Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЙ №1-3**

**Тема: Разработка страницы авторизации пользователей. Создание словаря стилей**

Студент: М.Д. Кравченко

Группа: 619

Преподаватель: Т.Г.Аксенова

Дата: 22.09.2022

**Цель работы:** приобрести практические навыки проектирования интерфейса приложения WPF, навигации по страницам WPF, подключения базы данных с помощью технологии Entity Framework ADO.NET, реализации страницы авторизации пользователей, создания и настройки словаря стилей.

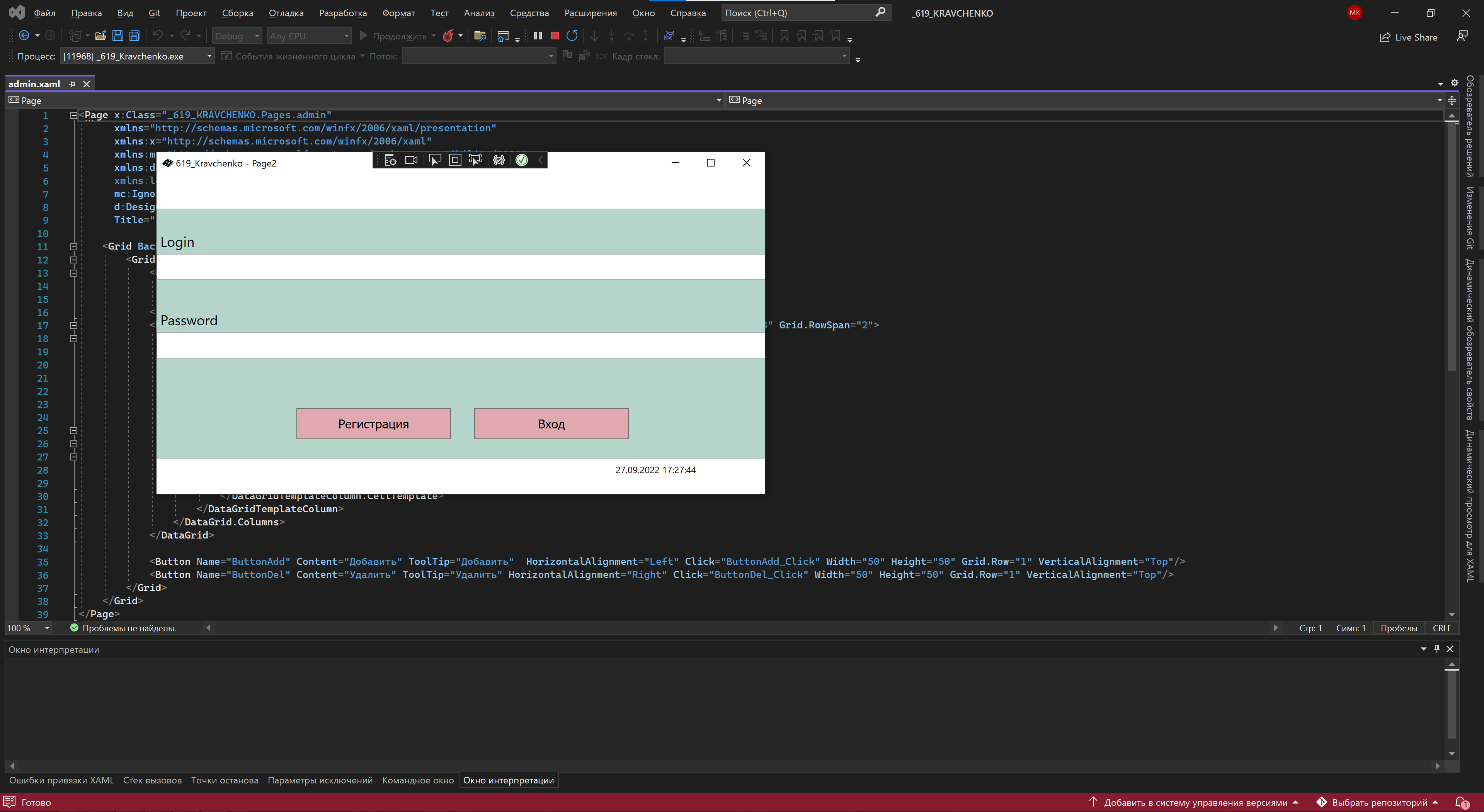


Рисунок 1. Авторизация

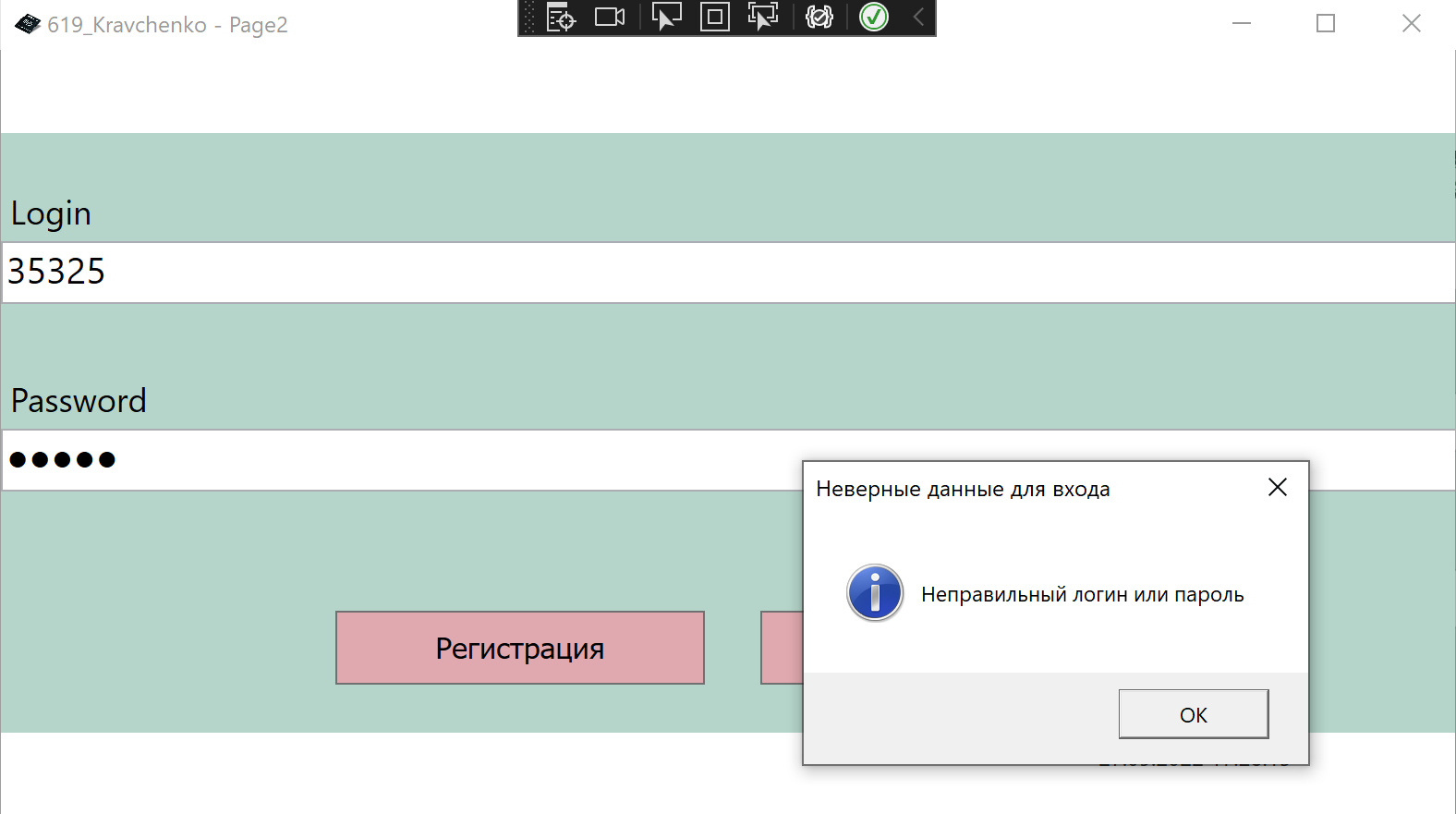


Рисунок 2. Неправильная авторизация

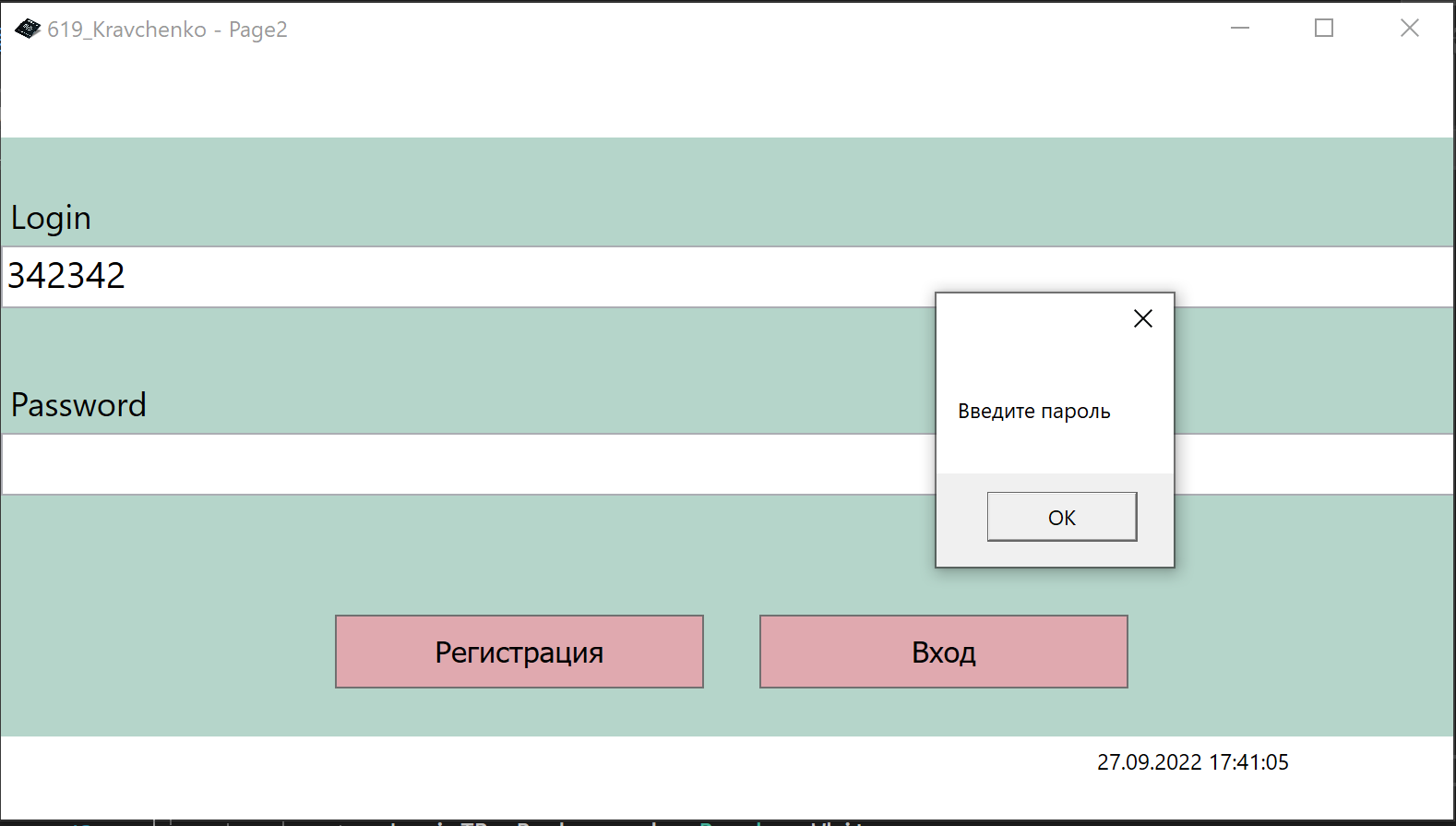


Рисунок 3. Проверка ввода данных

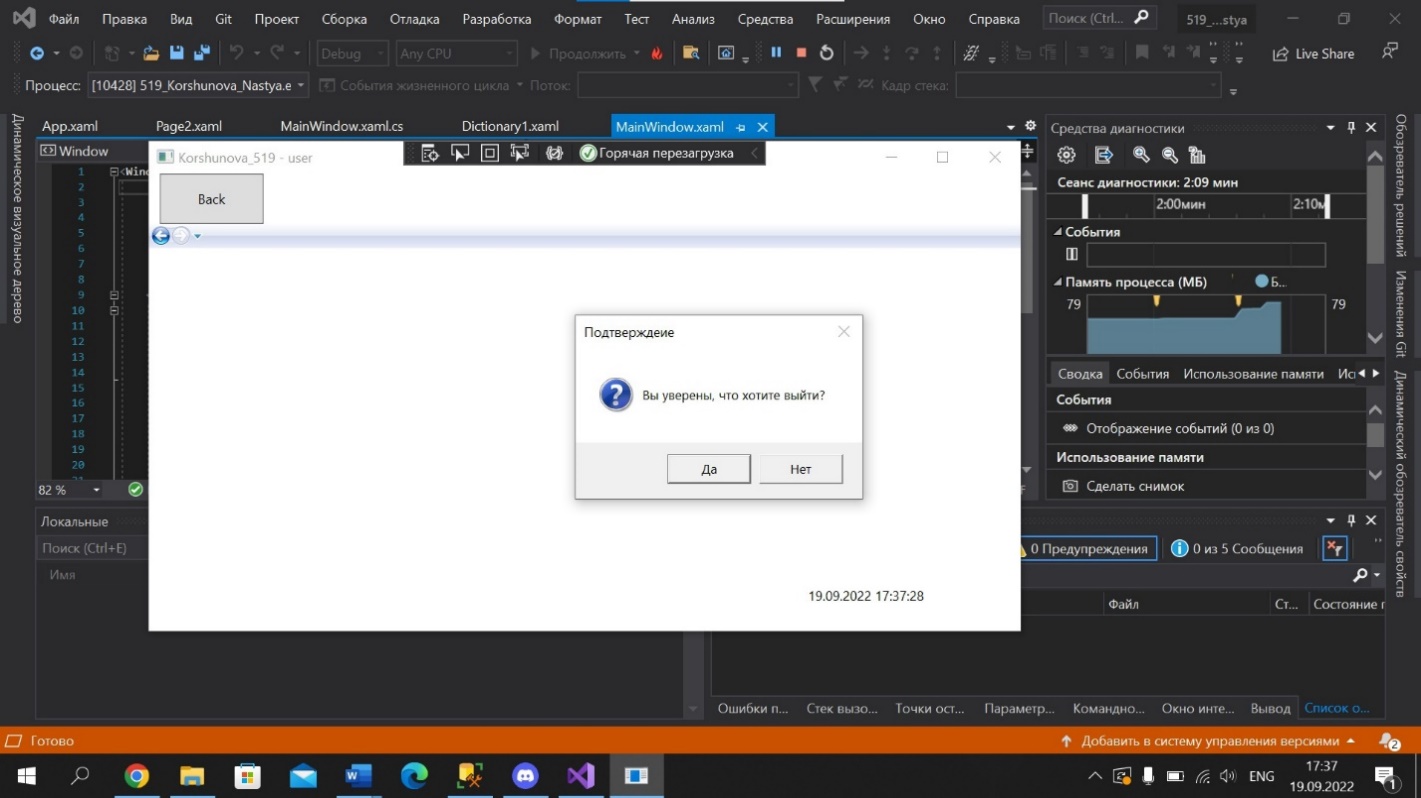


Рисунок 4. Подтверждение выхода

Изображение выглядит как текст, монитор, снимок экрана

Автоматически созданное описание

Рисунок 5. Окно MainWindow

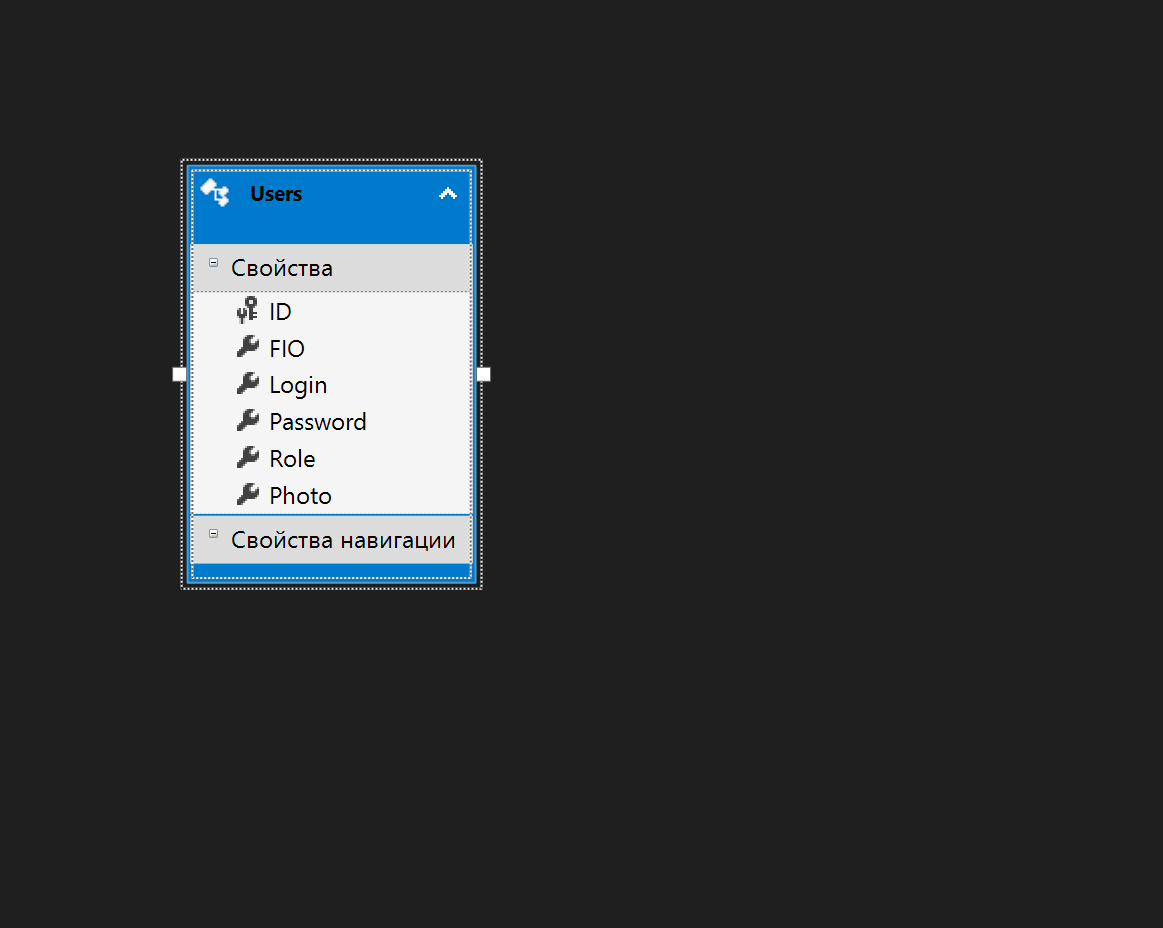


Рисунок 6. Подключенная база данных

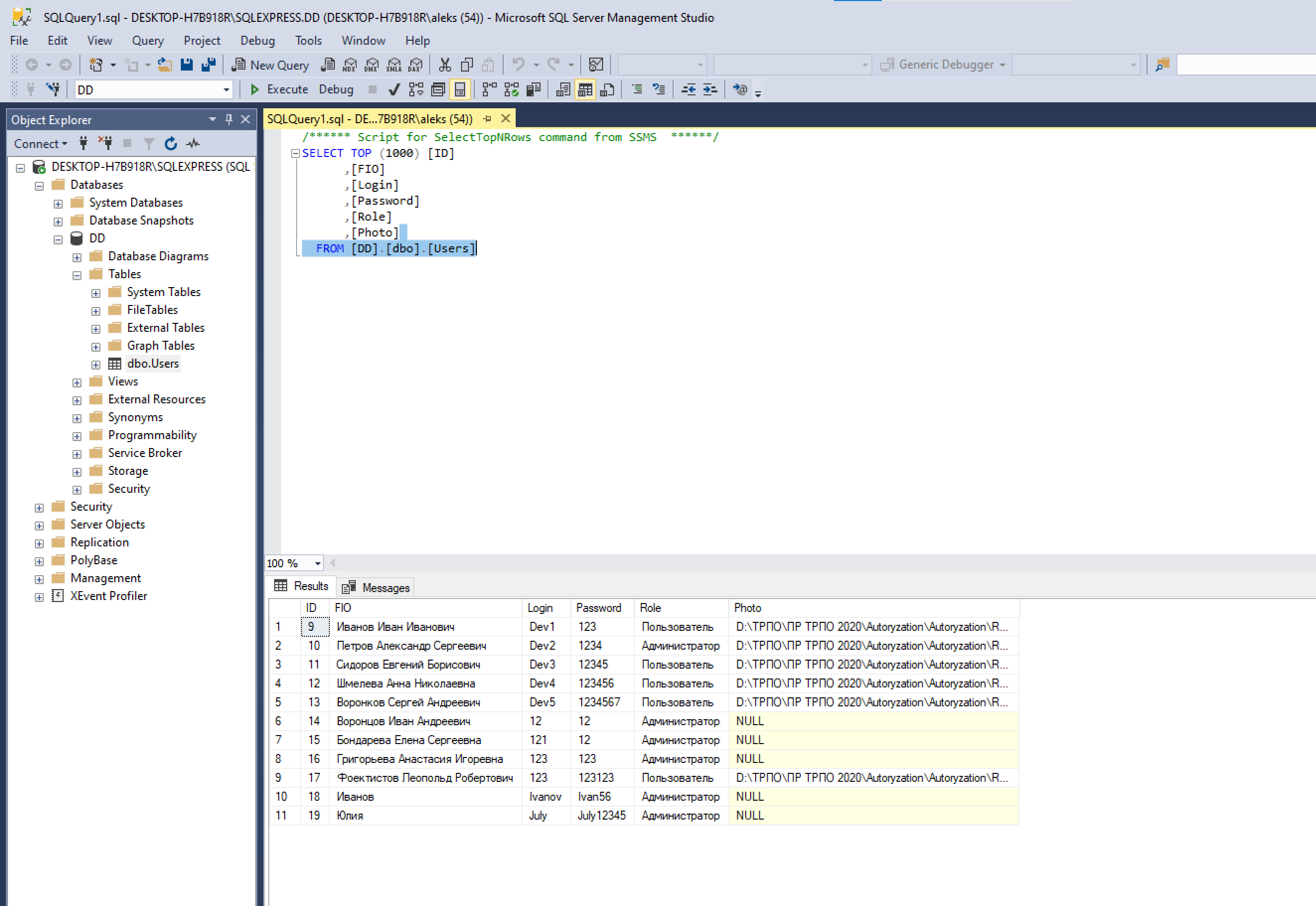


Рисунок 7. Таблица базы данных

Page2.xaml

<Page x:Class="\_619\_KRAVCHENKO.Pages.Page2"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:\_619\_KRAVCHENKO.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Page2" Background="Black">

<Grid Style="{DynamicResource Grid}">

<Grid.RowDefinitions>

<RowDefinition Height="25"/>

<RowDefinition Height="1\*"/>

<RowDefinition Height="1\*"/>

<RowDefinition Height="1\*"/>

</Grid.RowDefinitions>

<Label Content="Login" Grid.Row="1" FontSize="18"/>

<TextBox x:Name="LoginTBx" Height="34" FontSize="20" ToolTip="Введите логин" Grid.Row="1" />

<Label Content="Password" Grid.Row="2" FontSize="18"/>

<PasswordBox x:Name="PasswwordPBx" Height="34" FontSize="20" ToolTip="Введите пароль" Grid.Row="2"/>

<Button Click="Button\_Click" HorizontalAlignment="Center" Style="{DynamicResource Button}" Grid.Row="3" Content="Регистрация" Width="200" Height="40" Margin="0,0,230,0"/>

<Button HorizontalAlignment="Center" Style="{DynamicResource Button}" Grid.Row="3" Content="Вход" Width="200" Height="40" Name="Full" Click="Full\_Click" Margin="230,0,0,0" />

</Grid>

</Page>

MaimWindow.xaml

<Window x:Class="\_619\_KRAVCHENKO.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:\_619\_KRAVCHENKO"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800" Closing="Window\_Closing">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="45"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="45"/>

</Grid.RowDefinitions>

<Frame Navigated="frame1\_Navigated" Name="frame1" HorizontalAlignment="Left" Height="329" Grid.Row="1" VerticalAlignment="Top" Width="792" Source="pages/Page2.xaml" />

<Label Name="label1" HorizontalAlignment="Left" Height="45" Margin="589,0,0,0" Grid.Row="2" VerticalAlignment="Top" Width="203"/>

<Button Click="exit\_Click" Name="exit" Content="Back" HorizontalAlignment="Left" VerticalAlignment="Top" Width="93" Height="45" Margin="10,0,0,0"/>

</Grid>

</Window>

MaimWindow.xaml.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using System.Windows.Forms;

namespace \_619\_KRAVCHENKO

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

var timer = new System.Windows.Threading.DispatcherTimer();

timer.Interval = new TimeSpan(0, 0, 1);

timer.IsEnabled = true;

timer.Tick += (o, t) => { label1.Content = DateTime.Now.ToString(); };

timer.Start();

}

private void Window\_Closing(object sender, System.ComponentModel.CancelEventArgs e)

{

DialogResult res = System.Windows.Forms.MessageBox.Show("Вы уверены, что хотите выйти?", "Подтверждеие", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (res != System.Windows.Forms.DialogResult.Yes)

{

e.Cancel = true;

}

}

private void frame1\_Navigated(object sender, NavigationEventArgs e)

{

if (!(e.Content is Page page)) return;

this.Title = $"619\_Kravchenko - {page.Title}";

if (page is Pages.Page2)

exit.Visibility = Visibility.Hidden;

else

exit.Visibility = Visibility.Visible;

}

private void exit\_Click(object sender, RoutedEventArgs e)

{

if (frame1.CanGoBack) frame1.GoBack();

}

// обработчик, подключаемый в XAML

}

}

App.xaml

<Application x:Class="\_\_619\_KRAVCHENKO.App"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:local="clr-namespace:\_\_619\_KRAVCHENKO"

StartupUri="MainWindow.xaml">

<Application.Resources>

<ResourceDictionary>

<ResourceDictionary.MergedDictionaries>

<ResourceDictionary Source="Dictionary1.xaml"/>

</ResourceDictionary.MergedDictionaries>

</ResourceDictionary>

</Application.Resources>

</Application>

Dictionary.xaml

<ResourceDictionary xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml">

<Style

TargetType="Button" x:Key="Button">

<Setter Property="Height" Value="35"/>

<Setter Property="Width" Value="100"/>

<Setter Property="Padding" Value="8"/>

<Setter Property="Margin" Value="12"/>

<Setter Property="FontFamily" Value="Tahoma"/>

<Setter Property="FontSize" Value="12pt"/>

<Setter Property="Background" Value="#FFE0A9AF"/>

<Setter Property="Foreground" Value="Black"/>

</Style>

<Style TargetType="Grid" x:Key="Grid">

<Setter Property="Background" Value="#FFB5D5CA"/>

</Style>

</ResourceDictionary>

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЙ №4-6**

**Тема: Разработка страницы регистрации пользователей экспорт данных из базы данных в таблицу DataGrid.**

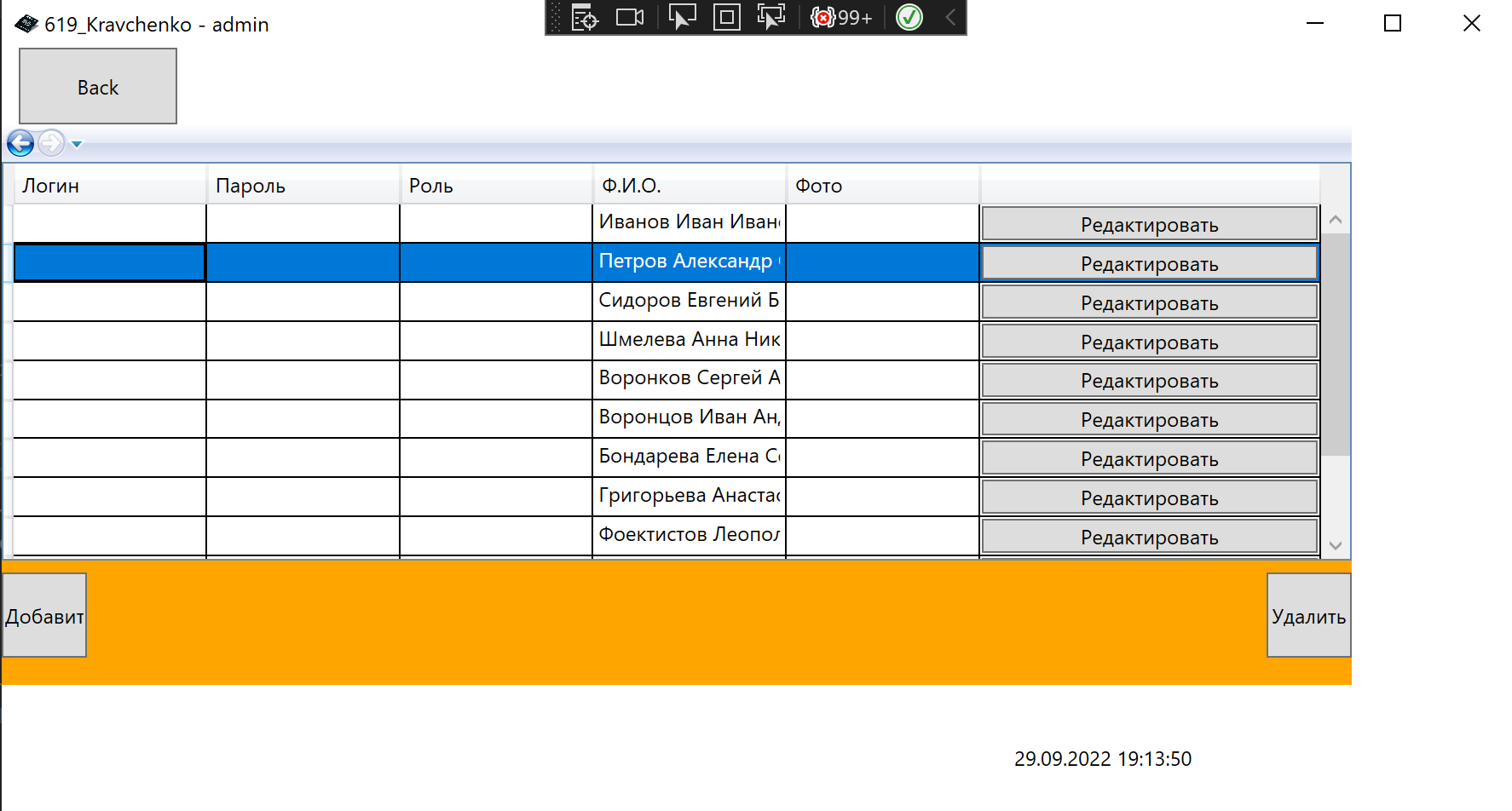


Рисунок 1. Таблица DataGrid

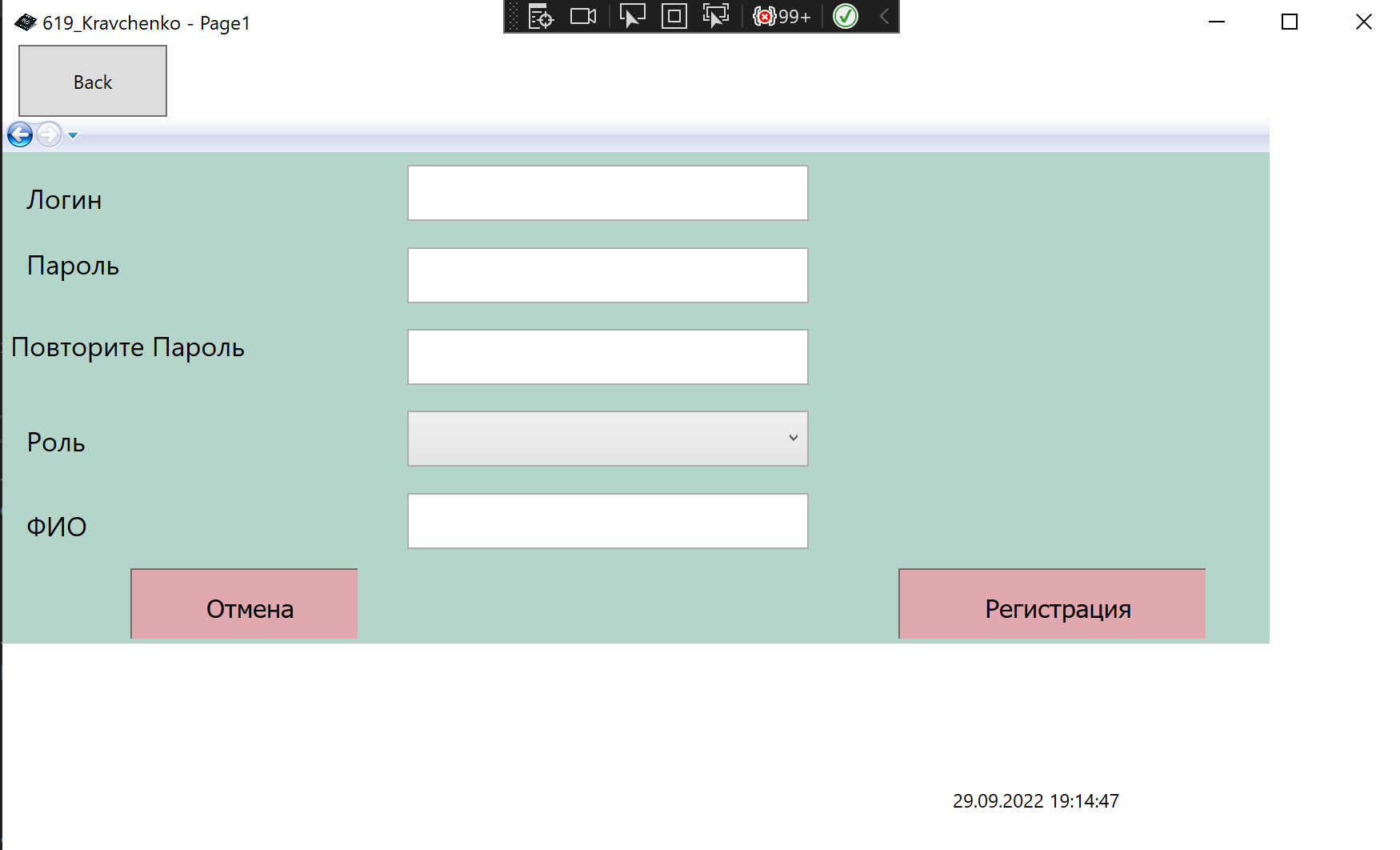


Рисунок 2. Регистрация

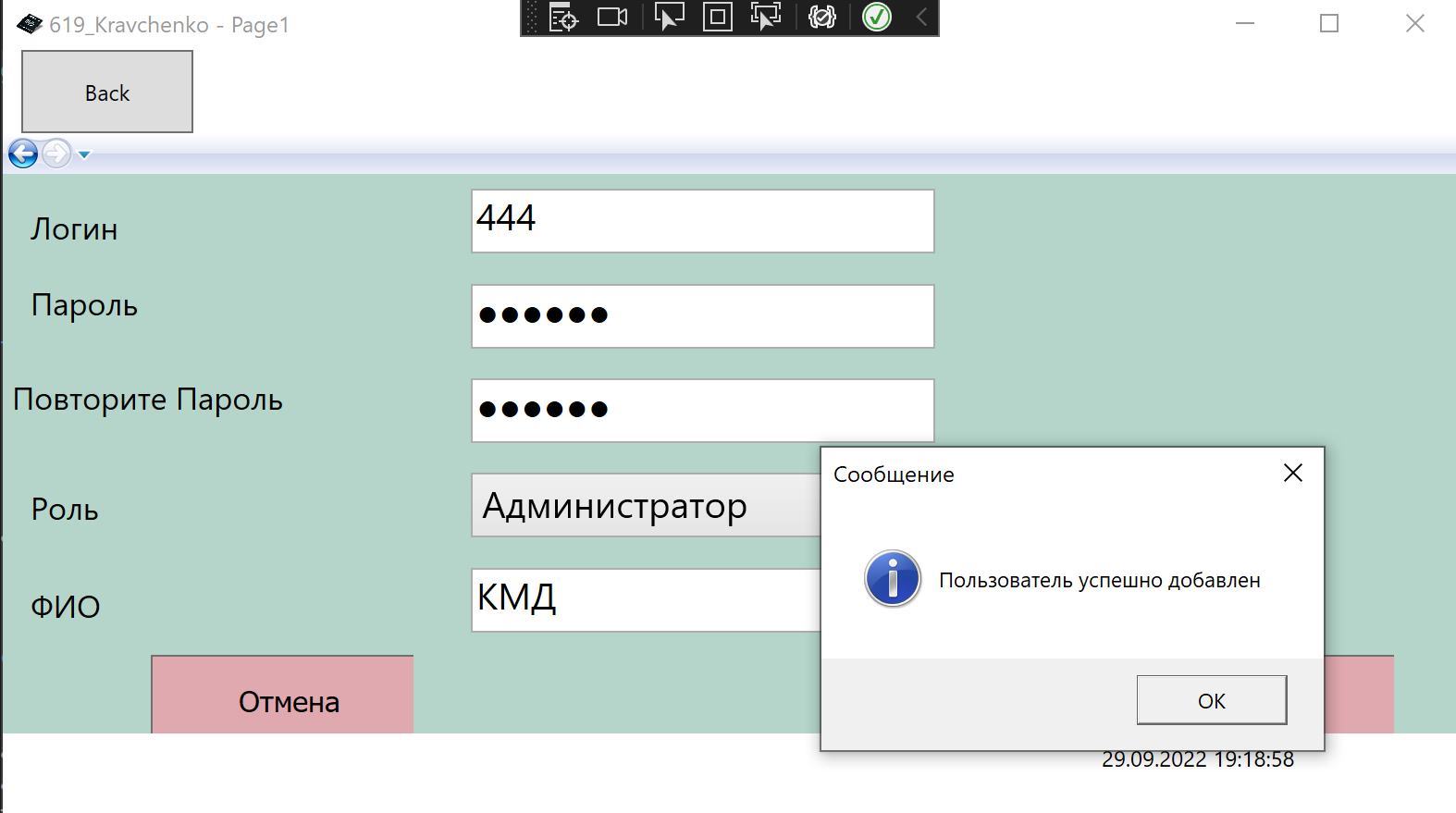


Рисунок 3. Пользователь зарегистрирован

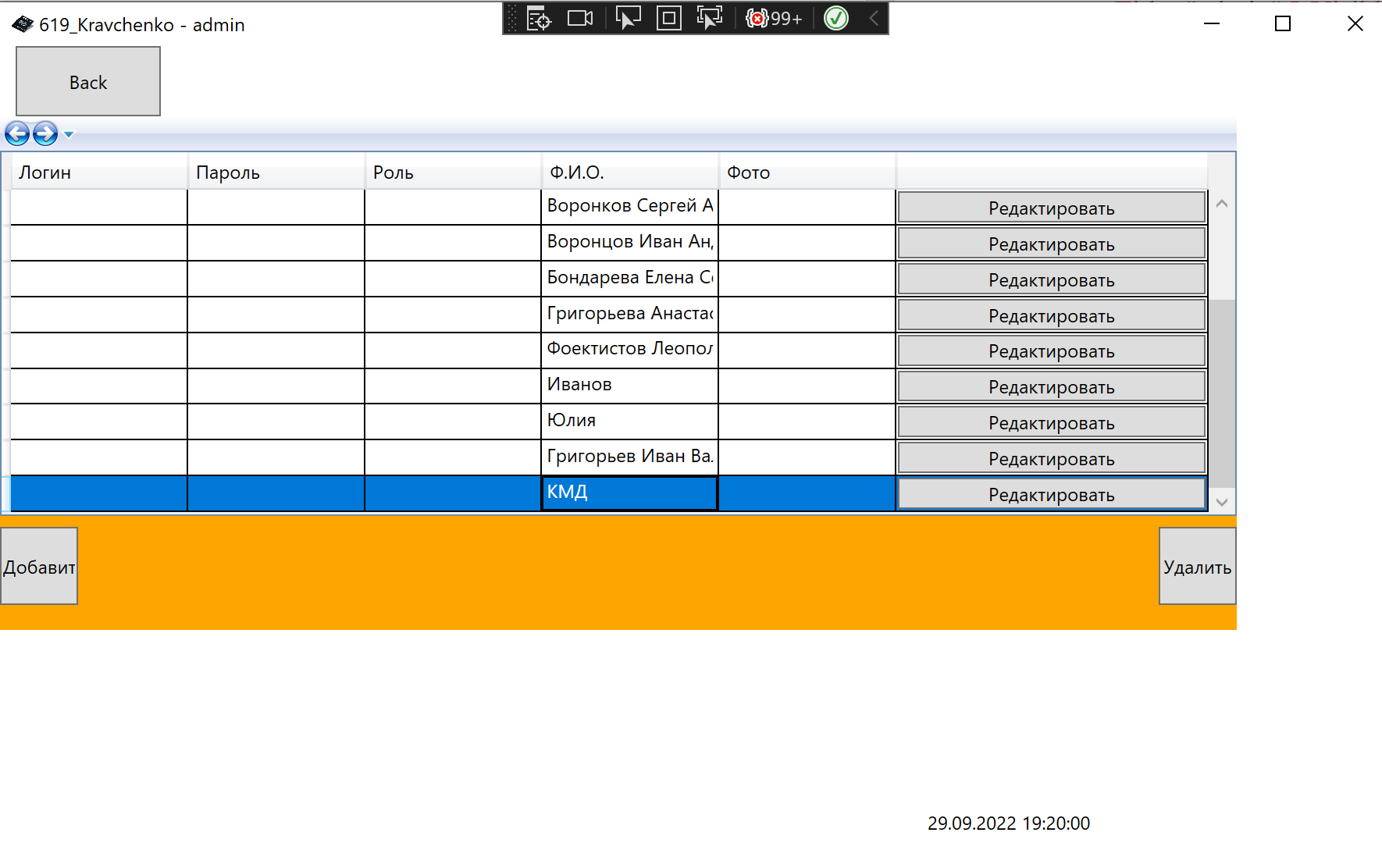


Рисунок 4. Добавился новый пользователь в таблицу после регистрации

Admin.xaml

<Page x:Class="\_619\_KRAVCHENKO.Pages.admin" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:\_619\_KRAVCHENKO.Pages"

mc:Ignorable="d"

d:DesignHeight="340" d:DesignWidth="800"

Title="admin" IsVisibleChanged="Page\_IsVisibleChanged">

<Grid Background="Aqua">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="267\*"/>

<RowDefinition Height="73\*"/>

</Grid.RowDefinitions>

<DataGrid Grid.Row="0" Name="DataGridUser" AutoGenerateColumns="False" IsReadOnly="True" Margin="0,0,0,73" Grid.RowSpan="2">

<DataGrid.Columns>

<DataGridTextColumn Header="Логин" Binding="{Binding login}" Width="1\*"/>

<DataGridTextColumn Header="Пароль" Binding="{Binding password}" Width="1\*"/>

<DataGridTextColumn Header="Роль" Binding="{Binding role}" Width="1\*"/>

<DataGridTextColumn Header="Ф.И.О." Binding="{Binding FIO}" Width="1\*"/>

<DataGridTextColumn Header="Фото" Binding="{Binding photo}" Width="1\*"/>

<DataGridTemplateColumn Width="200">

<DataGridTemplateColumn.CellTemplate>

<DataTemplate>

<Button Name="ButtonEdit" Click="ButtonEdit\_Click" Content="Редактировать"></Button>

</DataTemplate>

</DataGridTemplateColumn.CellTemplate>

</DataGridTemplateColumn>

</DataGrid.Columns>

</DataGrid>

<Button Name="ButtonAdd" Content="Добавить" ToolTip="Добавить" HorizontalAlignment="Left" Click="ButtonAdd\_Click" Width="50" Height="50" Grid.Row="1" VerticalAlignment="Top"/>

<Button Name="ButtonDel" Content="Удалить" ToolTip="Удалить" HorizontalAlignment="Right" Click="ButtonDel\_Click" Width="50" Height="50" Grid.Row="1" VerticalAlignment="Top"/>

</Grid>

</Grid>

</Page>

Admin.xaml.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using System.Windows.Forms;

using Button = System.Windows.Controls.Button;

using \_619\_Kravchenko;

namespace \_619\_KRAVCHENKO.Pages

{

/// <summary>

/// Логика взаимодействия для admin.xaml

/// </summary>

public partial class admin : Page

{

public admin()

{

InitializeComponent();

DataGridUser.ItemsSource = DataBaseConnection.entities.Users.ToList();

}

private void ButtonDel\_Click(object sender, RoutedEventArgs e)

{

DialogResult res = System.Windows.Forms.MessageBox.Show("Вы уверены, что хотите удалить выбранный элемент?", "Подтверждеие", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (res != System.Windows.Forms.DialogResult.No)

{

var delit = DataGridUser.SelectedItem as Users;

DataBaseConnection.entities.Users.Remove(delit);

DataBaseConnection.entities.SaveChanges();

DataGridUser.ItemsSource = DataBaseConnection.entities.Users.ToList();

}

}

private void ButtonAdd\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new Pages.Page3(null));

}

private void Page\_IsVisibleChanged(object sender, DependencyPropertyChangedEventArgs e)

{

if (Visibility == Visibility.Visible)

{

DataBaseConnection.entities.ChangeTracker.Entries().ToList().ForEach(x => x.Reload());

DataGridUser.ItemsSource = DataBaseConnection.entities.Users.ToList();

}

}

private void ButtonEdit\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new Pages.Page3((sender as Button).DataContext as Users));

}

}

}

Page2.xaml.cs

using \_619\_Kravchenko;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace \_619\_KRAVCHENKO.Pages

{

/// <summary>

/// Логика взаимодействия для Page2.xaml

/// </summary>

public partial class Page2 : Page

{

public Page2()

{

InitializeComponent();

}

private void Full\_Click(object sender, RoutedEventArgs e)

{

string login = LoginTBx.Text.Trim();

string pass = PasswwordPBx.Password.Trim();

if (login.Length == 0)

{

MessageBox.Show("Введите логин");

PasswwordPBx.Background = Brushes.White;

}

else if (pass.Length == 0)

{

MessageBox.Show("Введите пароль");

LoginTBx.Background = Brushes.White;

}

else

{

try

{

Users user = DataBaseConnection.entities.Users.Where(i => i.Login == LoginTBx.Text && i.Password == PasswwordPBx.Password).FirstOrDefault();

if (user != null)

{

LoginTBx.Text = "";

PasswwordPBx.Password = "";

switch (user.Role)

{

case "Администратор":

NavigationService.Navigate(new admin());

break;

case "Пользователь":

NavigationService.Navigate(new user());

break;

}

}

else

{

MessageBox.Show("Неправильный логин или пароль", "Неверные данные для входа", MessageBoxButton.OK, MessageBoxImage.Information);

PasswwordPBx.Password = "";

}

}

catch

{

MessageBox.Show("Обратитесь к администратору", "Ошибка входа", MessageBoxButton.OK, MessageBoxImage.Error);

}

}

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new Pages.Page1());

}

}

}

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЙ №7-9**

**Тема: МАНИПУЛИРОВАНИЕ ДАННЫМИ ИЗ БАЗЫ ДАННЫХ В ТАБЛИЦЕ DATAGRID.**

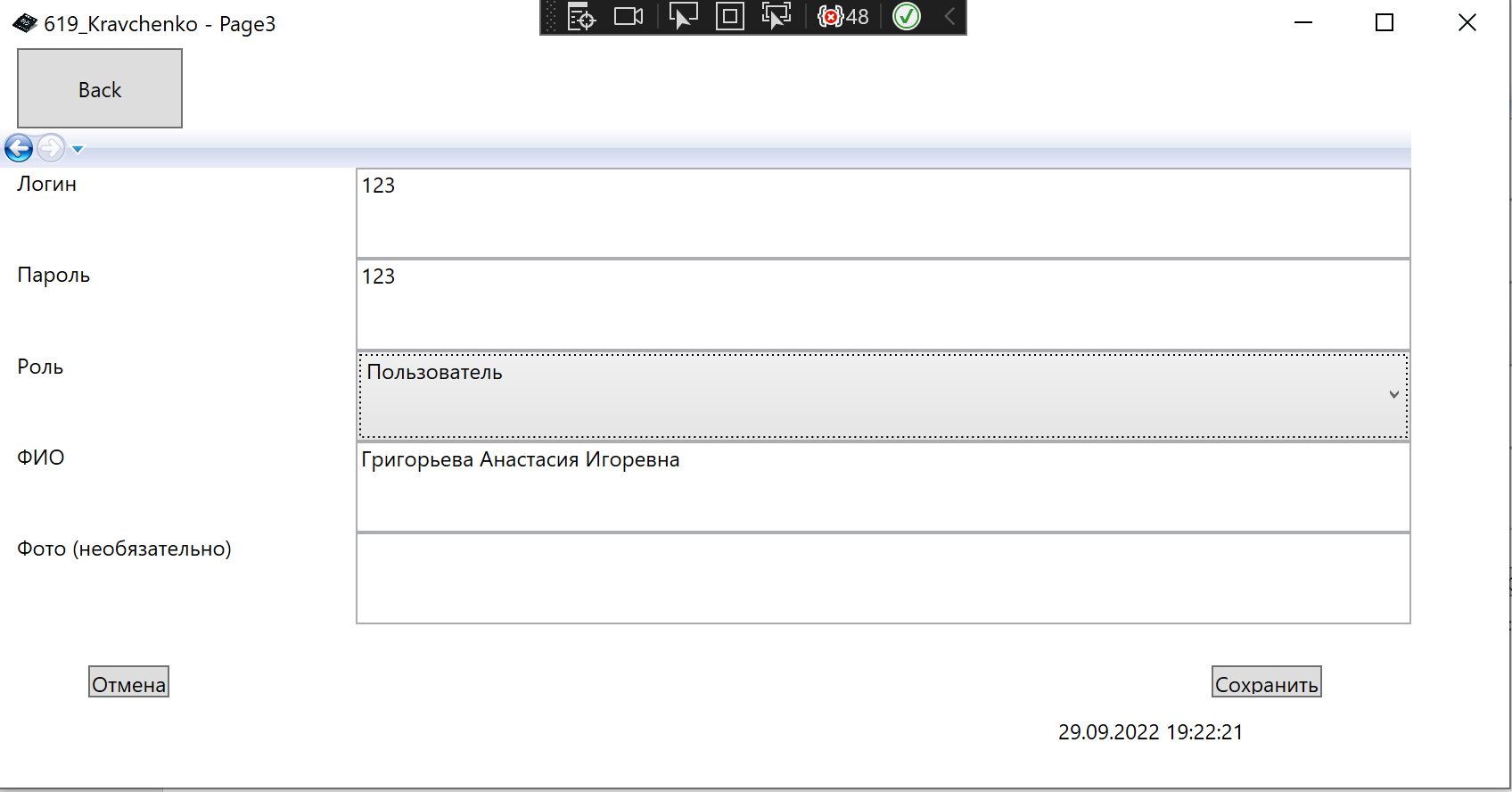


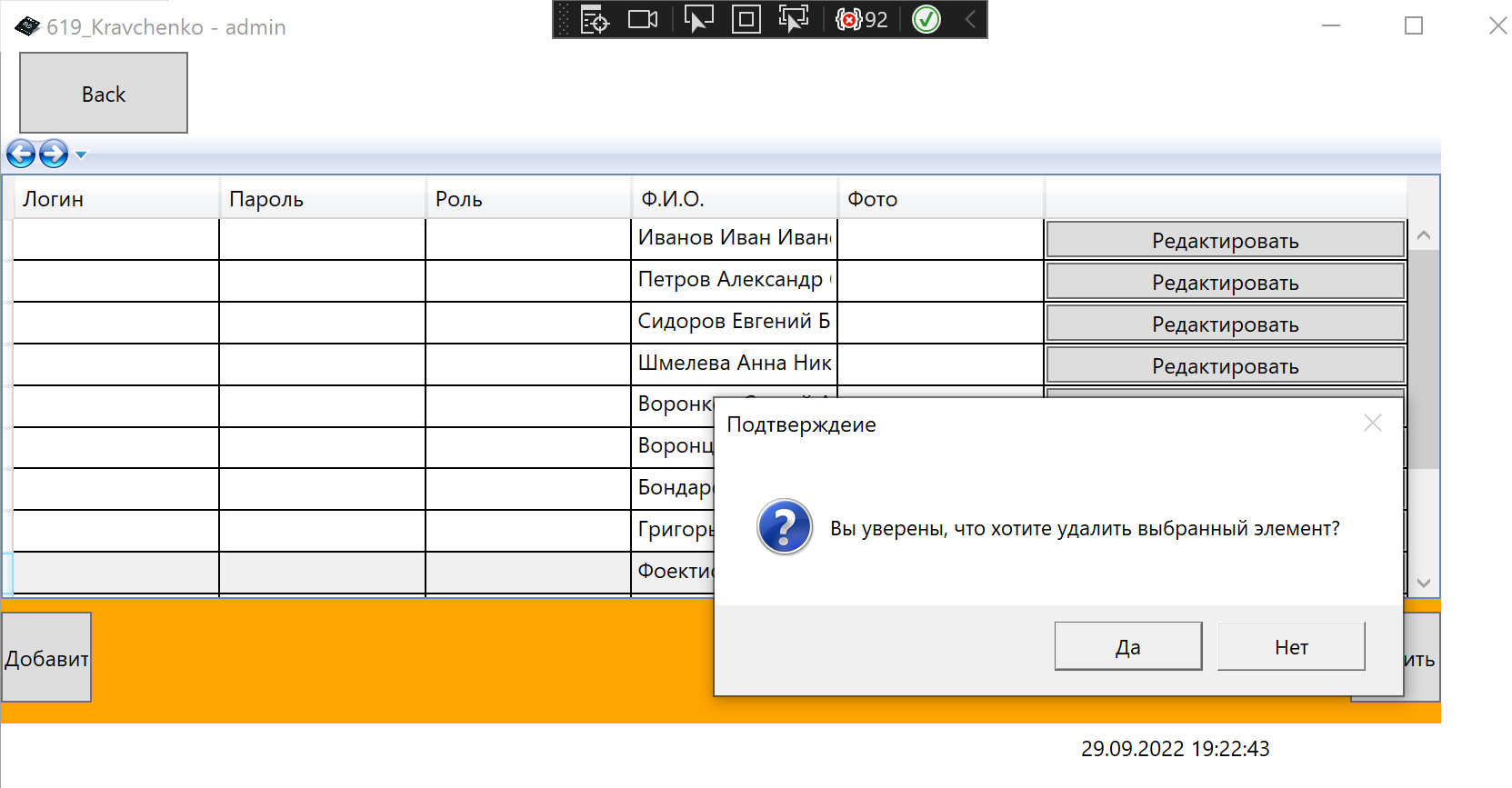
Рисунок 1. Редактирование данных

Рисунок 2. Удаление данных

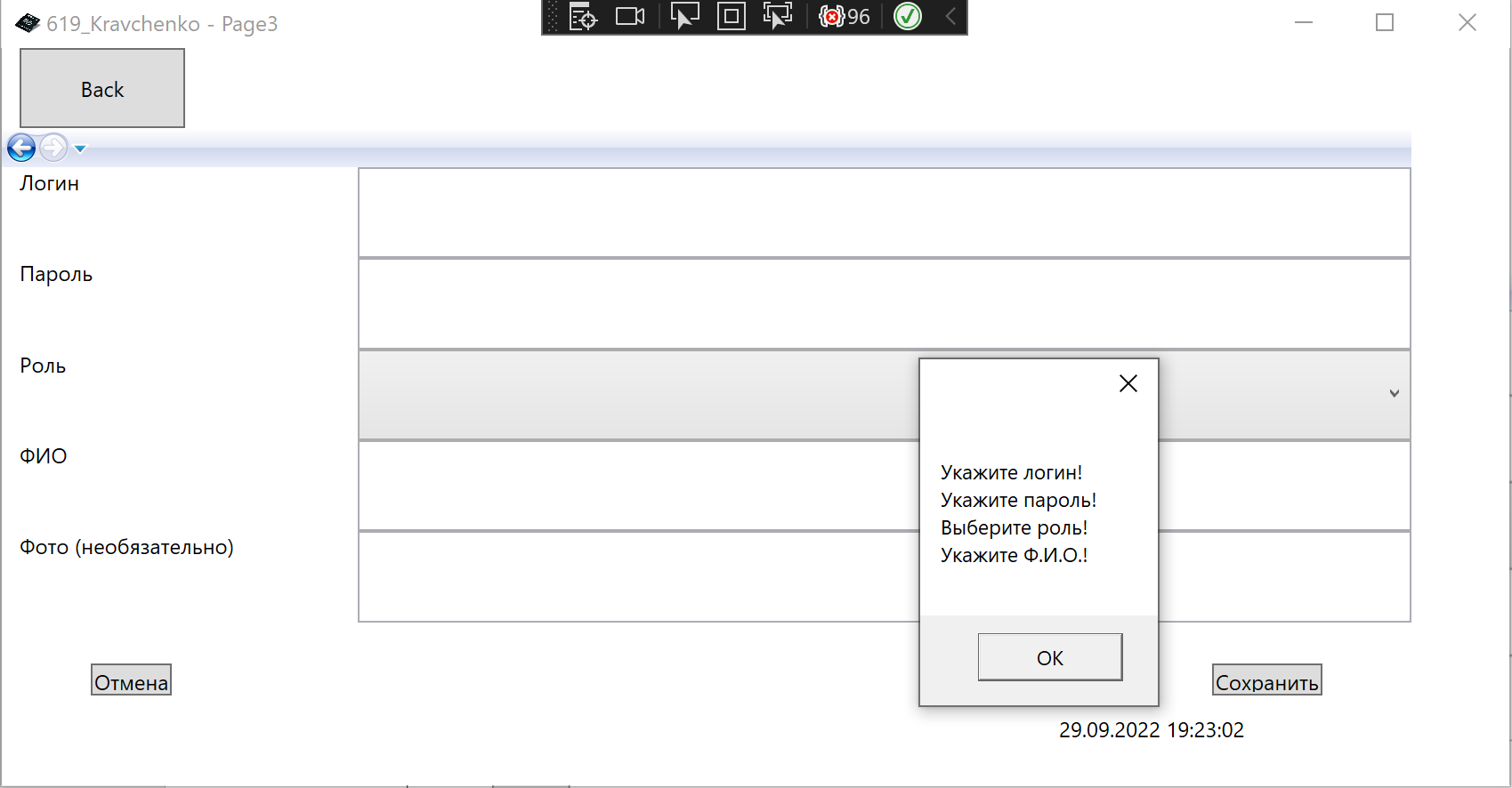


Рисунок 3. Проверка заполнения

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4. Сообщение об успешном сохранение

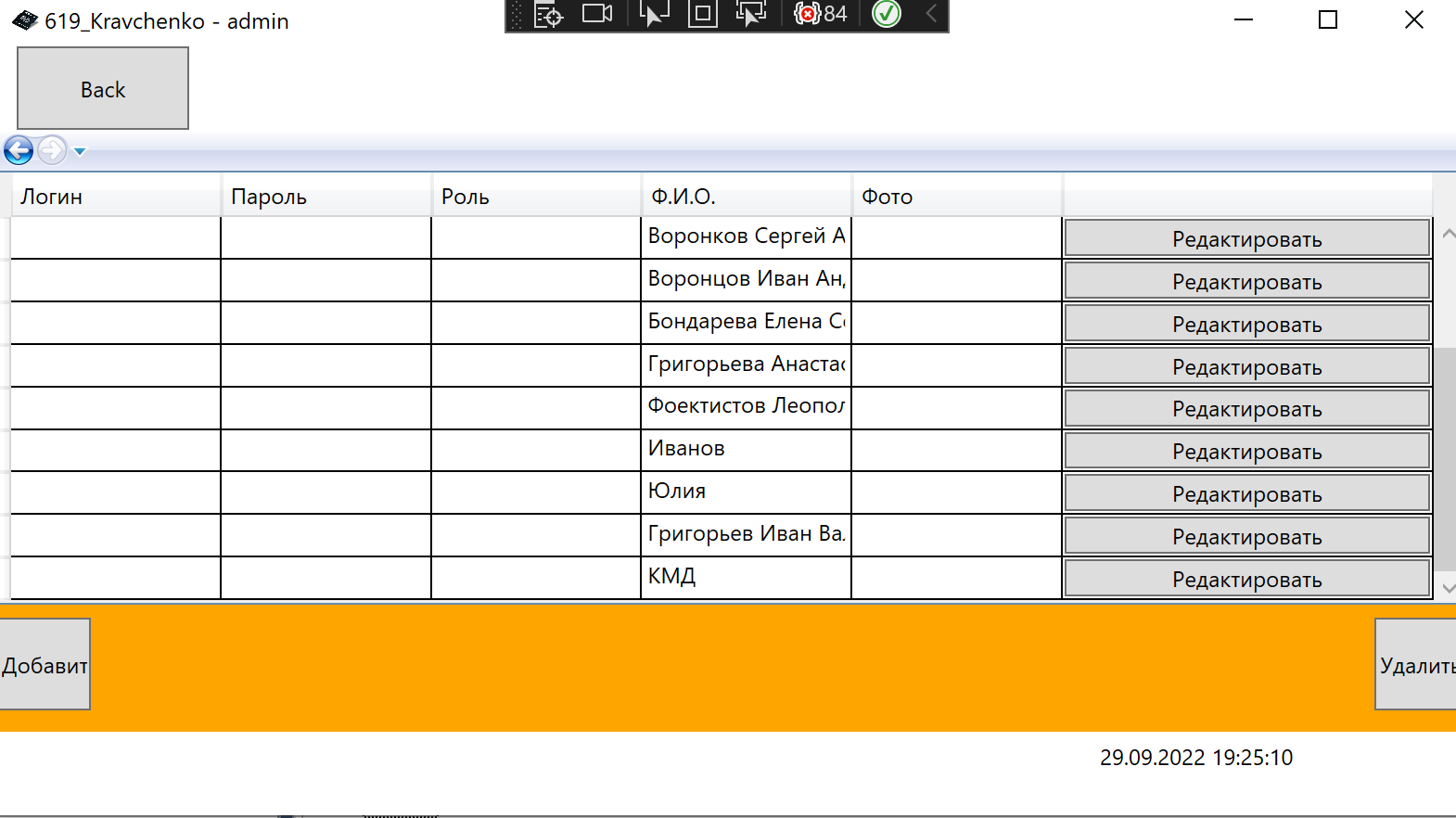


Рисунок 5. Обновление DataGrid после добавление данных

Page3.xaml

<Page x:Class="\_619\_KRAVCHENKO.Pages.Page3"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:\_619\_KRAVCHENKO.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Page3">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="200"/>

<ColumnDefinition Width="\*"/>

</Grid.ColumnDefinitions>

<TextBlock Text="Логин" Grid.Row="0" Grid.Column="0" Margin="10,0,0,0"/>

<TextBlock Text="Пароль" Grid.Row="1" Grid.Column="0" Margin="10,0,0,0"/>

<TextBlock Text="Роль" Grid.Row="2" Grid.Column="0" Margin="10,0,0,0"/>

<TextBlock Text="ФИО" Grid.Row="3" Grid.Column="0" Margin="10,0,0,0"/>

<TextBlock Text="Фото (необязательно)" Grid.Row="4" Grid.Column="0" Margin="10,0,0,0"/>

<TextBox ToolTip="Логин" Text="{Binding login}" Grid.Row="0" Grid.Column="1"/>

<TextBox ToolTip="Пароль" Text="{Binding password}" Grid.Row="1" Grid.Column="1"/>

<ComboBox ToolTip="Права" Name="cmbRole" SelectedItem="{Binding role}" Grid.Row="2" Grid.Column="1">

<ComboBoxItem Content="Администратор"/>

<ComboBoxItem Content="Пользователь"/>

</ComboBox>

<TextBox ToolTip="ФИО" Text="{Binding FIO}" Grid.Row="3" Grid.Column="1"/>

<TextBox ToolTip="Фото" Text="{Binding photo}" Grid.Row="4" Grid.Column="1"/>

<Button ToolTip="Сохранить" Name="ButtonSave" Content="Сохранить" Grid.Row="4" Click="ButtonSave\_Click" Margin="0,74.333,50,10" HorizontalAlignment="Right" Grid.Column="1" Grid.RowSpan="2"/>

<Button ToolTip="Отмена" Name="ButtonCancel" Content="Отмена" Grid.Row="4" Click="ButtonCancel\_Click" Margin="50,74.333,0,10" HorizontalAlignment="Left" Grid.RowSpan="2"/>

</Grid>

</Page>

Page3.xaml.cs

using \_619\_Kravchenko;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace \_619\_KRAVCHENKO.Pages

{

/// <summary>

/// Логика взаимодействия для Page3.xaml

/// </summary>

public partial class Page3 : Page

{

Users \_CurrentUser = new Users();

public Page3(Users selectedUser)

{

InitializeComponent();

if (selectedUser != null)

\_CurrentUser = selectedUser;

DataContext = \_CurrentUser;

}

private void ButtonSave\_Click(object sender, RoutedEventArgs e)

{

StringBuilder errors = new StringBuilder();

if (string.IsNullOrWhiteSpace(\_CurrentUser.Login))

errors.AppendLine("Укажите логин!");

if (string.IsNullOrWhiteSpace(\_CurrentUser.Password))

errors.AppendLine("Укажите пароль!");

if ((\_CurrentUser.Role == null) || (cmbRole.Text == ""))

errors.AppendLine("Выберите роль!");

else

\_CurrentUser.Role = cmbRole.Text;

if (string.IsNullOrWhiteSpace(\_CurrentUser.FIO))

errors.AppendLine("Укажите ФИО!");

if (errors.Length > 0)

{

MessageBox.Show(errors.ToString());

return;

}

if (\_CurrentUser.ID == 0)

DataBaseConnection.entities.Users.Add(\_CurrentUser);

try

{

DataBaseConnection.entities.SaveChanges();

MessageBox.Show("Данные успешно сохранены!");

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString());

}

}

private void ButtonCancel\_Click(object sender, RoutedEventArgs e)

{

NavigationService.GoBack();

}

}

}

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

# ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЙ №10-12

**Тема: ЭКСПОРТ ДАННЫХ ИЗ БАЗЫ ДАННЫХ В СПИСОК LISTVIEW. ПОИСК И ФИЛЬТРАЦИЯ ДАННЫХ**

Студенты: Кравченко М. Д.

Группа: 4ИСИП-619

Преподаватель: Т.Г.Аксёнова

Дата: 26.09.2022

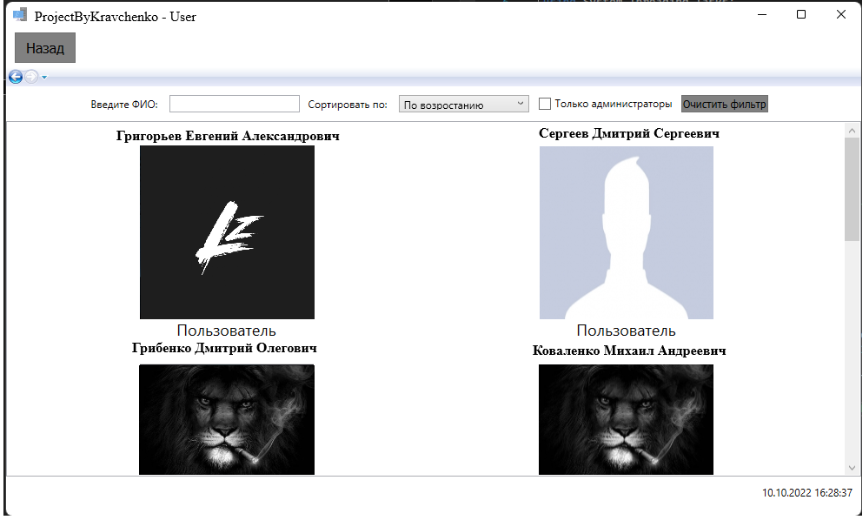


Рисунок 15. Страница Пользователя

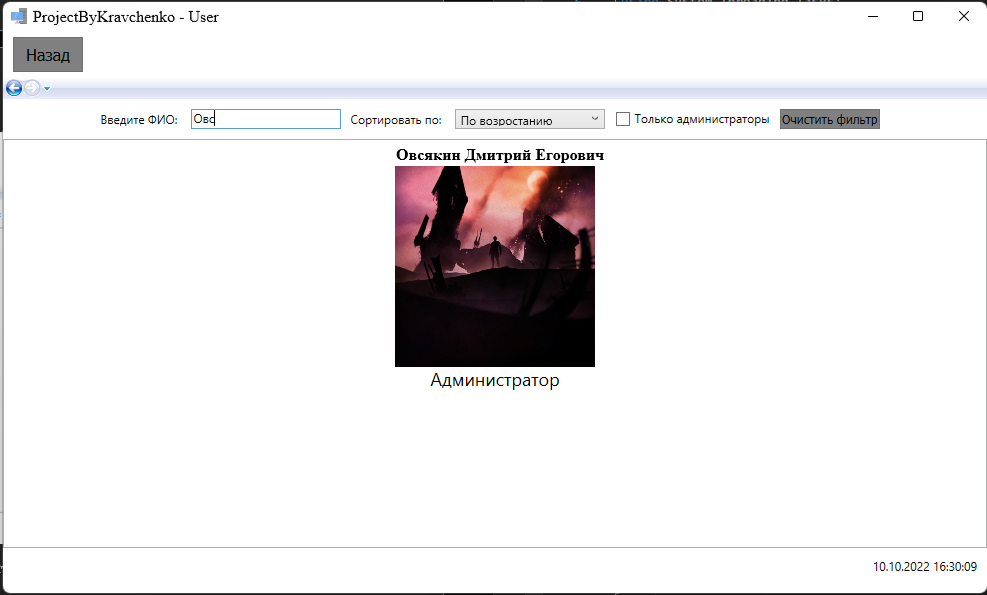


Рисунок 16. Поиск по ФИО

**Листинг программы**

User.xaml

<Page x:Class="\_619\_KRAVCHENKO.Pages.Users"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:\_619\_KRAVCHENKO.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="Users">

<Grid Background="white">

<Grid.RowDefinitions>

<RowDefinition Height="Auto"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<WrapPanel Grid.Row="0" HorizontalAlignment="Center">

<TextBlock Text="Введите ФИО: " VerticalAlignment="Center"/>

<TextBox Name="FindText" Height="20" Width="150" FontSize="12px" TextChanged="FindText\_TextChanged"/>

<TextBlock Text="Сортировать по: " VerticalAlignment="Center"/>

<ComboBox Height="20" Width="150" Name="Cmb" SelectionChanged="ComboBox\_SelectionChanged">

<ComboBoxItem>

<TextBlock Text="По возростанию"/>

</ComboBoxItem>

<ComboBoxItem>

<TextBlock Text="По убыванию"/>

</ComboBoxItem>

</ComboBox>

<CheckBox Content="Только администраторы"

VerticalAlignment="Center" Name="CheckAdm"

Checked="CheckBox\_Checked" Unchecked="CheckBox\_Unchecked"/>

<Button Content="Очистить фильтр" Height="20" Width="100" FontSize="12px" Click="Button\_Click"/>

</WrapPanel>

<ListView Name="ListUser" Grid.Row="1"

ScrollViewer.HorizontalScrollBarVisibility="Disabled"

HorizontalContentAlignment="Center">

<ListView.ItemsPanel>

<ItemsPanelTemplate>

<WrapPanel Orientation="Horizontal" HorizontalAlignment="Center"/>

</ItemsPanelTemplate>

</ListView.ItemsPanel>

<ListView.ItemTemplate>

<DataTemplate>

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="auto"/>

<RowDefinition Height="200"/>

<RowDefinition Height="auto"/>

</Grid.RowDefinitions>

<Image Width="200" Grid.Row="1" Stretch="UniformToFill"

HorizontalAlignment="Center" VerticalAlignment="Center">

<Image.Source>

<Binding Path="Photo">

<Binding.TargetNullValue>

<ImageSource>/Img/DefaultPhoto.png</ImageSource>

</Binding.TargetNullValue>

</Binding>

</Image.Source>

</Image>

<TextBlock VerticalAlignment="Center" TextAlignment="Center" Width="450"

Grid.Row="0" TextWrapping="Wrap" Text="{Binding FIO}"

HorizontalAlignment="Center" FontSize="18" FontWeight="Bold"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Center" Width="200"

Grid.Row="2" HorizontalAlignment="Center" FontSize="18"

Text="{Binding Role}"/>

</Grid>

</DataTemplate>

</ListView.ItemTemplate>

</ListView>

</Grid>

</Page>

User.xaml.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace \_619\_KRAVCHENKO.Pages

{

/// <summary>

/// Логика взаимодействия для Users.xaml

/// </summary>

public partial class Users : Page

{

public Users()

{

InitializeComponent();

var currentUsers = Entities.GetContext().USER.ToList();

ListUser.ItemsSource = currentUsers;

Cmb.SelectedIndex = 0;

CheckAdm.IsChecked = false;

UpdateUsers();

}

private void UpdateUsers()

{

var currentUsers = Entities.GetContext().USER.ToList();

currentUsers = currentUsers.Where

(x => x.FIO.ToLower().Contains(FindText.Text.ToLower())).ToList();

if (CheckAdm.IsChecked.Value)

currentUsers = currentUsers.Where(x => x.Role.Contains("Пользователь")).ToList();

if (Cmb.SelectedIndex == 0)

ListUser.ItemsSource = currentUsers.OrderBy(x => x.FIO).ToList();

else

ListUser.ItemsSource = currentUsers.OrderByDescending(x => x.FIO).ToList();

}

private void FindText\_TextChanged(object sender, TextChangedEventArgs e)

{

UpdateUsers();

}

private void ComboBox\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

UpdateUsers();

}

private void CheckBox\_Checked(object sender, RoutedEventArgs e)

{

UpdateUsers();

}

private void CheckBox\_Unchecked(object sender, RoutedEventArgs e)

{

UpdateUsers();

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

Cmb.SelectedIndex = 0;

CheckAdm.IsChecked = false;

FindText.Text = "";

UpdateUsers();

}

}

}

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

# ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЙ №10-12

**Тема: ЭКСПОРТ ДАННЫХ ИЗ БАЗЫ ДАННЫХ В СПИСОК LISTVIEW. ПОИСК И ФИЛЬТРАЦИЯ ДАННЫХ**

Студенты: Кравченко М. Д.

Группа: 4ИСИП-619

Преподаватель: Т.Г.Аксёнова

Дата: 26.09.2022

# Комаров Евгений Дмитриевич

|  |  |
| --- | --- |
| **Категория** | **Сумма расходов** |
| Транспорт | 200 руб. |
| Кино | 0 руб. |
| Ресторан | 0 руб. |
| ЖКХ | 0 руб. |
| Налог | 0 руб. |

Самый дорогостоящий платеж - Транспорт за 200 руб. от 01.01.2020 0:00:00

Самый дешевый платеж - Транспорт за 200 руб. от 01.01.2020 0:00:00

# Пахомов Александр Платонович

|  |  |
| --- | --- |
| **Категория** | **Сумма расходов** |
| Транспорт | 0 руб. |
| Кино | 1000 руб. |
| Ресторан | 0 руб. |
| ЖКХ | 0 руб. |
| Налог | 0 руб. |

Самый дорогостоящий платеж - Октяборькое за 1000 руб. от 02.01.2020 0:00:00

Самый дешевый платеж - Октяборькое за 1000 руб. от 02.01.2020 0:00:00

# Гордеев Дмитрий Дмитриевич

|  |  |
| --- | --- |
| **Категория** | **Сумма расходов** |
| Транспорт | 0 руб. |
| Кино | 0 руб. |
| Ресторан | 50000 руб. |
| ЖКХ | 0 руб. |
| Налог | 0 руб. |

Самый дорогостоящий платеж - Русский за 50000 руб. от 03.02.2020 0:00:00

Самый дешевый платеж - Русский за 50000 руб. от 03.02.2020 0:00:00

# Ефремов Максим Евгеньевич

|  |  |
| --- | --- |
| **Категория** | **Сумма расходов** |
| Транспорт | 0 руб. |
| Кино | 0 руб. |
| Ресторан | 0 руб. |
| ЖКХ | 10000 руб. |
| Налог | 0 руб. |

Самый дорогостоящий платеж - ЖКХ за 10000 руб. от 04.01.2020 0:00:00

Самый дешевый платеж - ЖКХ за 10000 руб. от 04.01.2020 0:00:00

# Косарев Кирилл Дмитриевич

|  |  |
| --- | --- |
| **Категория** | **Сумма расходов** |
| Транспорт | 0 руб. |
| Кино | 0 руб. |
| Ресторан | 0 руб. |
| ЖКХ | 0 руб. |
| Налог | 50000 руб. |

Самый дорогостоящий платеж - Налог за 50000 руб. от 05.01.2020 0:00:00

Самый дешевый платеж - Налог за 50000 руб. от 05.01.2020 0:00:00

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.07 «Информационные системы и программирование»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Выполнил:

обучающийся учебной группы №\_\_\_619\_\_\_

М.Д. Кравченко

*(подпись) (И.О. Фамилия)*

*(подпись) (И.О. Фамилия)*

Проверили:

Руководители практики от колледжа:

Сибирев И. В.

*(должность) (И.О. Фамилия)*

*(оценка) (подпись)*

**Москва**

**20­­­­22**

**2022г**

**Лабораторная работа №1-2**

**Цель работы:**

**Задание:** Спроектируйте базу данных. Создайте необходимые объекты базы данных

**Ход работы:** Создаем базу данных



**Таблица 1**

|  |  |  |
| --- | --- | --- |
| **Сотрудники** | | |
| **Название** | **Тип** | **Разрешить** |
| Код страдника | Int | + |
| Фамилия | varchar(50) | - |
| Имя | varchar(50) | - |
| Отчество | varchar(50) | - |
| Оклад | Int | - |

Таблица «Сотрудники» хранит в себе информацию: ФИО сотрудника, его оклад

**Таблица 2**

|  |  |  |
| --- | --- | --- |
| **Работы** | | |
| **Название** | **Тип** | **Разрешить** |
| Код страдника | Int | + |
| Код вида | varchar(50) | - |
| Дата начал | Date | - |
| Дата окончания | Date | - |

Таблица хранит в себе информацию о доп.работе сотрудника, вид его работы, код отрдника.

**Таблица 3**

|  |  |  |
| --- | --- | --- |
| **Работы** | | |
| **Название** | **Тип** | **Разрешить** |
| Код страдника | Int | + |
| Описание | varchar(50) | - |
| Оплата за день | Money | - |

Таблица хранит в себе информацию о виде работы и ее оплаты

**Вывод:** научились проектировать базу данных, создали ER-диаграмму и описали сущности

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.07 «Информационные системы и программирование»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Выполнил:

обучающийся учебной группы №\_\_\_619\_\_\_

М.Д. Кравченко

*(подпись) (И.О. Фамилия)*

*(подпись) (И.О. Фамилия)*

Проверили:

Руководители практики от колледжа:

Сибирев И. В.

*(должность) (И.О. Фамилия)*

*(оценка) (подпись)*

**Москва**

**20­­­­22**

**Лабораторная работа №4**

**Цель работы:** Изучить создание запросов и представлений, использование механизма объединений.

**Задание:**

• Для базы данных согласно варианту задания (из Лабораторной

работы №1) написать следующие SQL запросы из Приложения 2

• Реализовать запросы из лабораторной работы 1 по варианту,

оформить два из них в виде представлений.

**Ход работы:** Создаем зпросы

a)

select \* from [dbo].[Вид\_работы]

select \* from [dbo].[Работы]

select \* from [dbo].[Сотрудники]

b)

SELECT [ID\_Sotrudnika],[Last\_name],[Name],[Patronic],[ID\_vida], [opisanie]

FROM [dbo].[Сотрудники]

INNER JOIN [dbo].[Вид\_работы] ON [ID\_vida] = [ID\_Sotrudnika]

c)

select [Last\_name], [Name], [Patronic] from [dbo].[Сотрудники]

order by [Last\_name], [Name], [Patronic]

**Вывод:** Научились создавать запросы для Бд

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.07 «Информационные системы и программирование»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Выполнил:

обучающийся учебной группы №\_\_\_619\_\_\_

М.Д. Кравченко

*(подпись) (И.О. Фамилия)*

*(подпись) (И.О. Фамилия)*

Проверили:

Руководители практики от колледжа:

Сибирев И. В.

*(должность) (И.О. Фамилия)*

*(оценка) (подпись)*

**Москва**

**20­­­­22**

**Цель работы:**

Освоить методы переноса и генерации скриптов базы данных

**Ход работы:**

Для Базы данных из лабораторной работы 2 реализовать перенос

базы данных через механизм резервного копирования, сгенерировать

скрипт создания базы данных.

Создаем скрипт бд

USE [master]

GO

/\*\*\*\*\*\* Object: Database [KravchemkoM1] Script Date: 27.10.2022 12:36:02 \*\*\*\*\*\*/

CREATE DATABASE [KravchemkoM1]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'KravchemkoM1', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSSQL\DATA\KravchemkoM1.mdf' , SIZE = 5120KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )

LOG ON

( NAME = N'KravchemkoM1\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSSQL\DATA\KravchemkoM1\_log.ldf' , SIZE = 1024KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)

GO

ALTER DATABASE [KravchemkoM1] SET COMPATIBILITY\_LEVEL = 120

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [KravchemkoM1].[dbo].[sp\_fulltext\_database] @action = 'enable'

end

GO

ALTER DATABASE [KravchemkoM1] SET ANSI\_NULL\_DEFAULT OFF

GO

ALTER DATABASE [KravchemkoM1] SET ANSI\_NULLS OFF

GO

ALTER DATABASE [KravchemkoM1] SET ANSI\_PADDING OFF

GO

ALTER DATABASE [KravchemkoM1] SET ANSI\_WARNINGS OFF

GO

ALTER DATABASE [KravchemkoM1] SET ARITHABORT OFF

GO

ALTER DATABASE [KravchemkoM1] SET AUTO\_CLOSE OFF

GO

ALTER DATABASE [KravchemkoM1] SET AUTO\_SHRINK OFF

GO

ALTER DATABASE [KravchemkoM1] SET AUTO\_UPDATE\_STATISTICS ON

GO

ALTER DATABASE [KravchemkoM1] SET CURSOR\_CLOSE\_ON\_COMMIT OFF

GO

ALTER DATABASE [KravchemkoM1] SET CURSOR\_DEFAULT GLOBAL

GO

ALTER DATABASE [KravchemkoM1] SET CONCAT\_NULL\_YIELDS\_NULL OFF

GO

ALTER DATABASE [KravchemkoM1] SET NUMERIC\_ROUNDABORT OFF

GO

ALTER DATABASE [KravchemkoM1] SET QUOTED\_IDENTIFIER OFF

GO

ALTER DATABASE [KravchemkoM1] SET RECURSIVE\_TRIGGERS OFF

GO

ALTER DATABASE [KravchemkoM1] SET DISABLE\_BROKER

GO

ALTER DATABASE [KravchemkoM1] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [KravchemkoM1] SET DATE\_CORRELATION\_OPTIMIZATION OFF

GO

ALTER DATABASE [KravchemkoM1] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [KravchemkoM1] SET ALLOW\_SNAPSHOT\_ISOLATION OFF

GO

ALTER DATABASE [KravchemkoM1] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [KravchemkoM1] SET READ\_COMMITTED\_SNAPSHOT OFF

GO

ALTER DATABASE [KravchemkoM1] SET HONOR\_BROKER\_PRIORITY OFF

GO

ALTER DATABASE [KravchemkoM1] SET RECOVERY SIMPLE

GO

ALTER DATABASE [KravchemkoM1] SET MULTI\_USER

GO

ALTER DATABASE [KravchemkoM1] SET PAGE\_VERIFY CHECKSUM

GO

ALTER DATABASE [KravchemkoM1] SET DB\_CHAINING OFF

GO

ALTER DATABASE [KravchemkoM1] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )

GO

ALTER DATABASE [KravchemkoM1] SET TARGET\_RECOVERY\_TIME = 0 SECONDS

GO

ALTER DATABASE [KravchemkoM1] SET DELAYED\_DURABILITY = DISABLED

GO

USE [KravchemkoM1]

GO

/\*\*\*\*\*\* Object: Table [dbo].[vid\_rabot] Script Date: 27.10.2022 12:36:03 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[vid\_rabot](

[ID\_vida] [int] NOT NULL,

[opisanie1] [varchar](1) NULL,

[pay\_for\_work] [int] NULL,

UNIQUE NONCLUSTERED

(

[pay\_for\_work] 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

SET ANSI\_PADDING OFF

**Вывод:** Сделали бакап БД

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.07 «Информационные системы и программирование»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Выполнил:

обучающийся учебной группы №\_\_\_619\_\_\_

М.Д. Кравченко

*(подпись) (И.О. Фамилия)*

*(подпись) (И.О. Фамилия)*

Проверили:

Руководители практики от колледжа:

Сибирев И. В.

*(должность) (И.О. Фамилия)*

*(оценка) (подпись)*

**Москва**

**20­­­­22**

**Лабораторная работа №4**

**Цель работы:** Изучение механизма динамических запросов с использованием

хранимых процедур и функций

**Задание:**

Для базы данных согласно варианту задания (из Лабораторной работы

№2) написать:

**Ход работы:** Создаем хранимые процедуры

a)

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Author,,Name>

-- Create date: <Create Date,,>

-- Description: <Description,,>

-- =============================================

alter PROCEDURE zarplata

-- Add the parameters for the stored procedure here

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

-- Insert statements for procedure here

select SUM([Salary]) + (select sum([pay\_for\_work]) from [dbo].[Вид\_работы])

from [dbo].[Сотрудники]

END

GO

**Вывод:** Повторил создание хромых процедур для Бд

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.07 «Информационные системы и программирование»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Выполнил:

обучающийся учебной группы №\_\_\_519\_\_\_

М.Д. Кравченко

*(подпись) (И.О. Фамилия)*

*(подпись) (И.О. Фамилия)*

Проверили:

Руководители практики от колледжа:

Сибирев И. В.

*(должность) (И.О. Фамилия)*

*(оценка) (подпись)*

**Москва**

**20­­­­22**

**Цель работы:**

Изучение механизмов обеспечения целостности данных SQL

**Задание:**

Для базы данных из лабораторной работы 2

1 Добавить ограничение DEFAULT

2 Создать триггеры индикаторы:

Добавление, выводящий на экран сообщение "Запись добавлена"

при добавлении новой записи в таблицу

Изменение, выводящий на экран с сообщение "Запись изменена"

при изменении записи в таблице

Удаление, выводящий на экран с сообщение "Запись удалена" при

удалении записи из таблицы

3 Создать триггер по удалении, когда данные удаляются сразу из двух связанных таблиц.

4 Проверить и продемонстрировать созданных триггеров.

**Ход работы:** Создаем триггер

1. Создаем новую таблицу

create table vid\_rabot([ID\_vida] [int] not null,

[opisanie1] [varchar] DEFAULT (0),

[pay\_for\_work] [int] UNIQUE)

1. Новая запись

Create TRIGGER tr\_add on [dbo].[Сотрудники] after insert

as

begin SET NOCOUNT ON;

PRINT 'НОВАЯ ЗАПИСЬ'

END

GO

2. Изменения

CREATE TRIGGER tr\_change ON [dbo].[Сотрудники]

AFTER UPDATE AS BEGIn

SET NOCOUNT ON;

PRINT 'Запись изменена'

END

1. Удаление записи

CREATE TRIGGER tr\_DELETE ON [dbo].[Работы]

AFTER DELETE AS BEGIN

SET NOCOUNT ON;

PRINT 'Запись удалена'

END

GO

**Вывод:** Создали триггеры в бд

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

09.02.07 Информационные системы и программирование

## **ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ Профессиональный модуль - УП.01.01 Учебная практика**

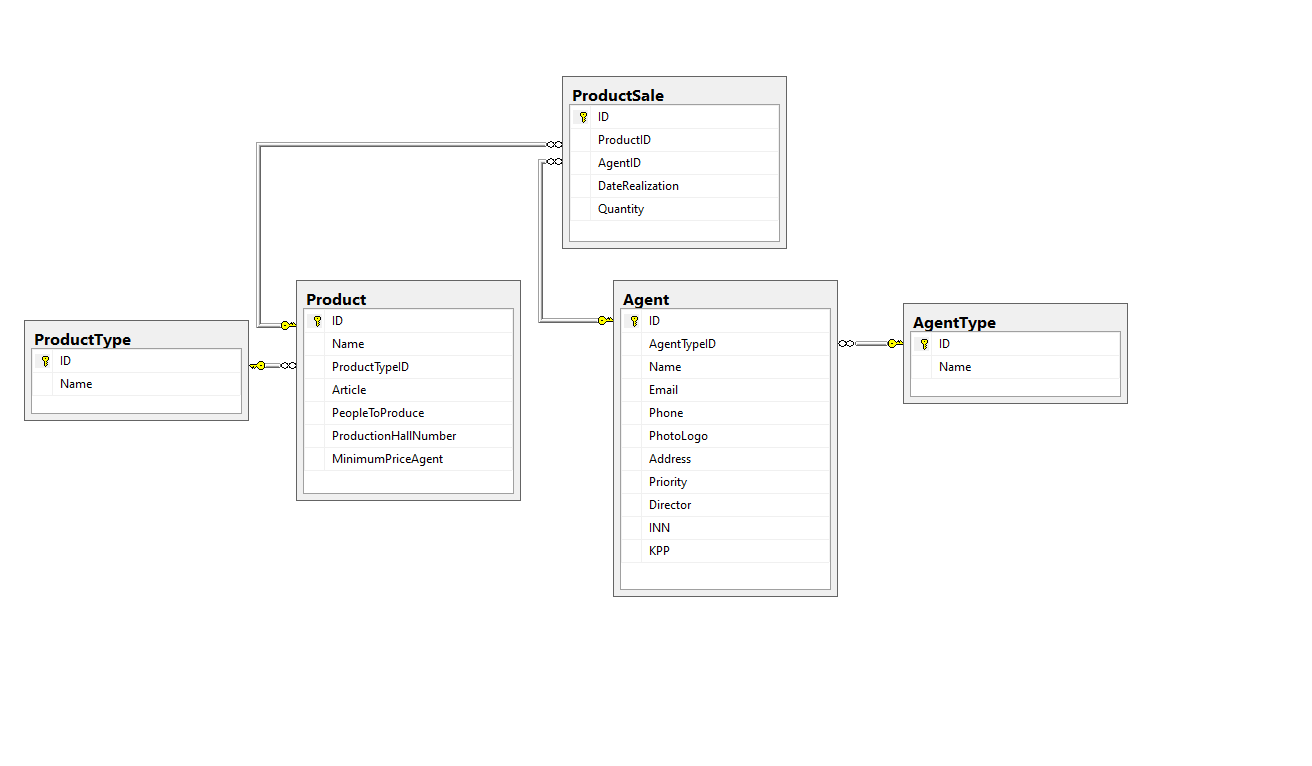
Выполнила: студент учебной группы 4ИСИП-619

Кравченко М. Д.

Руководитель практики от колледжа: Альшакова.Е.Л

**Москва – 2021**

**Цель работы:** отформатировать Excel файлы, создать базу данных и импортировать

 Рисунок 1. Диаграмма базы данных

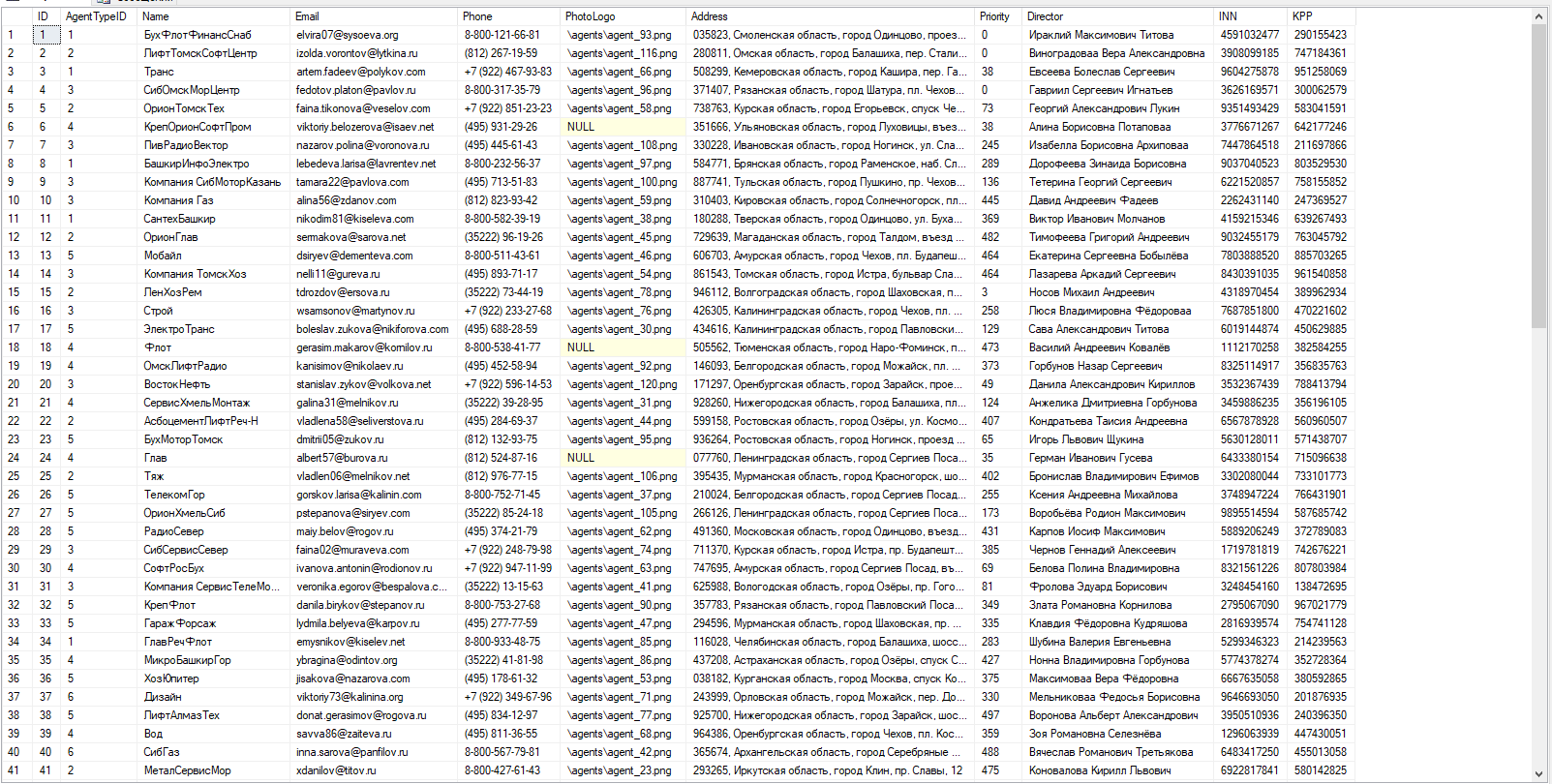


Рисунок 2. Таблица agent

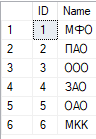


Рисунок 3. Таблица AgentType

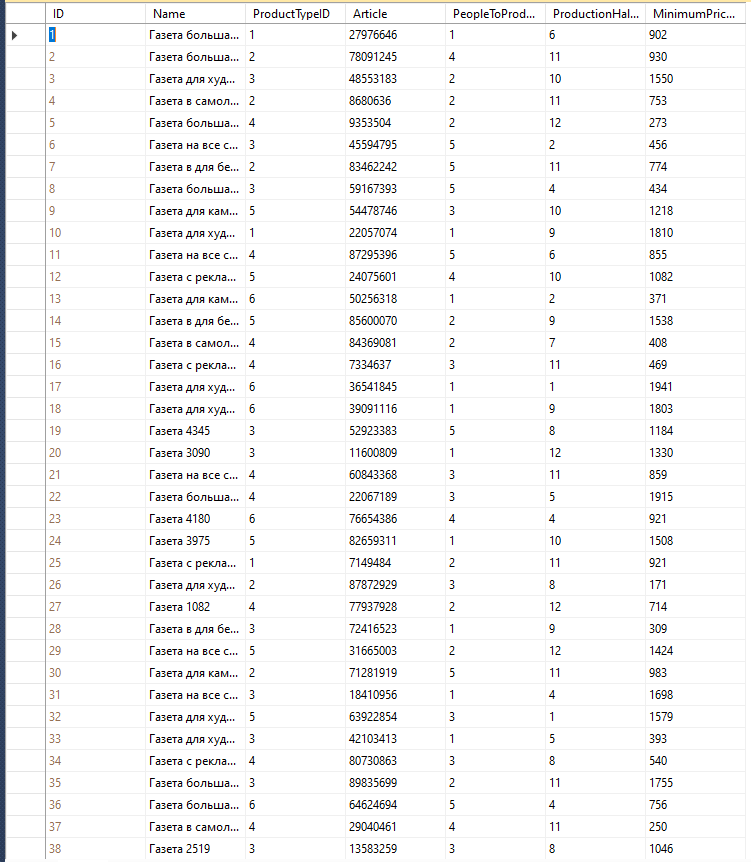


Рисунок 4. Таблица Product

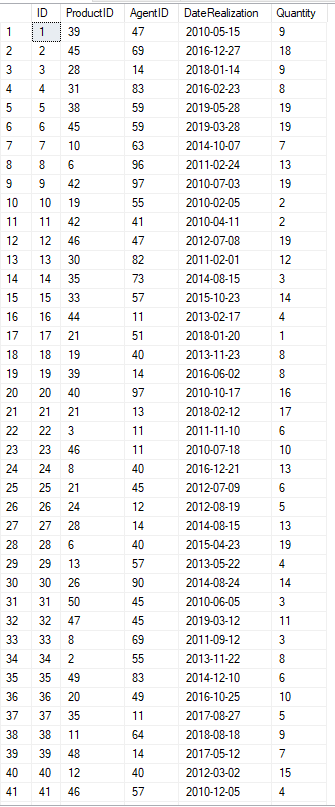


Рисунок 5. Таблица ProductSale

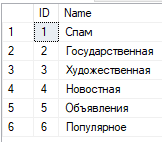


Рисунок 6. Таблица ProductType

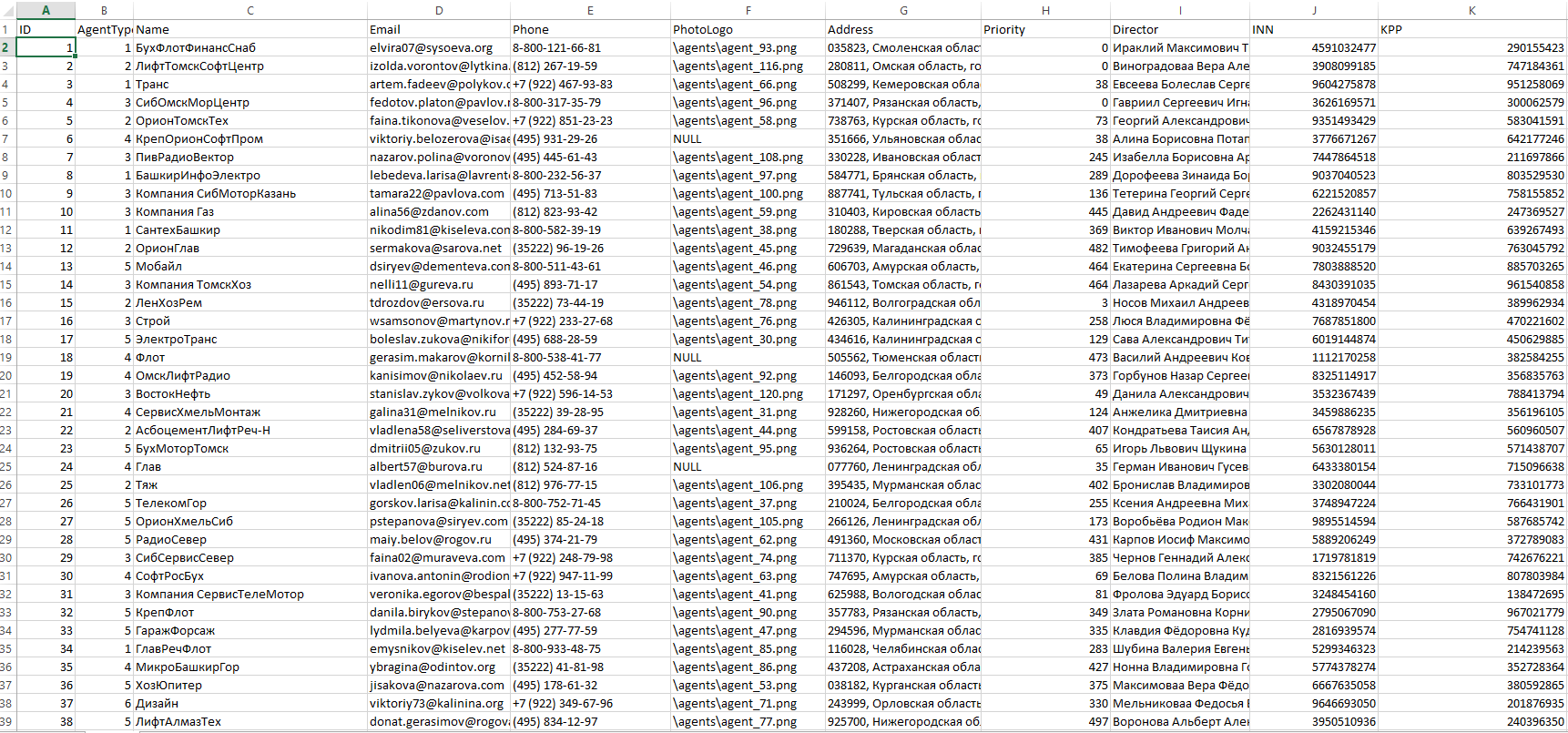


Рисунок 7. Данные из Excel файла Agent

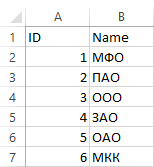


Рисунок 8. Данные из Excel файла AgentType

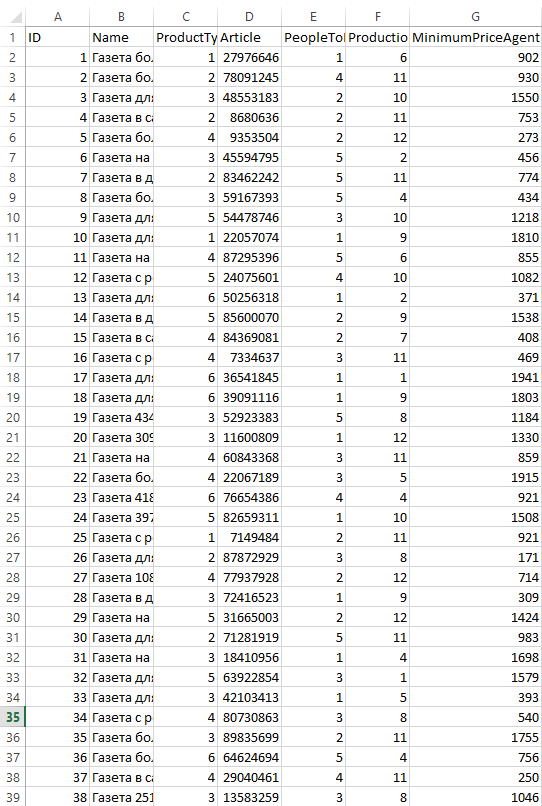


Рисунок 9. Данные из Excel файла Product

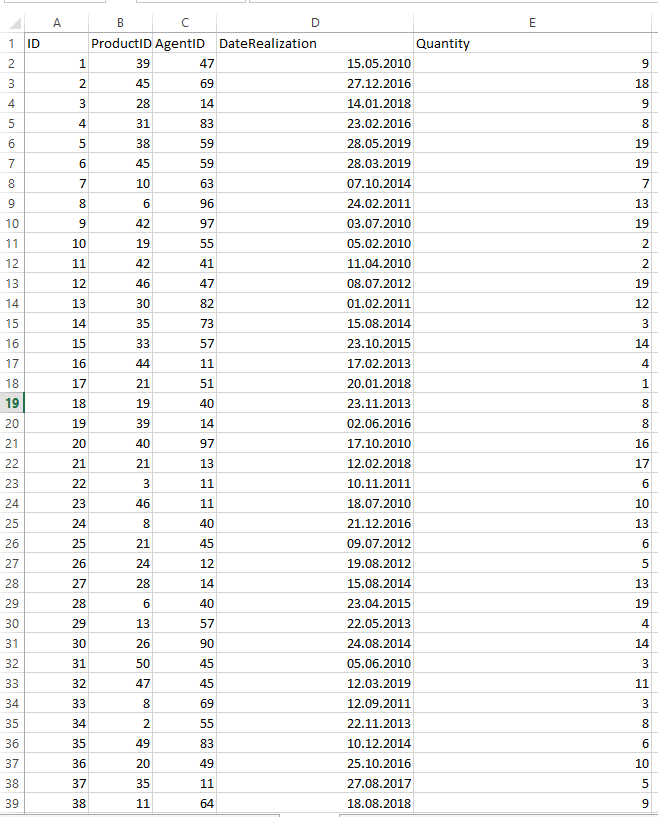


Рисунок 10. Данные из Excel файла ProductSale

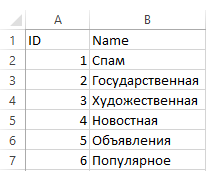


Рисунок 11. Данные из Excel файла ProductType

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.07 «Информационные системы и программирование»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Выполнил:

обучающийся учебной группы №\_\_\_619\_\_\_

М.Д. Кравченко

*(подпись) (И.О. Фамилия)*

*(подпись) (И.О. Фамилия)*

Проверили:

Руководители практики от колледжа:

*(должность) (И.О. Фамилия)*

*(должность) (И.О. Фамилия)*

*(должность) (И.О. Фамилия)*

**Москва**

**20­­­­22**

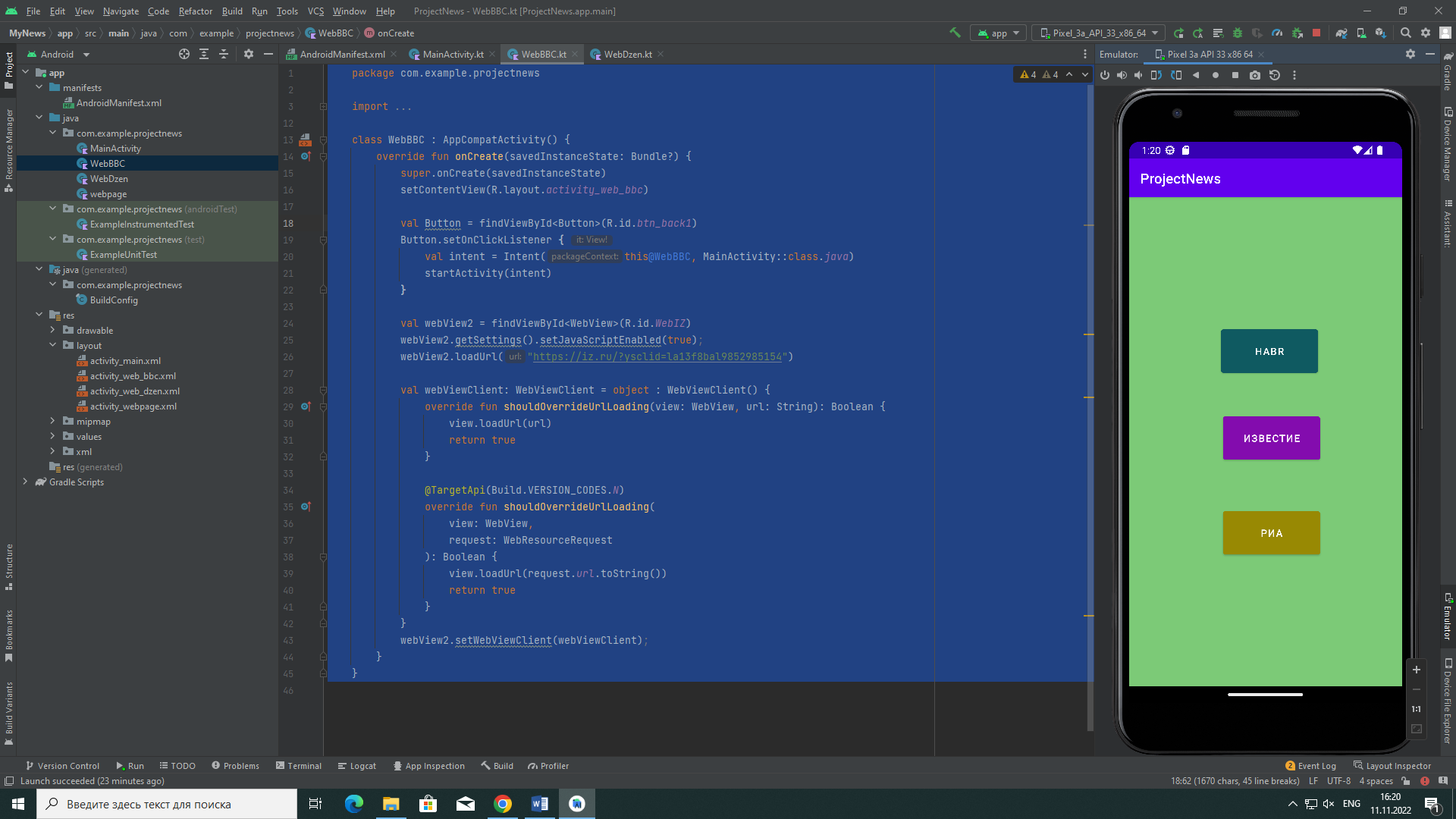


Рисунок 1. Главное меню

Программный листинг MainActivity представлен ниже:

package com.example.projectnews

import android.content.Intent

import android.os.Bundle

import android.widget.Button

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val button = findViewById<Button>(R.id.btn\_habr)

button.setOnClickListener {

val intent = Intent(this@MainActivity, webpage::class.java)

startActivity(intent)

};

val button1 = findViewById<Button>(R.id.btn\_IZ)

button1.setOnClickListener {

val intent = Intent(this@MainActivity, WebBBC::class.java)

startActivity(intent)

};

val button2 = findViewById<Button>(R.id.btn\_dzen)

button2.setOnClickListener {

val intent = Intent(this@MainActivity, WebDzen::class.java)

startActivity(intent)

};

}

}

Программный листинг WebBBC представлен ниже:

package com.example.projectnews

import android.annotation.TargetApi

import android.content.Intent

import android.os.Build

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.webkit.WebResourceRequest

import android.webkit.WebView

import android.webkit.WebViewClient

import android.widget.Button

class WebBBC : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_web\_bbc)

val Button = findViewById<Button>(R.id.btn\_back1)

Button.setOnClickListener {

val intent = Intent(this@WebBBC, MainActivity::class.java)

startActivity(intent)

}

val webView2 = findViewById<WebView>(R.id.WebIZ)

webView2.getSettings().setJavaScriptEnabled(true);

webView2.loadUrl("https://iz.ru/?ysclid=la13f8bal9852985154")

val webViewClient: WebViewClient = object : WebViewClient() {

override fun shouldOverrideUrlLoading(view: WebView, url: String): Boolean {

view.loadUrl(url)

return true

}

@TargetApi(Build.VERSION\_CODES.N)

override fun shouldOverrideUrlLoading(

view: WebView,

request: WebResourceRequest

): Boolean {

view.loadUrl(request.url.toString())

return true

}

}

webView2.setWebViewClient(webViewClient);

}

}

Программный листинг WebDzen представлен ниже:

package com.example.projectnews

import android.annotation.TargetApi

import android.content.Intent

import android.os.Build

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.webkit.WebResourceRequest

import android.webkit.WebView

import android.webkit.WebViewClient

import android.widget.Button

class WebDzen : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_web\_dzen)

val Button = findViewById<Button>(R.id.button)

Button.setOnClickListener {

val intent = Intent(this@WebDzen, MainActivity::class.java)

startActivity(intent)

}

val webView3 = findViewById<WebView>(R.id.WebDzen)

webView3.loadUrl("https://ria.ru/")

val webViewClient: WebViewClient = object : WebViewClient() {

override fun shouldOverrideUrlLoading(view: WebView, url: String): Boolean {

view.loadUrl(url)

return true

}

@TargetApi(Build.VERSION\_CODES.N)

override fun shouldOverrideUrlLoading(

view: WebView,

request: WebResourceRequest

): Boolean {

view.loadUrl(request.url.toString())

return true

}

}

webView3.setWebViewClient(webViewClient)

}

}

Программный листинг Webpage представлен ниже:

package com.example.projectnews

import android.annotation.TargetApi

import android.content.Intent

import android.os.Build

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.webkit.WebResourceRequest

import android.webkit.WebView

import android.webkit.WebViewClient

import android.widget.Button

class webpage : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_webpage)

val button = findViewById<Button>(R.id.btn\_back)

button.setOnClickListener {

val intent = Intent(this@webpage, MainActivity::class.java)

startActivity(intent)

}

val webView1 = findViewById<WebView>(R.id.webView2);

webView1.loadUrl("https://habr.com/ru/all/");

val webViewClient: WebViewClient = object : WebViewClient() {

override fun shouldOverrideUrlLoading(view: WebView, url: String): Boolean {

view.loadUrl(url)

return true

}

@TargetApi(Build.VERSION\_CODES.N)

override fun shouldOverrideUrlLoading(

view: WebView,

request: WebResourceRequest

): Boolean {

view.loadUrl(request.url.toString())

return true

}

}

webView1.setWebViewClient(webViewClient);

}

}