

## Тема: Создание приложения с использованием окон, стилей

**Задание 1.** Создать приложение, позволяющее выполнить:

- 1) Авторизацию пользователя
- 2) При успешной авторизации получить список пользователей, представленный в табличном виде (элемент DataGridView)

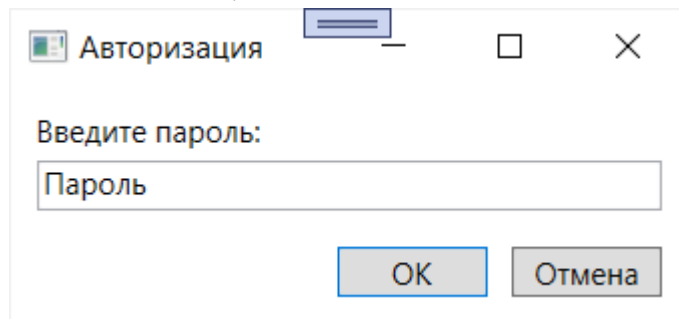
**Примечание:** Образцы окон даны без учета требований по стилю

При создании приложения предусмотреть:

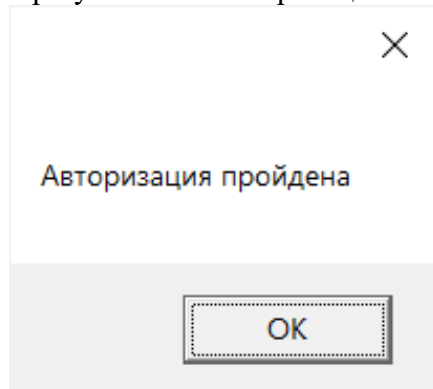
- 1) все окна приложения должны находиться в одной папке, например, *Windows*
- 2) при запуске системы первым окном будет главное окно, реализованное с заставкой. Главное окно имеет нестандартную форму (форму создать на свое усмотрение), кнопки для перехода на окно авторизации и закрытия приложения

Для окна реализовать режим изменения его размера и перемещения

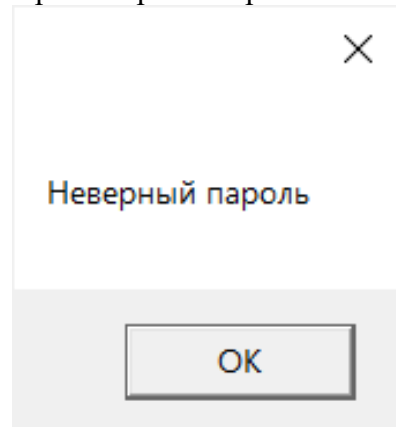
- 3) Реализуйте окно авторизации пользователя согласно макета:



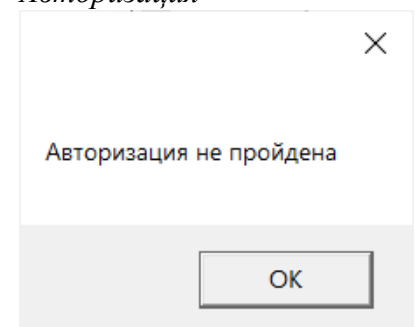
При успешной авторизации



При неверном пароле



При нажатии кнопка **Отмена** в окне авторизации и есть возможность на главном окне снова нажать кнопку *Авторизация*



- 4) После успешной авторизации пользователь получает возможность просмотра списка пользователей системы:

Пользователи					
Имя	Фамилия	Роль	Логин	Пароль	
Петров	Илья	админ	Pladm	789	
Иванов	Федор	гость	IFgost	456	
Горохова	Анна	менеджер	GAmanegr	123	

- 5) Визуальные компоненты должны соответствовать руководству по стилю: цветовая схема, использование шрифтов (описать в отдельном файле **App.xaml**).
- 6) Вид кнопок описать в словаре ресурсов
- 7) Все компоненты системы должны иметь единый согласованный внешний вид, соответствующий руководству по стилю, а также следующим требованиям:
  - разметка и дизайн (предпочтение отдается масштабируемой компоновке; должно присутствовать ограничение на минимальный размер окна; должна присутствовать возможность изменения размеров окна, где это необходимо; увеличение размеров окна должно увеличивать размер контентной части, например, таблицы с данными
  - последовательный переход фокуса по элементам интерфейса (по нажатию клавиши TAB);
  - последовательный пользовательский интерфейс, позволяющий перемещаться между существующими окнами в приложении (в том числе обратно, например, с помощью кнопки «Назад»);
  - соответствующий заголовок на каждом окне приложения (не должно быть значений по умолчанию типа MainWindow, Form1 и т.п).

### Руководство по стилю

**Шрифт:** Используйте шрифт Comic Sans MS.

**Цветовая схема:**

В качестве **основного фона** используется **белый цвет**; в качестве дополнительного: **RGB (231, 250, 191)**. Для **акцентирования внимания** пользователя на целевое действие интерфейса используйте цвет **RGB (4, 160, 255)**.

Основной фон	Дополнительный фон	Акцентирование внимания
RGB (255, 255, 255)	RGB (231, 250, 191)	RGB (4, 160, 255)

**Примечания:**

1. При перемещении MainWindow в папку со всеми окнами может возникнуть необработанное исключение типа в PresentationFramework.dll

Как исправить?

В файле App.xaml измените:

StartupUri="MainWindow.xaml" на StartupUri="FolderName/MainWindow.xaml"

где FolderName – это имя вашей папки с окнами

Также при переименовании могут возникнуть такая же ситуация. Это можно решить используя синтаксис URI пакета, чтобы установить свойство StartUri файла App.xaml, нужно указать на новое местоположение MainWindow.xaml, так как:

```
<Application x:Class="TrafficLights.Controller.App"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
StartupUri="pack://application:;./View/MainWindow.xaml">
```