|  |
| --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  «Национальный исследовательский ядерный университет «МИФИ» |
| **Димитровградский инженерно-технологический институт –**  филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ»  **(ДИТИ НИЯУ МИФИ)** |

**Специальность 09.02.07 Информационные системы и программирование**

**МДК.01.01 Разработка программных модулей**

«Дифференцированный зачёт»

Выполнил студент 3 курса 331 группы

Жуйков Богдан Адамович

(Ф.И.О. студента)

Преподаватель ДИТИ НИЯУ МИФИ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.В. Надеждина

Работа сдана 08.03.2020г

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Димитровград

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc101029476)

[ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 4](#_Toc101029477)

[ПРАКТИЧЕСКАЯ ЧАСТЬ 5](#_Toc101029478)

[Задание №1. Создание дизайна для окна авторизации 5](#_Toc101029479)

[Задание №2. Подключение MySQL и создание базы данных 8](#_Toc101029480)

[Задание №3. Авторизация пользователя через базу данных 11](#_Toc101029481)

[Задание №4. Авторизация пользователя через базу данных 12](#_Toc101029482)

[Задание №5. Регистрация пользователя в базе данных 13](#_Toc101029483)

[Задание №6. Переход между окнами в приложении 15](#_Toc101029484)

[Задание №7. Контрольное: Создать в БД таблицу, по вашей теме, чтобы она выводилась на форме "Главный экран", после авторизации. 17](#_Toc101029485)

[ЗАКЛЮЧЕНИЕ 19](#_Toc101029486)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 20](#_Toc101029487)

**ВВЕДЕНИЕ**

Приложение «ZhuykovSQLApp» является актуальным в наше время, так как оно написано на современном языке программирования C#, имеет подключение к базе данных, находящейся на сервере, что позволяет вносить изменения и сразу получать результат действий, также данное приложение в дальнейшем может обновляться, могут появляться новые полезные функции или же если появятся баги, то они могут быть оперативно исправлены.

Главной целью этой работы было: научиться составлять БД и запросы к ней, после чего совместить её с проектом, в котором будет реализованы окна авторизации/регистрации и окна для вывода таблиц из созданной БД. Научиться создавать интерфейс программы и функционал.

Данное приложение предназначено для работы в кондитерском магазине.

Оно упрощает ведение учёта журнала поступления, содержит информацию о поставщиках, позволяет добавлять новый товар, вся информация находится на удалённом сервере с базой данных, что упрощает внесение изменений.

Благодаря данной программе можно будет делать похожие проекты и для других заведений, благодаря тому, что сама программа позволяет дополнять её функционал.

**ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

Во время разработки программы использовался популярный и актуальный на момент разработки язык программирования C#. С помощью данного языка программирования легко можно создать собственное приложение, так как можно использовать при создании приложения шаблон Windows Forms, что существенно облегчает разработку. Также с помощью функций и методов, которые присутствуют в данном языке программирования легко подключиться к базе данных SQL, которая расположена на локальном сервере.

Для создания базы данных использовали SQL, а точнее с помощью программы MAMP версии 4.2.0 я создал свой локальный сервер, на котором расположена база данных, что существенно упростило мою работу и сэкономило мне много времени, так как после создания и запуска локального сервера, я смог перейти в phpmyadmin и там уже продолжить создание базы данных, это довольно удобный и простой в освоении сайт по созданию базы данных на SQL, с этого сайта легко экспортировать саму базу данных, чтобы перенести её на другой компьютер или сервер и также импортировать обратно.

Для загрузки своего проекта в общий доступ выбрали GitHub, в данный момент это актуальная платформа для загрузки своих приложений или чего-то подобного, очень простой и удобный интерфейс, загрузка проекта осуществляется через GitHub Desktop, что упрощает загрузку приложения в репозиторий.

C# — объектно-ориентированный язык программирования. Разработан в 1998—2001 годах группой инженеров компании Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота как язык разработки приложений для платформы Microsoft .NET Framework и .NET Core. C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java.

SQL — декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных. Является, прежде всего, информационно-логическим языком, предназначенным для описания, изменения и извлечения данных, хранимых в реляционных базах данных.

GitHub — крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Веб-сервис основан на системе контроля версий Git и разработан на Ruby on Rails и Erlang компанией GitHub, Inc. Сервис бесплатен для проектов с открытым исходным кодом и (с 2019 года) небольших частных проектов, предоставляя им все возможности (включая SSL), а для крупных корпоративных проектов предлагаются различные платные тарифные планы.

**ПРАКТИЧЕСКАЯ ЧАСТЬ**

**Задание №1. Создание дизайна для окна авторизации**

Создали новое приложения под именем ZhuykovSQLApp, добавили формы с именем LoginForm, поместили на форму элементы: 4 PictureBox, 2 Label, 2 TextBox, Button. Подписали, настроили элементы (рис. 1):

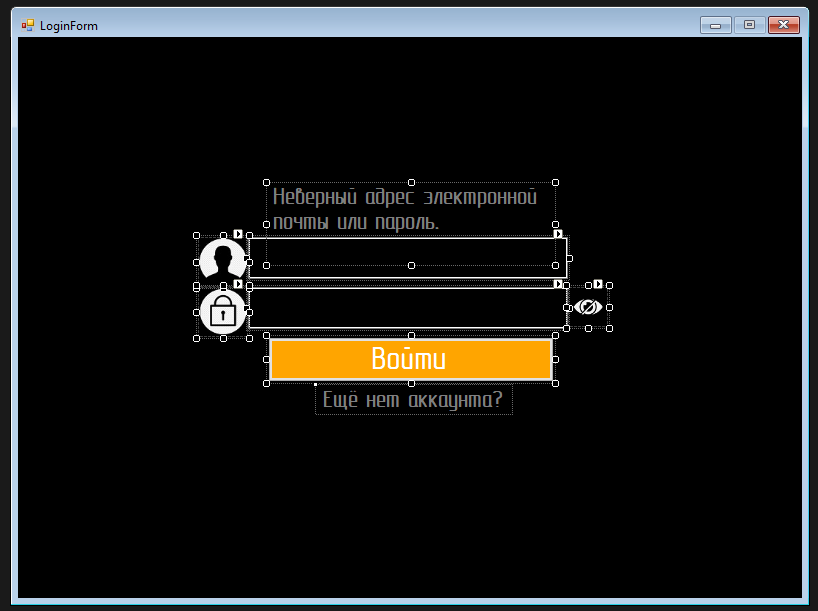


Рисунок 1 – Форма авторизации

Создали в папке проекта папку Images для картинок (рис. 2):

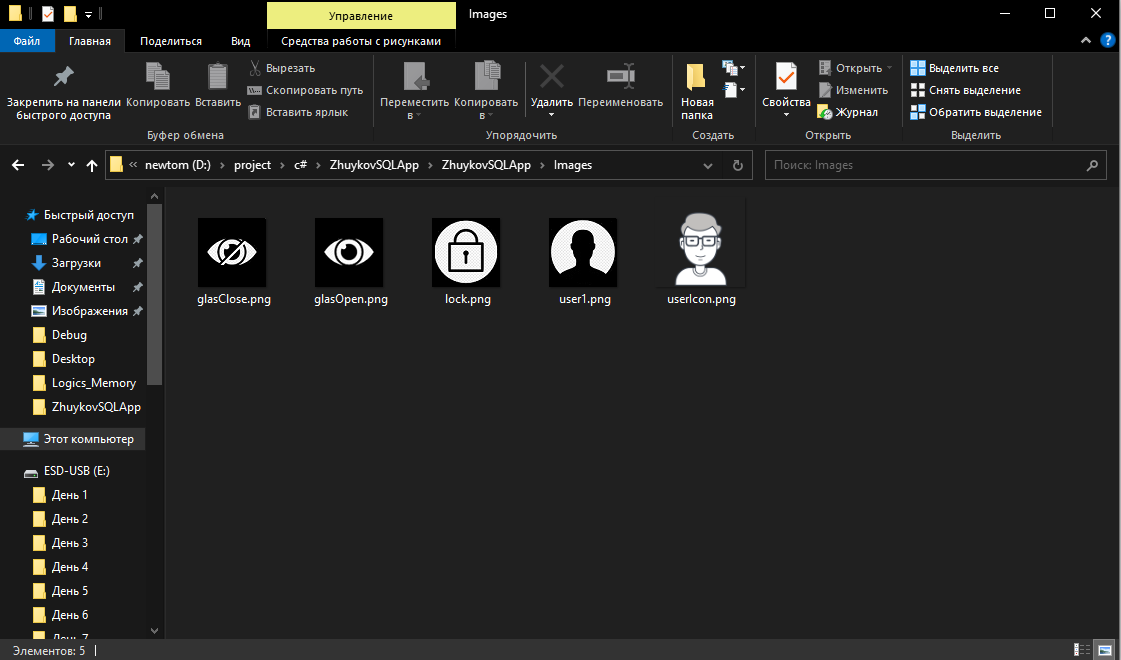


Рисунок 2 – Папка с картинками

Для компонентов TextBox написали код, который будет срабатывать при наведении курсора на элемент (рис. 3):

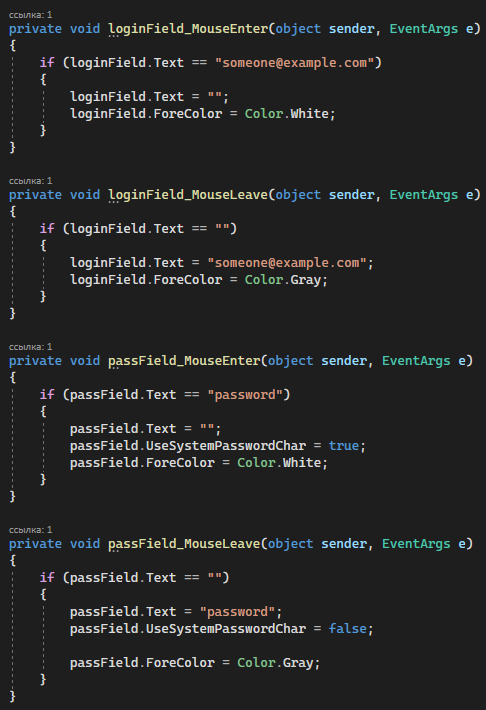


Рисунок 3 – Поля для ввода

Написали код, который срабатывает по нажатию на картинку вследствие чего свойства Visible и UseSystemPasswordChar меняются на противоположные, чтобы мы могли увидеть введённый текст, и также наоборот, если мы видим текст, то он скроется (рис. 4):

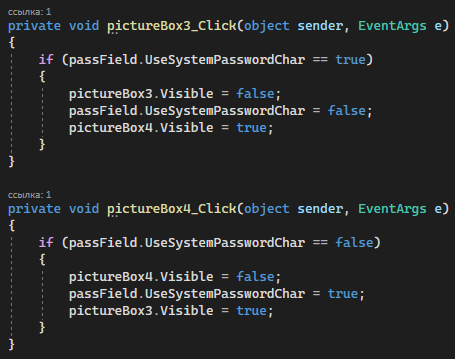
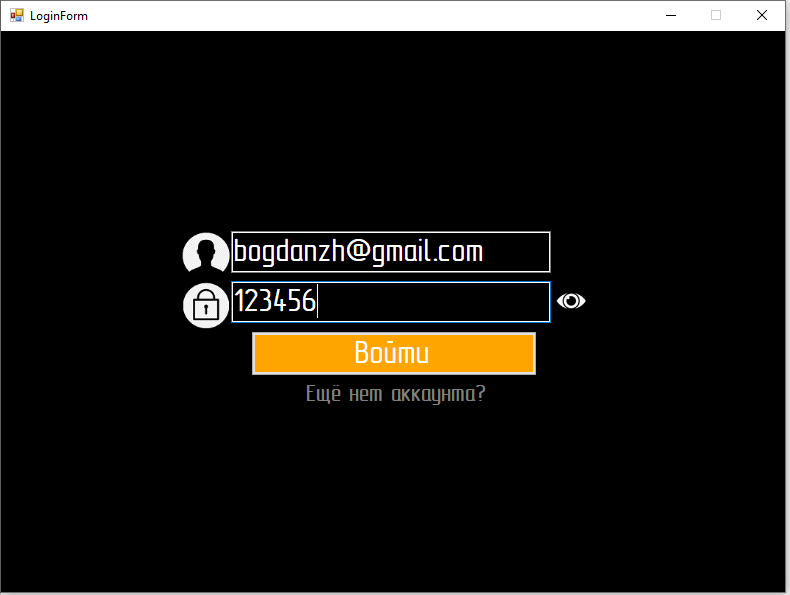
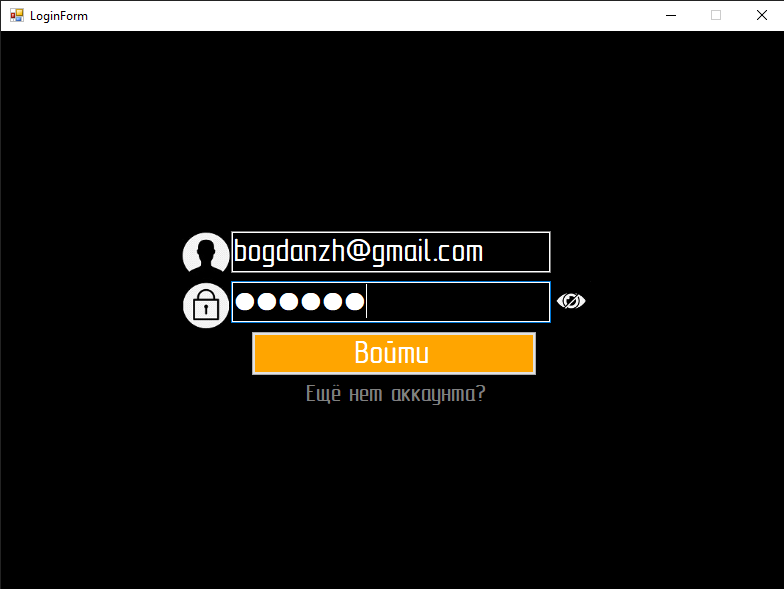


Рисунок 4 – Смена отображения текста в поле для ввода пароля по нажатию на картинку

Проверили работу кода, который меняет отображение текста (рис. 5, а), (рис. 5, б):



а б

Рисунок 5 – Проверка уже проделанной работы

## Задание №2. Подключение MySQL и создание базы данных

Добавили ссылку на файл MySql.Data.dll (рис. 6):

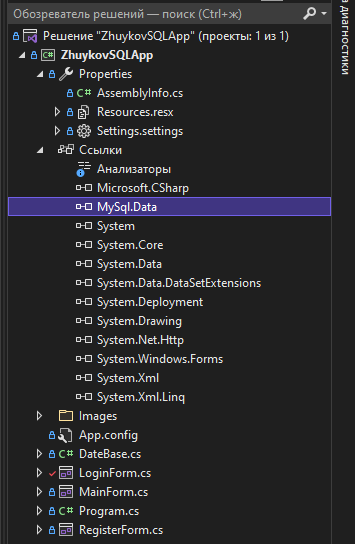


Рисунок 6 – Ссылка на MySql.Data.dll

Установили MAMP версии 4.2.0 и запустили локальный сервер (рис. 7):

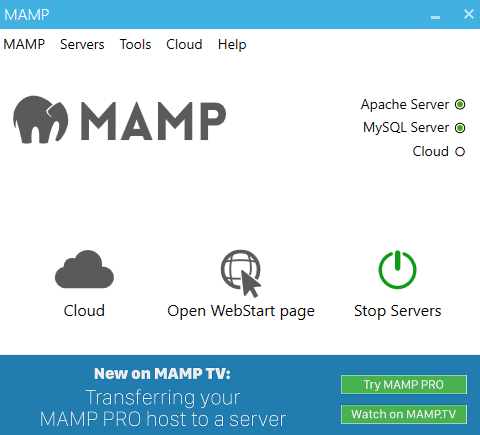


Рисунок 7 – Создание сервера MAMP

Перешли на PhpMyAdmin и создали там базу данных, а в ней 4 таблицы: конфеты, поставщики, журнал поступления и таблицу для личной информации (рис. 8):

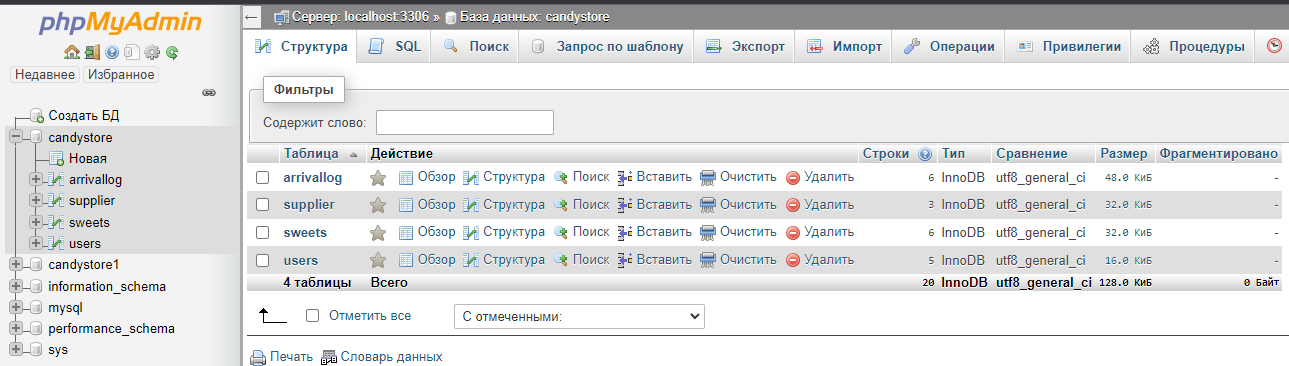


Рисунок 8 – База данных

Таблица sweets которая хранит тип конфеты, единицу измерения и цену (рис. 9):

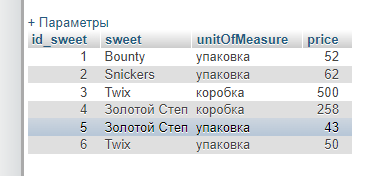


Рисунок 9 – Таблица sweets

Таблица supplier которая хранит название поставщика, телефон и адрес (рис. 10):

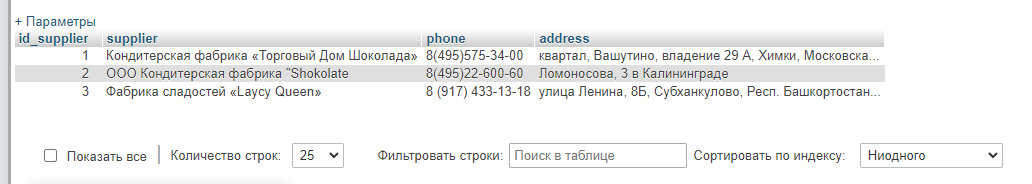


Рисунок 10 – Таблица supplie

Таблица arrivallog которая хранит номер поставки, дату, два внешних ключа и количество (рис. 11):

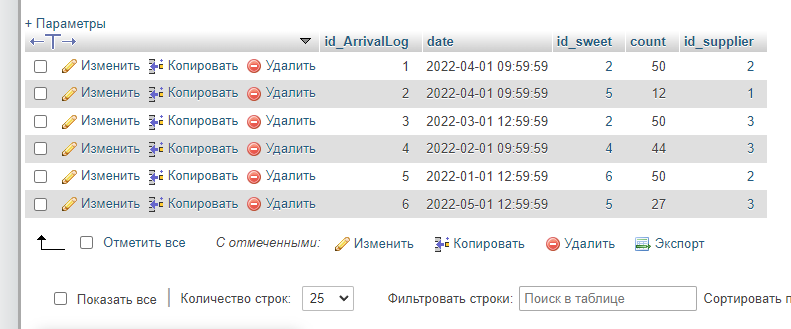


Рисунок 11 – Таблица arrivallog

Создали новый класс DateBase, который хранит подключение к нашей БД, а также имеет 3 метода, для включения и выключения нашей БД, а также для передачи строки подключения (рис. 12):

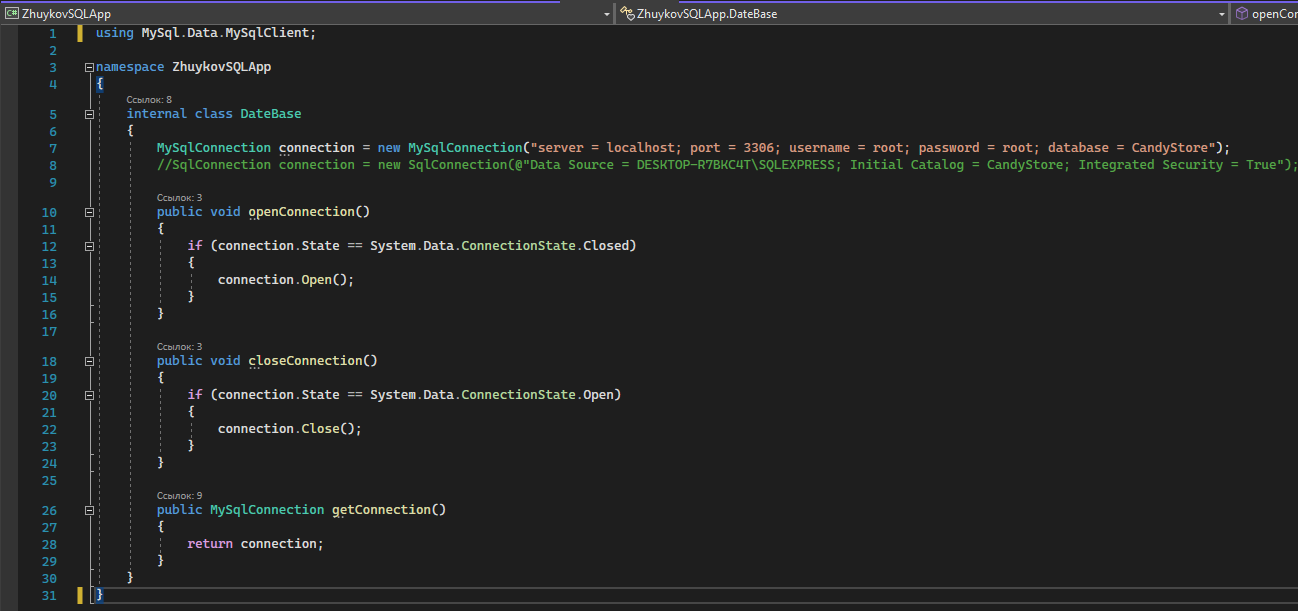


Рисунок 12 – Класс DateBase

**Задание №3. Авторизация пользователя через базу данных**

Написали код, который отправляет запрос при нажатии на кнопку «Войти» и проверяет наличие записи в БД, если запись была найдена, то вход успешный, иначе нет, также добавлено приветственное окно при успешном входе (рис. 13):

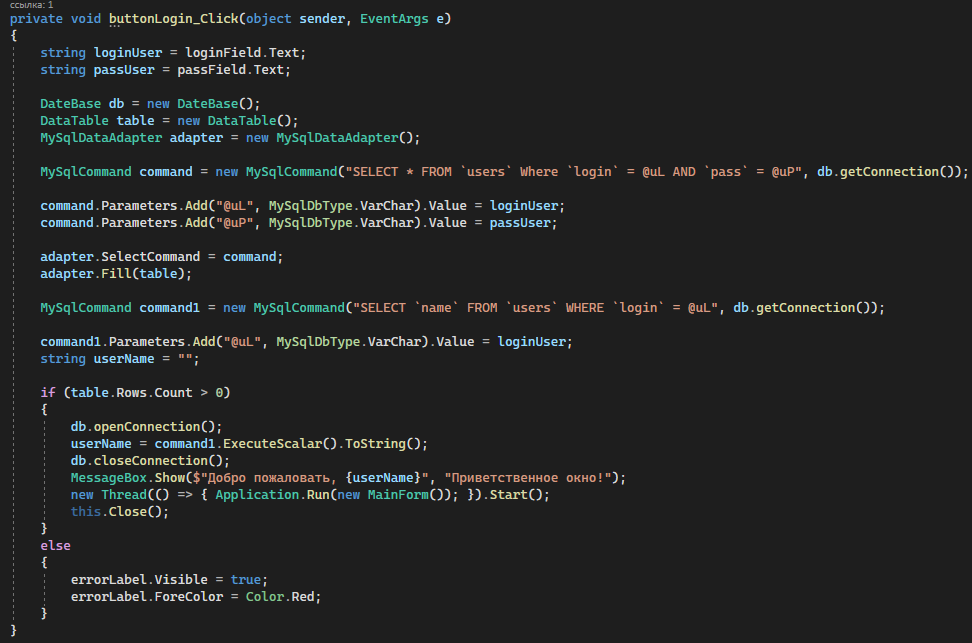
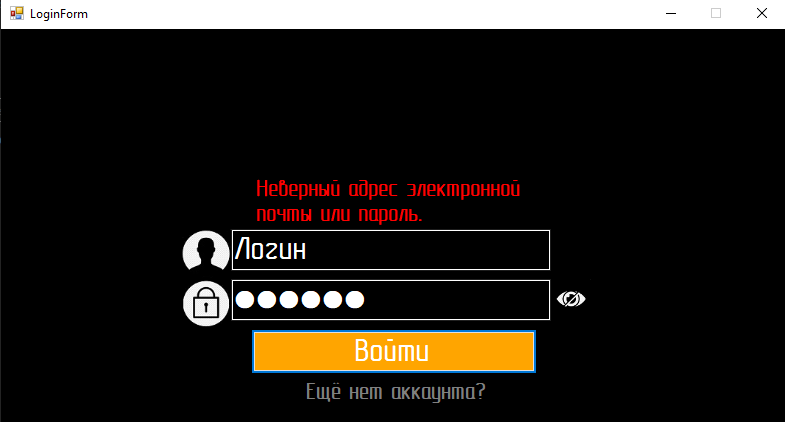
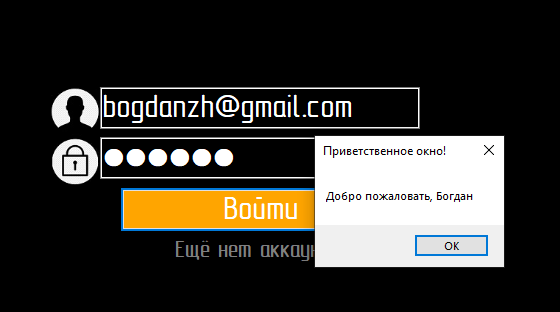


Рисунок 13 – Запросы для проверки наличия пользователя с введёнными данными

Проверка работоспособности написанного кода (рис. 14, а), (рис. 14, б):

а б

Рисунок 14 – Вывод сообщений

**Задание №4. Авторизация пользователя через базу данных**

Создали форму регистрации нового пользователя, добавили поля для ввода имени, фамилии, почты, пароля и подтверждения пароля (рис. 15):

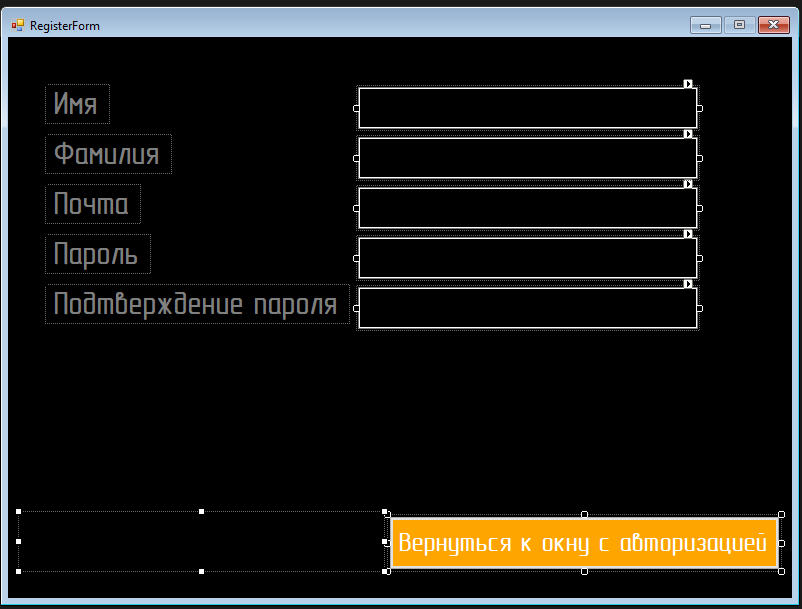


Рисунок 15 – Форма регистрации

Написали код, который проверяет заполнены ли все поля (рис. 16):

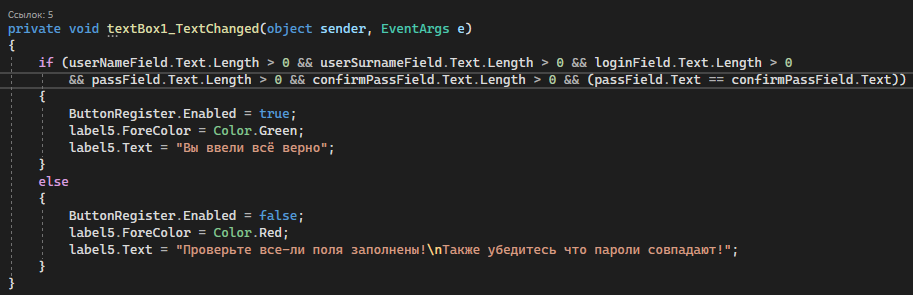


Рисунок 16 – Листинг для текстовых полей

**Задание №5. Регистрация пользователя в базе данных**

Создали событие для всех textbox на форме регистрации, для проверки их заполнения, если хотя бы один не заполнен, то кнопка «Вернуться к окну с авторизацией» будет неактивна (рис. 17):

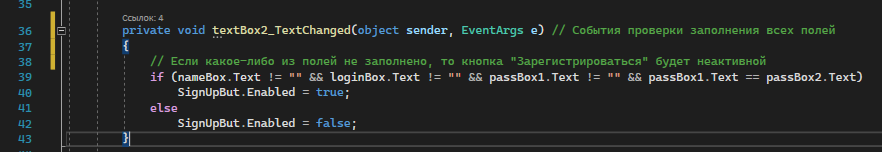


Рисунок 17 – Проверка заполнения всех полей

Написали метод, который проверяет внесённые данные на наличие пользователя с таким логином в базе данных, если он есть, то создание пользователя будет невозможным (рис. 18):

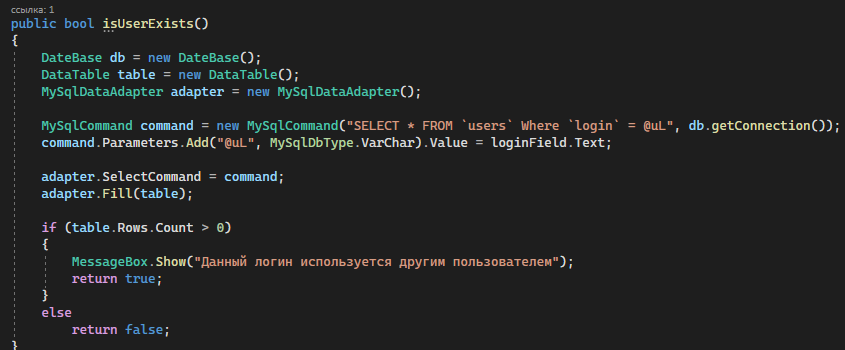


Рисунок 18 – Проверка на наличие пользователя с таким же логином

Написали код для кнопки «Вернуться к окну с авторизацией», код позволяет зарегистрировать нового пользователя, либо выведет сообщение о том, что пользователь с таким логином уже существует (рис.19):



Рисунок 19 – Событие клик для кнопки «Вернуться к окну с авторизацией»

Проверим код на работоспособность (рис. 20):

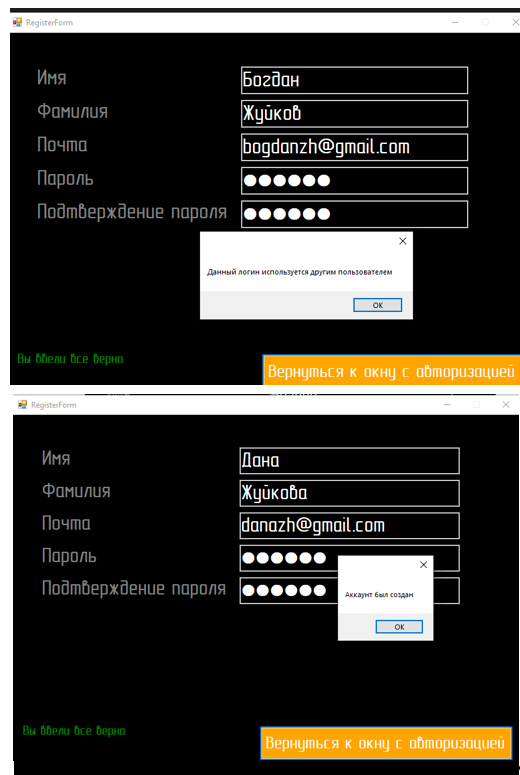


Рисунок 20 – Проверка работоспособности

**Задание №6. Переход между окнами в приложении**

Переход между формой авторизации и регистрации осуществляется нажатием на соответствующую запись «Ещё нет аккаунта?» (рис. 21):

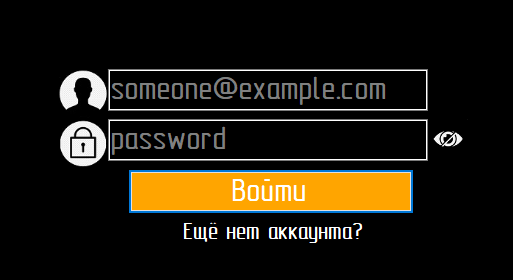


Рисунок 21 – Кнопка для перехода

Чтобы осуществить переход с формы регистрации на форму входа нужно заполнить все поля и нажать «Вернуться к окну с авторизацией» (рис. 22):

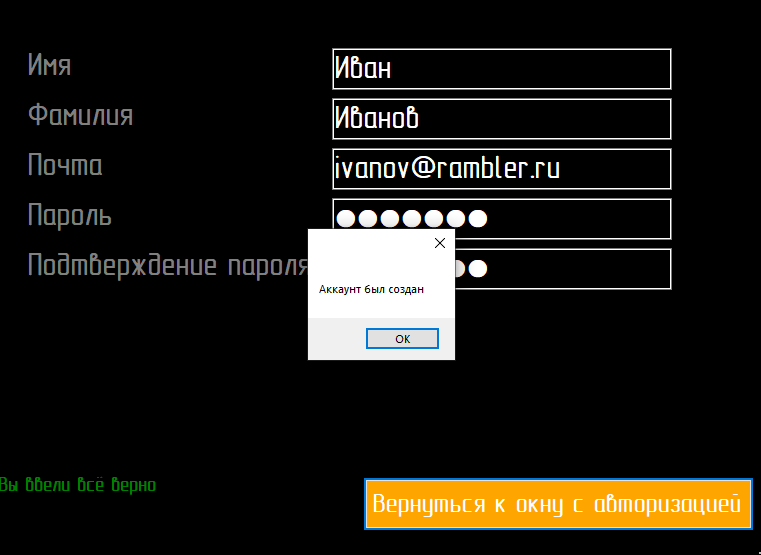


Рисунок 22 – Кнопка для перехода

Чтобы перейти на главную форму, необходимо пройти авторизацию и нажать кнопку «Войти» (рис. 23):

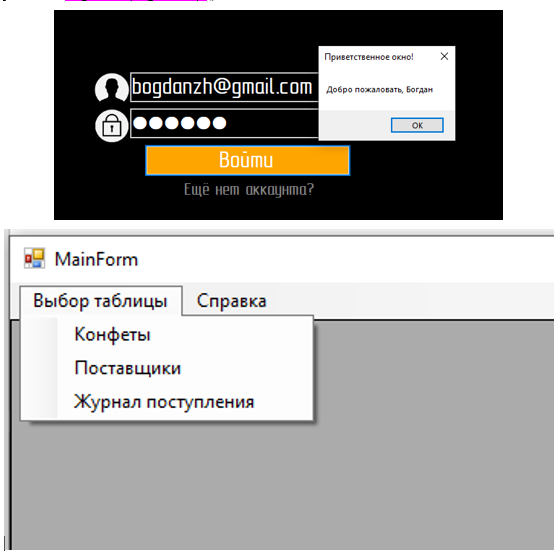


Рисунок 23 – Кнопка и результат нажатия по кнопке

**Задание №7. Контрольное: Создать в БД таблицу, по вашей теме, чтобы она выводилась на форме "Главный экран", после авторизации**

Создали главную форму, добавили компоненты: menuStrip и DataGridView, в нём будет отображаться информация из базы данных, добавили справку, в menuStrip можно выбрать отображаемую таблицу, а также при двойном клике можно просмотреть внешний ключ (рис. 24):

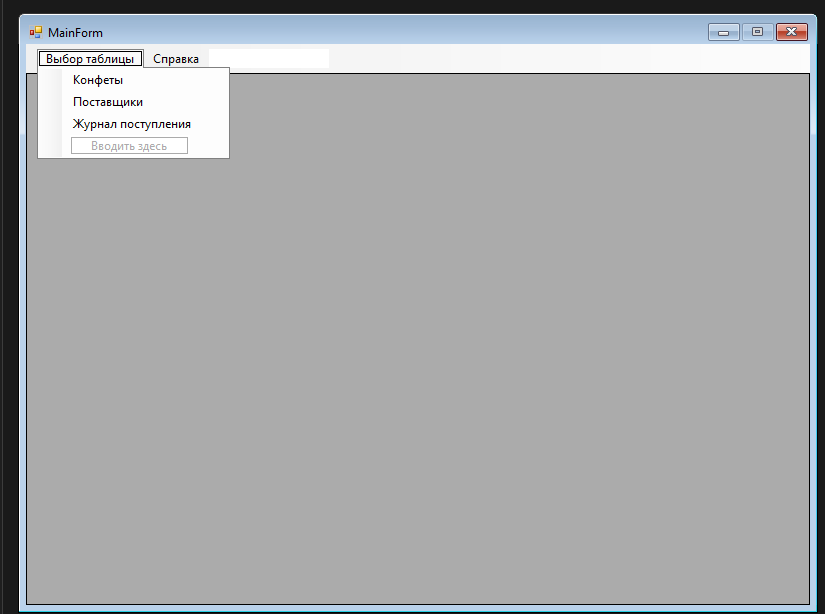


Рисунок 24 – Дизайн главной формы

Написали код для справки (рис. 25):

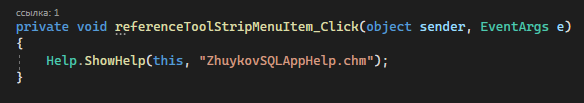


Рисунок 25 – Справка

В контекстном меню выбираем таблицу «Конфеты» (рис. 26):

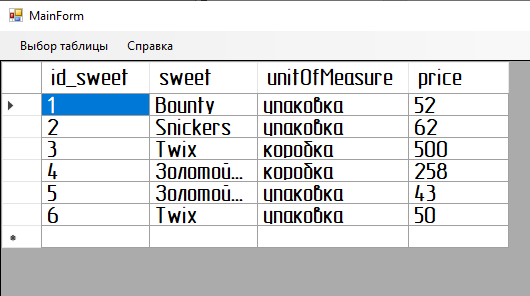


Рисунок 26 – Таблица Конфеты

В контекстном меню выбираем таблицу «Поставщики» (рис. 27):

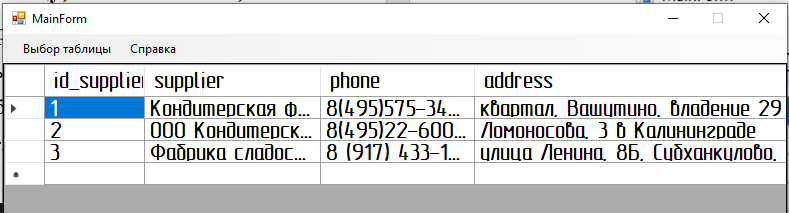


Рисунок 27 – Таблица Поставщики

В контекстном меню выбираем таблицу «Журнал поступления» (рис. 28):

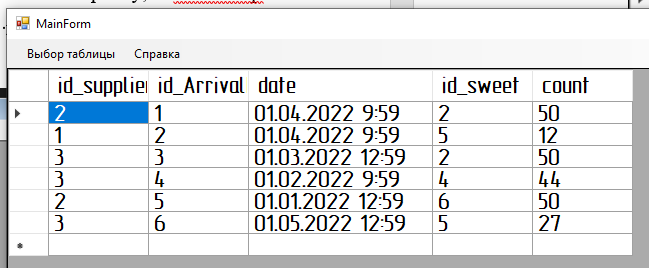


Рисунок 28 – Таблица Журнал поступления

Написали код, который отображает информацию из БД (рис. 31):

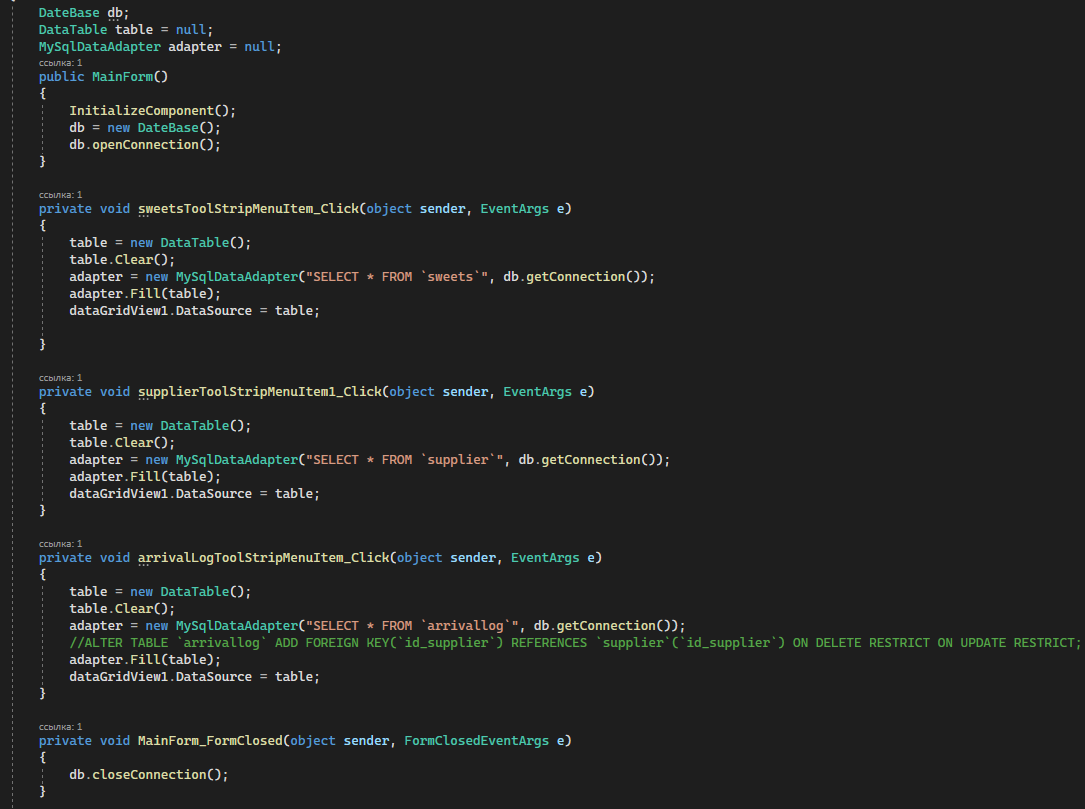


Рисунок 31 – Листинг кнопок

Написали код для внешних ключей (рис. 32):

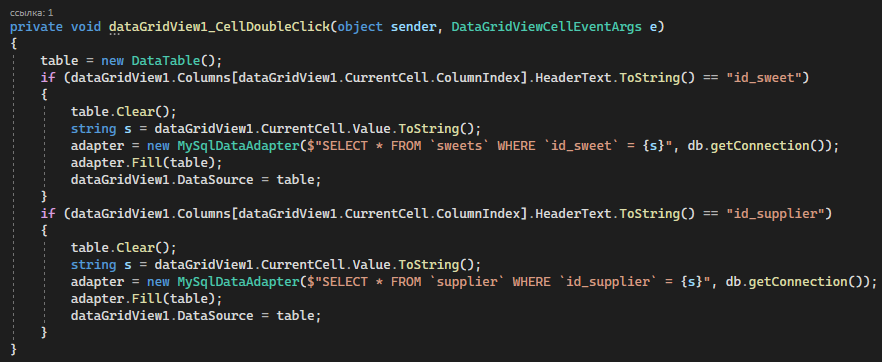


Рисунок 32 – Листинг внешних ключей

**ЗАКЛЮЧЕНИЕ**

На производстве в последнее время считается актуальным возможность упрощения обработки определённых данных. С этой наитруднейшей задачей успешно справляются табличные процессоры.

Приложение было выполнено в срок, оно имеет все заявленные в Введении функции, работает стабильно и без багов, была реализована функция авторизации пользователя, его регистрации, создана возможность подключения к базе данных на локальном сервере. На главной форме реализована возможность просмотра таблиц из базы данных, а также переключение между ними.

Во время выполнения работы я узнал много новой полезной информации, то как работать с GITHUB и как публиковать там свои проекты, научился создавать несложные SQL запросы для БД и соединять их в дальнейшем с программой. Ознакомился с локальным сервером MAMP, в котором была создана сама БД и осуществлялся поток с сервера в программу.

Данная работа оказалась не сложной, было получено очень много хорошей и полезной информацией, что позволит выполнять последующие проекты ещё лучше и качественнее.

Осмыслили сущность и социальную значимость своей будущей профессии, проявили к ней устойчивый интерес.

Научились использовать информационно-коммуникационные технологии в профессиональной деятельности.

Научились самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

Научились осуществлять оптимизацию программного кода модуля, осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Metanit: сайт посвящен различным языкам и технологиям программирования, компьютерам, мобильным платформам и ИТ-технологиям. – Москва, 2012. – URL: https://metanit.com/sharp/general.php (дата обращения: 04.04.2022)
2. Понамарев, В. Программирование на C++/C# в Visual Studio .NET 2003 / В. Понамарев. - М.: БХВ-Петербург, 2015. – 917 c
3. Документация по C# – 2016 – URL: https://docs.microsoft.com/ru-ru/dotnet/csharp/ (дата обращения: 06.04.2022)
4. Мюллер, Джон Пол, Семnф, Билл, Сфер, Чак. М98 С# для чайников.: Пер. с англ. - СПб.: ООО "Диалектика", 2019. - 608 с.: ил. - Парал. тит. анr.11.