


Рисунок 1. ER диаграмма

Получилось выделить несколько основных сущностей: диспетчер, клиент, заказ, заявка, водитель, машина и тариф.

После по ER модели создаем в MsSQL диаграмму базы данных «Грузоперевозки» (Рис.2). Все таблицы соединены между собой первичный ключом, связь один ко многим.

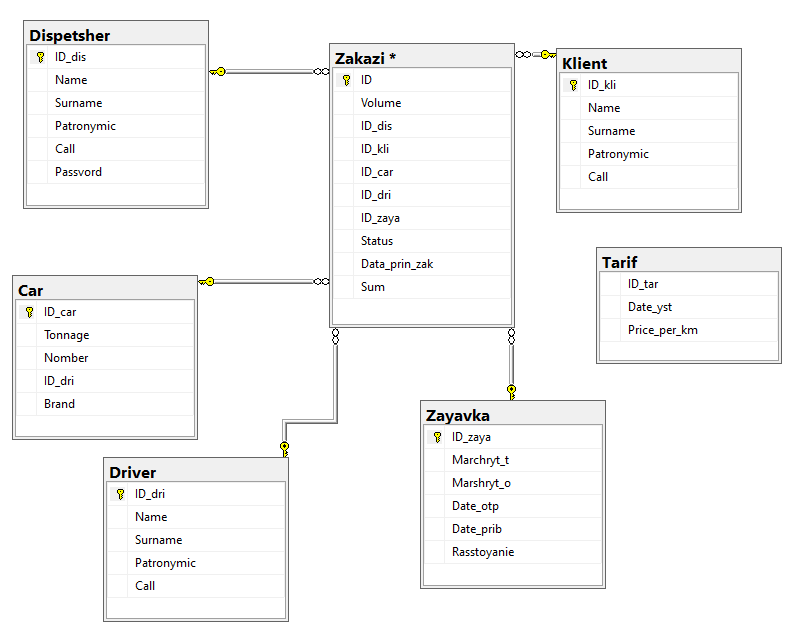


Рисунок 2 Диаграмма БД

После сохраняем все изменения и переходим к созданию скриптов. Для этого откроем меню, выберем «Создать скрипт для базы данных»-> «Используя CREATE» (Рис.3).

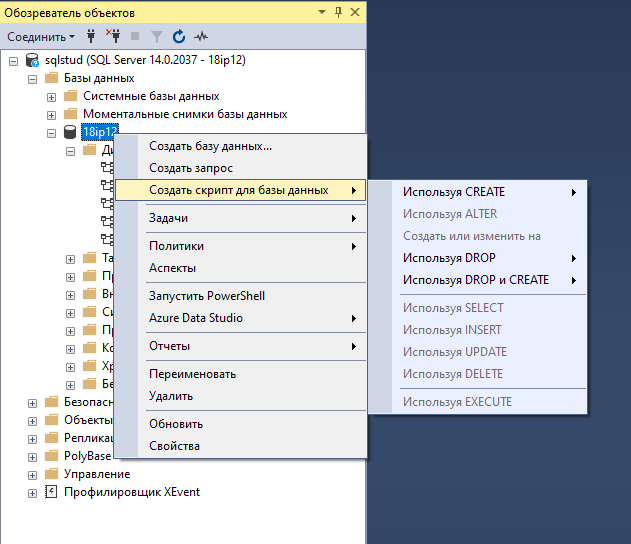


Рисунок 3 Создание скрипта

После, с помощью сочетания клавиш Ctrl+F, вызовем окошко со строкой и изменим название базы (Рис.4).

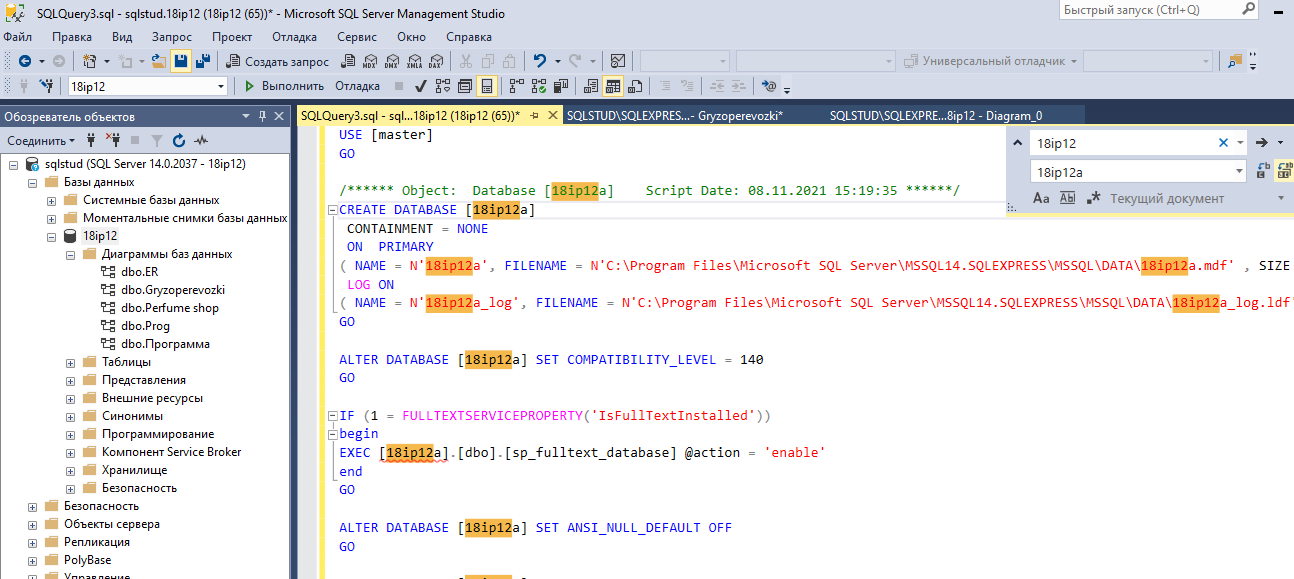


Рисунок 4 Скрипт базы данных в новом окне

Так же нужно создать скрипт базы данных с помощью команды «Создать скрипт» и сохранить в папку. Так же вызываем меню, выбираем «Задачи»-> «Сформировать скрипты», а далее указываем какую базу/таблицы нужно записать(Рис.5-6).

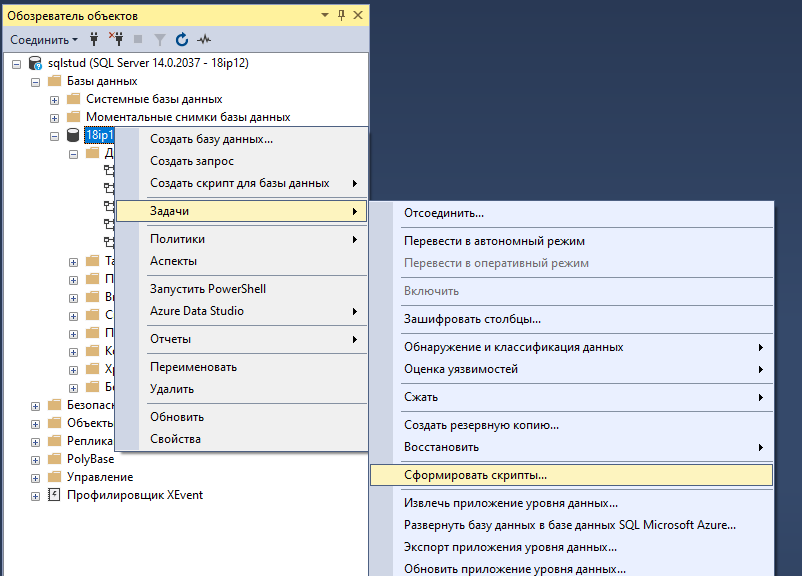


Рисунок 5 Формирование скрипта

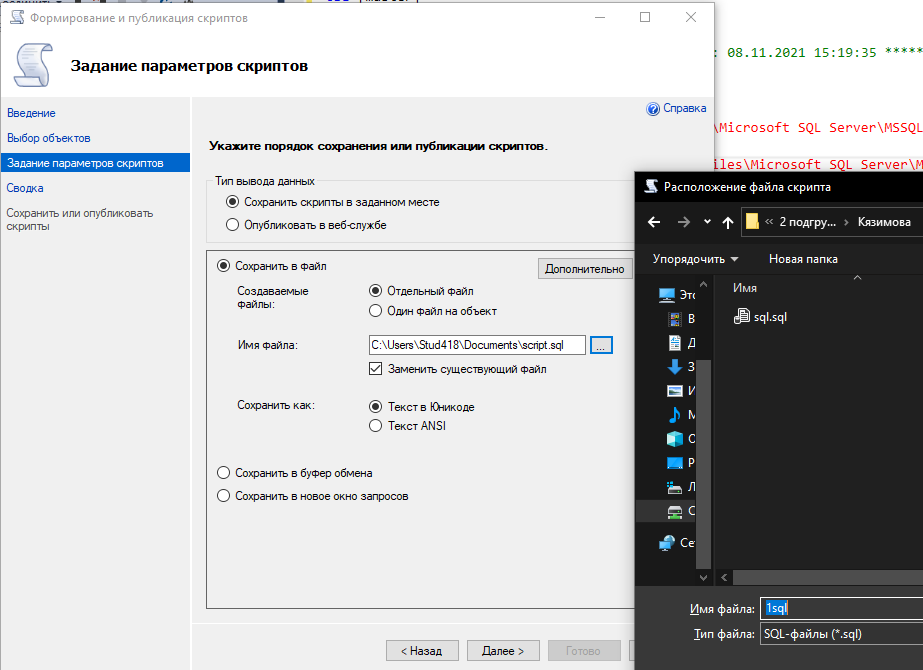


Рисунок 6 Сохранение скрипта

Перетащив скрипт в пустое поле в программе SQL он откроется и после сохранения появиться в базе (Рис.7)

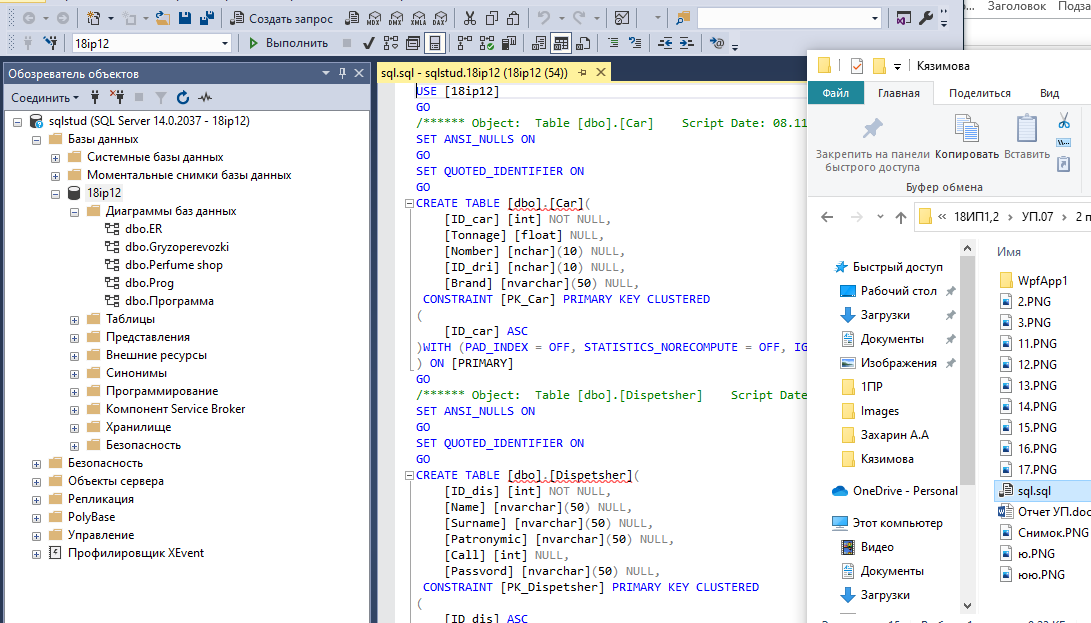


Рисунок 7 Разворачивание скрипта базы данных

**Вывод**

В результате проделанной работы реализовали ER-диаграмму в базе данных, создали скрипт и разместили все в репозитории.

**Контрольные вопросы**

1. Что такое «база данных»?

База данных - набор сведений, хранящихся некоторым упорядоченным способом.

1. Из каких этапов состоит проектирование базы данных?

Проектирование базы данных осуществляется в три этапа:

* концептуальное проектирование;
* логическое проектирование;
* физическое проектирование.

1. Что подразумевается под таблицей и полем в SQL?

Таблицы являются объектами, которые содержат все данные в базах данных. В таблицах данные логически организованы в виде строк и столбцов по аналогии с электронной таблицей.

Поля – это компоненты, представляющие структуру таблицы.

1. Что такое первичный ключ (Primary key)?

Первичный ключ (сокращенно РК - primary key) - столбец, значения которого во всех строках различны. Первичные ключи могут быть логическими (естественными) и суррогатными (искусственными).

1. Что такое внешний ключ (Foreign key)?

Внешний ключ SQL — это ключ, используемый для объединения двух таблиц. Иногда его также называют ссылочным ключом. Внешний ключ — это столбец или комбинация столбцов, значения которых соответствуют Первичному ключу в другой таблице.

1. Какие связи существуют в реляционных базах данных?

Существует три вида связей между таблицами баз данных:

* связь один к одному;
* связь один ко многим;
* связь многие ко многим.

1. Что подразумевается под целостностью данных?

Под целостностью баз данных понимается некоторый набор требований нарушение которых приведет к противоречию между базой данных  предметной областью которую она отражает

1. Что такое нормализация и каковы ее преимущества?

Нормализация – это метод проектирования базы данных, который позволяет привести базу данных к минимальной избыточности.

Избыточность устраняется, как правило, за счёт декомпозиции отношений (таблиц), т.е. разбиения одной таблицы на несколько.

Преимущества:

* устранение аномалий;
* повышении производительности;
* повышение удобства управления данными.

**Практическая работа №2**

**Тема**

Создание руководства по стилю, логотипа, иконки. Размещение файлов в репозитории.

**Цель работы**

Разместить скрипт с базой данных, руководство по стилю, логотип и иконку в удаленный репозиторий проекта.

**Ход работы**

**Руководство по стилю**

Общие требования.

При создании приложения руководствуйтесь требованиями, описанными ниже. Не допускайте орфографические и грамматические ошибки.

Использование логотипа

Все экранные формы пользовательского интерфейса должны иметь заголовок с логотипом. Не искажайте логотип (не изменяйте изображение, его пропорции, цвет). Также для приложений должна быть установлена иконка (Рис. 9).



Рисунок 8 Иконка

Шрифт

Используйте шрифт Comic Sans MS.

Цветовая схема

В качестве основного фона используется белый цвет; в качестве дополнительного: RGB (128, 128, 128). Для акцентирования внимания пользователя на целевое действие интерфейса используйте цвет RGB (255, 140, 0).

|  |  |  |
| --- | --- | --- |
| Основной фон | Дополнительный фон | Акцентирование внимания |
| RGB (255, 255, 255) | RGB (128, 128, 128) | RGB (255, 140, 0) |
|  |  |  |

**Работа с Git-ом**

Существует два способа добавления файлов в Git: локально (через Git Bash) и непосредственно на самом сайте в своем аккаунте.

Второй способ будет более удобен (НО при наличии интернета), первый же нужно прописывать в консоли команды (можно работать при отсутствии интернета). Воспользуемся первым способом. Чтобы внести наши файлы создадим репозиторий и добавим в него (Рис.9-10)

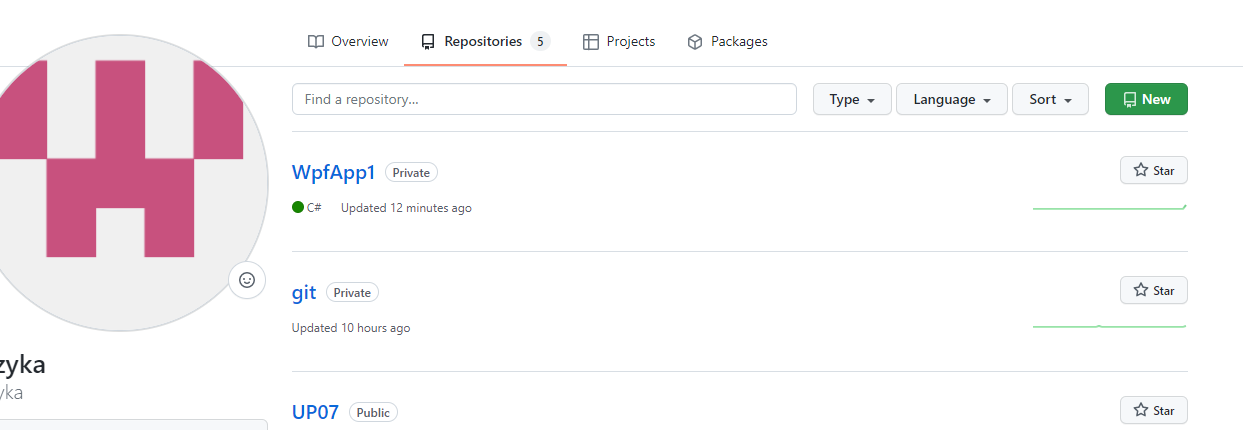


Рисунок 9 Создание нового репозитория

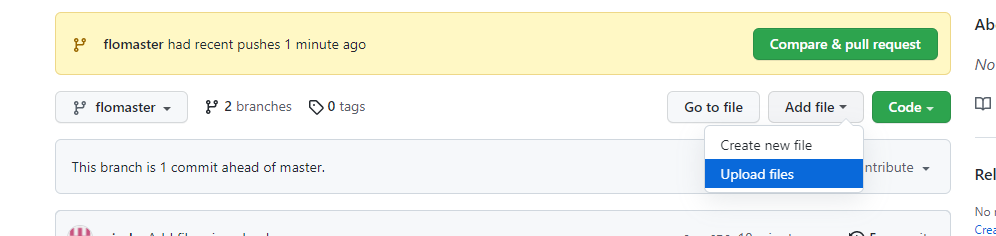


Рисунок 10 Добавление файлов в репозиторий

Результат (Рис.11)

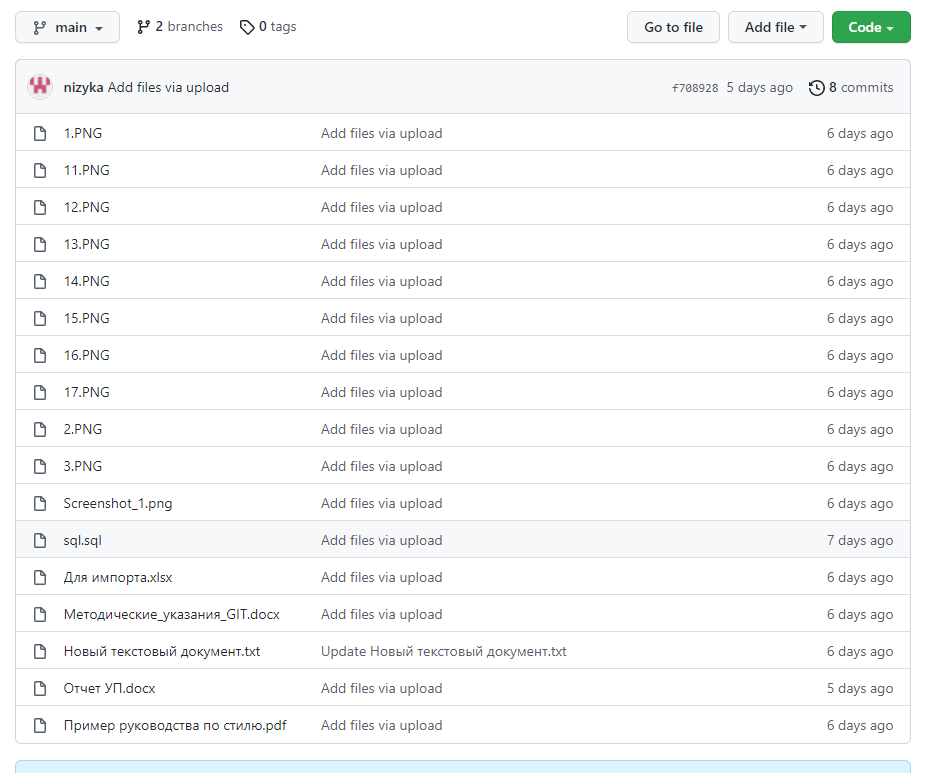


Рисунок 11 Добавленные файлы

**Вывод**

В результате проделанной работы научились создавать руководства по стилю, логотипа, иконки и разместили файлы в репозитории.

**Контрольные вопросы**

1. В чем заключается экономия времени при использовании системы контроля версий?

Внеся изменения в один или несколько файлов проекта, программист записывает изменения в репозиторий – хранилище всех версий и изменений проекта. Стоит отметить, что сохраняется не весь проект целиком, а, в целях экономии места и времени сохранения изменений (сервер с репозиторием может быть удаленным, и, если проект - очень большой, то требуется достаточно большое время для передачи всех его файлов по сети), в репозиторий добавляются только файлы, претерпевшие изменения.

1. В чем преимущества использования системы контроля версий?

* Проста в использовании и хорошо подходит для ознакомления с принципами работы систем контроля версий.
* Хорошо подходит для резервного копирования отдельных файлов, не требующих частого изменения группой пользователей.
* Широко распространена и предустановлена в большинстве свободно распространяемых операционных системах

1. Что такое Git?

Git - это консольная утилита, для отслеживания и ведения истории изменения файлов, в вашем проекте

1. Как начать использовать Git?

Для работы нужно скачать и установитьGit. Далее программу нужно настроить: наше имя пользователя и адрес электронной почты. Открыть терминал и запустите команды.

1. Как начать использовать GitHub?

Нужно начать с регистрации. Переходим по ссылке [github.com/signup/free](https://github.com/signup/free) и вводим свои данные.

После регистрации попадаем на Dashboard нашего аккаунта.

Изначально у нас нет ни одного репозитория, и можно либо создать новый репозиторий, либо ответвиться (fork) от уже существующего чужого репозитория и вести собственную ветку разработки. Затем, при желании, свои изменения можно предложить автору исходного репозитория (Pull request).

1. Основные (наиболее часто используемые) команды Git.

**git add** - добавляет содержимое рабочего каталога;

**git status** - показывает состояния файлов в рабочем каталоге;

**git diff -** используется для вычисления разницы между любыми двумя Git деревьями;

git commit - берёт все данные, добавленные в индекс с помощью git add, и сохраняет их слепок во внутренней базе данных;

**git rm** - используется в Git для удаления файлов из рабочей копии;

**git clean** - используется для удаления мусора из рабочего каталога.

1. Какие сервисы существуют для Git?

Ряд сервисов предоставляют хостинг для git-репозиториев, среди наиболее известных — GitHub, Codebase, SourceForge, SourceHut, Gitea, Bitbucket, GitLab.

1. Как работать с локальным репозиторием?

Для того, чтобы внести вклад в какой-либо Git-проект, необходимо уметь работать с удалёнными репозиториями. Удалённые репозитории представляют собой версии вашего проекта, сохранённые в интернете или ещё где-то в сети. У вас может быть несколько удалённых репозиториев, каждый из которых может быть доступен для чтения или для чтения-записи. Взаимодействие с другими пользователями предполагает управление удалёнными репозиториями, а также отправку и получение данных из них.

Для локальной работы с репозиторием его нужно клонировать.

Для этого используется команда git clone.

1. Как работать с распределенным репозиторием?

 В РСКВ клиенты не просто скачивают снимок всех файлов (состояние файлов на определенный момент времени) - они полностью копируют репозиторий. В этом случае, если один из серверов, через который разработчики обмениваются данными, умрёт, любой клиентский репозиторий может быть скопирован на другой сервер для продолжения работы. Каждая копия репозитория является полным бэкапом всех данных.

**Практическая работа №3**

**Тема**

Создание проекта в Visual Studio. Настройка проекта. Создание словаря стилей. Создание Git репозитория с помощью Visual Studio.

**Цель работы**

Создать и настроить проект в Visual Studio, создать репозиторий, используя средства Visual Studio.

**Ход работы**

Для работы нам понадобиться создать несколько папок в «Обозревателе решений» которые позднее понадобятся (Рис.12).

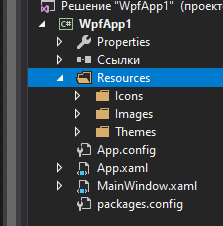


Рисунок 12 Созданные папки

В созданные папки добавим иконку (в папку Icons) для нашей программы, изображения и словарь стилей (Рис.13)

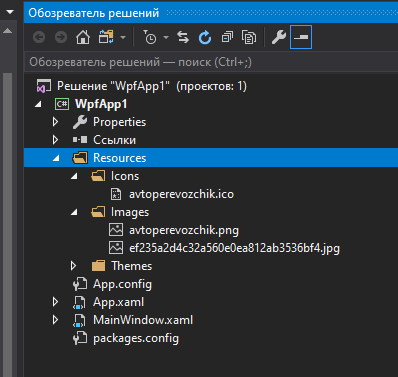


Рисунок 13 Добавление файлов в папки

В папке Themes создадим словарь стилей (Рис.14).

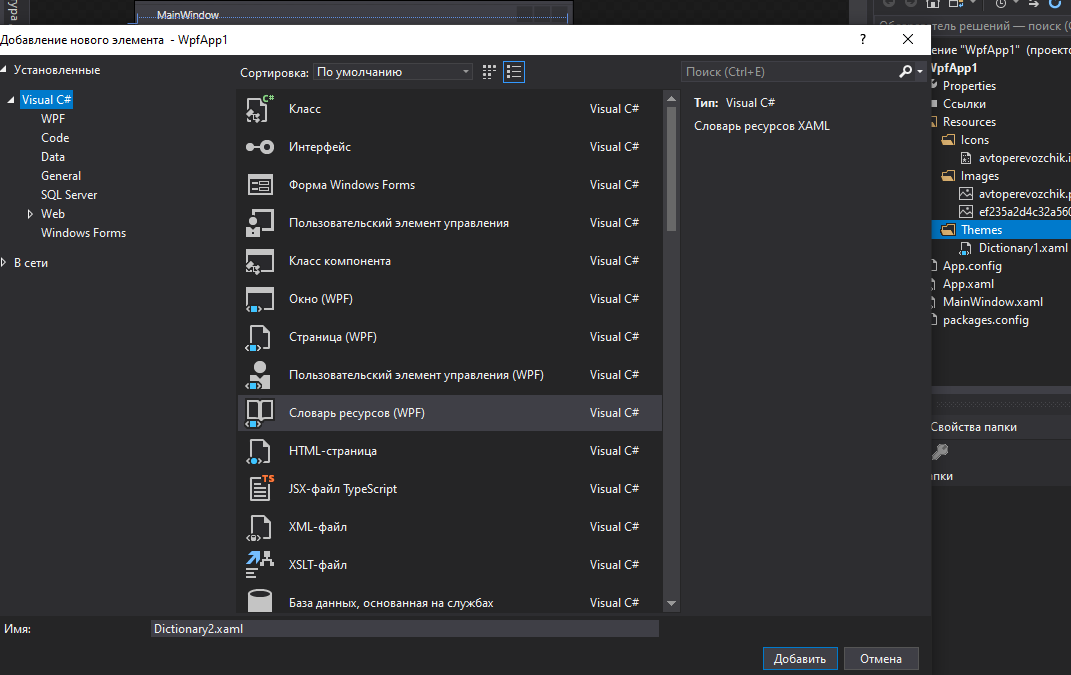


Рисунок 14 Добавление словаря стилей

Внутри тега ResourceDictionary прописываем стили (цвет, границы, размер) для необходимых элементов в созданном словаре ресурсов (Рис.15).

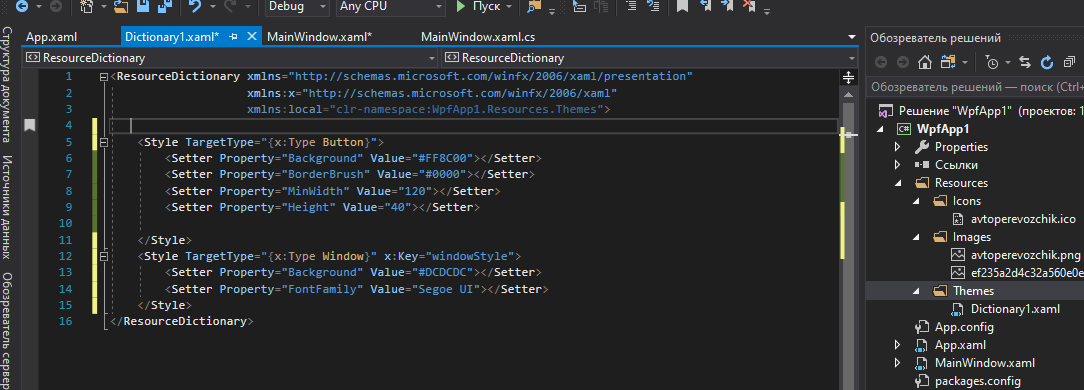


Рисунок 15 Указание стилей элементов

После этого, нужно применить созданный словарь к программе. Для этого откроем файл App.xaml. Внутри тега Application.Resources пропишем путь для нашего справочника ресурсов (который располагается в Resources/Themes) и так же указать в конструкторе. (Рис.16-17).

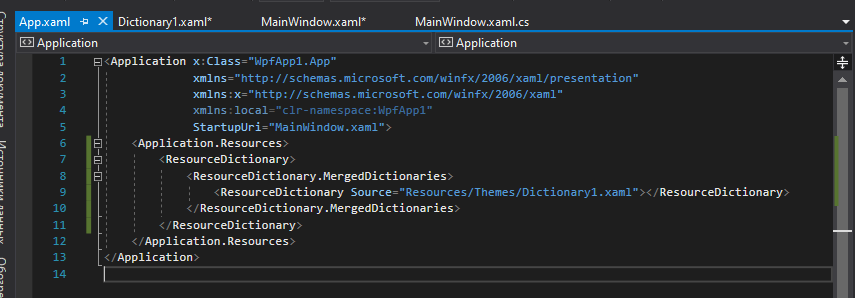


Рисунок 16 Применение словаря к программе

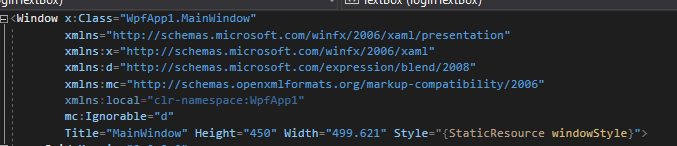


Рисунок 17 Применение стиля к окну

Далее в окно приложения размещаем элементы. Результат применения стилей (Рис.18)

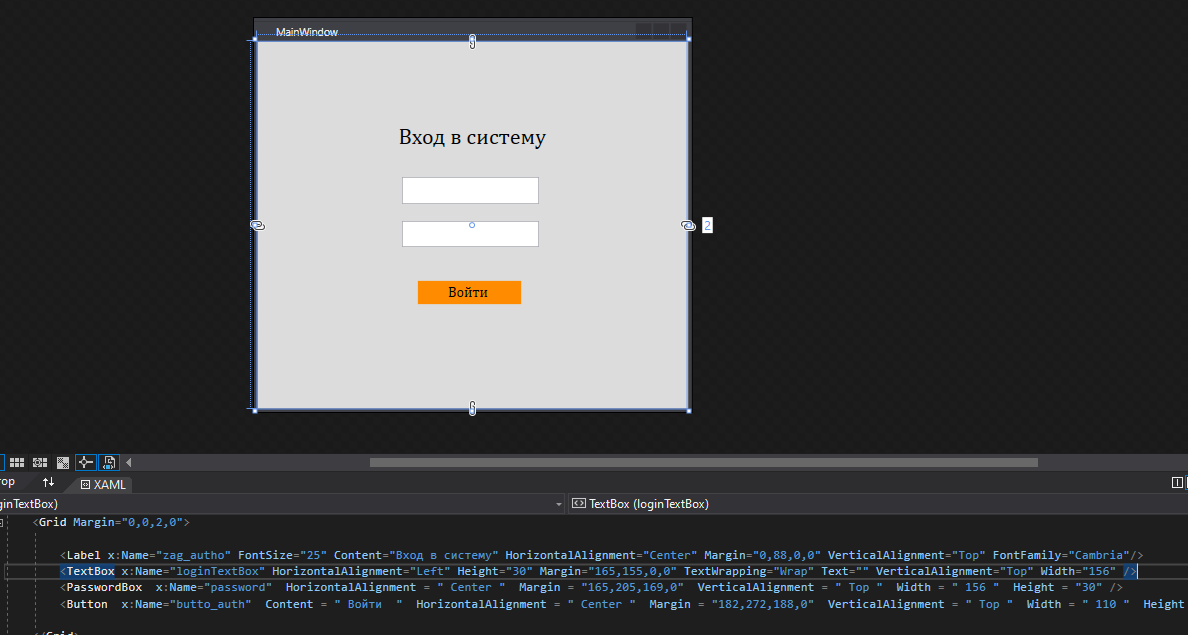


Рисунок 18 Результат

После, в Visual Studio нужно создать Git репозиторий. Во вкладке «Изменение Git» нажимаем кнопку «Создать репозиторий» (Рис.19).

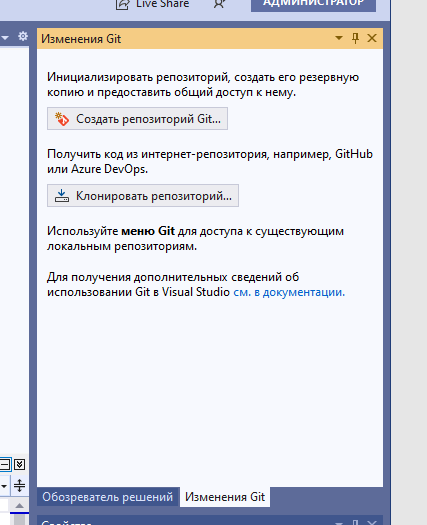


Рисунок 19 Создание репозитория в GitHub

Далее нужно авторизироваться в Github (Рис.20)

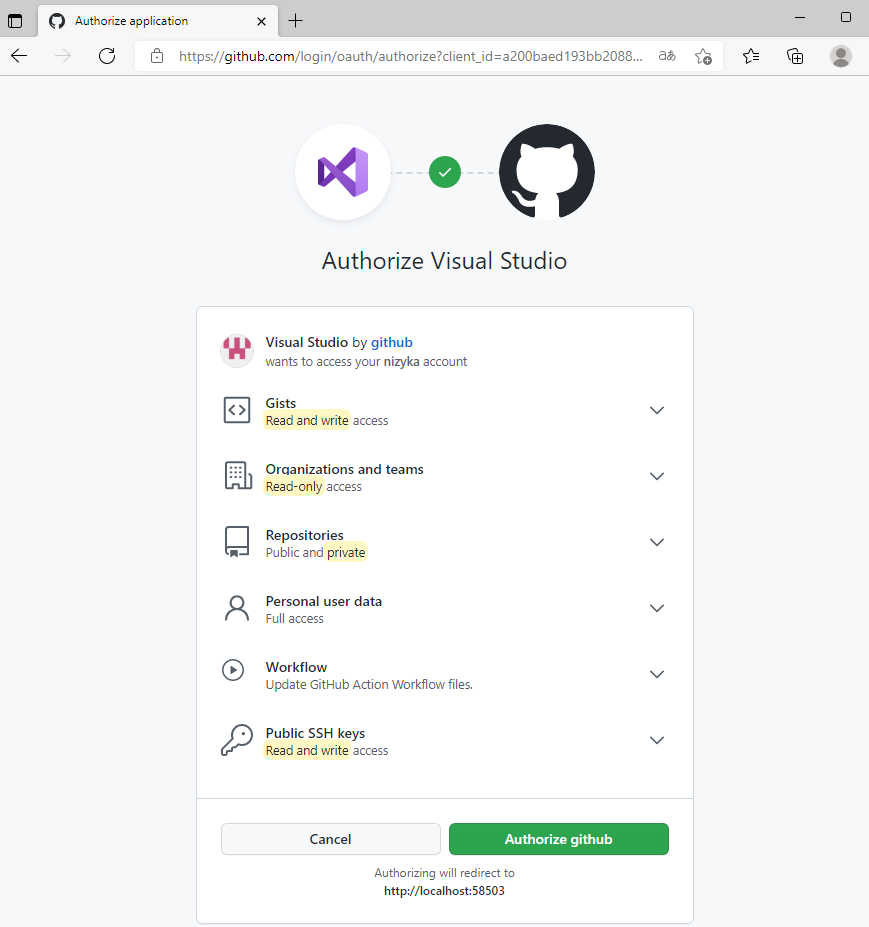


Рисунок 20 Авторизация в GitHub

После входа, в окне «Создание репозитория» появится информация об учетной записи, в которую ранее мы авторизировались (Рис.21)

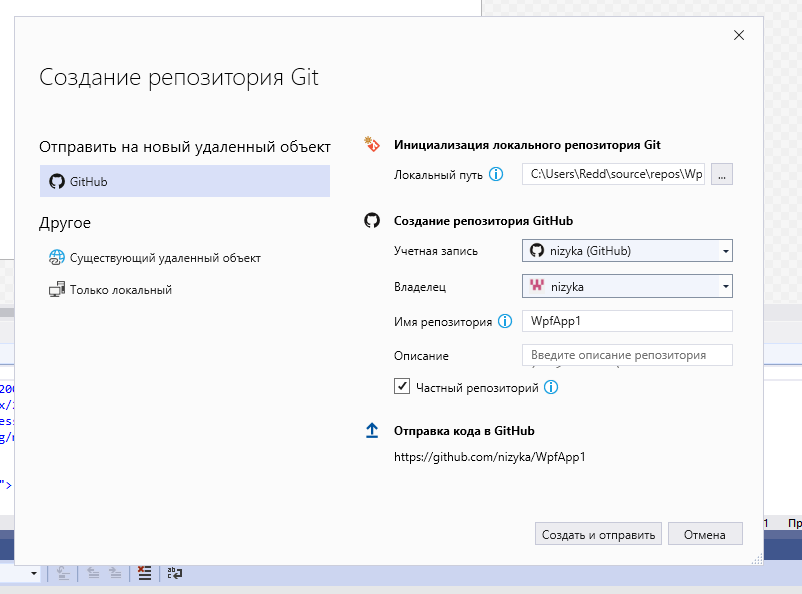


Рисунок 21 Подключенная учетная запись

Теперь в нашем аккаунте на Github появился новый репозиторий с нашим приложением (Рис.22)

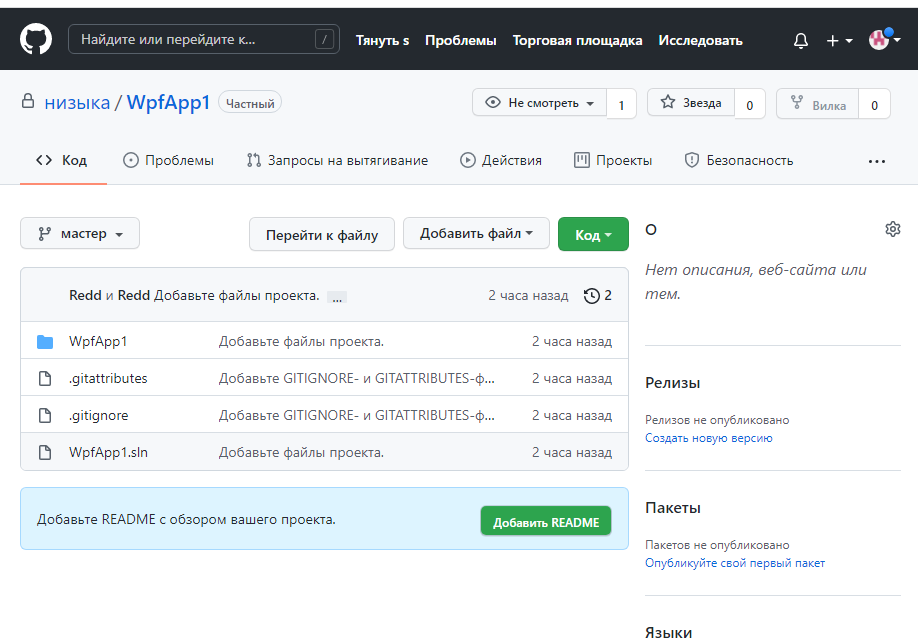


Рисунок 22 Созданный репозиторий

Теперь в проекте нахватает только файла readme.md. Кликнув ПКМ по решению нажмем «Открыть папку в проводнике» и в открывшуюся папку добавим файл readme.md (Рис.23-24)

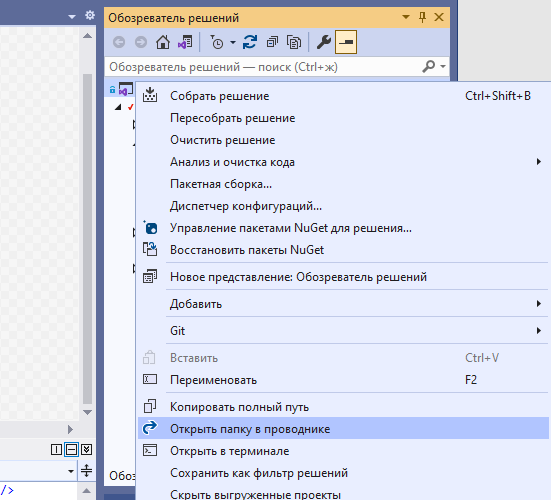


Рисунок 23 Обозреватель решений

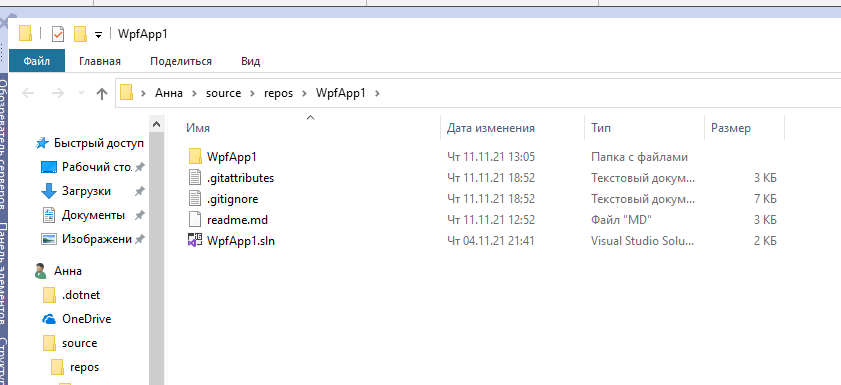


Рисунок 24 Добавление файла readme.md

Теперь создадим коммит с изменениями в проекте. Для этого в текстовое поле введём комментарий к коммиту и нажмем на кнопку «Зафиксировать все» (Рис.25)

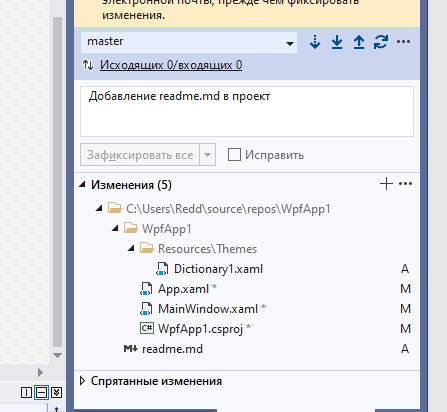


Рисунок 25 Создание коммита

Результат изменений (Рис.26)

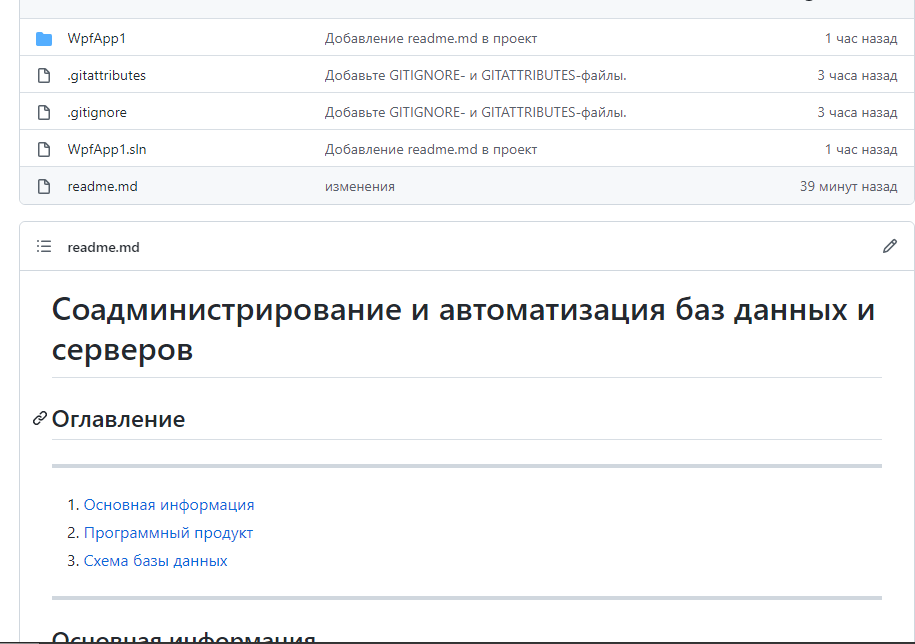


Рисунок 26 Результат

**Вывод**

В результате проделанной работы создали и настроили проект в Visual Studio, создали репозиторий, используя средства Visual Studio.

**Контрольные вопросы**

1. Для чего предназначен словарь ресурсов в проекте WPF?

Чтобы разделить ресурсы между множеством проектов можно создать словарь ресурсов. Словарь ресурсов представляет собой просто XAML-документ, который всего лишь хранит необходимые ресурсы.

1. Как определить стиль для элемента управления в WPF?

Для начала нужно создать словарь ресурсов и там прописать для каждого элемента свой стиль, а после подключить его к проекту

1. За что отвечает атрибут x:Key у стиля?

Однозначно идентифицирует элементы, которые создаются и на которые ссылаются в словаре, определяемом XAML. Добавление данного элемента является наиболее распространенным способом идентификации ресурсов в словаре.

1. Как применить стиль, у которого указан атрибут x:Key, к элементу на форме?

Если у стиля указан атрибут x:Key, необходимо у окна явно указать данный стиль ( в файле MainWindow.xaml и во всех остальных формах приложения)

**Практическая работа №4**

**Тема**

Подключение базы данных. Авторизация в системе.

**Цель работы**

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

**Ход работы**

В ходе практики была разработана база данных (см 1 практическую работу).

Теперь эту базу нужно подключить к нашей программе.

В обозревателе решений ПКМ вызовем меню и создадим элемент «Модель ADO.NET» (Рис.27)

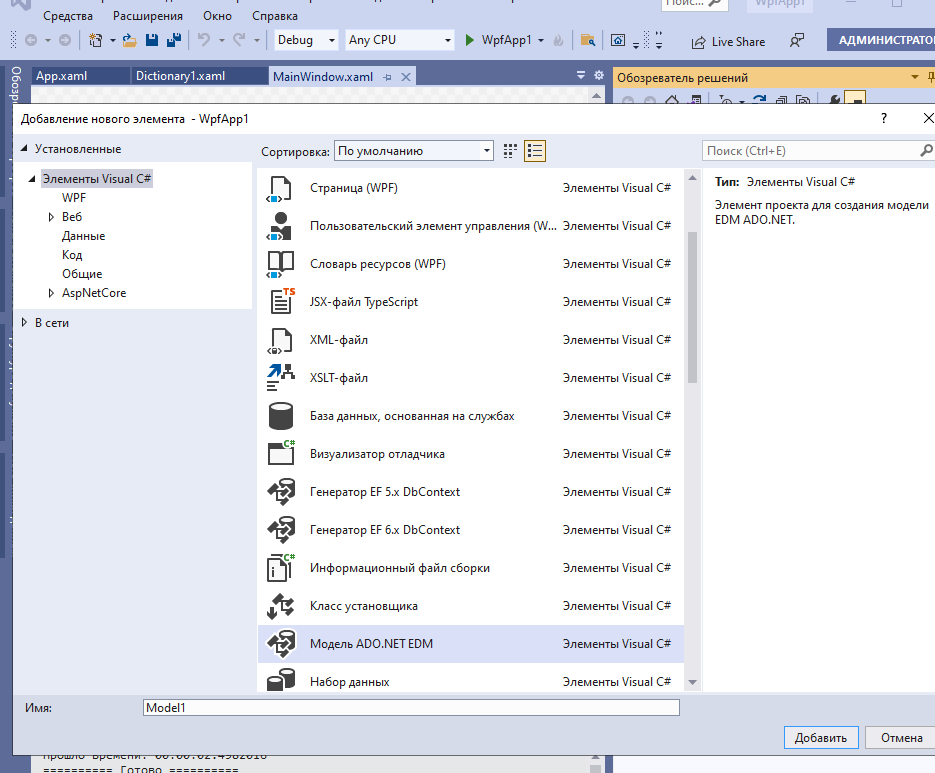


Рисунок 27 Создание элемента

Далее укажем нужную базу и выберем таблицы, которые в дальнейшем нам пригодяться (Рис.27)

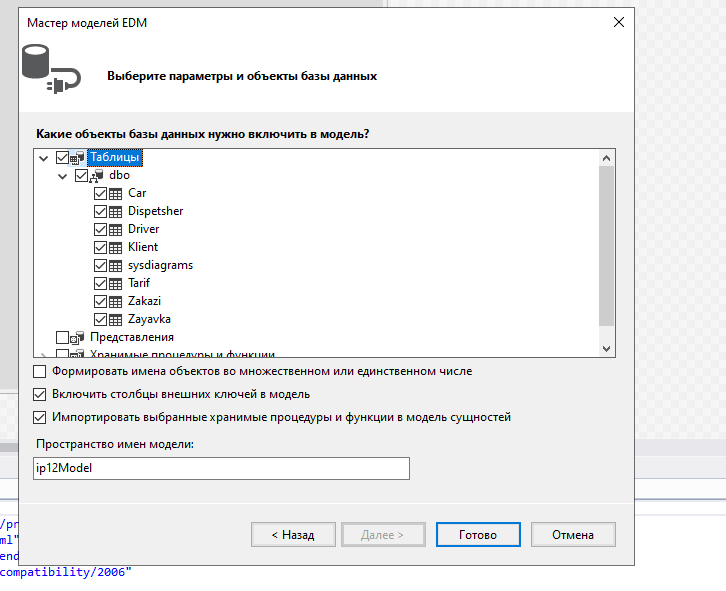


Рисунок 28 Выбранные таблицы

Результат (Рис.29)

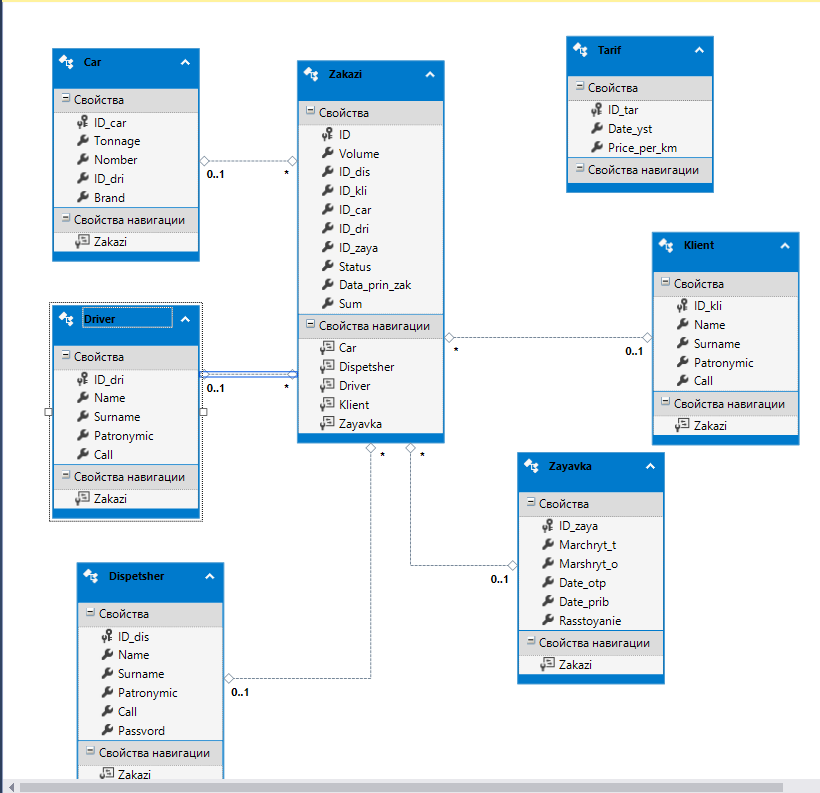


Рисунок 29 Подключенная база данных

После, чтобы БД правильно работала, нужно прописать небольшой кусочек кода (Рис.30)

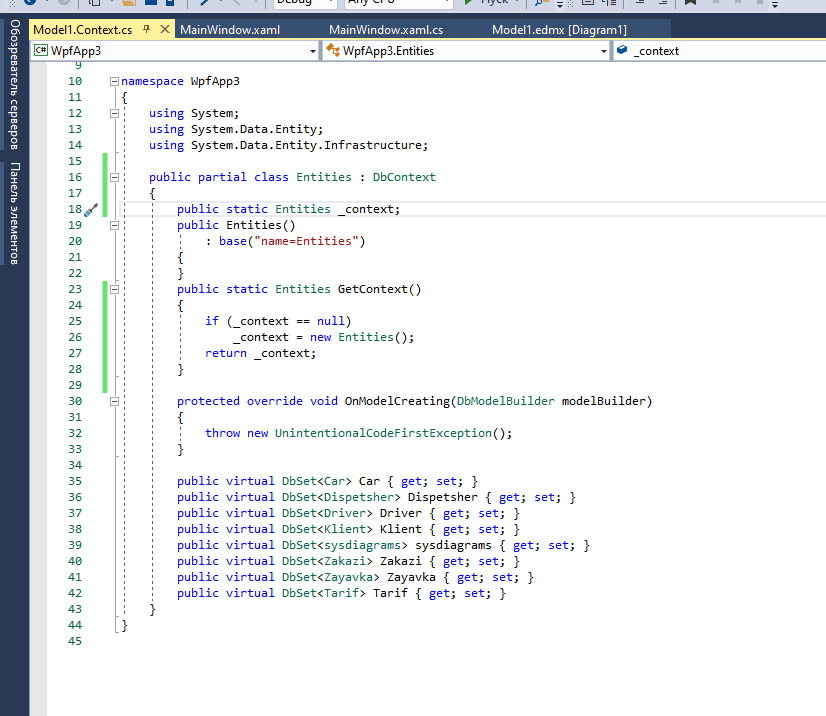


Рисунок 30 Код для работы БД

Ранее мы создали форму для авторизации пользователя, теперь, чтобы она правильно работала, нужно к ней подключить базу из которой будут браться данные для авторизации. Двойным нажатием по созданной кнопке «вход» перейдем в поле для написания кода для данного эдемента (Рис.31)

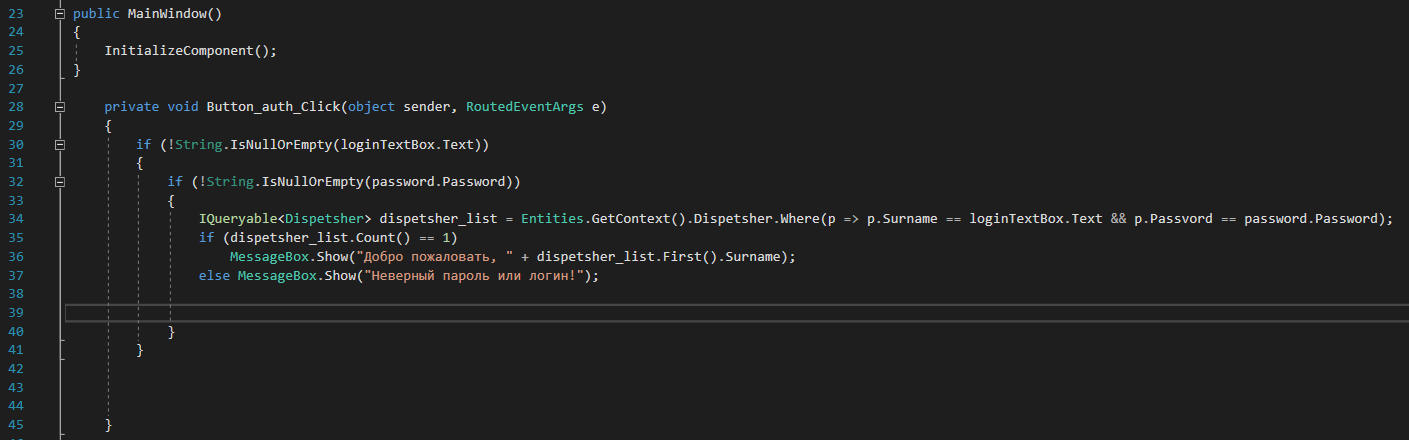


Рисунок 31 Код для окна авторизации

Запускаем нашу программу и наблюдаем за результатом (Рис.32)

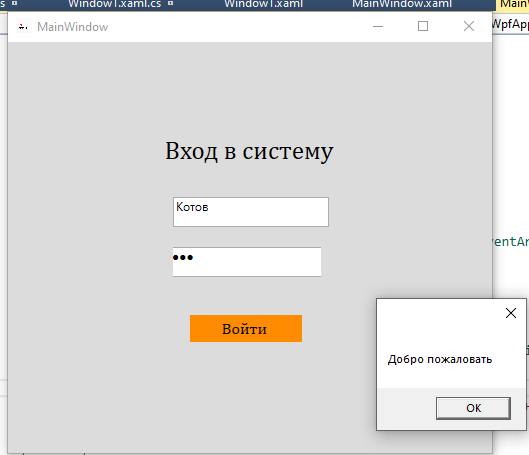


Рисунок 32 Работа приложения (авторизация)

**Вывод**

В результате проделанной работы к созданному приложению была подключена база данных, и создана авторизация к будущей системе .

**Контрольные вопросы**

1. Что такое «Entity Framework»?

Entity Framework — это решение для работы с базами данных, которое используется в программировании на языках семейства.NET. Оно позволяет взаимодействовать с СУБД с помощью сущностей (entity), а не таблиц. Также код с использованием EF пишется гораздо быстрее.

1. В чем плюсы и в чем минусы «Entity Framework»?

Entity Framework позволяет значительно сократить код работы с базами данных. Однако стоит учитывать, что Entity Framework выступает прослойкой между приложением и базой данных, поэтому может ухудшаться производительность.

1. Какие возможности предоставляет «Entity Framework»?

Entiti Framework предоставляет возможность использовать:

* foreign keys;
* связи one-to-one, one-to-many и many-to-many;
* параметризацию запросов;
* хранимые процедуры.

1. Как создать подключение к базе данных?

При помощи создания элемента ADO.EDM.NET, настройке подключения к необходимой базе данных и выбора требуемых таблиц.

**Практическая работа №5**

**Тема**

Отображение набора данных, навигация.

**Цель работы**

Реализовать отображение данных базы данных на форме. Сделать подсветку строк по условию.

**Ход работы**

Ранее для системы была создана авторизация, теперь нужно реализовать главную страницу данной программы.

Реализовав интерфейс в новом окне WPF (расположив все необходимые элементы) (Рис. 33), в коде созданной форме нужно указать все данные из базы, которые выводим (Рис. 34)

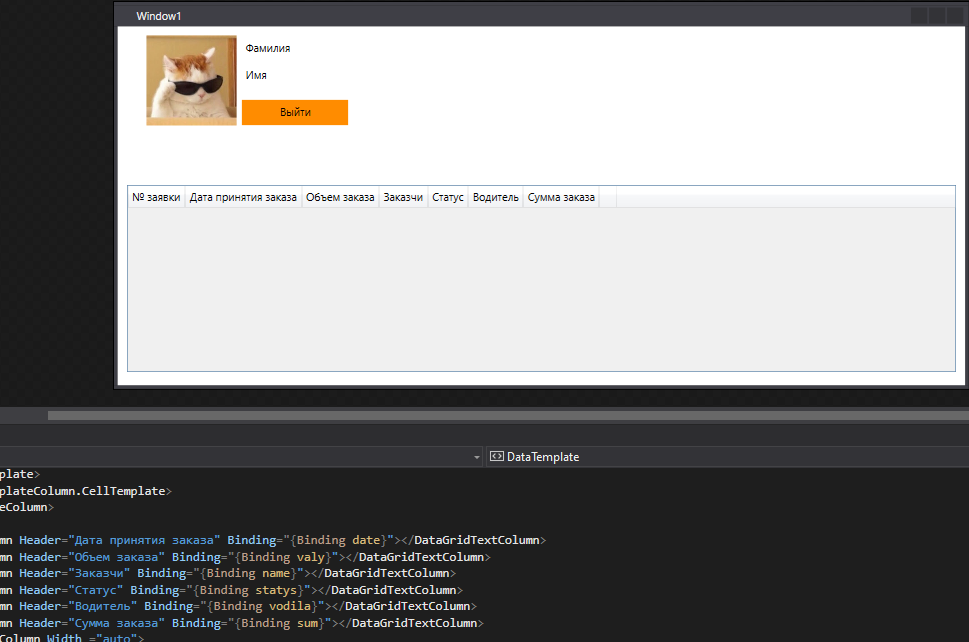


Рисунок 33 Реализация интерфейса

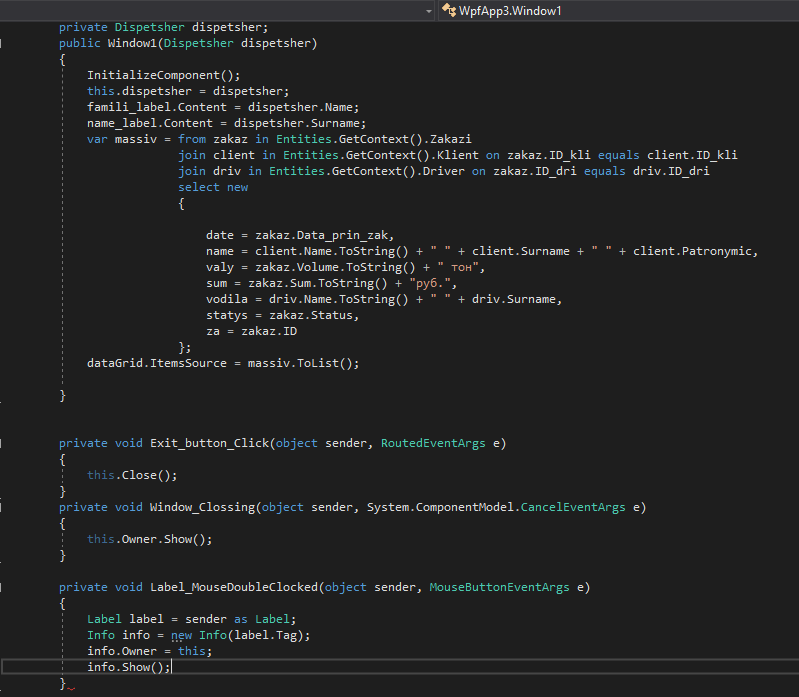


Рисунок 34 Реализация кода

