1. Создание проекта

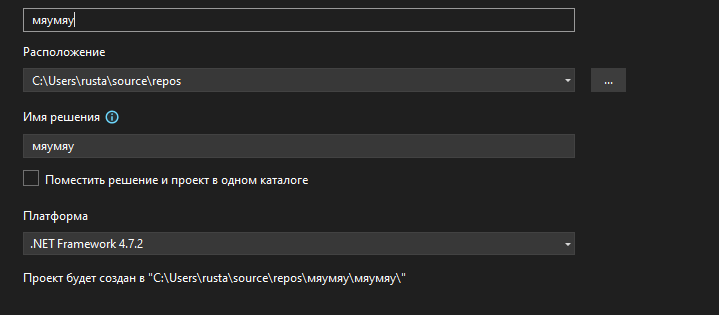


Рисунок - Создание проекта

1. Созданный проект

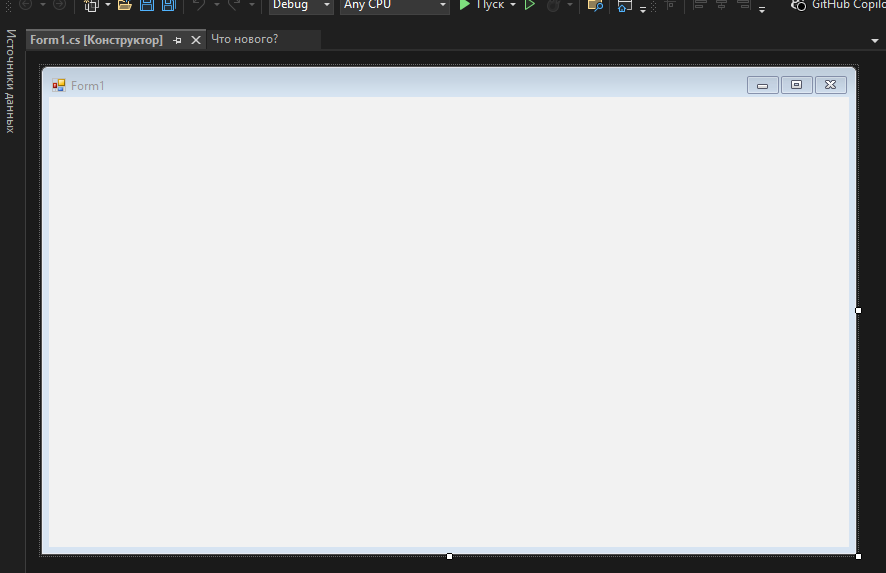


Рисунок - Созданный проект

1. Удаление Form1

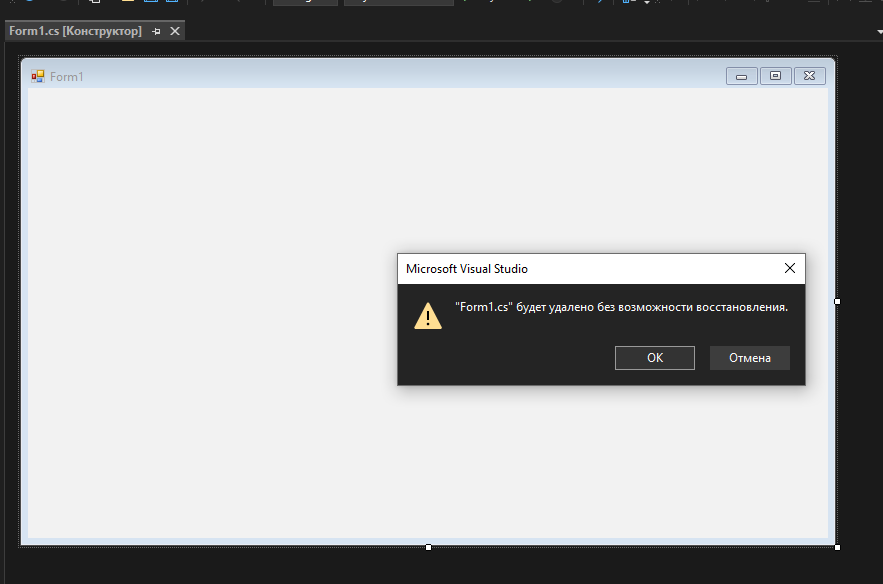


Рисунок - Удаление формы

1. Создание LoginForm

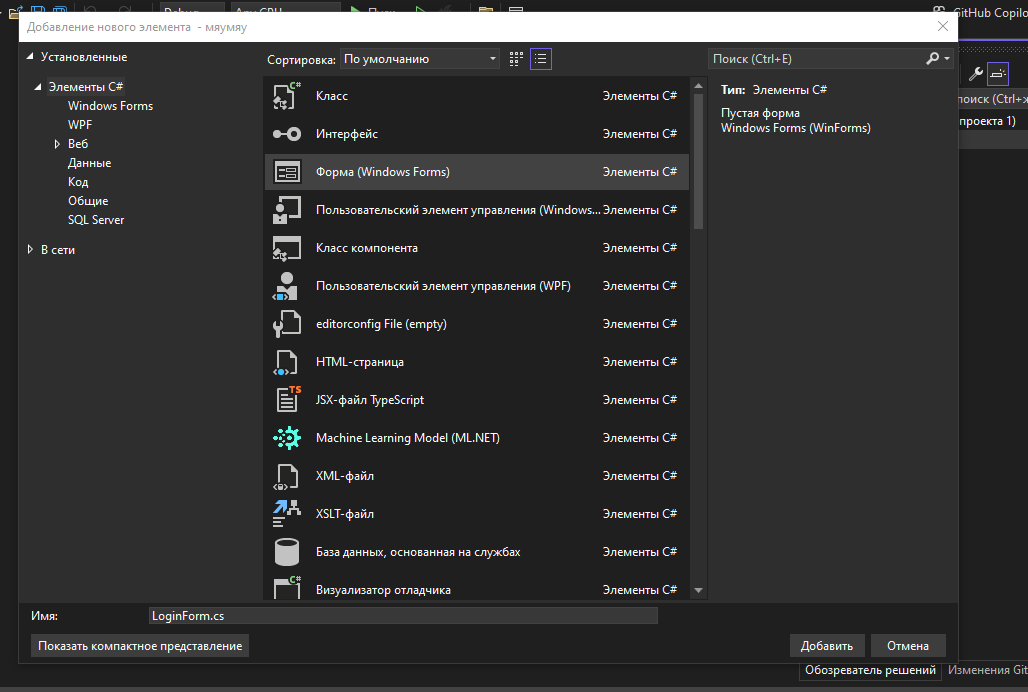


Рисунок - Создание формы

1. Открываем панель элементов

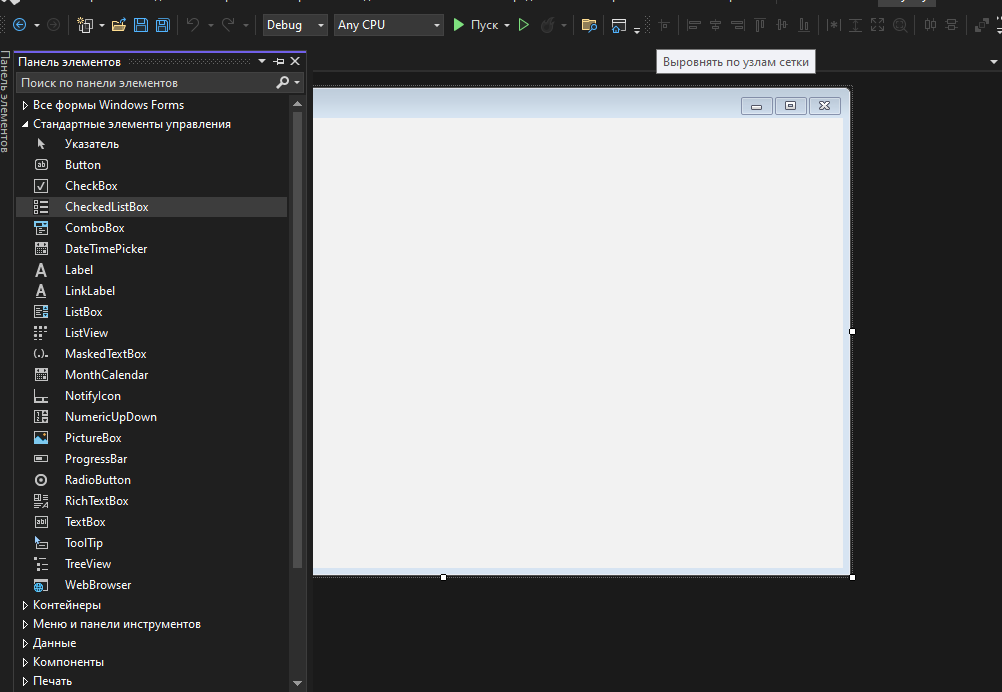


Рисунок - Панель элементов

1. Выбираем в панели элементов элемент panel

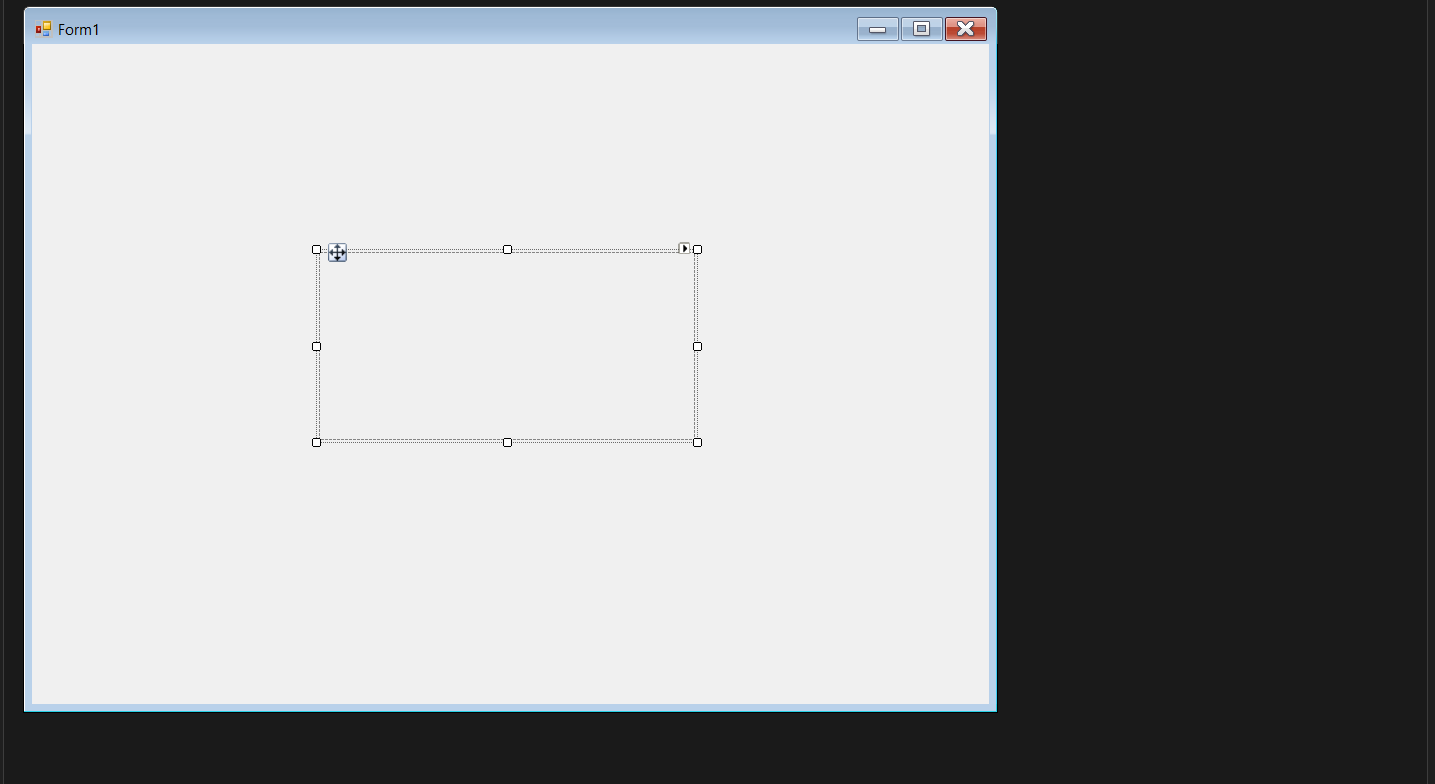


Рисунок - Элемент panel

1. Закрепляем panel в родительском контейнере

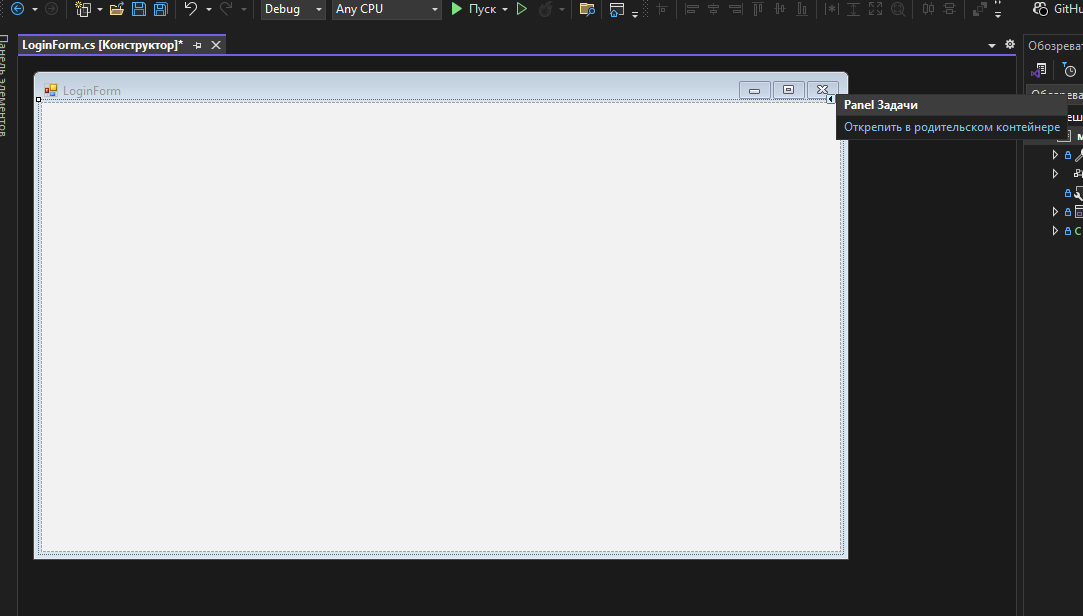


Рисунок – Закрепление panel

1. Меняем значения свойства BackColor элемента Panel на 17;24;34

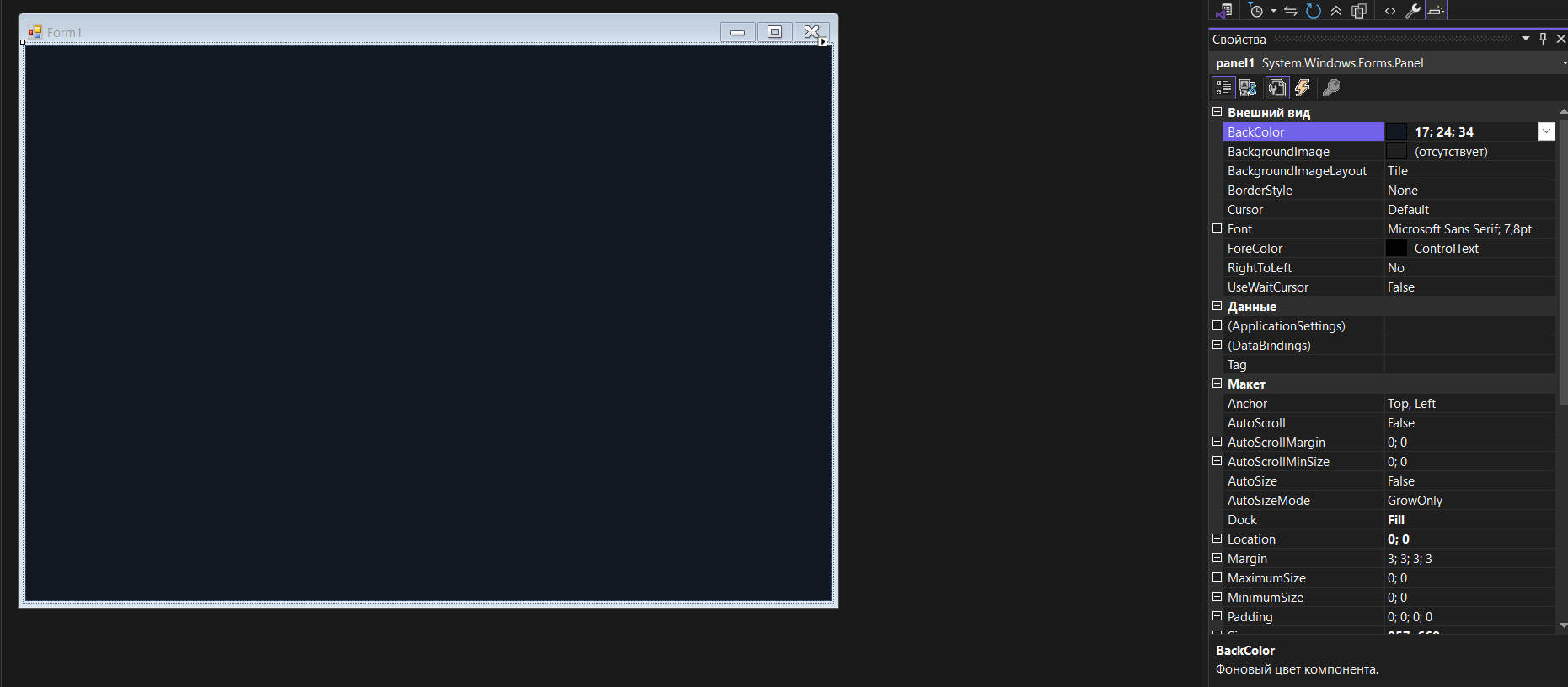


Рисунок - Цвет панели

1. Создаем новую панель и располагаем ее сверху формы по ширине

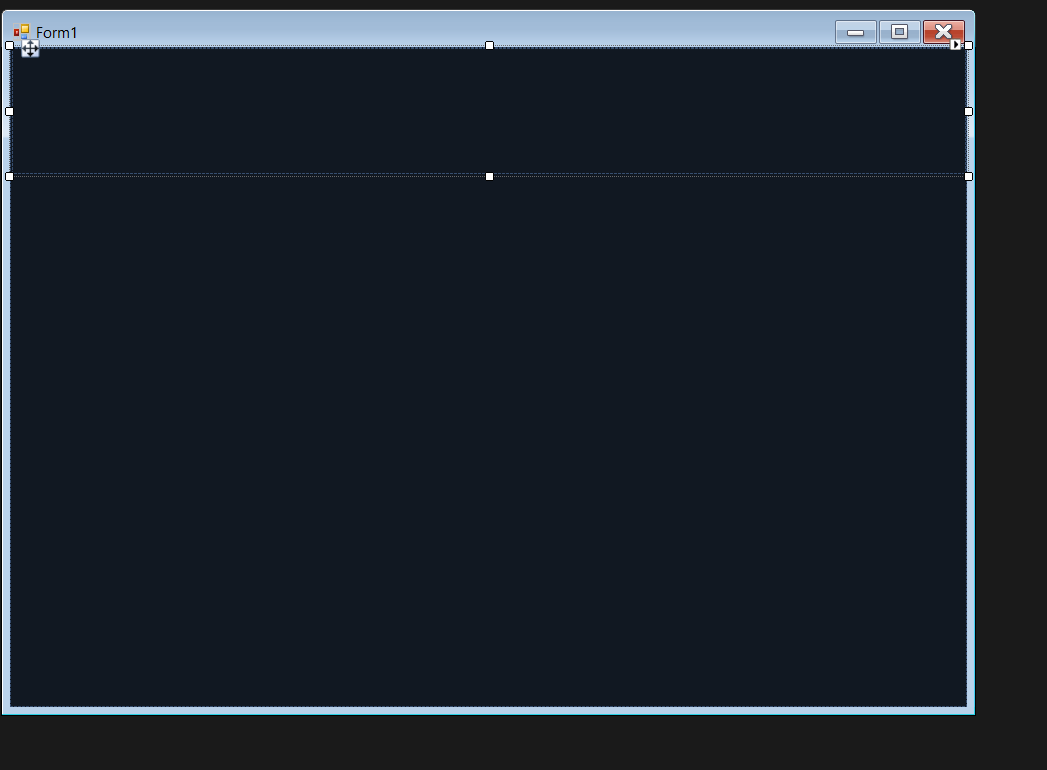


Рисунок - Создание второй панели

1. Меняем значения свойства BackColor элемента Panel на 255;183;19

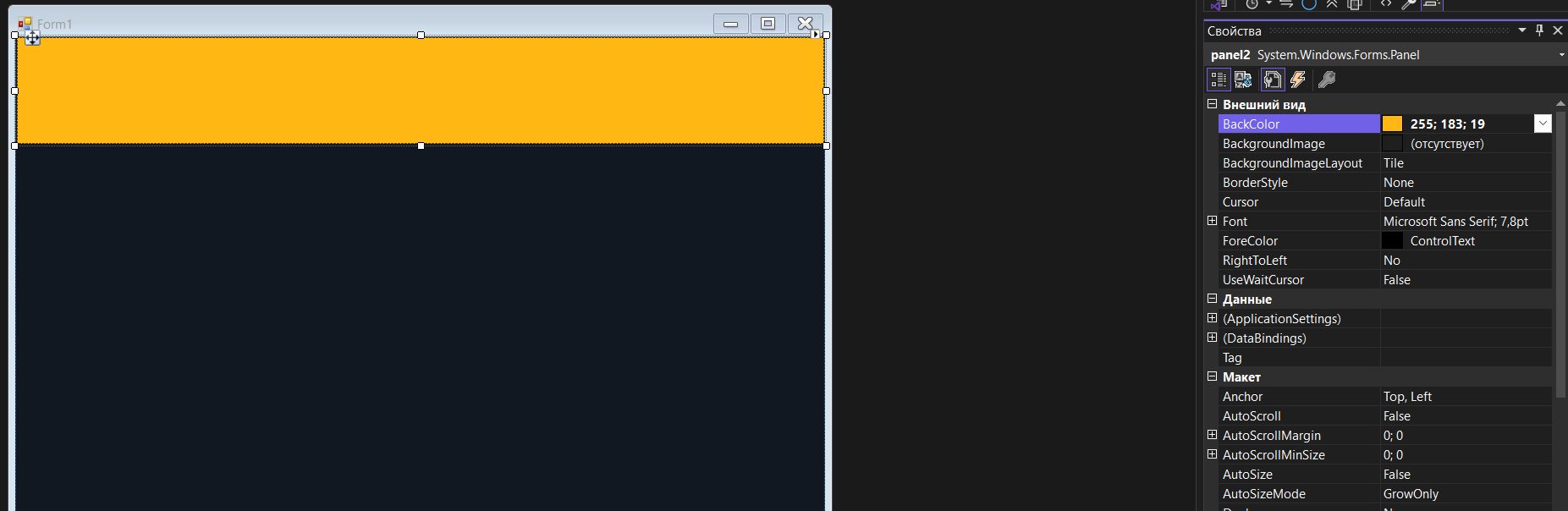


Рисунок - Цвет панели

1. Создаем папку с изображениями под названием image, копируем изображения и вставляем в эту папку.

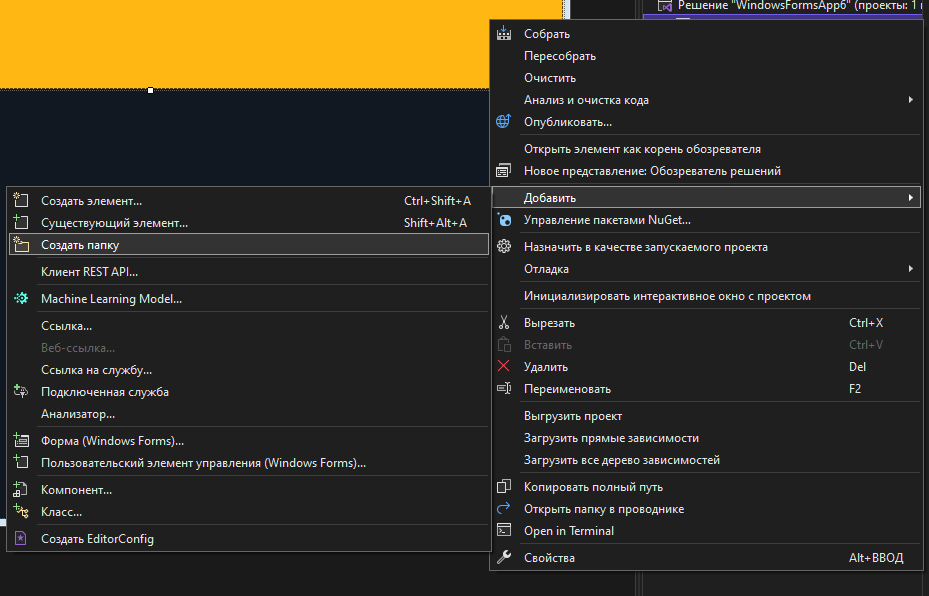


Рисунок - Создание папки

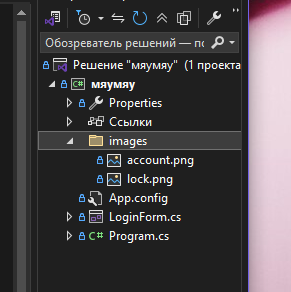


Рисунок - Созданная папка

1. Создаем label и настраиваем шрифт и текст

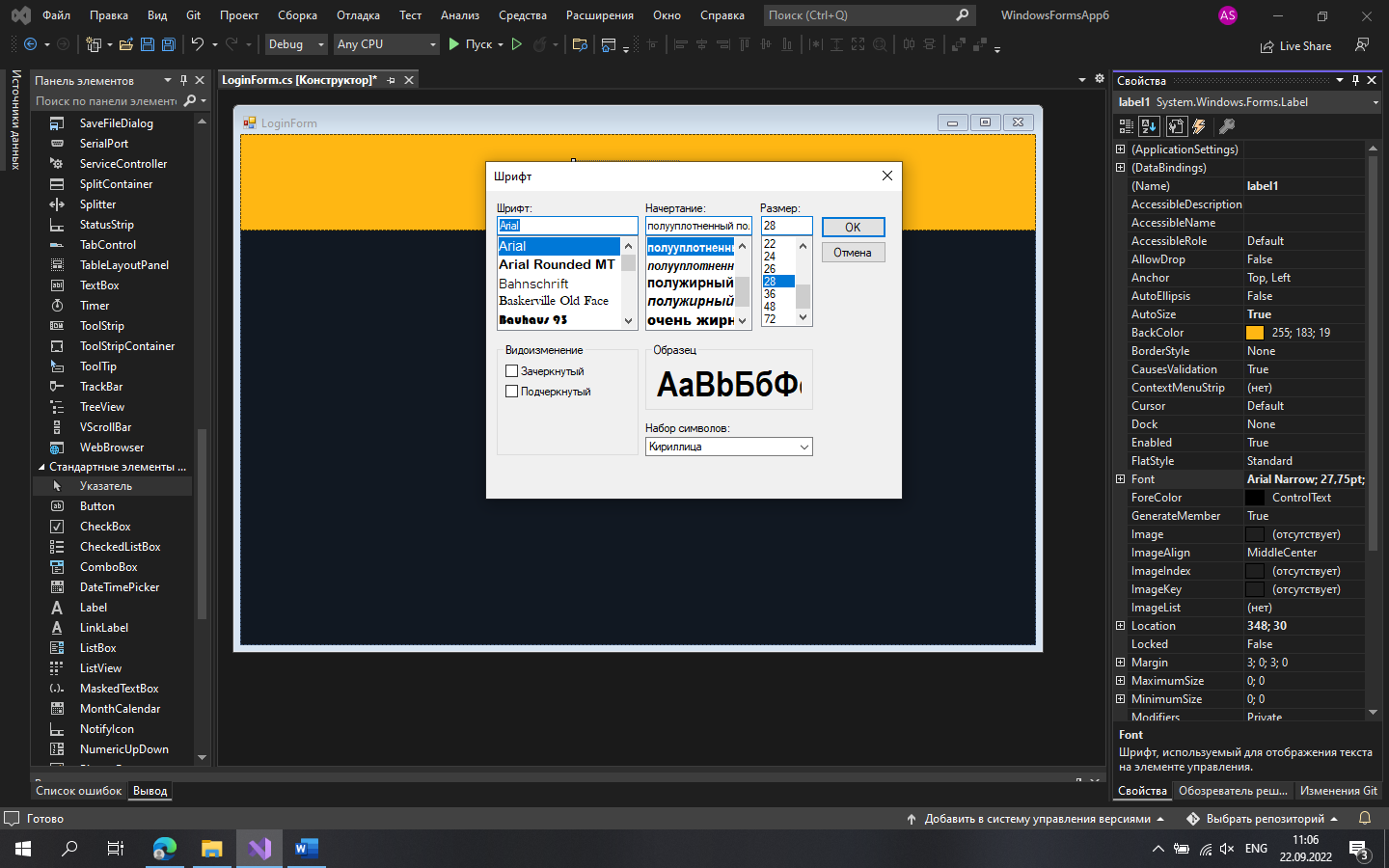


Рисунок - Настройки шрифта

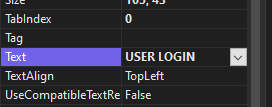


Рисунок - Текст label

1. Меняем свойство dock на значение fill, и textAlign на MiddleCenter

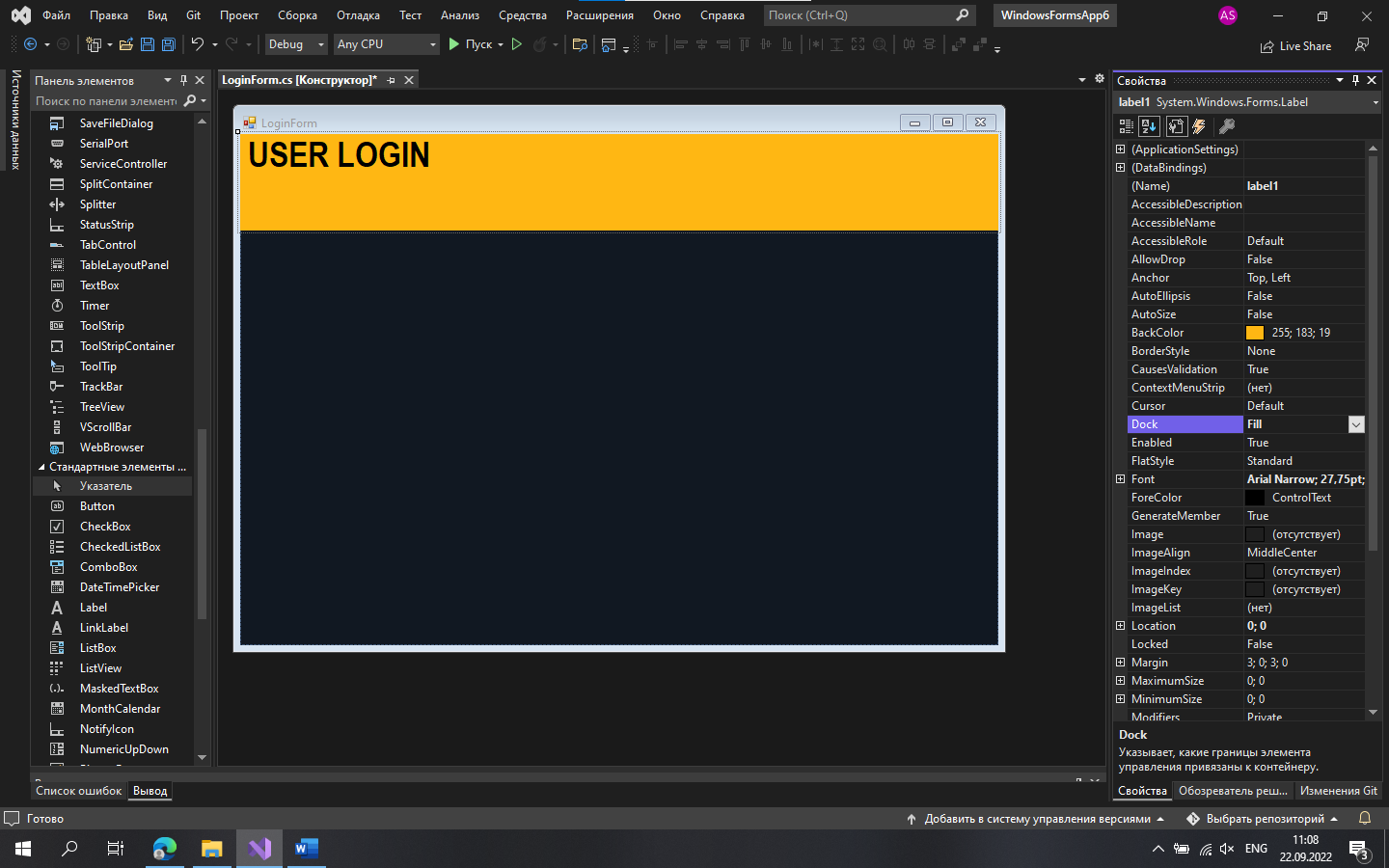


Рисунок - Свойство Dock

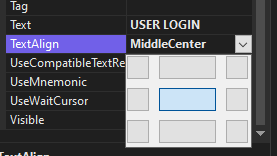


Рисунок - Свойство TextAlign

1. Создаем еще один label, который будет кнопкой закрытия приложения, меняем текст на х и имя на labelClose, а так же настраиваем шрфит

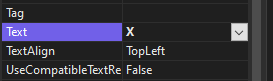


Рисунок - Текст lable



Рисунок - Имя label

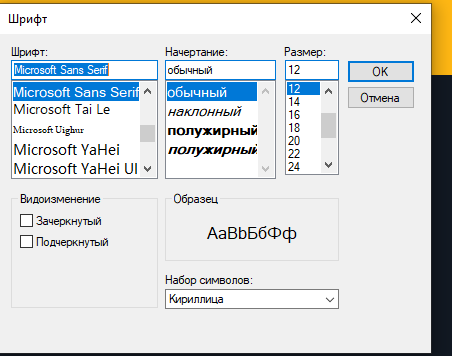


Рисунок - Настройки шрифта

1. Заходим в Program.cs и меняем Form1 на LoginForm, чтобы при включении приложения загружалась LoginForm

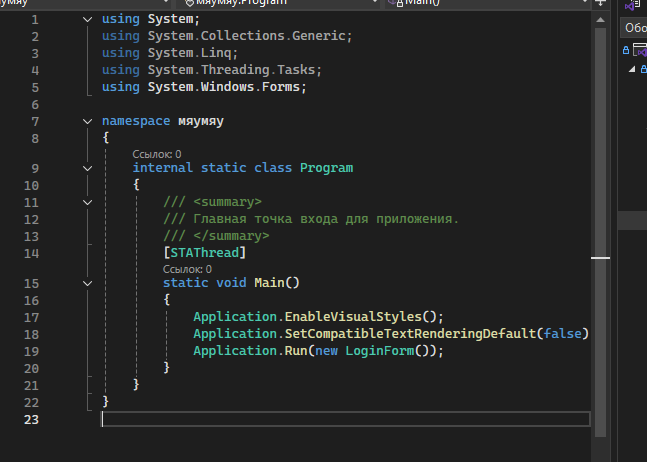


Рисунок - Program.cs

1. Промежуточный результат

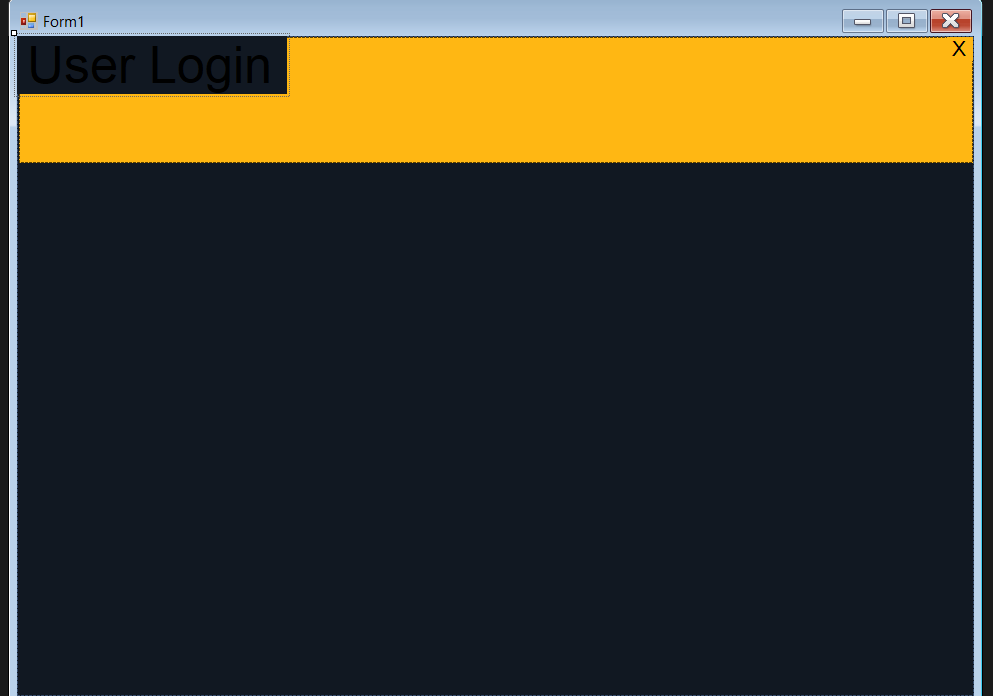


Рисунок - Промежуточный результат

1. Заходим на панель элементов и создаем pictureBox, задаем ему размер и загружаем в него изображение из папки image

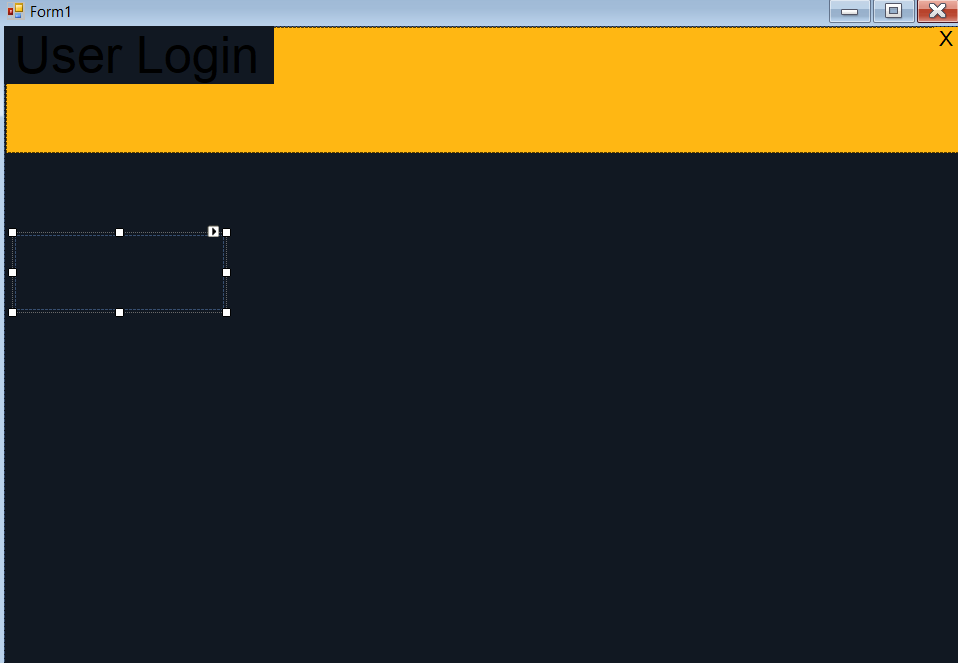


Рисунок - pictureBox

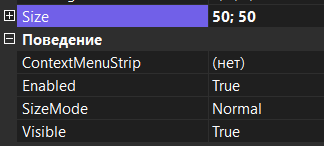


Рисунок - Размер pictureBox

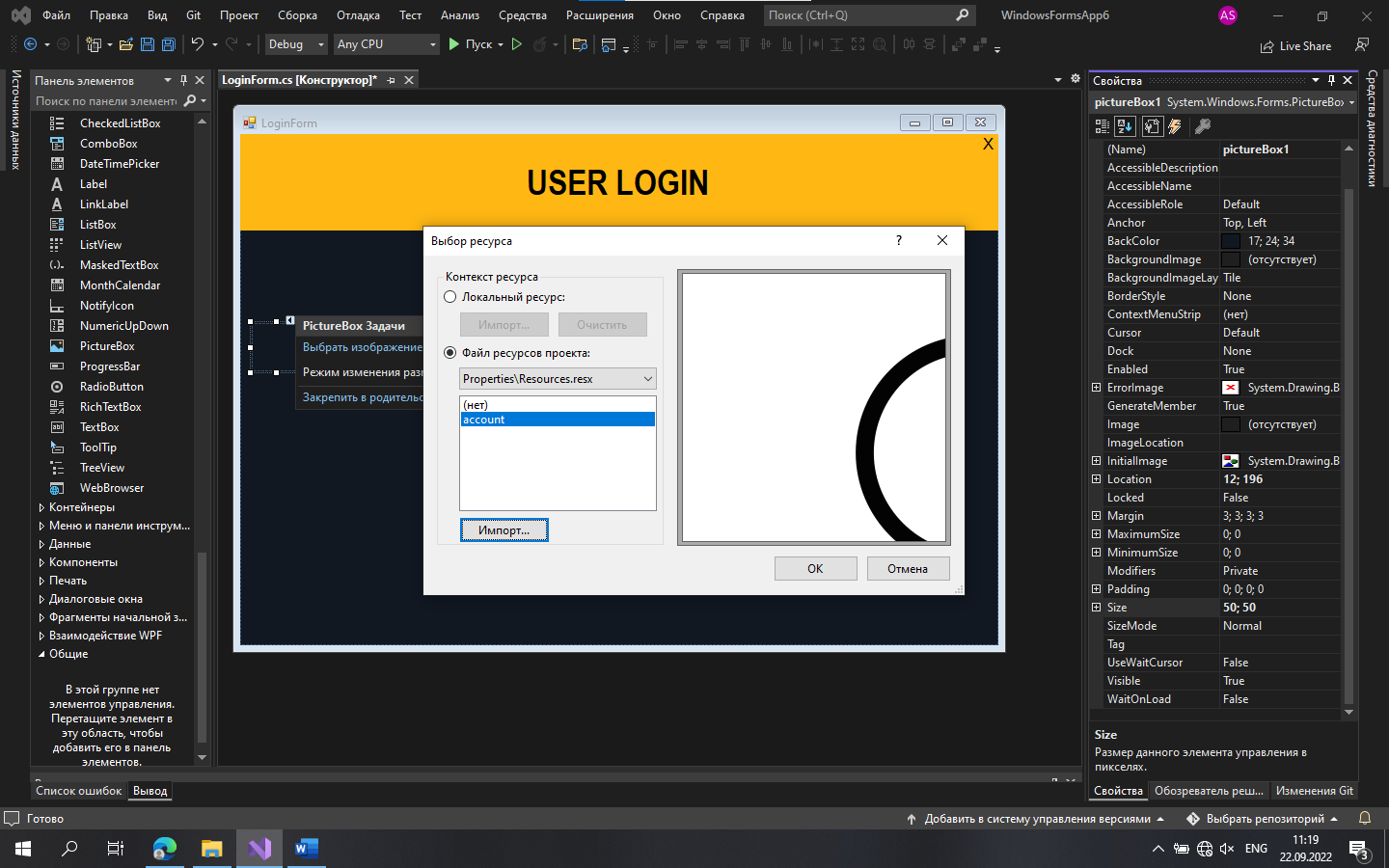


Рисунок - Выбор изображения для pictureBox

1. Готовый pictureBox



Рисунок - Готовый pictureBox

1. Создаем textbox, в свойстве MultiLine ставим галочку и задаем размеры

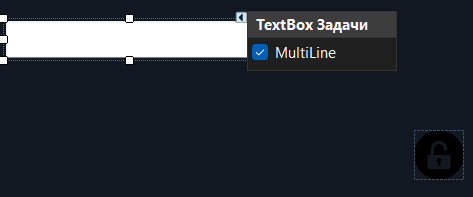


Рисунок - Свойство MultiLine



Рисунок - Размеры textbox

1. Копируем сделанный pictureBox и textbox и меняем изображение в pictureBox на замок

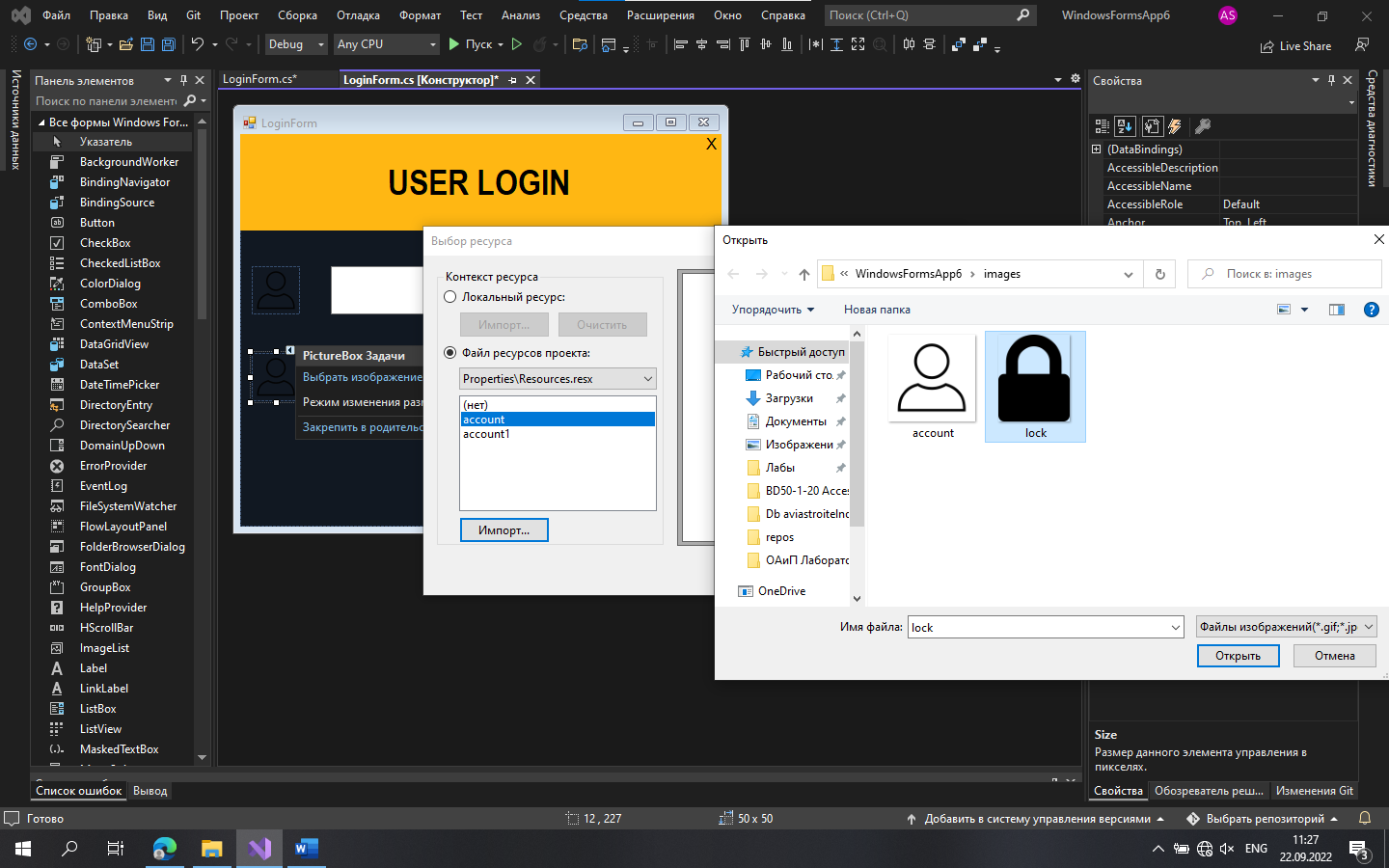


Рисунок - Выбор изображения для pictureBox

1. Промежуточный результат



Рисунок - Промежуточный результат

1. Создаем кнопку и настраиваем следующие свойства: BackColor и FlatStyle



Рисунок - Кнопка



Рисунок - Цвет кнопки



Рисунок - Свойство FlatStyle

1. Настраиваем шрифт кнопки и цвет текста

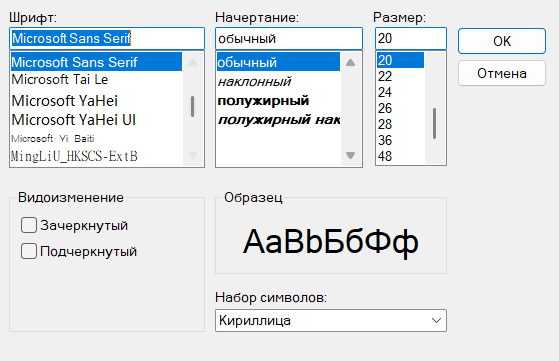


Рисунок - Настройки шрифта



Рисунок - Цвет текта

1. Убираем контур у кнопки



Рисунок - Контур кнопки

1. У текстбоксов настраиваем шрифты

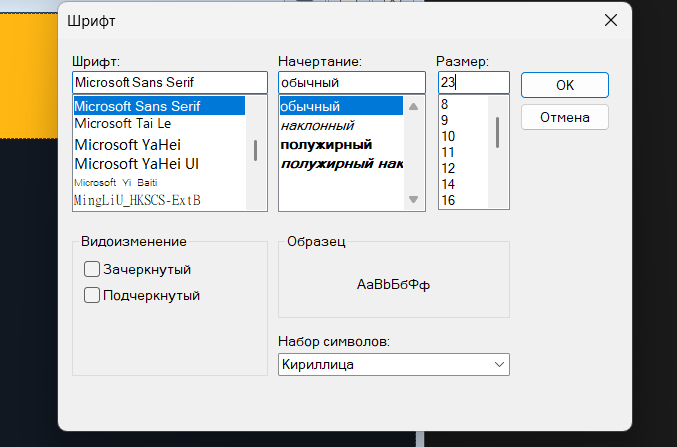


Рисунок - Настройка шрифта

1. Далее переименовываем элементы



Рисунок - Название textboxName



Рисунок - Название textBoxPassword



Рисунок - Название buttonLogin

1. Заходим в настройки textboxPassword и совйство UseSystemPassword меняем на true и в свойстве MultiLine убираем галочку



Рисунок - Свойство UseSystemPassword

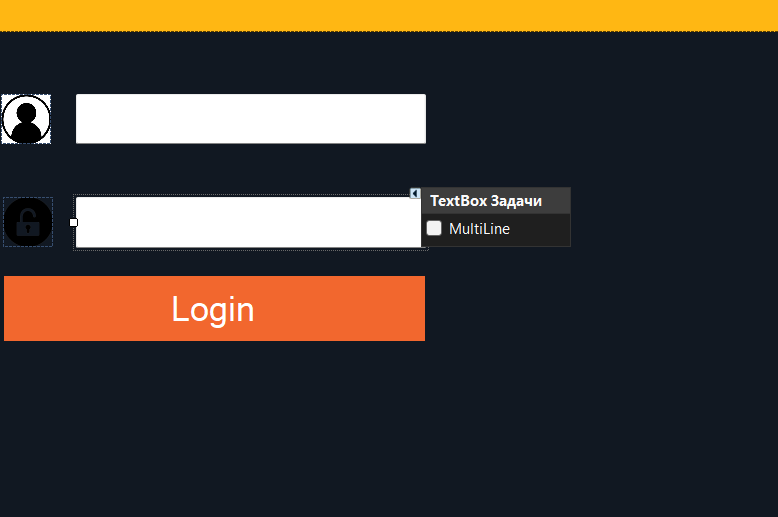


Рисунок - Свойство MultiLine

1. Промежуточный результат

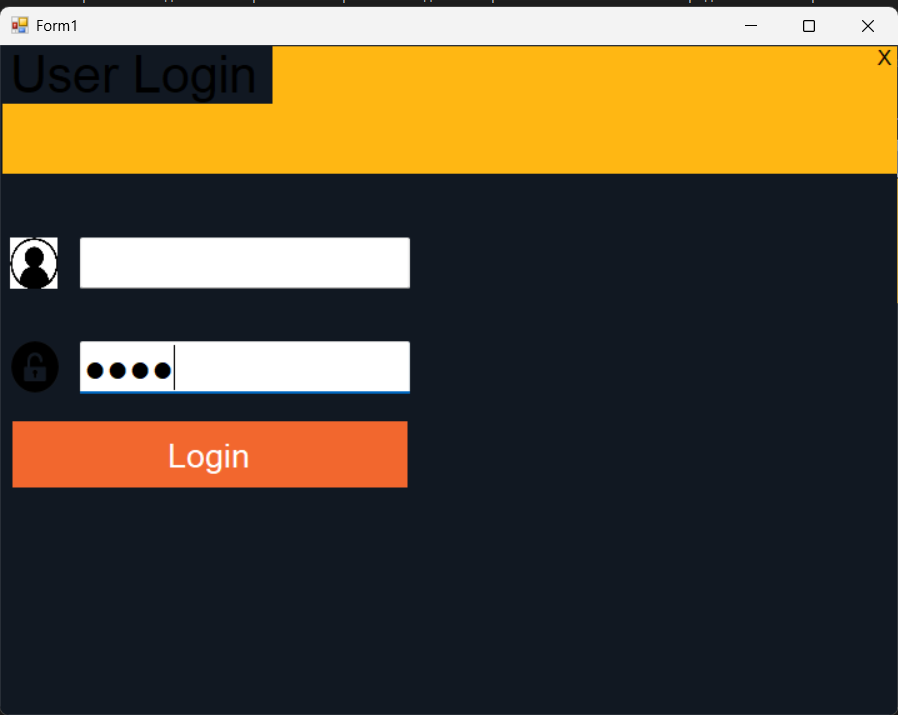


Рисунок - Промежуточный результат

1. Переходим к коду, задаем textBoxPassword размер

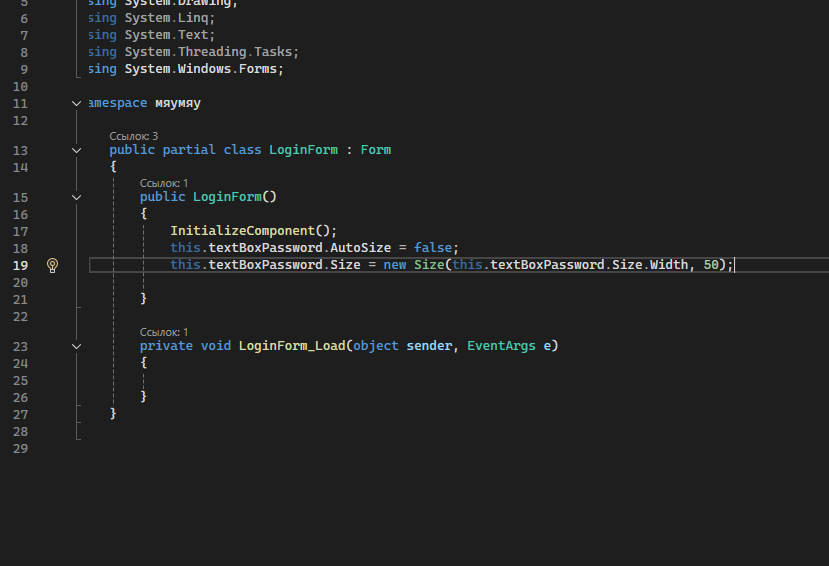


Рисунок - Размер textbox

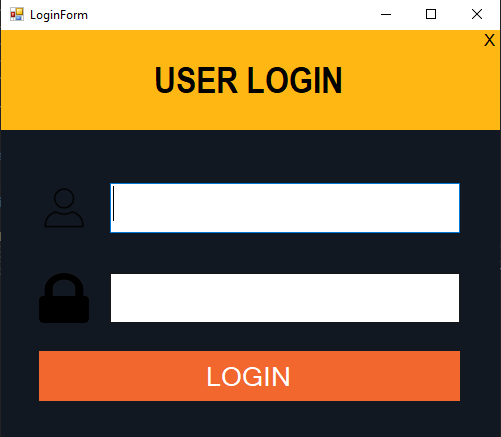


Рисунок - Промежуточный результат

1. Заходим в свойства LoginForm и настраиваем, чтобы форма открывалась по середине экрана и у нее не было конутра



Рисунок - Свойство CenterScreen



Рисунок - Свойство FormBorderStyle

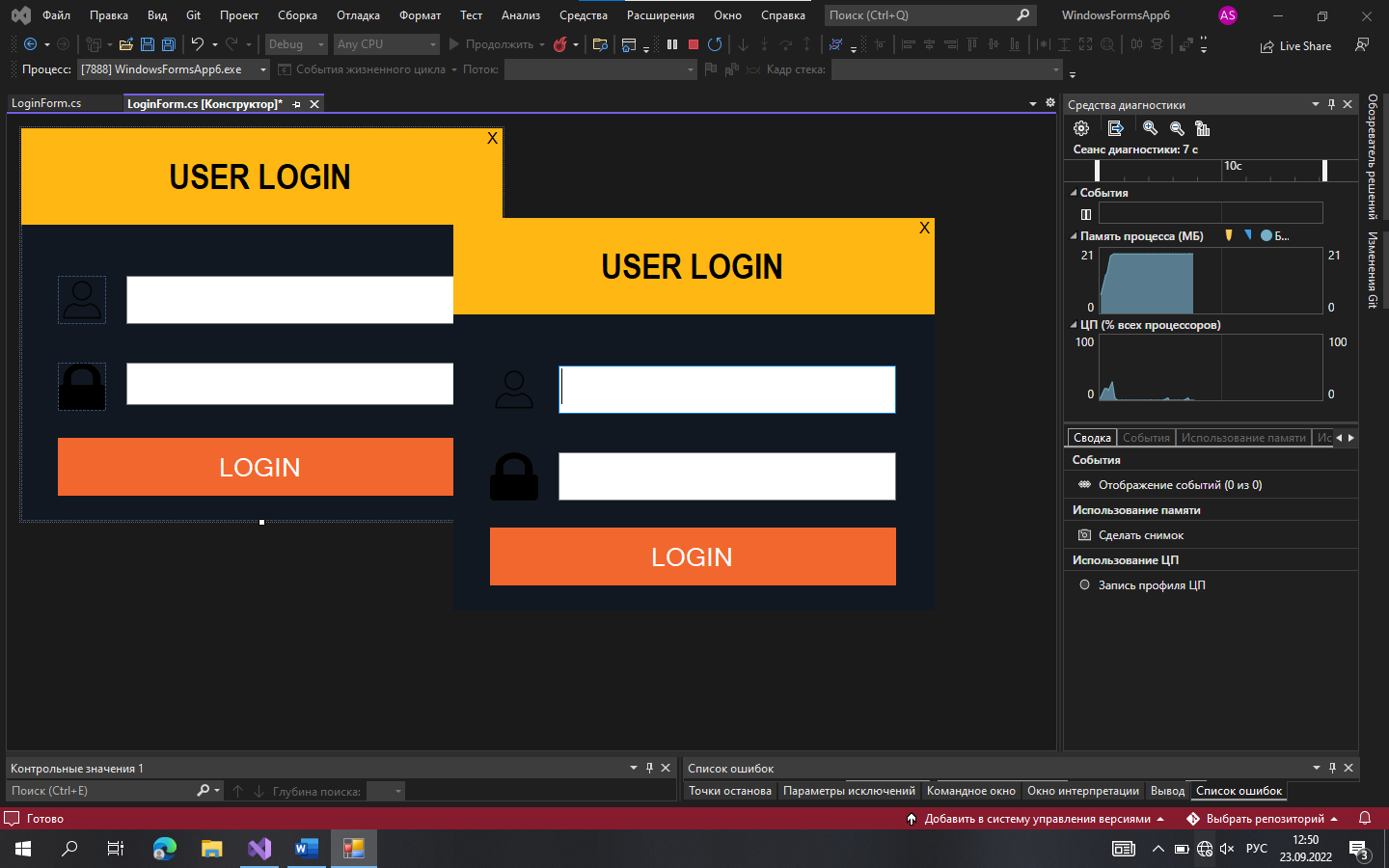


Рисунок - Промежуточный результат

1. Заходим в настройки свойств labelClose и настраиваем цвет и курсор, шрифт



Рисунок - Свойство Cursor



Рисунок - Цвет

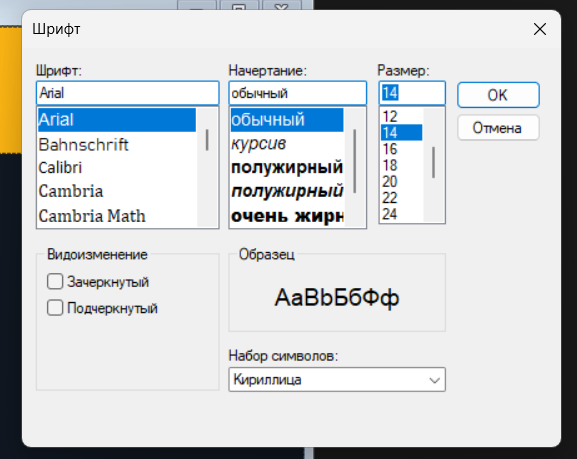


Рисунок - Настройка шрифта

1. Заходим в события labelClose и находим MouseEnter и MouseLeave и задаем цвета, чтобы label был белым, а при наведении курсора черным

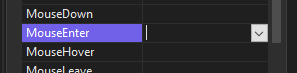


Рисунок - Событие MouseEnter

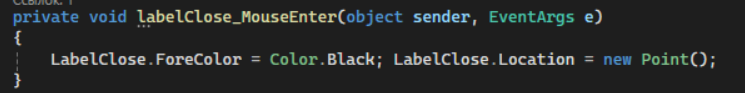


Рисунок - Код события MouseEnter



Рисунок - Событие MouseLeave

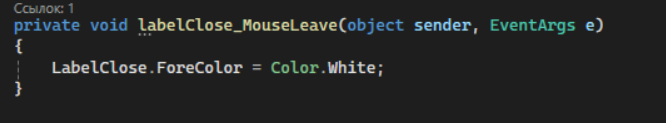


Рисунок – Код события MouseLeave

1. Готовая форма

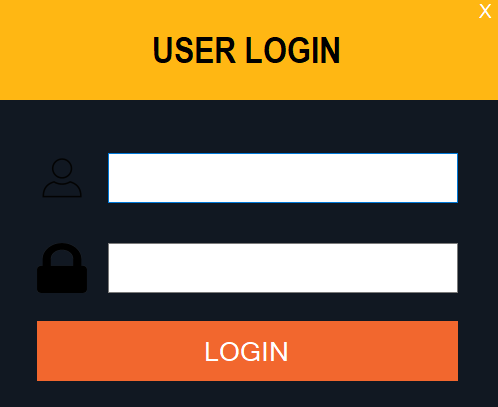


Рисунок -Готовая форма

1. Форма при наведении курсора на кнопку закрыть

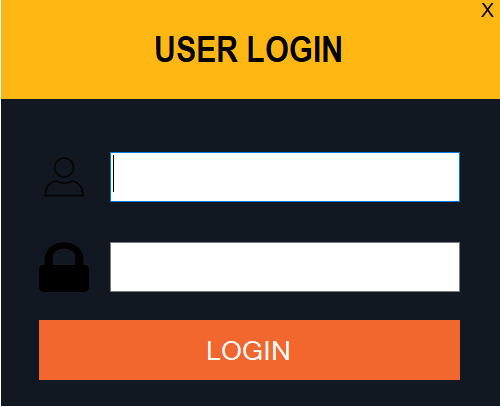


Рисунок - Форма при наведении курсора на кнопку закрыть

1. Заходим в события label, который у нас выполняет роль выхода из приложения (крестик), и пишем событие, чтобы при нажатии на него, приложение закрылось

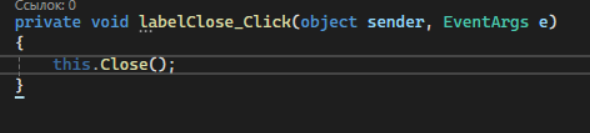


Рисунок 57 - Функция labelClose\_Click

1. Заходим на сайт, представленный на скрине и скачиваем mysql connector

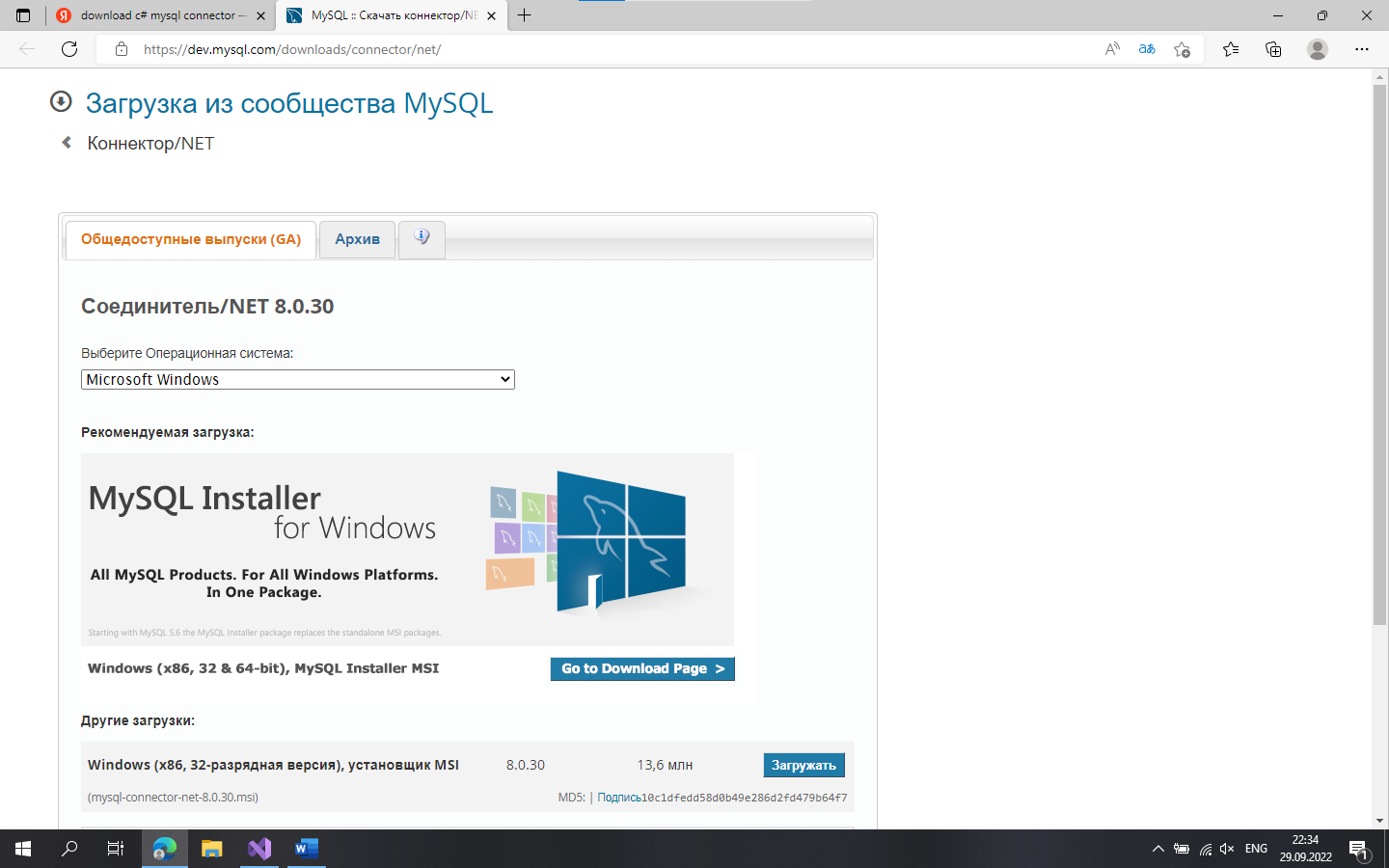


Рисунок 58 - Сайт mysql connector

1. Далее необходимо выполнить установку

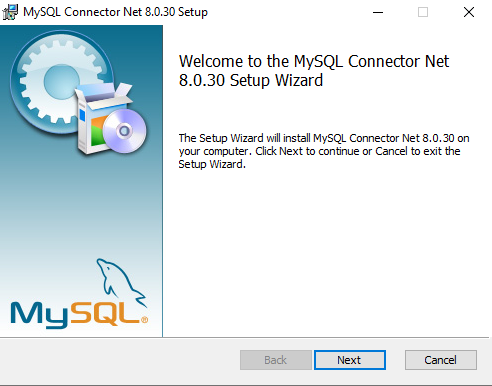


Рисунок 59 - Установка

1. Заходим в менеджер ссылок проекта и добавляем MySql.Data

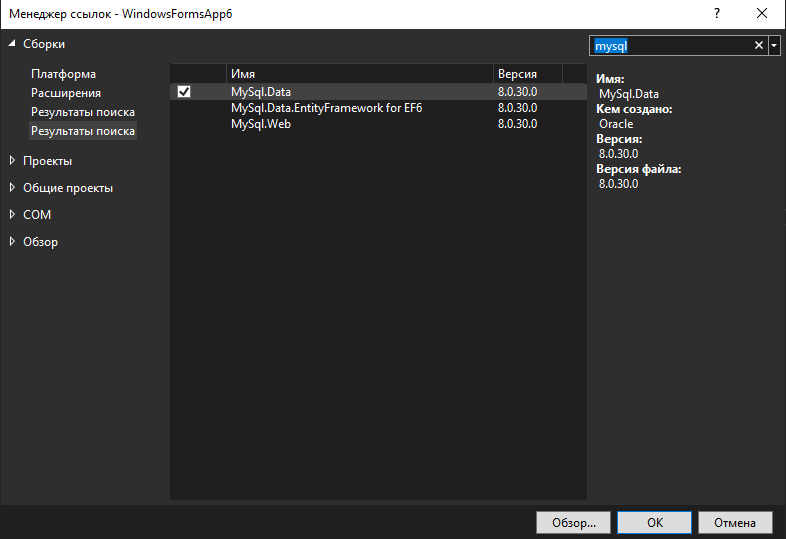


Рисунок 59 - Менеджер ссылок

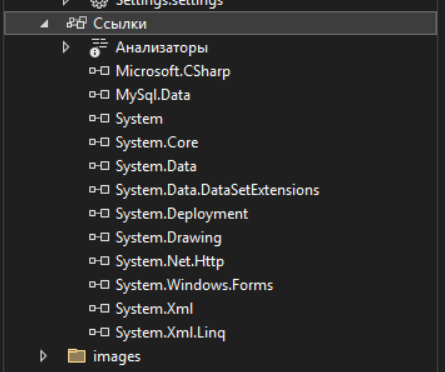


Рисунок 60 - Добавленные ссылки

1. Создаем подключение и в пространство имен подключаем MySql.Data.MySqlClient



Рисунок 61 - Пространство имен

1. Запускаем сервер и заходим в PhpMyAdmin

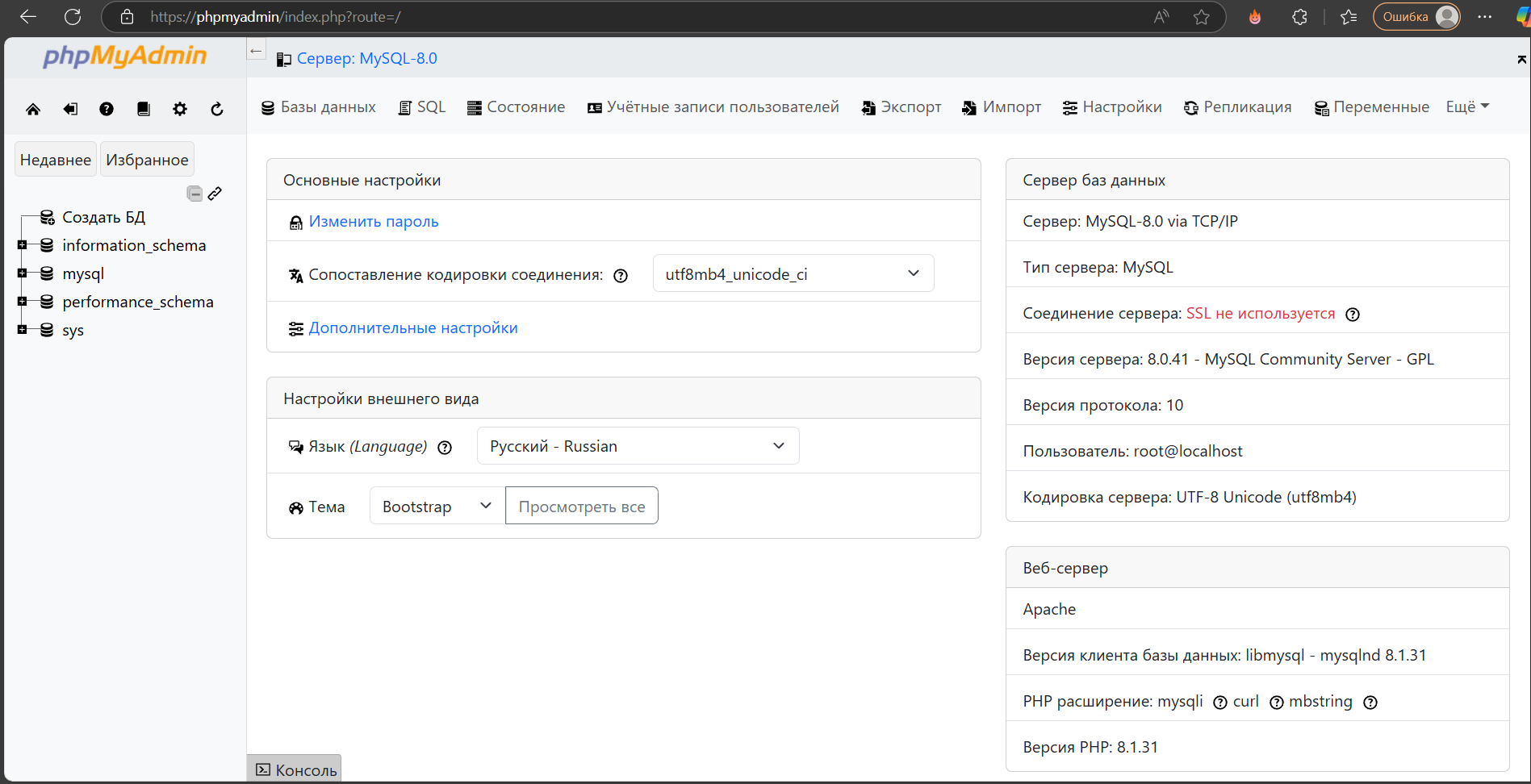


Рисунок 62 - PhpMyAdmin

1. Создаем новую бд под названием csharp\_users\_db

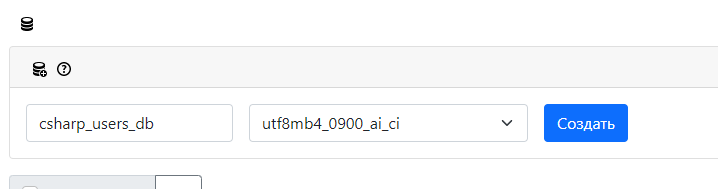


Рисунок 63 - Создание бд

1. Создаем новую таблицу, указывая имя таблицы и количество столбцов

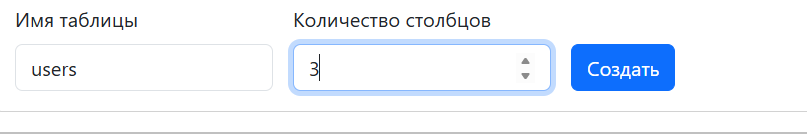


Рисунок 64 - Создание таблицы

1. Заполняем в таблице атрибуты

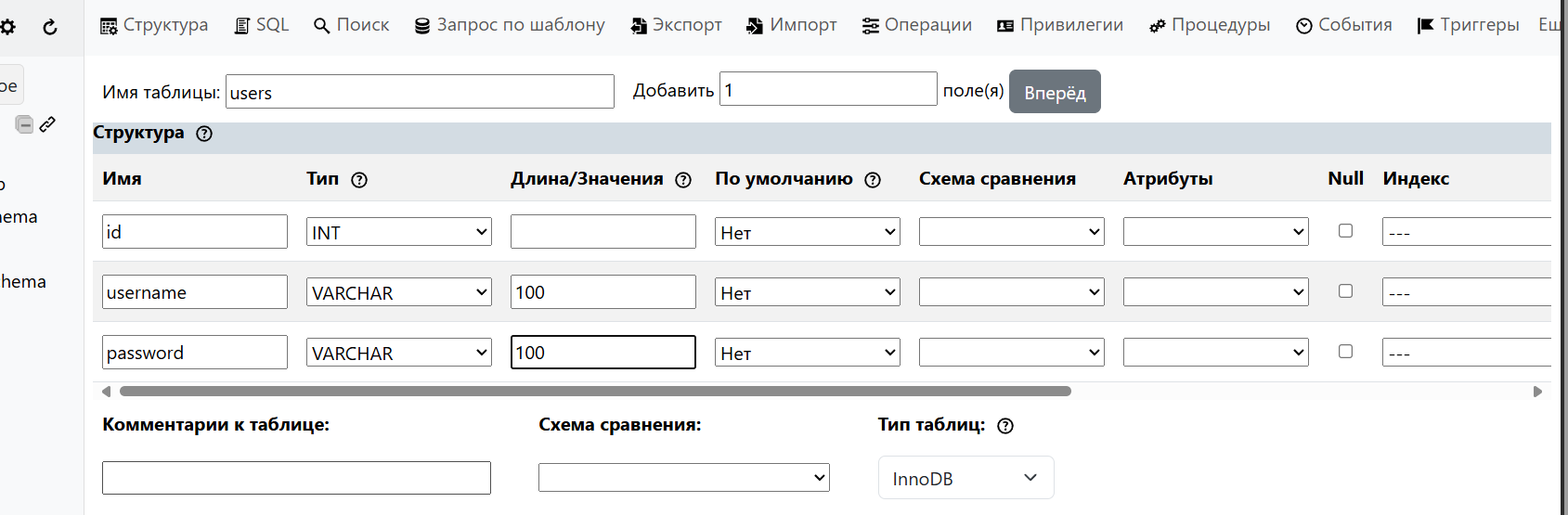


Рисунок 65 - Заполнение таблицы

1. Создаем подключение к бд, указывая сервер, порт, логин, пароль и название, созданной ранее бд



Рисунок 66 - Подключение бд

1. Создаем новый класс

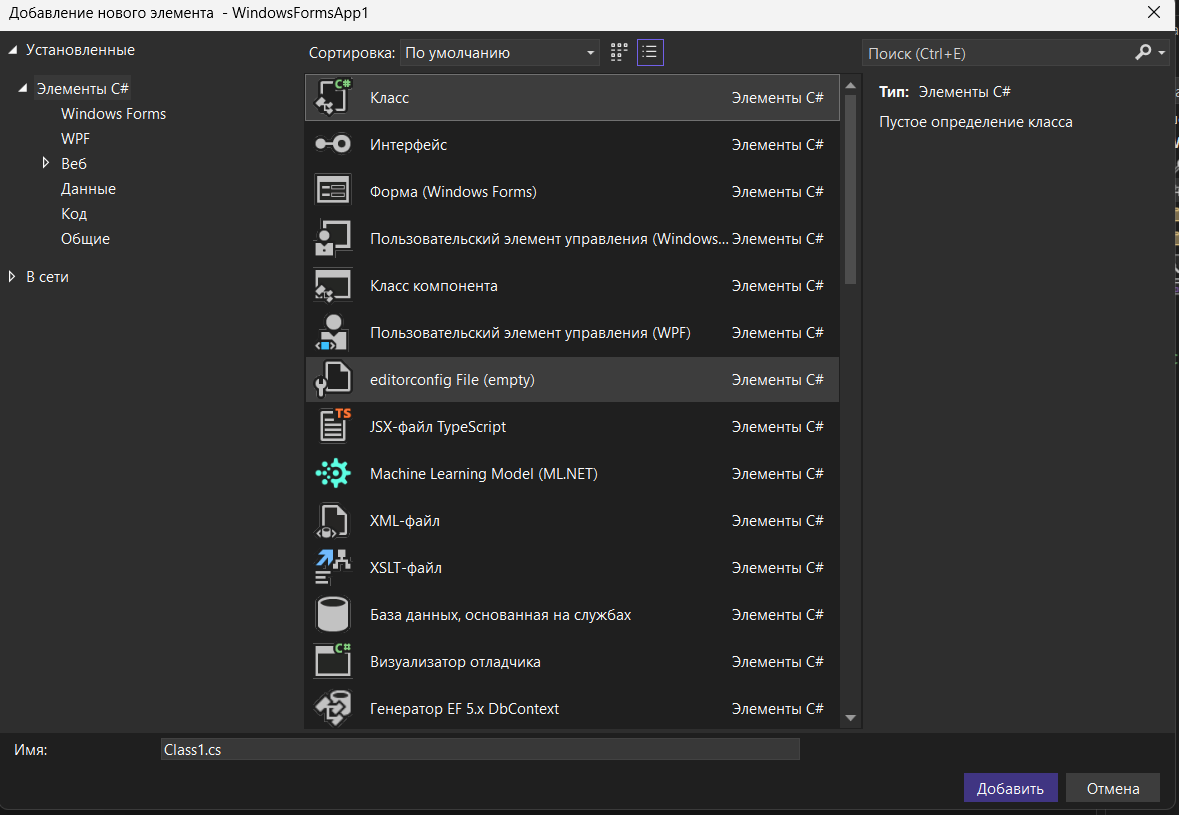


Рисунок 67 - Создание нового класса

1. Копируем наше подключение бд в новый созданный класс

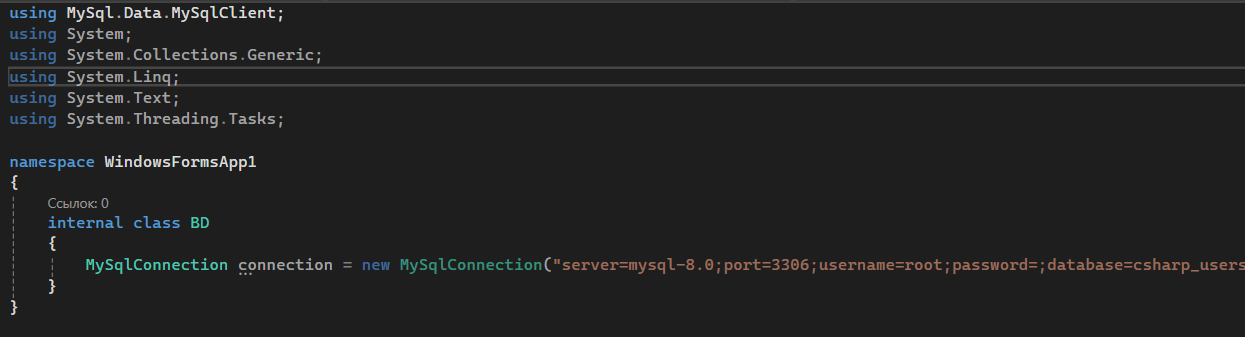


Рисунок 68 - Созданный класс

1. Пишем функцию для открытия соединений

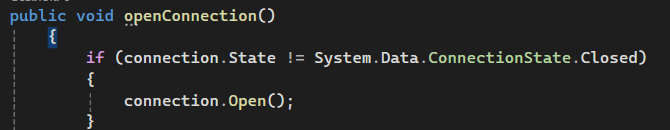


Рисунок 69 - Открытие соединения

1. Создаем функцию для закрытия соединения

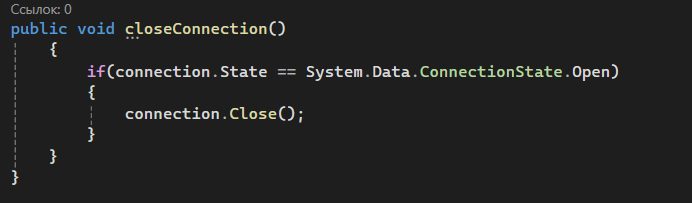


Рисунок 70 - Закрыие соединения

1. Создаем функцию для возврата соединения

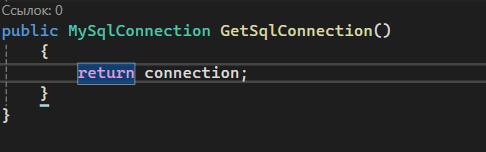


Рисунок 71 - Возврат соединения

1. Пишем событие кнопки login, в котором происходит обращение к бд с данными пользователя для получения пароля и логина, для входа и их проверки

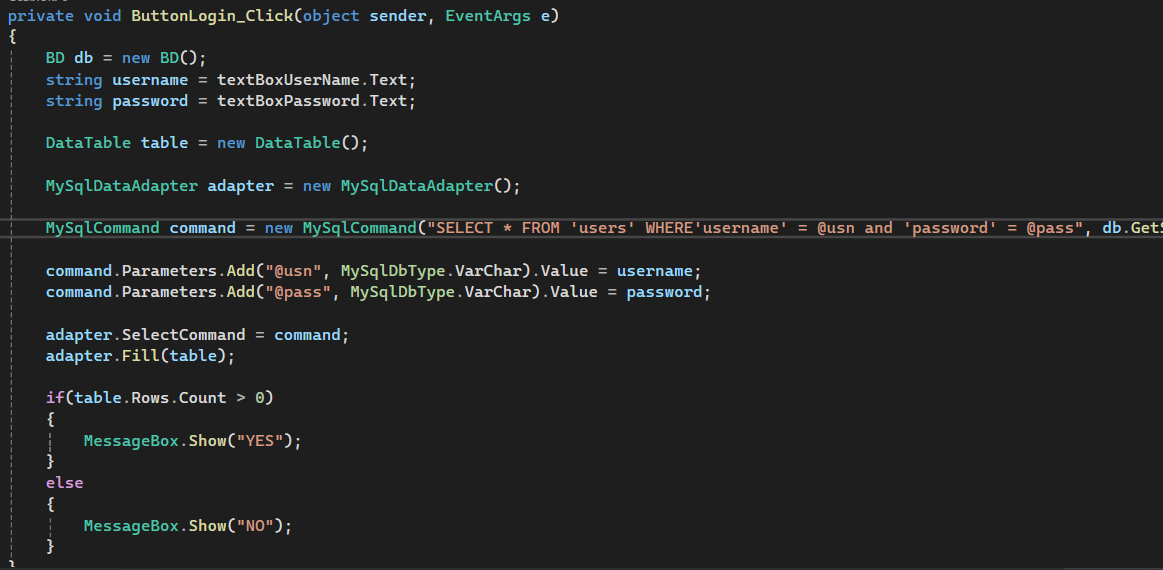


Рисунок 72 - Функция кнопки Login

1. Вносим в нашу бд данные пользователя для проверки авторизации

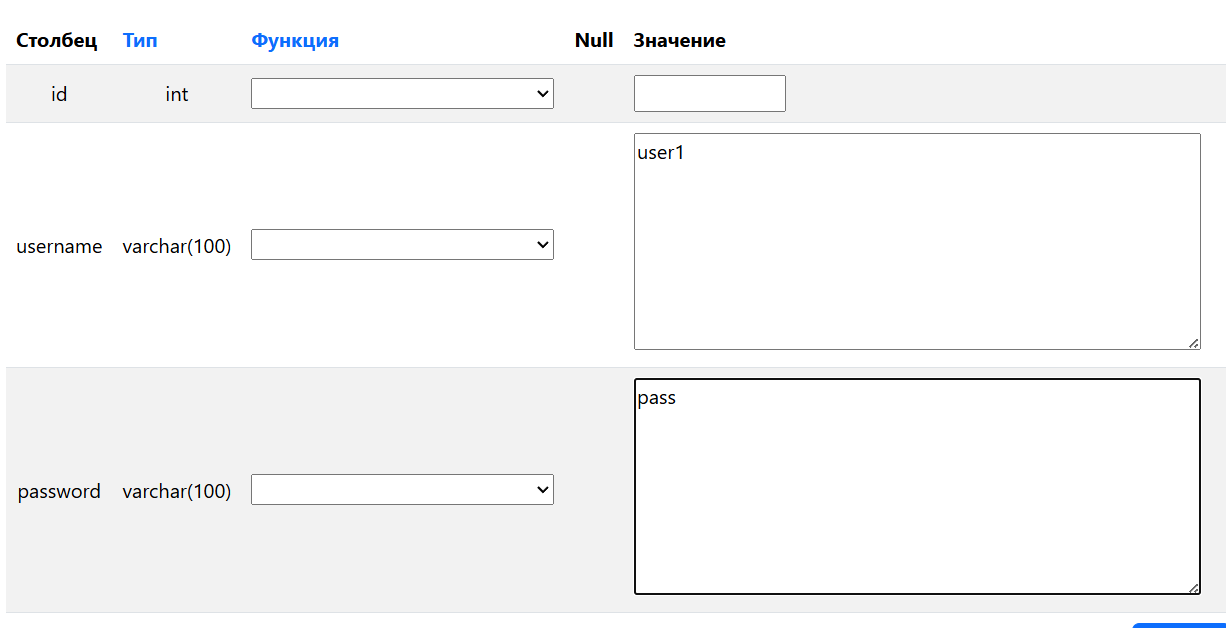


Рисунок 73 - Внесения данных в бд

Тестируем результат

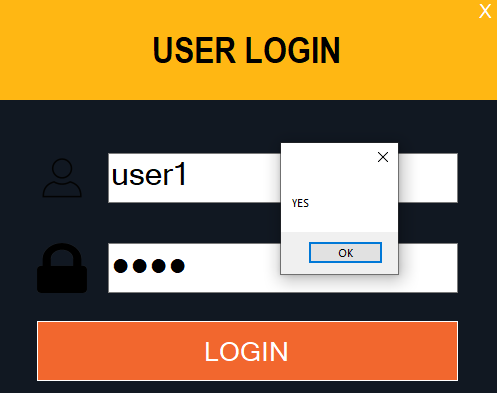


Рисунок 74 - Результат при введении верных логин и пароль

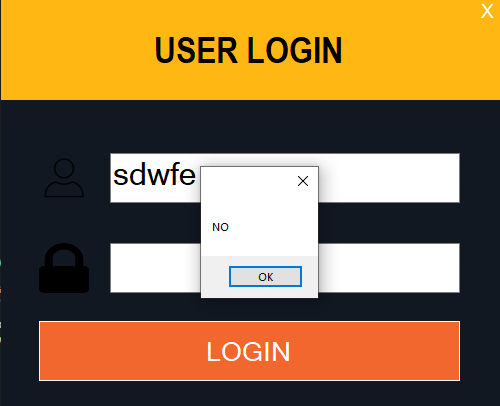


Рисунок 75 - Результат при введении неверных логин и пароль

Добавляем новую форму в проект

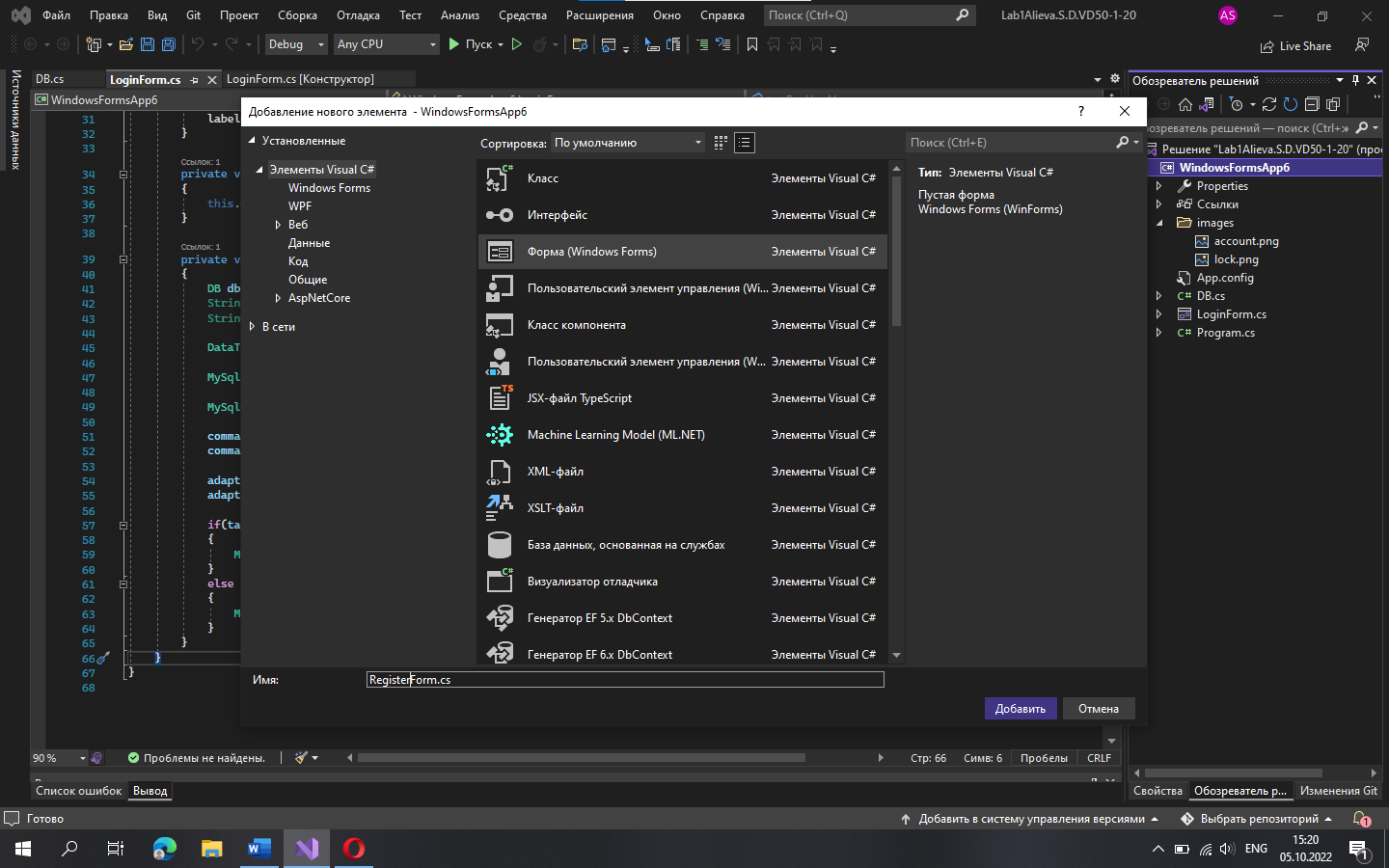


Рисунок - Новая форма

Копируем содержимое LoginForm на новую созданную форму RegisterForm

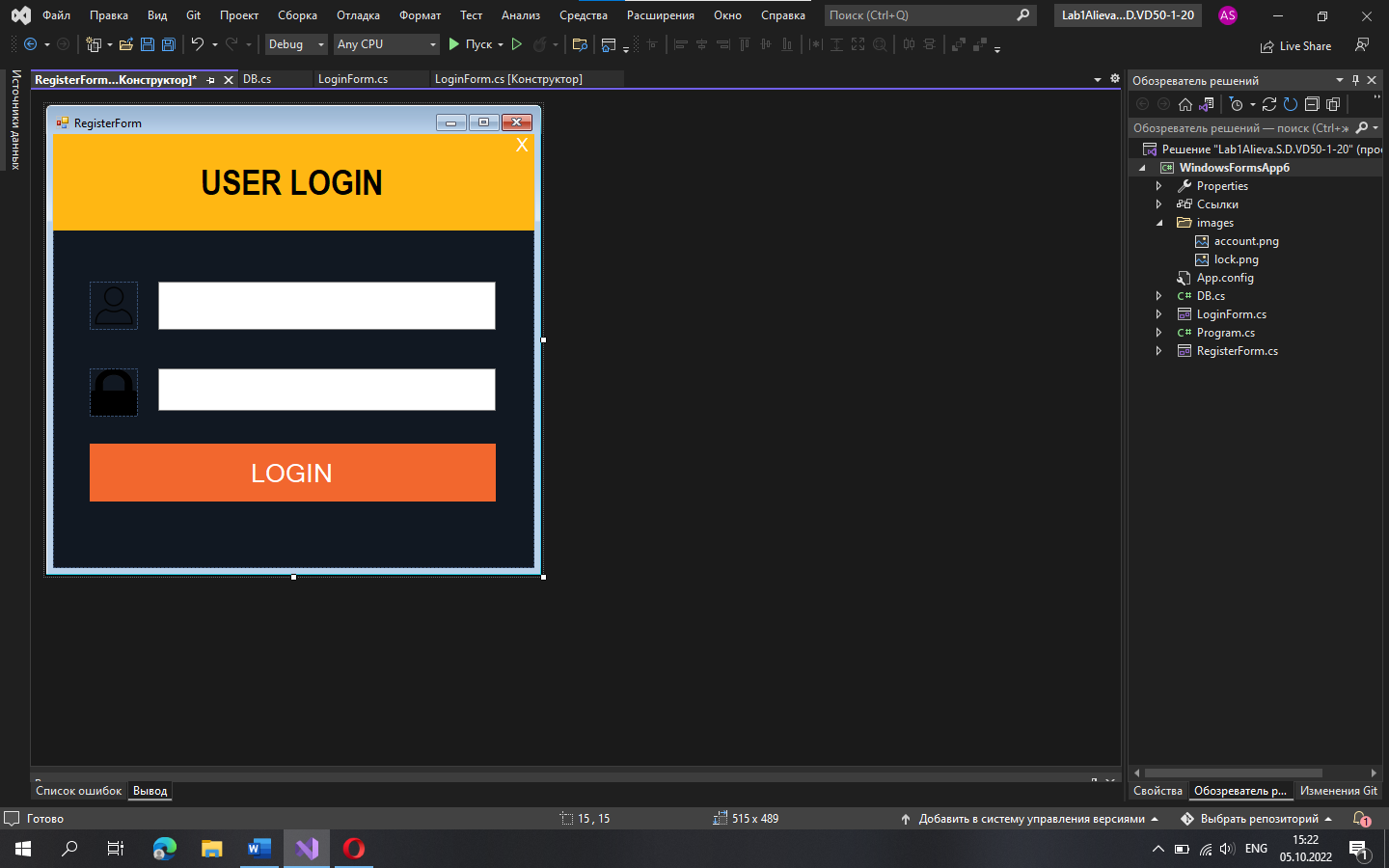


Рисунок - Скопированная LoginForm

Редактируем, добавляем textbox

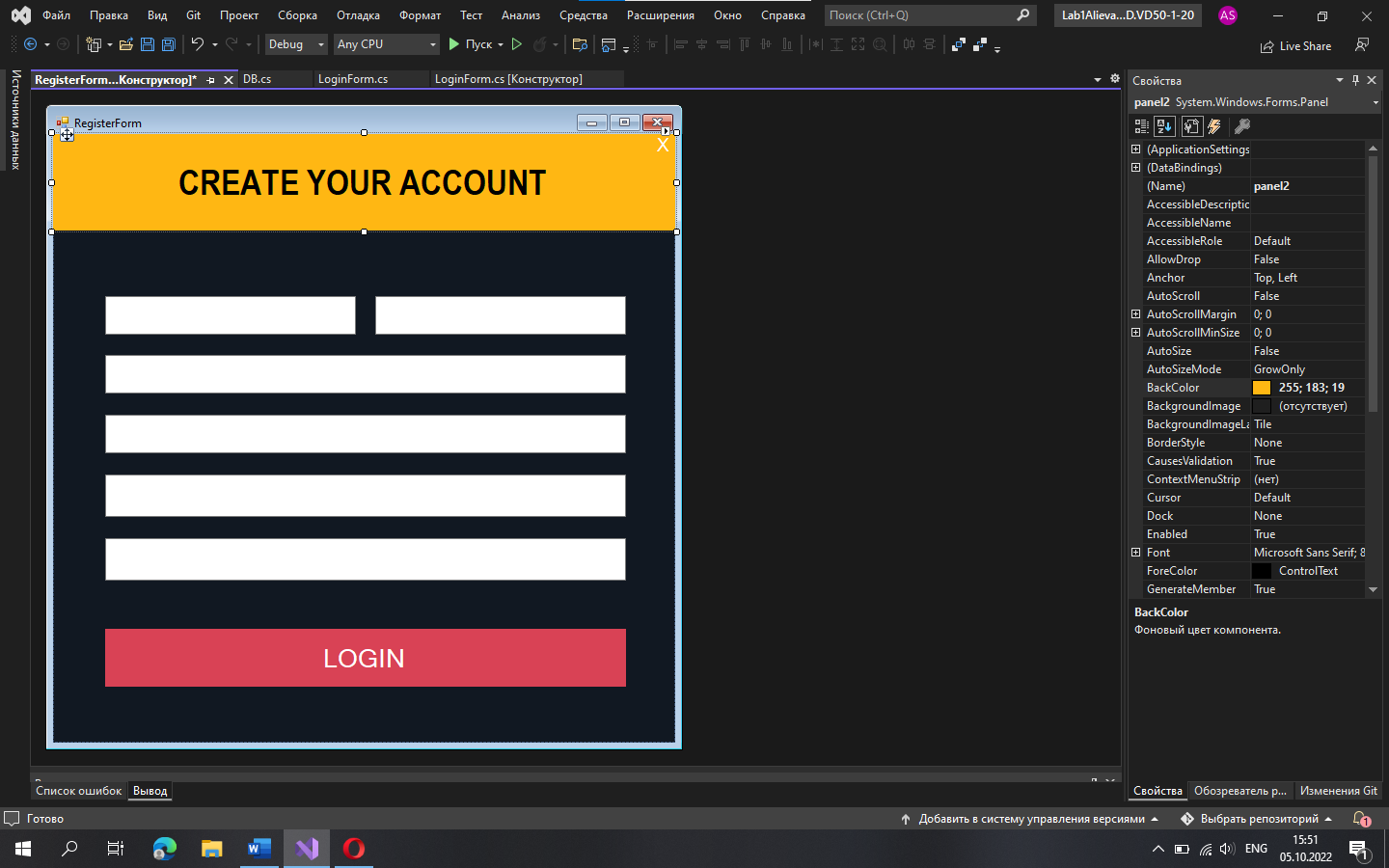


Рисунок - Отредактированная форма RegisterForm

В Program.cs меняем LoginForm на RegisterForm, чтобы при запуске приложения запускалась она



Рисунок - Program.cs

Заходим в свойства формы и задаем, чтобы она запускалась по центру экрана



Рисунок – StartPosition

Переименовываем все textbox



Рисунок - textBoxFirstName



Рисунок - textBoxLastName



Рисунок - textBoxEmail



Рисунок - textBoxUsername



Рисунок - textBoxPasswordConfirm



Рисунок – buttonCreateAccountx

Промежуточный результат

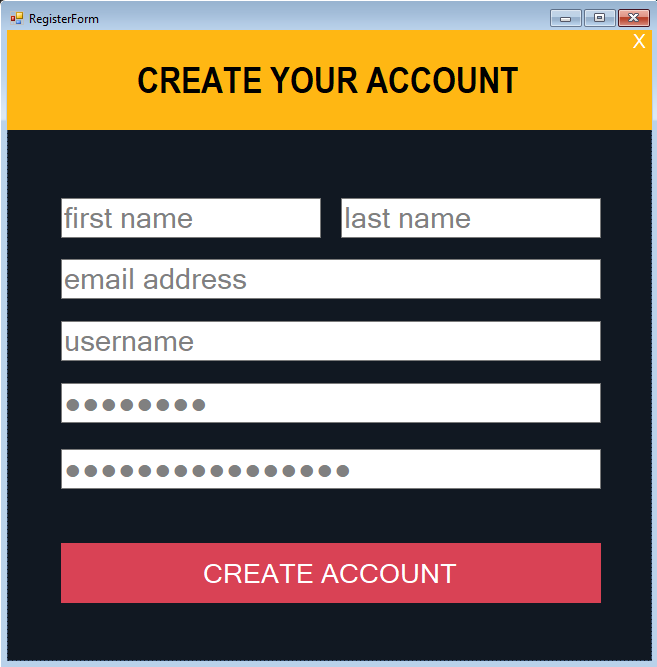


Рисунок - Промежуточный результат

Для textBoxPassword и textBoxPasswordConfirm меняем свойство UseSystemPassword, чтобы в полях было видно, что написано, а не скрыто



Рисунок - UseSystemPassword

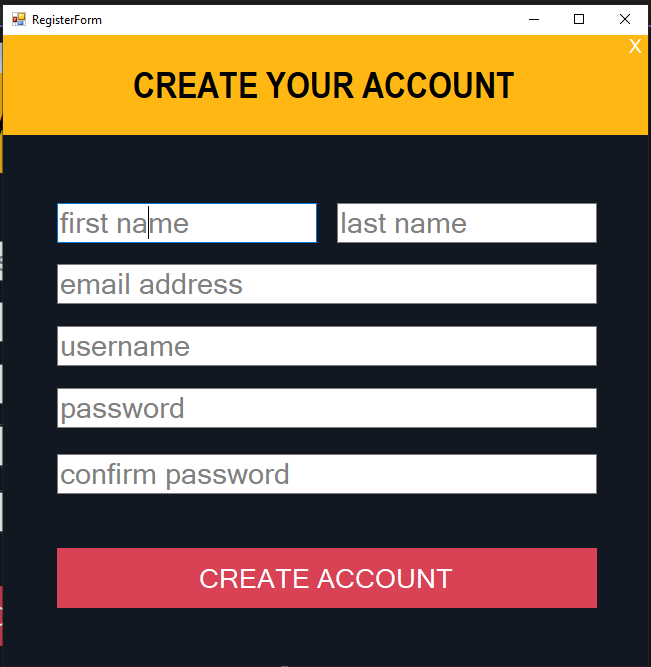


Рисунок – Результат после изменения свойства UseSystemPassword

Для textboxFirstName пишем функции, чтобы при нажатии на данное поле надпись скрывалась, если поле остается пустым, то надпись опять появляется

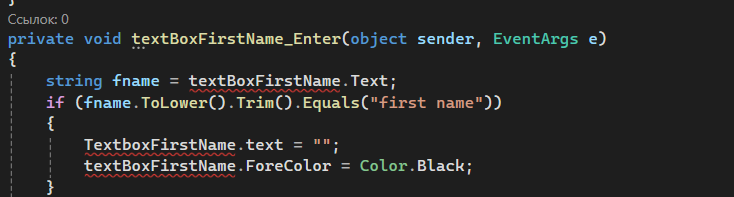


Рисунок - Функция textBoxFirstName\_Enter

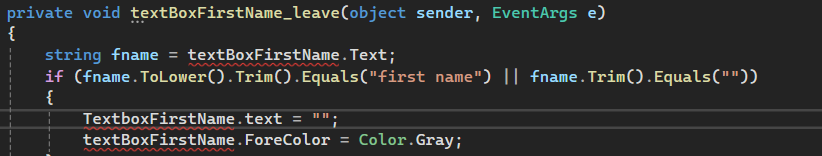


Рисунок - Функция textBoxFirstName\_Leave

Чтобы при запуске текстбоксы не были активными пишем следующую функцию

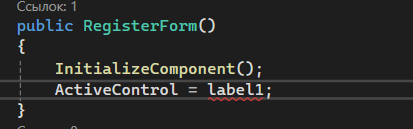


Рисунок – Активный элемент

Для textboxLastName пишем функции, чтобы при нажатии на данное поле надпись скрывалась, если поле остается пустым, то надпись опять появляется



Рисунок – textboxLastName

Для textboxEmail пишем функции, чтобы при нажатии на данное поле надпись скрывалась, если поле остается пустым, то надпись опять появляется



Рисунок – textboxEmail

Для textboxUsername пишем функции, чтобы при нажатии на данное поле надпись скрывалась, если поле остается пустым, то надпись опять появляется



Рисунок – textboxUsername

Для textboxPassword пишем функции, чтобы при нажатии на данное поле надпись скрывалась, если поле остается пустым, то надпись опять появляется



Рисунок – textboxPassword

Для textboxPasswordConfirm пишем функции, чтобы при нажатии на данное поле надпись скрывалась, если поле остается пустым, то надпись опять появляется



Рисунок – textboxPasswordConfirm

Для label, который выполняет функцию закрытия приложения пишем следующие события, чтобы при нажатии на него оно закрывалось, а при наведении label менял цвет на черный



Рисунок - События кнопки закрытия приложения

Убираем у формы рамки меняя свойство FormBorderStyle



Рисунок - FormBorderStyle

После выполнения всех действий получаем данный результат

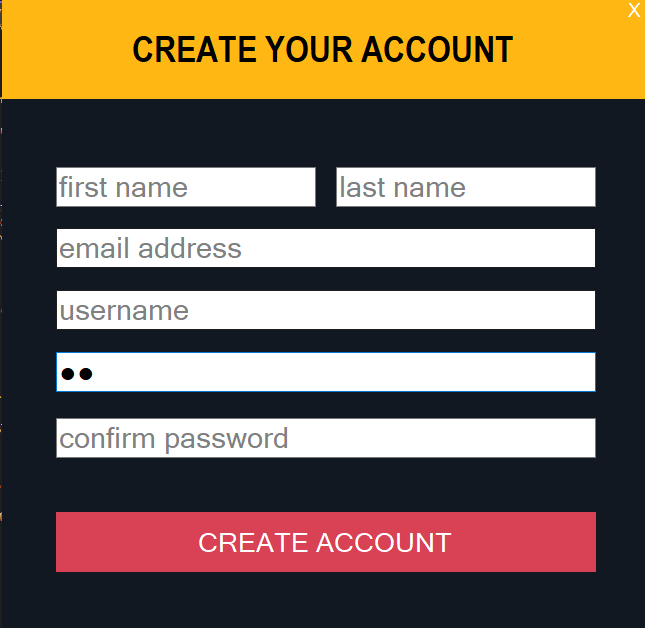


Рисунок - Результат

Заходим в PhpMyAdmin и добавляем в таблице столбцы с атрибутами firstname, lastname и emailaddress

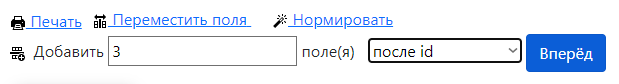


Рисунок - Добавление столбцов в таблицу

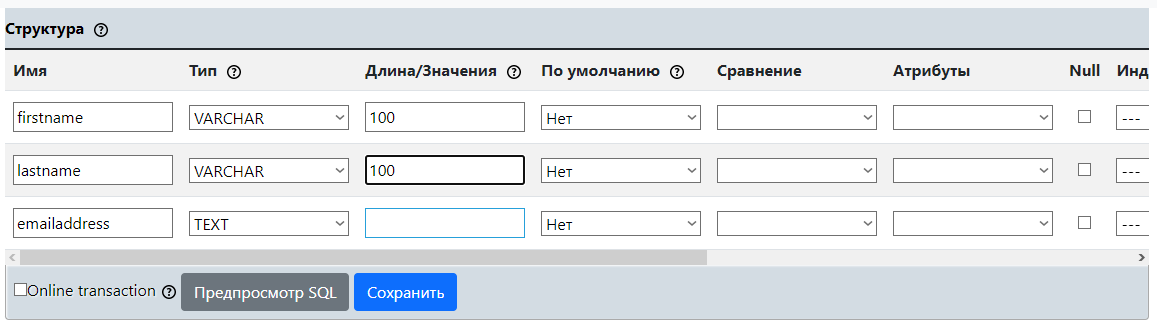


Рисунок - Структура таблицы

В пространство имен подключаем MySql

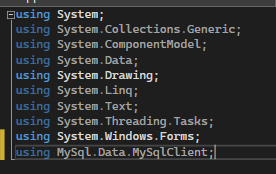


Рисунок - Пространстрво имен

Для кнопки создание аккаунт пишем функцию, в которой данные с текстбоксов считываются и заносятся в бд и выводится сообщение «ACCOUNT CREATED»

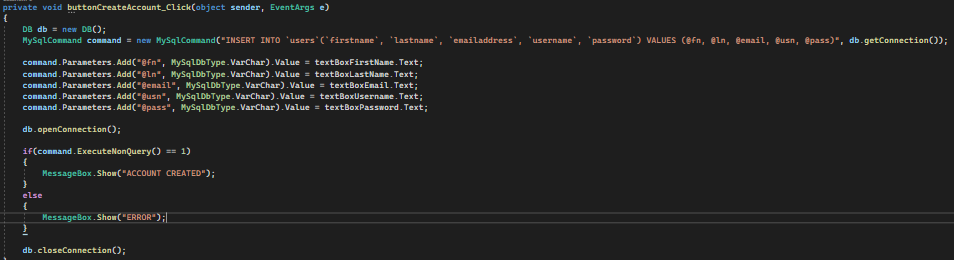


Рисунок - Функция кнопки создания аккаунта

Промежуточный результат



Рисунок - Промежуточный результат

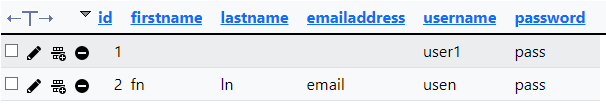


Рисунок - Данные в бд

Пишем событие, в котором будет проверяться username, чтобы невозможно было создать пользователя с таким username, который уже используется

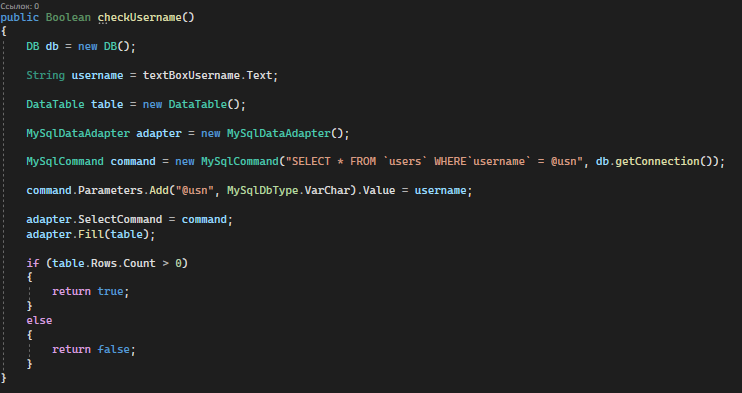


Рисунок - Функция checkUsername

Пишем условие, в котором определяется создан такой пользователь уже или нет, если не создан то выводим месседжбокс, с надписью «account created»



Рисунок - Проверка, существует такой аккаунт или нет

Пишем функцию, в которой проверяются поля, пустые они или заполненные



Рисунок - Проверка, пустые поля регистрации или нет

Проверка создания аккаунта с данными уже существующего пользователя

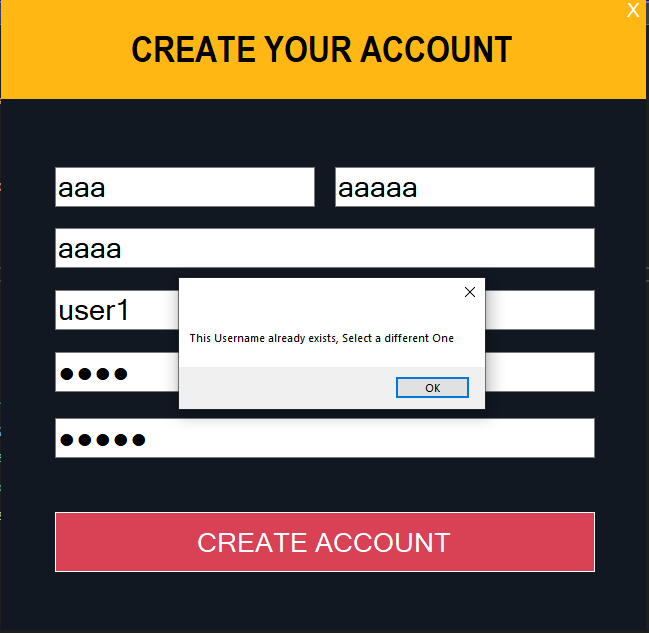


Рисунок - Создание уже существующего пользователя

Проверка создания аккаунт с данными еще существующего пользователя

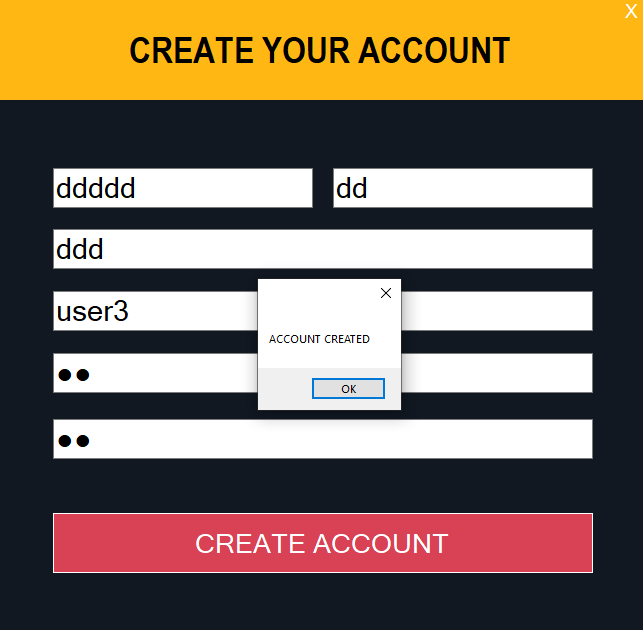


Рисунок - Создание аккаунта

Добавление нового пользователя в базе данных



Рисунок - Новый пользователь в бд

Пишем функцию, в которой проверяется введены какие-то данные пользователем или нет, если нет, то выводится мееседжбокс с сообщением «Enter your information first»



Рисунок - Вывод меседжбокса если поля пустые

Проверка регистрации аккаунта без введения данных



Рисунок - Создание аккаунта с пустыми полями

Пишем условие проверки паролей, чтобы при вводе подтверждения пароля оно совпадало с введенным паролем.



Рисунок - Проверка пароля

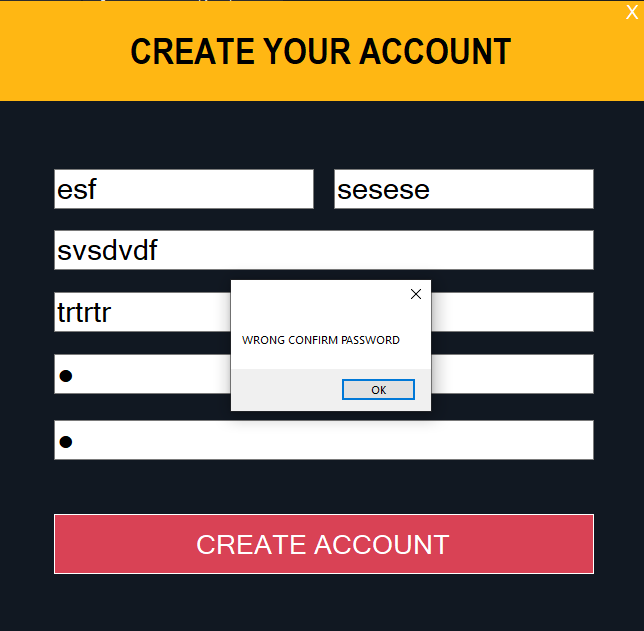


Рисунок – Отрицательный результат проверки пароля

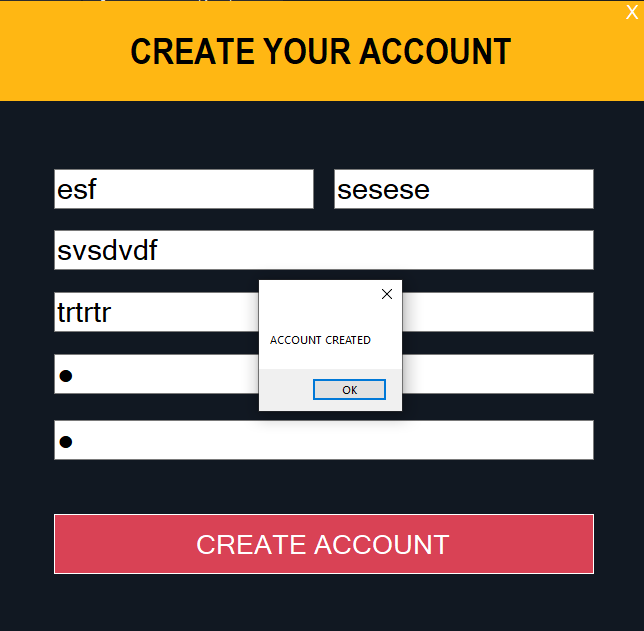


Рисунок – Положительный результат проверки пароля

Добавляем в месседжбокс, который показывается при использовании существующего username кнопки «ОК», «Отмена» и изображение ошибки, а так же в тот, который показывается при создании аккаунта с изображением информации и кнопкой «ОК».



Рисунок - Доработка MessageBox



Рисунок - Готовые MessageBox с ошибкой



Рисунок - Готовые MessageBox без ошибки

Добавляем в месседжбокс, который показывается при вводе в пароль и подтверждения пароля разные данные кнопки «ОК», «Отмена» и изображение ошибки.



Рисунок - Доработка MessageBox



Рисунок – Готовый MessageBox с ошибкой

Добавляем в месседжбокс, который показывается при оставлении всех полей воода пустыми кнопки «ОК», «Отмена» и изображение ошибки.



Рисунок - Доработка MessageBox



Рисунок - Готовый MessageBox с ошибкой

Так же добавляем в мессджбоксы на LoginForm кнопки «ОК» и изображение ошибки.



Рисунок - Доработка MessageBox

Создаем новую форму – MainForm и копируем в нее все элементы с RegisterForm.

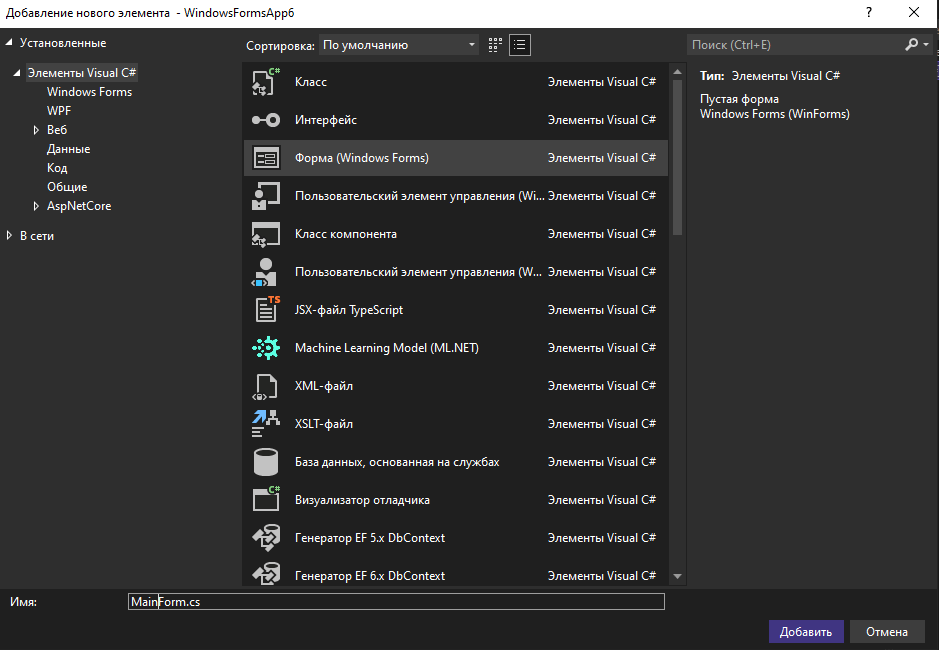


Рисунок - Создание новой формы

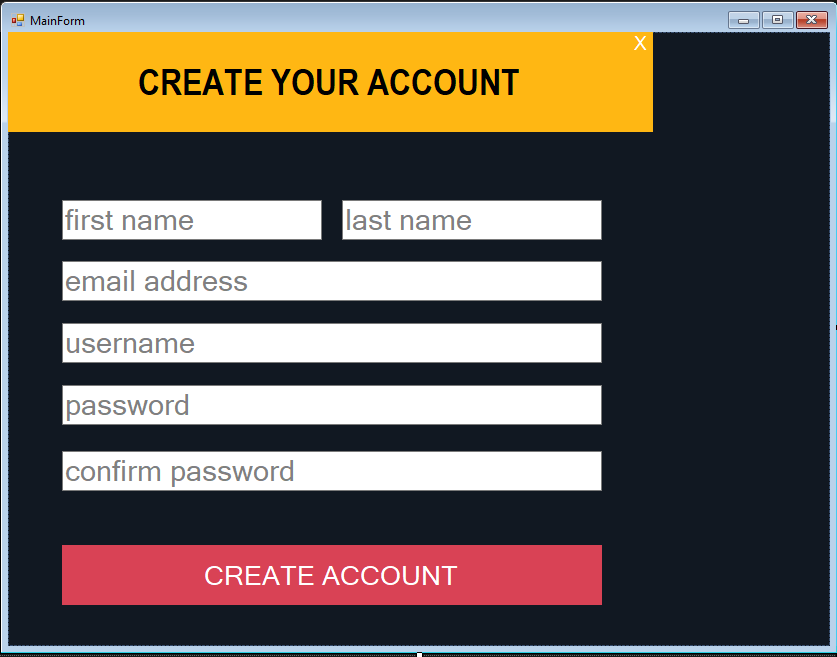


Рисунок - Новая форма

Удаляем с нее некоторые элементы и меняем текст в label1, задаем форме свойства, чтобы она открывалась по середине экрана.



Рисунок - Отредактировання форма



Рисунок - Свойство StartPosition

В форму LoginForm добавляем label с текстом Don’t have account? Sign up.



Рисунок - Текст label

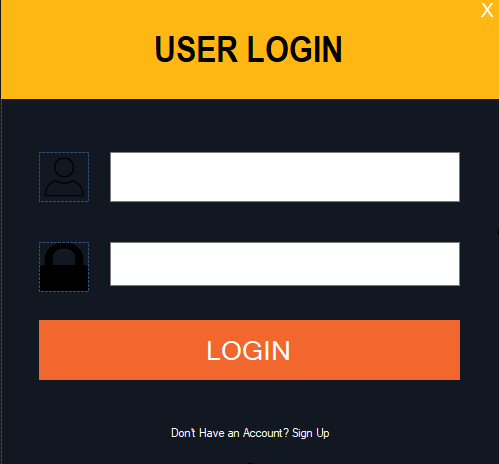


Рисунок - Отредактированная LoginForm

Для созданного label пишем событие на клик, чтобы открывалась форма RegisterForm.



Рисунок - Событие открытия RegisterForm

Заходим в Program.cs и меняем открывающеюся форму на LoginForm.



Рисунок - Program.cs

В Register form на label, который отвечает за закрытие приложения пишем Application.Exit(), чтобы оно действительно закрывалось полностью, а не просто скрывалась форма.



Рисунок - Событие закрытия приложения

На всех формах у кнопок и label, которые выполняют роль кнопки, меняем свойство cursor на hand, чтобы при наведении на них курсор менялся на руку.



Рисунок - Свойтсво Cursor

Пишем события для labelGoSingUp, чтобы в спокойном состоянии тест был белого цвета, а при наведении желтого



Рисунок - Событие labelGoSingUp

В настройках шрифта у labelGoSingUp ставим галочку напротив «Подчерктнутый»



Рисунок - Настройки шрифта

Копируем этот label на форму RegisterForm, меняем у него название и текст

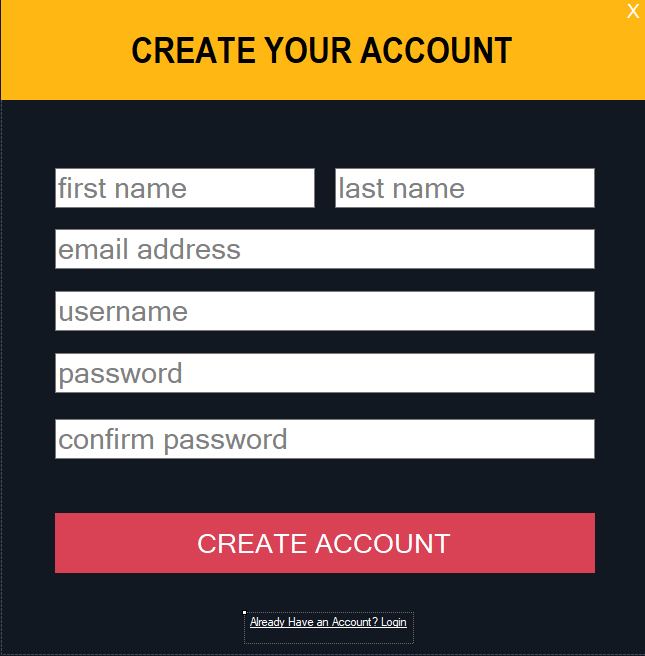


Рисунок - RegisterForm



Рисунок - Название label

Пишем события для него, чтобы в спокойном состоянии тест был белого цвета, а при наведении желтого, а также чтобы при клике на него открывалась форма LoginForm.



Рисунок - Событие смены цвета при наведении на label



Рисунок - Событие открытия LoginForm

В LoginForm на label, который отвечает за закрытие, приложение пишем Application.Exit(), чтобы оно действительно закрывалось полностью, а не просто скрывалась форма.



Рисунок - Событие закрытия приложения

В событие проверки правильности ввода username и password пишем, открытие MainForm.



Рисунок - Открытие MainForm

В MainForm пишем событие закртия приложения.



Рисунок - Событие закрытия приложения

В Login form при проверке введения данных в текстбоксы меняем username на password



Рисунок - Проверка введения данных в текстбоксы

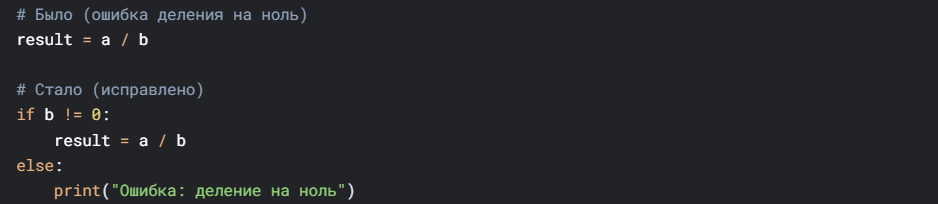
**Определение:**

**Отладка — это важный этап разработки программного обеспечения, направленный на выявление и исправление ошибок в коде. Ниже приведено пошаговое руководство по эффективной отладке**

**Цель:**

**Цель: Создать условия, при которых ошибка проявляется**

**Задание:**

1. 

**Инструменты:**

|  |  |
| --- | --- |
| **Инстумент** | **Описание** |
| **Отладчики** | **Пошаговое выполнение, просмотр переменных** |
| **Статические анализаторы** | **Поиск ошибок до запуска программы** |
| **Логирование** | **Запись событий во время выполнения** |
| **Профилировщики** | **Анализ производительности** |

**Методы:**

Пошаговая отладка — выполнение кода строка за строкой.

Точки останова (Breakpoints) — остановка в нужном месте.

Трассировка — запись истории выполнения.

Профилирование — поиск узких мест в производительности.

Навыки для успешной отладки

Чтение кода — понимание чужого и своего кода.

Логическое мышление — анализ возможных причин ошибки.

Внимательность — замечать мелкие детали.

Настойчивость — не сдаваться, даже если ошибка сложная.

Вывод

Отладка — это итеративный процесс, требующий системного подхода. Используйте инструменты, анализируйте код, тестируйте исправления — и ваша программа станет надежнее!