ГУАП

ФАКУЛЬТЕТ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

ОТЧЁТ  
ЗАЩИЩЕН С ОЦЕНКОЙ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| преподаватель |  |  |  | И. А. Юрьева |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЁТ О ЛАБОРАТОРНЫХ РАБОТАХ |
|  |
| по дисциплине МДК 04.01 ТЕХНОЛОГИЯ РАЗРАБОТКИ И ЗАЩИТЫ БАЗ ДАННЫХ |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | С022 |  |  |  | Графова К.Л. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

Лабораторная работа №1

Тема: «Взаимодействие с базами данных в MS Visual Studio»

Цель работы: изучить вспомогательные классы для взаимодействия с базой данных через клиентское приложение(использование технологии ADO.NET).

ЧАСТЬ 1

Задание 1. Используя вспомогательные классы, предоставляемые поставщиком OLE DB (Object Linking and Embedding), установить соединение с БД Колледж и реализовать выполнение запросов, перечисленных ниже.

Запрос 1. Вывести в DataGrid все записи из таблицы Students.



Рис. 1 – вывод записей из таблицы Students в DataGrid

**Запрос 2.** Для студента с фамилией, заданной в текстовое поле, вывести поля Фамилия (Familia), №группы(№gr) и Бюджет (Budget), т. е. реализовать запрос на выборку, в котором фамилия будет вводиться как параметр. Запрос выполняется вызовом обработчика события нажатия на соответствующую кнопку.

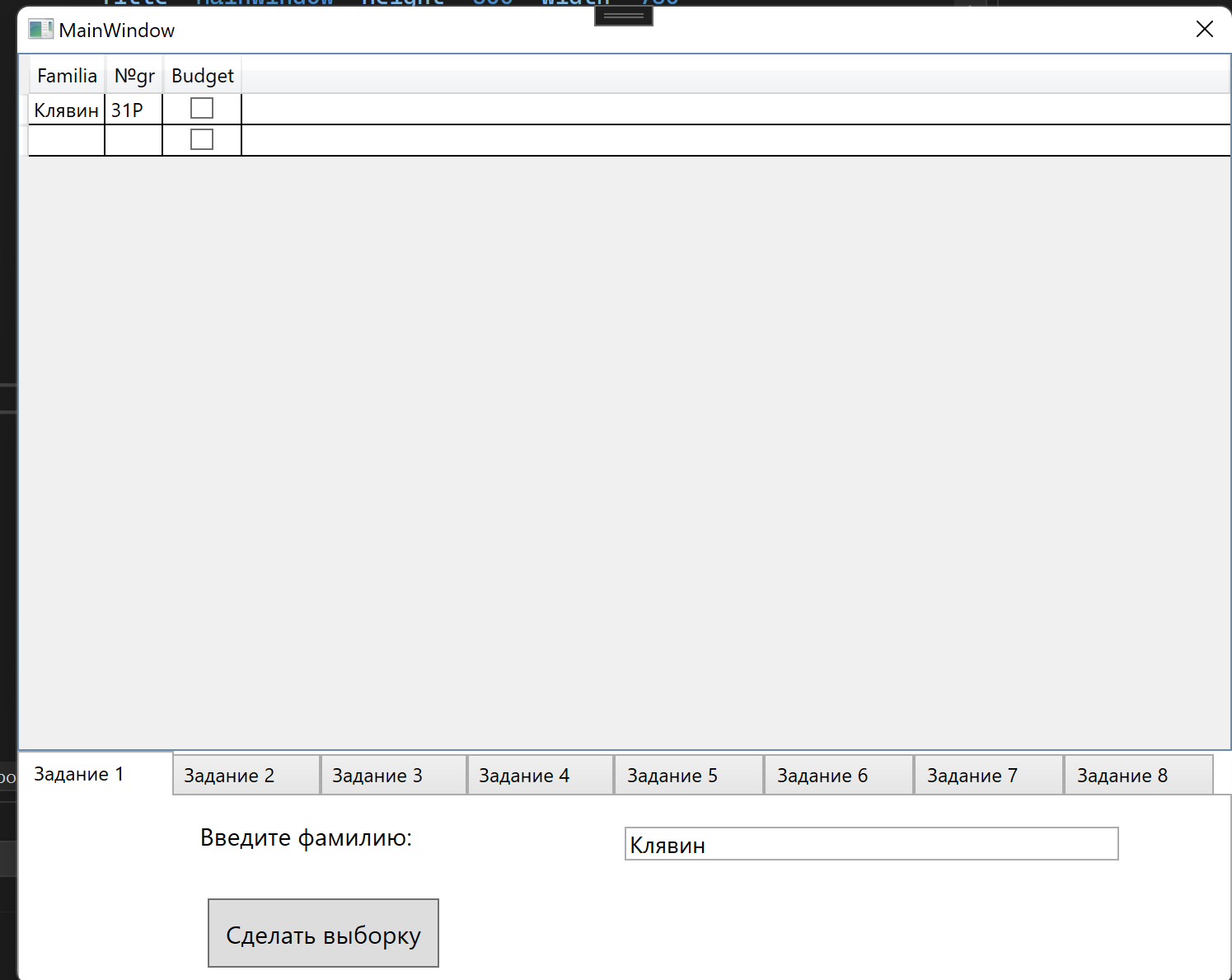
****

Рисунок 1.Вывод фамилии, номера группы для заданного студента

**Запрос 3.** В предыдущем запросе добавить в код конструкцию try..catch и предусмотреть случай, когда нет соединения с БД.

**Запрос 4.** Вычислить возраст студентов-юношей.

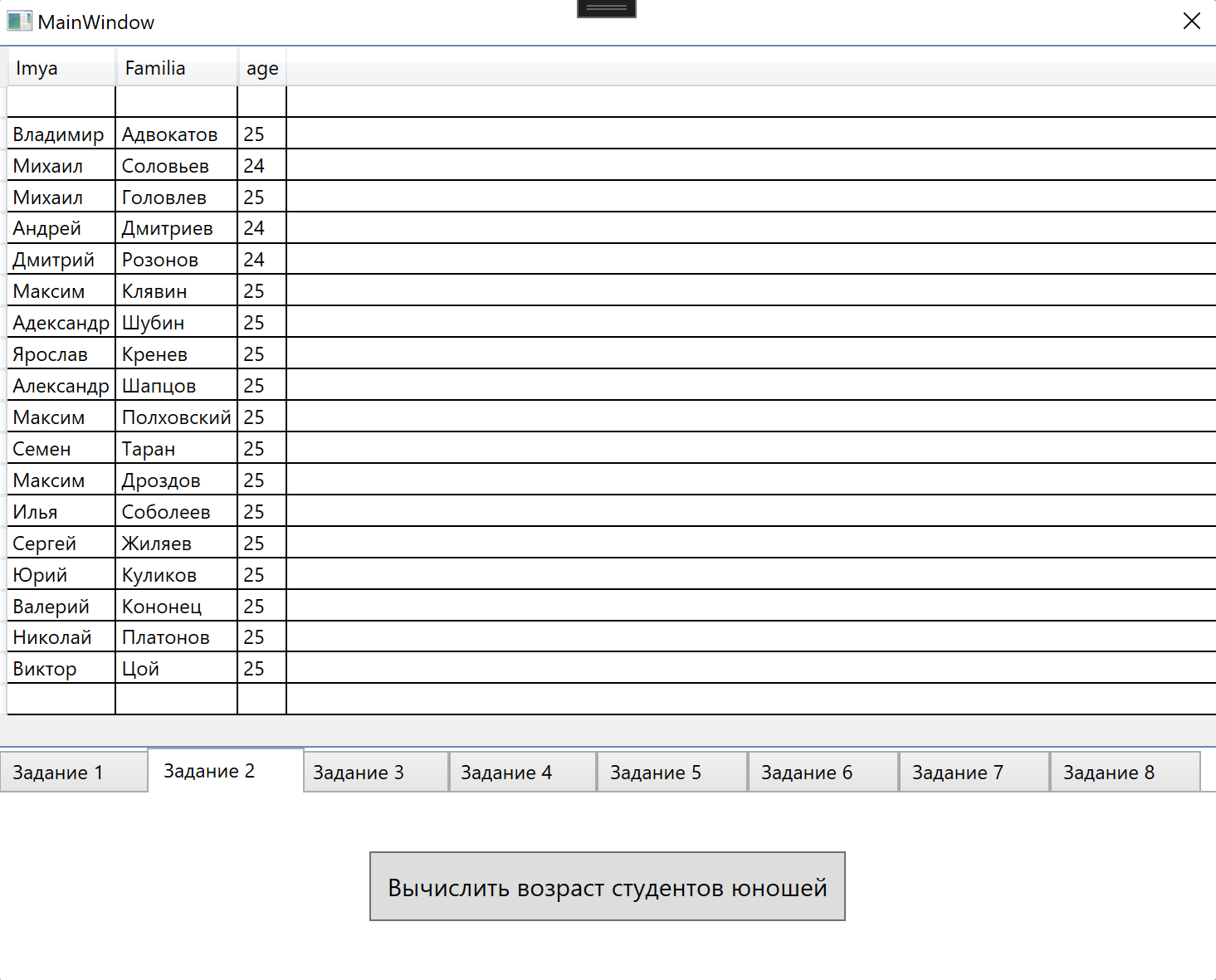


Рисунок 2.Возраст студентов-юношей

**Запрос 5.** Для студентов из группы, введенной в дополнительный TextBox, вывести Фамилию и инициалы, разделенные точками.

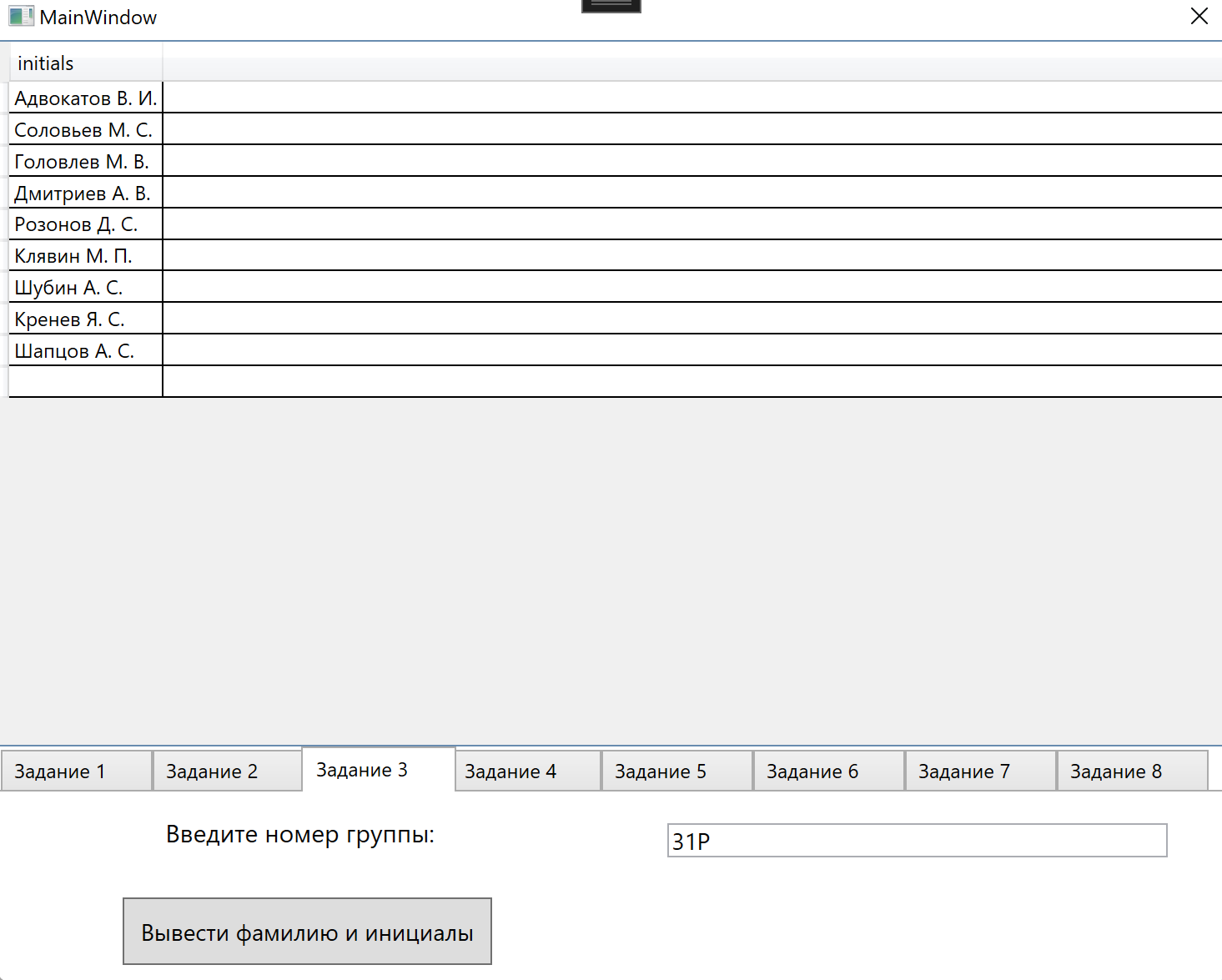


Рисунок 3.Фамилия и инициалы студентов заданной группы

**Запрос 6.** Для студентов с заданным годом рождения (дополнительный TextBox) вывести фамилию и №группы.

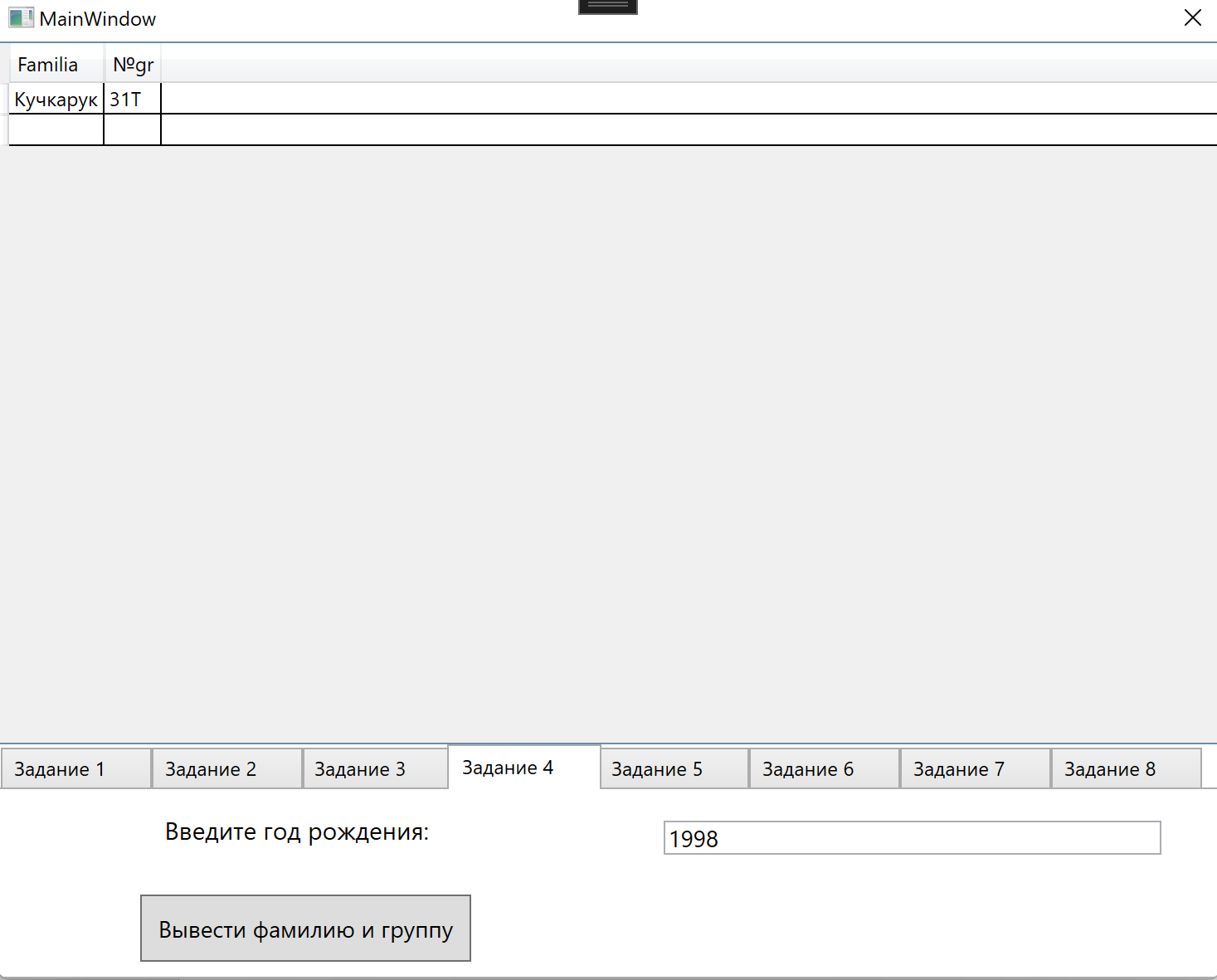
****

Рисунок 4.Фамилия и номер группы для студентов из заданной группы

**Запрос 7.** Вывести фамилии и №группы иногородних студентов.

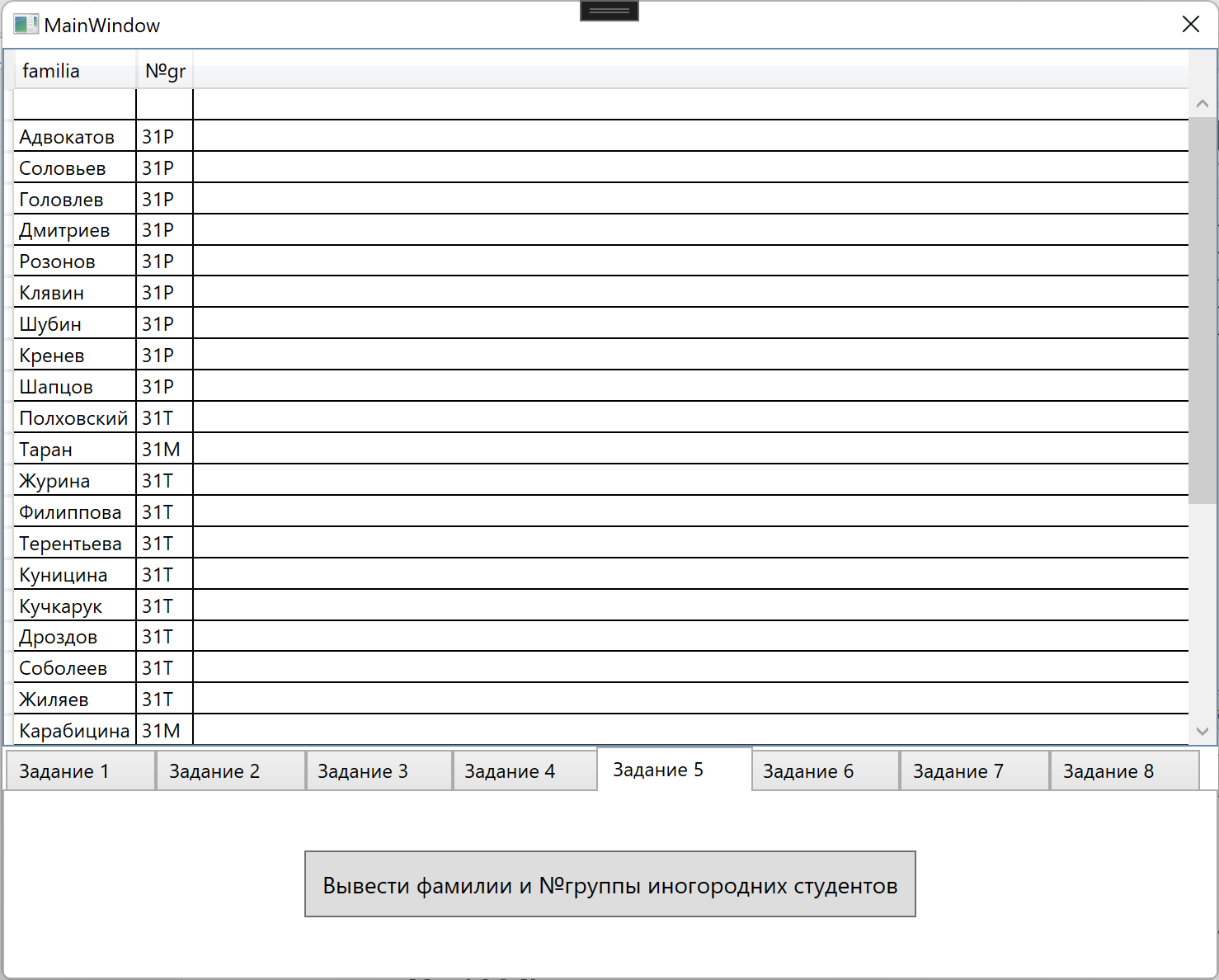
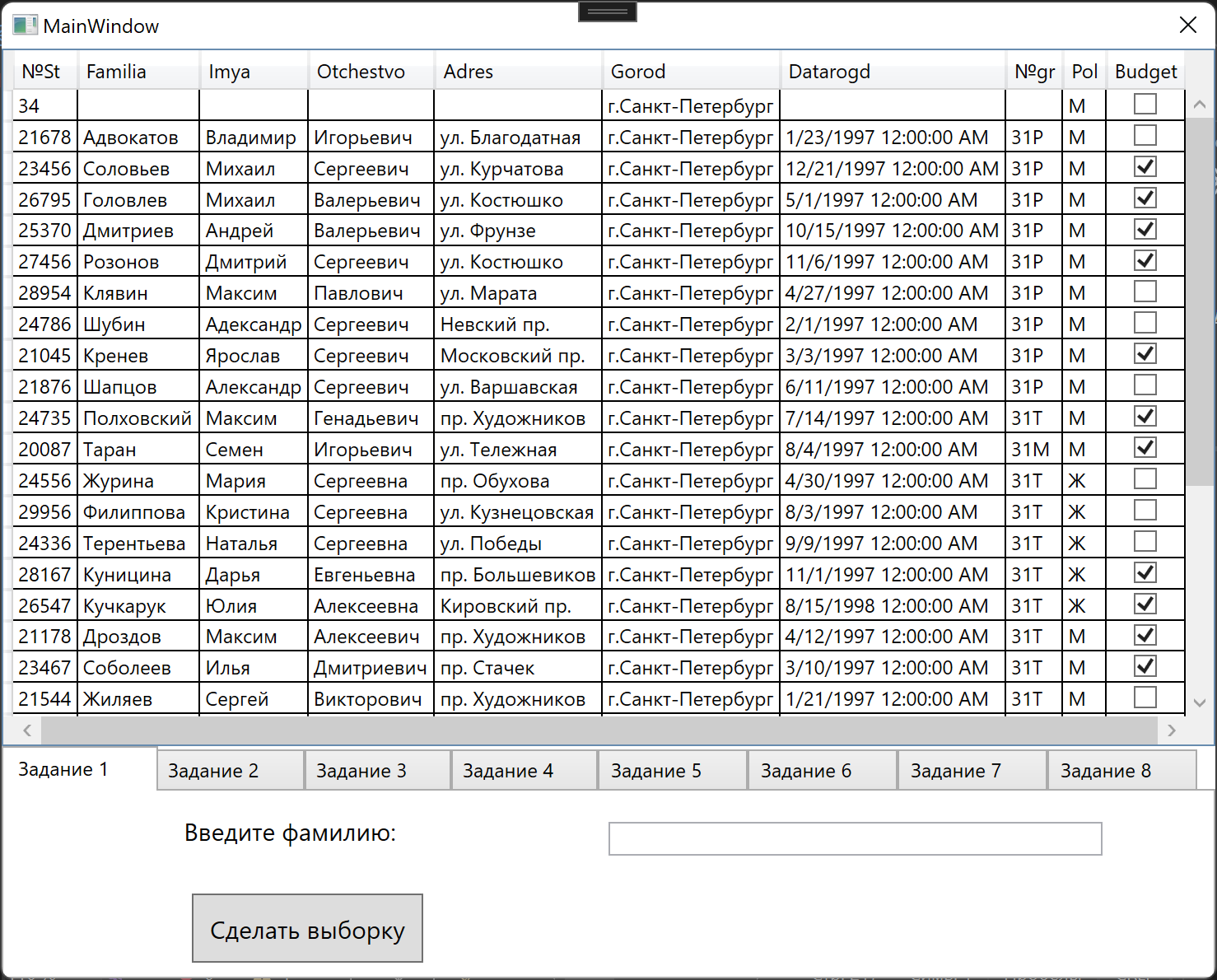


Рисунок 5.Вывод фамилии и №группы иногородних студентов.

**Запрос 8.**

1. Реализуйте запрос на обновление



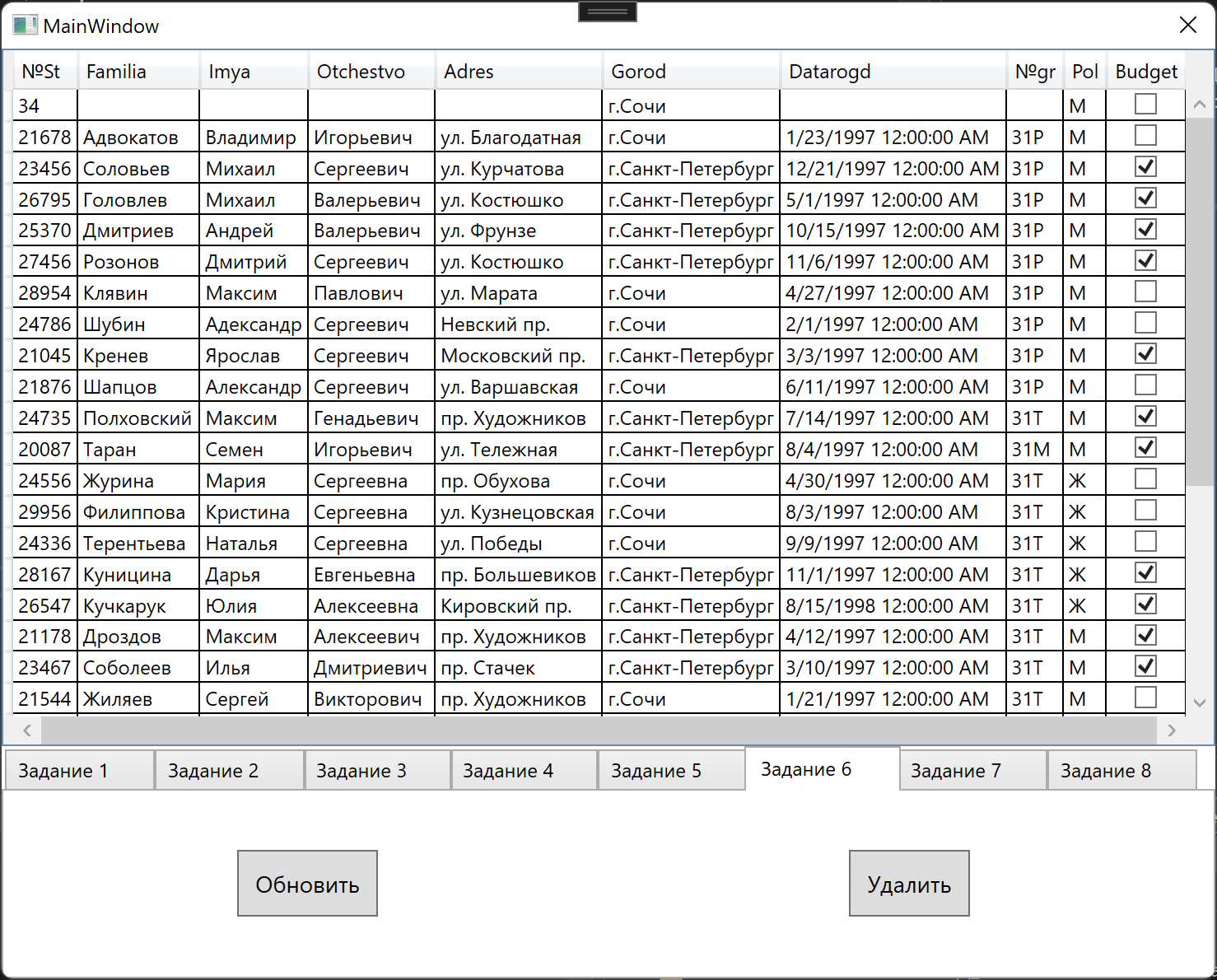


Рисунок 6.Замена города у всех студентов на контракте

1. Реализуйте запрос на удаление

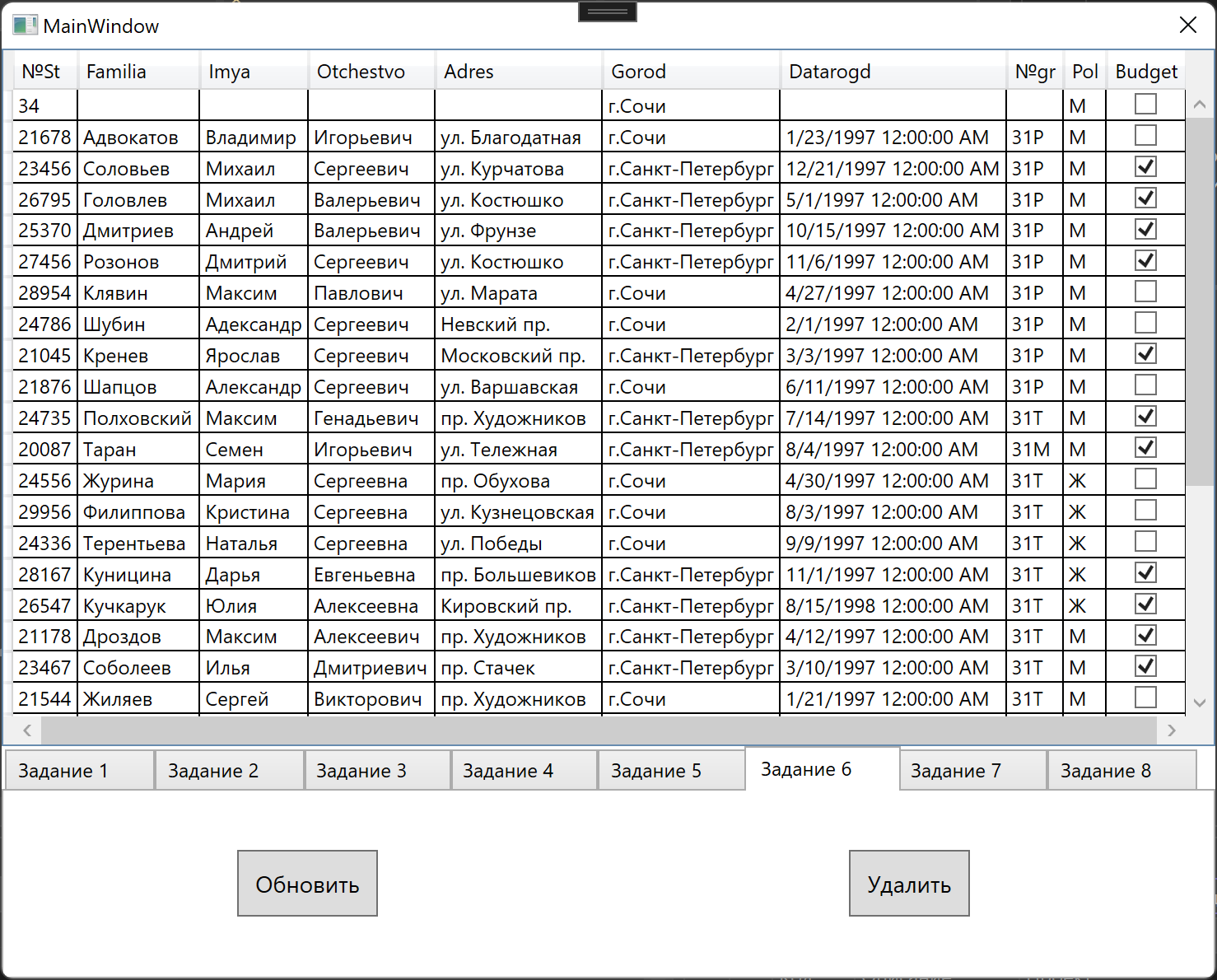


Рисунок 7.Удаление студента с фамилией Похловский

**Запрос 9.**

а) Используя элемент **Calendar** или **DatePicker** выполнить запрос, в котором будут найдены студенты с заданной датой рождения.

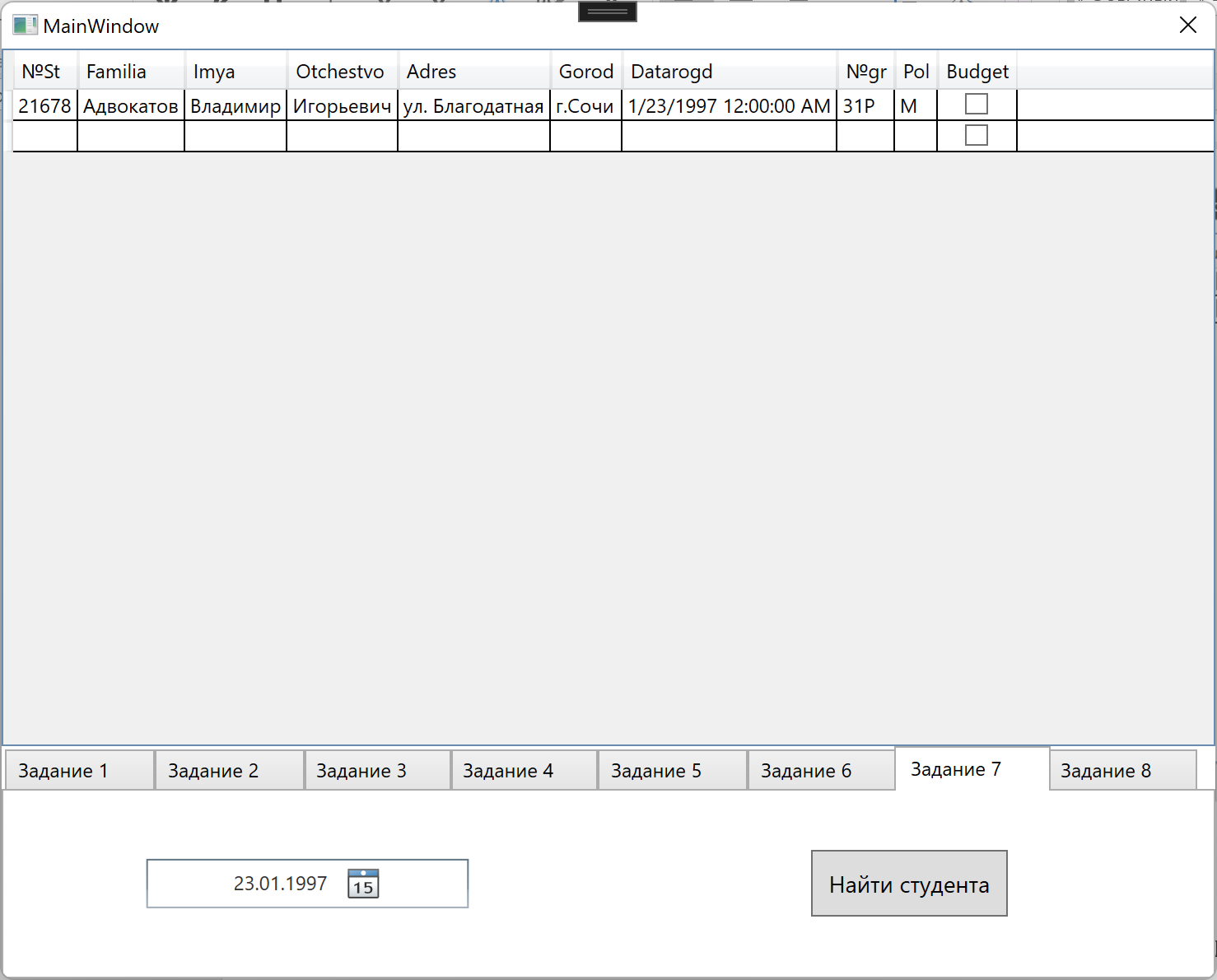


Рисунок 8.Найти студента с заданной датой рождения.

б) Изменить предыдущий запрос: найдены студенты с датой рождения, которая принадлежит выбранному диапазону

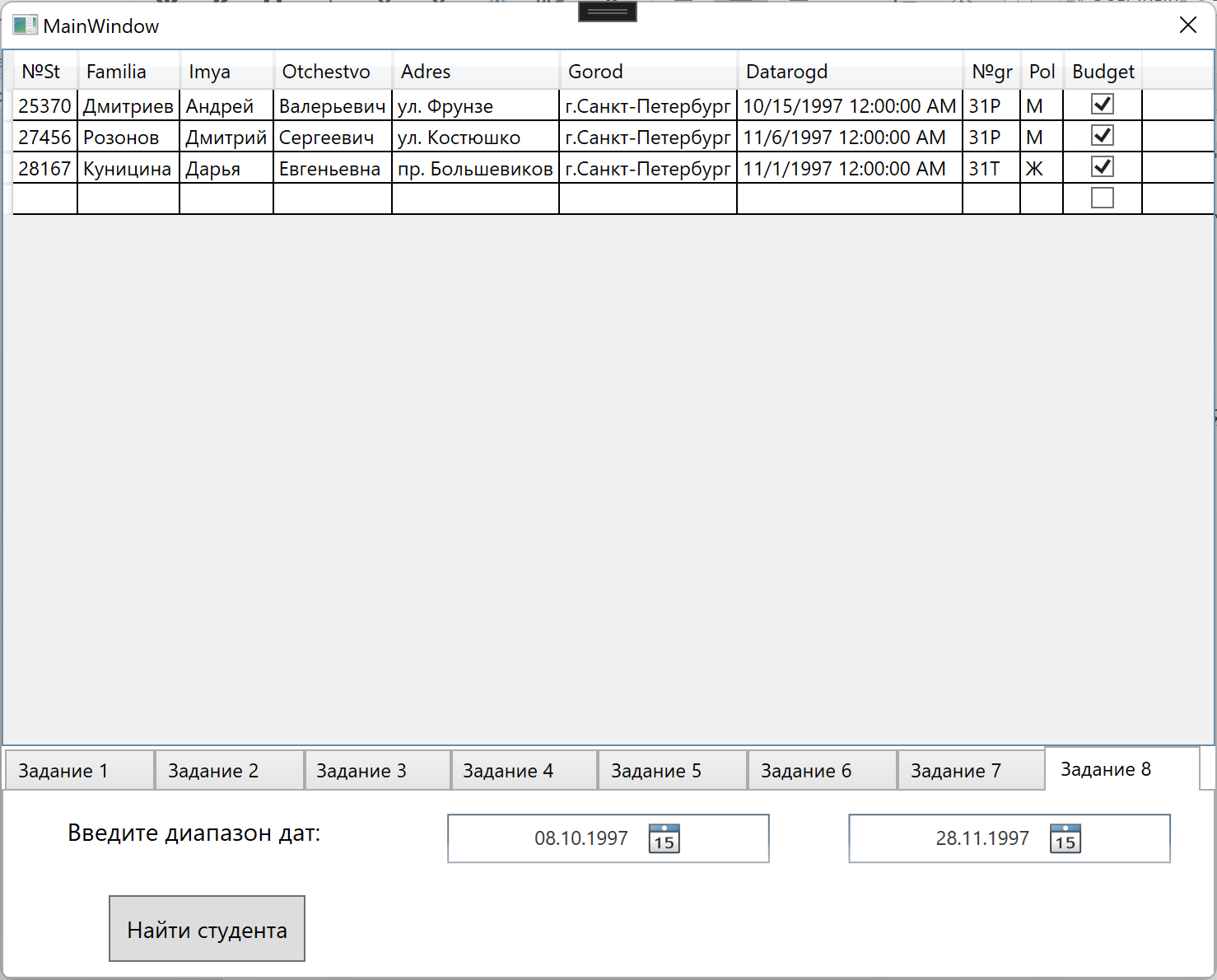


Рисунок 9.Найти студентов в заданном диапазоне дат рождения

**Листинг:**

**Разметка:**

<Window x:Class="БД\_Лаб1.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:БД\_Лаб1"

mc:Ignorable="d"

Title="MainWindow" Height="600" Width="750"

ResizeMode="NoResize">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="3\*"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<DataGrid Name="data" Grid.Column="0"

>

</DataGrid>

<TabControl

Grid.ColumnSpan="2"

Grid.Column="0"

Grid.Row ="1"

SelectionChanged="TabControl\_SelectionChanged">

<TabItem x:Name="ex1"

>

<TabItem.Header>

<StackPanel Orientation="Horizontal">

<TextBlock Width="75" Margin="2">Задание 1</TextBlock>

</StackPanel>

</TabItem.Header>

<TabItem.Content>

<UniformGrid Columns="2">

<TextBlock

Width="150"

Height="30"

FontSize="15"

Text="Введите фамилию:"/>

<TextBox

x:Name="Excercise1"

HorizontalAlignment="Left"

VerticalAlignment="Center"

Width="300"

FontSize="14" />

<Button

x:Name="makeSelectButton"

Content="Сделать выборку"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="15"

Click="makeSelectButton\_Click"

/>

</UniformGrid>

</TabItem.Content>

</TabItem>

<TabItem x:Name="ex2">

<TabItem.Header>

<StackPanel Orientation="Horizontal">

<TextBlock Width="75" Margin="2">Задание 2</TextBlock>

</StackPanel>

</TabItem.Header>

<TabItem.Content>

<UniformGrid Columns="1">

<Button x:Name="ex2Button"

Content="Вычислить возраст студентов юношей"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="15"

Click="ex2Button\_Click"/>

</UniformGrid>

</TabItem.Content>

</TabItem>

<TabItem x:Name="taskThree">

<TabItem.Header>

<StackPanel Orientation="Horizontal">

<TextBlock Width="75" Margin="1">Задание 3</TextBlock>

</StackPanel>

</TabItem.Header>

<TabItem.Content>

<UniformGrid Columns="2">

<TextBlock

Width="170"

Height="30"

FontSize="15"

Text="Введите номер группы:"/>

<TextBox

x:Name="Excercise3"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Width="300"

FontSize="14" />

<Button

x:Name="ex3Button"

Content="Вывести фамилию и инициалы"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="14"

Click="ex3Button\_Click"/>

</UniformGrid>

</TabItem.Content>

</TabItem>

<TabItem x:Name="taskFour">

<TabItem.Header>

<StackPanel Orientation="Horizontal">

<TextBlock Width="75" Margin="1">Задание 4</TextBlock>

</StackPanel>

</TabItem.Header>

<TabItem.Content>

<UniformGrid Columns="2">

<TextBlock

Width="170"

Height="30"

FontSize="15"

Text="Введите год рождения:"/>

<TextBox

x:Name="Excercise4"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Width="300"

FontSize="14" />

<Button

x:Name="ex4Button"

Content="Вывести фамилию и группу"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="14"

Click="ex4Button\_Click"

/>

</UniformGrid>

</TabItem.Content>

</TabItem>

<TabItem x:Name="taskFive">

<TabItem.Header>

<StackPanel Orientation="Horizontal">

<TextBlock Width="75" Margin="2">Задание 5</TextBlock>

</StackPanel>

</TabItem.Header>

<TabItem.Content>

<UniformGrid Columns="1">

<Button x:Name="ex5Button"

Content="Вывести фамилии и №группы иногородних студентов"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="14"

Click="ex5Button\_Click"

/>

</UniformGrid>

</TabItem.Content>

</TabItem>

<TabItem x:Name="taskSix">

<TabItem.Header>

<StackPanel Orientation="Horizontal">

<TextBlock Width="75" Margin="2">Задание 6</TextBlock>

</StackPanel>

</TabItem.Header>

<TabItem.Content>

<UniformGrid Columns="2">

<Button x:Name="updateButtonEx6"

Content="Обновить"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="14"

Click="updateButtonEx6\_Click"/>

<Button x:Name="deleteButtonEx6"

Content="Удалить"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="14"

Click="deleteButtonEx6\_Click"/>

</UniformGrid>

</TabItem.Content>

</TabItem>

<TabItem x:Name="taskSeven">

<TabItem.Header>

<StackPanel Orientation="Horizontal">

<TextBlock Width="75" Margin="2">Задание 7</TextBlock>

</StackPanel>

</TabItem.Header>

<TabItem.Content>

<UniformGrid Columns="2">

<DatePicker

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="50,5"

Name="ChooseDate"

>

</DatePicker>

<Button

x:Name="findStudentButton"

Content="Найти студента"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="14"

Click="findStudentButton\_Click"/>

</UniformGrid>

</TabItem.Content>

</TabItem>

<TabItem x:Name="taskEight">

<TabItem.Header>

<StackPanel Orientation="Horizontal">

<TextBlock Width="75" Margin="2">Задание 8</TextBlock>

</StackPanel>

</TabItem.Header>

<TabItem.Content>

<UniformGrid Columns="3"

>

<TextBlock

Width="170"

Height="30"

FontSize="15"

Text="Введите диапазон дат:"/>

<DatePicker

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="50,5"

Name="ChooseDateRange1"

>

</DatePicker>

<DatePicker

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="50,5"

Name="ChooseDateRange2"

DisplayDateStart="{Binding ElementName=ChooseDateRange1, Path=SelectedDate}">

</DatePicker>

<Button

x:Name="findStudentEx8"

Content="Найти студента"

HorizontalAlignment="Center"

VerticalAlignment="Center"

Padding="10"

FontSize="14"

Click="findStudentEx8\_Click"

/>

</UniformGrid>

</TabItem.Content>

</TabItem>

</TabControl>

</Grid>

</Window>

**Код:**

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.Data.OleDb;

using System.Data;

using System.Windows.Markup;

using System.Globalization;

namespace БД\_Лаб1

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

string sql = "Select \* from students;";

string connectionstring = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + @"\\Mac\Home\Desktop\3 курс\Лабораторные работы\БД\Задания\Лабораторная работа №1\Колледж(для студентов).mdb;Persist Security Info=False";

public MainWindow()

{

InitializeComponent();

}

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

{

DataBase(sql);

}

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

{

string makeSelect = $"Select Familia, №gr, Budget from students where familia = @familia;";

string lastName = Excercise1.Text;

OleDbParameter param = new OleDbParameter("@familia", lastName);

DataBase(makeSelect, param);

}

private void ex2Button\_Click(object sender, RoutedEventArgs e)

{

string countAge = $"Select Imya, Familia,format(date()-datarogd,'yy') as age from students where pol='М';";

DataBase(countAge);

}

private void ex3Button\_Click(object sender, RoutedEventArgs e)

{

string makeSelectByGroup = $"Select Familia & ' ' & left(imya, 1) & '. ' & left(otchestvo, 1) & '.' " +

$"as initials from students where №gr = @№gr; ";

string groupNumber = Excercise3.Text;

OleDbParameter param = new OleDbParameter("@№gr", groupNumber);

DataBase(makeSelectByGroup, param);

}

private void ex4Button\_Click(object sender, RoutedEventArgs e)

{

int yearOfBorn = int.Parse(Excercise4.Text);

string makeSelect = $"Select Familia,№gr from students where year(datarogd)=@yearOfBorn ;";

OleDbParameter param = new OleDbParameter("@yearOfBorn", yearOfBorn);

DataBase(makeSelect, param);

}

private void ex5Button\_Click(object sender, RoutedEventArgs e)

{

string makeSelect = $"Select familia, №gr from students where gorod is not null;";

DataBase(makeSelect);

}

private void updateButtonEx6\_Click(object sender, RoutedEventArgs e)

{

string update = $"Update students set gorod='г.Сочи' where budget=0;";

DbChange(update);

}

private void deleteButtonEx6\_Click(object sender, RoutedEventArgs e)

{

string delete = $"Delete from students where familia='Похловский';";

DbChange(delete);

}

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

{

string choosenDate = ChooseDate.Text;

string select = $"Select \* from students where datarogd=@choosenDate;";

OleDbParameter param = new OleDbParameter("@choosenDate", choosenDate);

DataBase(select, param);

}

private void findStudentEx8\_Click(object sender, RoutedEventArgs e)

{

if (string.IsNullOrEmpty(ChooseDateRange2.Text))

{

MessageBox.Show("Ошибка выбора диапазона", "Error", MessageBoxButton.OK, MessageBoxImage.Error);

return;

}

if (DateTime.Parse(ChooseDateRange1.Text) > DateTime.Parse(ChooseDateRange2.Text))

{

MessageBox.Show("Ошибка выбора диапазона", "Error", MessageBoxButton.OK, MessageBoxImage.Error);

return;

}

DateTime startRange = DateTime.Parse(ChooseDateRange1.Text);

DateTime endRange = DateTime.Parse(ChooseDateRange2.Text);

string sql = $"Select \* from students where datarogd between @startRange and @endRange";

OleDbParameter param1 = new OleDbParameter("@startRange", startRange);

OleDbParameter param2 = new OleDbParameter("@endRange", endRange);

try

{

DataTable table = new DataTable();

OleDbConnection connection = new OleDbConnection(connectionstring);

connection.Open();

OleDbCommand oleDbCommand = new OleDbCommand(sql, connection);

oleDbCommand.Parameters.Add(param1);

oleDbCommand.Parameters.Add(param2);

OleDbDataAdapter dataAdapter = new OleDbDataAdapter(oleDbCommand);

dataAdapter.Fill(table);

data.ItemsSource = table.DefaultView;

connection.Close();

}

catch (OleDbException ex)

{

MessageBox.Show("Ошибка доступа", "Error", MessageBoxButton.OK, MessageBoxImage.Error);

}

}

public void DataBase(string sql, OleDbParameter param)

{

try

{

DataTable table = new DataTable();

OleDbConnection connection = new OleDbConnection(connectionstring);

connection.Open();

OleDbCommand oleDbCommand = new OleDbCommand(sql, connection);

oleDbCommand.Parameters.Add(param);

OleDbDataAdapter dataAdapter = new OleDbDataAdapter(oleDbCommand);

dataAdapter.Fill(table);

data.ItemsSource = table.DefaultView;

connection.Close();

}

catch (OleDbException ex)

{

MessageBox.Show("Ошибка доступа", "Error", MessageBoxButton.OK, MessageBoxImage.Error);

}

}

public void DataBase(string sql)

{

try

{

DataTable table = new DataTable();

OleDbConnection connection = new OleDbConnection(connectionstring);

connection.Open();

OleDbCommand oleDbCommand = new OleDbCommand(sql, connection);

OleDbDataAdapter dataAdapter = new OleDbDataAdapter(oleDbCommand);

dataAdapter.Fill(table);

data.ItemsSource = table.DefaultView;

connection.Close();

}

catch (OleDbException ex)

{

MessageBox.Show("Ошибка доступа", "Error", MessageBoxButton.OK, MessageBoxImage.Error);

}

}

public void DbChange(string sql1)

{

try

{

DataTable table = new DataTable();

OleDbConnection connection = new OleDbConnection(connectionstring);

connection.Open();

OleDbCommand dbCommand = connection.CreateCommand();

dbCommand.CommandText = sql1;

dbCommand.ExecuteNonQuery();

OleDbCommand oleDbCommand = new OleDbCommand(sql, connection);

OleDbDataAdapter dataAdapter = new OleDbDataAdapter(oleDbCommand);

dataAdapter.Fill(table);

data.ItemsSource = table.DefaultView;

connection.Close();

}

catch (OleDbException ex)

{

MessageBox.Show("Ошибка доступа", "Error", MessageBoxButton.OK, MessageBoxImage.Error);

}

}

}

}