Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Практическая 13\_2**»

Выполнил: Карамов Д.Э

Группа: ПР-24

Преподаватель: Мирошниченко Г.В

2025

**Задание:**

Создать приложение для записи, редактирования и удаления студентов

Подпрограмма addStudent //заносит студента в таблицу

**Входные данные**

Класс Student

{

private string name;

private string surname;

private int bookNumber;

private int age; //возраст студента

private string group; //группа студента

}

**Выходные данные**

Класс Student

{

private string name;

private string surname;

private int bookNumber;

private int age; //возраст студента

private string group; //группа студента

}

Подпрограмма deleteStudent //удаляет студента из таблицы

**Входные данные**

Класс Student

{

private string name;

private string surname;

private int bookNumber;

private int age; //возраст студента

private string group; //группа студента

}

**Выходные данные**

-

Подпрограмма redactStudent //редактирует информацию о студенте

**Входные данные**

Класс Student

{

private string name;

private string surname;

private int bookNumber;

private int age; //возраст студента

private string group; //группа студента

}

**Выходные данные**

Класс Student

{

private string name;

private string surname;

private int bookNumber;

private int age; //возраст студента

private string group; //группа студента

}

**Листинг программы**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace pr13\_2\_karamov

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

initDataGridView();

}

private DataGridViewColumn dataGridViewColumn1 = null;

private DataGridViewColumn dataGridViewColumn2 = null;

private DataGridViewColumn dataGridViewColumn3 = null;

private DataGridViewColumn dataGridViewColumn4 = null;

private DataGridViewColumn dataGridViewColumn5 = null;

private List<Student> studentList = new List<Student>();

private List<int> booknumbers = new List<int>();

private void initDataGridView()

{

dataGridView1.DataSource = null;

dataGridView1.Columns.Add(getDataGridViewColumn1());

dataGridView1.Columns.Add(getDataGridViewColumn2());

dataGridView1.Columns.Add(getDataGridViewColumn3());

dataGridView1.Columns.Add(getDataGridViewColumn4());

dataGridView1.Columns.Add(getDataGridViewColumn5());

dataGridView1.AutoResizeColumns();

}

private DataGridViewColumn getDataGridViewColumn1()

{

if (dataGridViewColumn1 == null)

{

dataGridViewColumn1 = new DataGridViewTextBoxColumn();

dataGridViewColumn1.Name = "";

dataGridViewColumn1.HeaderText = "Имя";

dataGridViewColumn1.ValueType = typeof(string);

dataGridViewColumn1.Width = dataGridView1.Width / 3;

}

return dataGridViewColumn1;

}

private DataGridViewColumn getDataGridViewColumn2()

{

if (dataGridViewColumn2 == null)

{

dataGridViewColumn2 = new DataGridViewTextBoxColumn();

dataGridViewColumn2.Name = "";

dataGridViewColumn2.HeaderText = "Фамилия";

dataGridViewColumn2.ValueType = typeof(string);

dataGridViewColumn2.Width = dataGridView1.Width / 3;

}

return dataGridViewColumn2;

}

private DataGridViewColumn getDataGridViewColumn3()

{

if (dataGridViewColumn3 == null)

{

dataGridViewColumn3 = new DataGridViewTextBoxColumn();

dataGridViewColumn3.Name = "";

dataGridViewColumn3.HeaderText = "Зачётка";

dataGridViewColumn3.ValueType = typeof(string);

dataGridViewColumn3.Width = dataGridView1.Width / 3;

}

return dataGridViewColumn3;

}

private DataGridViewColumn getDataGridViewColumn4()

{

if (dataGridViewColumn4 == null)

{

dataGridViewColumn4 = new DataGridViewTextBoxColumn();

dataGridViewColumn4.Name = "";

dataGridViewColumn4.HeaderText = "Возраст";

dataGridViewColumn4.ValueType = typeof(int);

dataGridViewColumn4.Width = dataGridView1.Width / 3;

}

return dataGridViewColumn4;

}

private DataGridViewColumn getDataGridViewColumn5()

{

if (dataGridViewColumn5 == null)

{

dataGridViewColumn5 = new DataGridViewTextBoxColumn();

dataGridViewColumn5.Name = "";

dataGridViewColumn5.HeaderText = "Группа";

dataGridViewColumn5.ValueType = typeof(int);

dataGridViewColumn5.Width = dataGridView1.Width / 3;

}

return dataGridViewColumn5;

}

private void addStudent(string name, string surname, int bookNumber, int age, string group)

{

if (booknumbers.Contains(bookNumber) == false)

{

booknumbers.Add(bookNumber);

Student student = new Student();

studentList.Add(student);

student.Name = name;

student.Surname = surname;

student.BookNumber = bookNumber;

student.Age = age;

student.Group = group;

}

else

{

MessageBox.Show("Такой номер зачётки уже существует");

}

showListInGrid();

}

private void deleteStudent(int index)

{

studentList.Remove(studentList[index]);

booknumbers.Remove(booknumbers[index]);

showListInGrid();

}

private void redactStudent(int index, string name, string surname, int bookNumber, int age, string group)

{

studentList[index].Name = name;

studentList[index].Surname = surname;

studentList[index].BookNumber = bookNumber;

studentList[index].Age = age;

studentList[index].Group = group;

showListInGrid();

}

private void showListInGrid()

{

dataGridView1.Rows.Clear();

foreach (Student s in studentList)

{

DataGridViewRow row = new DataGridViewRow();

DataGridViewTextBoxCell cell1 = new

DataGridViewTextBoxCell();

DataGridViewTextBoxCell cell2 = new

DataGridViewTextBoxCell();

DataGridViewTextBoxCell cell3 = new

DataGridViewTextBoxCell();

DataGridViewTextBoxCell cell4 = new

DataGridViewTextBoxCell();

DataGridViewTextBoxCell cell5 = new

DataGridViewTextBoxCell();

cell1.ValueType = typeof(string);

cell1.Value = s.Name;

cell2.ValueType = typeof(string);

cell2.Value = s.Surname;

cell3.ValueType = typeof(int);

cell3.Value = s.BookNumber;

cell4.ValueType = typeof(int);

cell4.Value = s.Age;

cell5.ValueType = typeof(string);

cell5.Value = s.Group;

row.Cells.Add(cell1);

row.Cells.Add(cell2);

row.Cells.Add(cell3);

row.Cells.Add(cell4);

row.Cells.Add(cell5);

dataGridView1.Rows.Add(row);

}

}

private void button1\_Click(object sender, EventArgs e)

{

try

{

addStudent(textBox1.Text, textBox2.Text, Convert.ToInt32(numericUpDown2.Value), Convert.ToInt32(numericUpDown1.Value), textBox5.Text);

}

catch

{

MessageBox.Show("Неверный формат данных");

}

}

private void УдалитьToolStripMenuItem\_Click(object sender, EventArgs e)

{

int selectedRow = dataGridView1.SelectedCells[0].RowIndex;

DialogResult dr = MessageBox.Show("Удалить студента?", "", MessageBoxButtons.YesNo);

if (dr == DialogResult.Yes)

{

try

{

deleteStudent(selectedRow);

}

catch

{

MessageBox.Show("Ошибка");

}

}

}

private void РедактироватьToolStripMenuItem\_Click(object sender, EventArgs e)

{

int selectedRow = dataGridView1.SelectedCells[0].RowIndex;

textBox1.Text = studentList[selectedRow].Name;

textBox2.Text = studentList[selectedRow].Surname;

numericUpDown2.Value = studentList[selectedRow].BookNumber;

numericUpDown1.Value = studentList[selectedRow].Age;

textBox5.Text = studentList[selectedRow].Group;

button1.Visible = false;

button1.Enabled = false;

button2.Visible = true;

button2.Enabled = true;

}

private void button2\_Click(object sender, EventArgs e)

{

int selectedRow = dataGridView1.SelectedCells[0].RowIndex;

try

{

redactStudent(selectedRow, textBox1.Text, textBox2.Text, Convert.ToInt32(numericUpDown2.Value), Convert.ToInt32(numericUpDown1.Value), textBox5.Text);

MessageBox.Show("Данные о студенте изменены");

}

catch

{

MessageBox.Show("Ошибка");

}

button1.Visible = true;

button1.Enabled = true;

button2.Visible = false;

button2.Enabled = false;

}

}

}

Класс student:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace pr13\_2\_karamov

{

class Student

{

private string name;

private string surname;

private int bookNumber;

private int age; //возраст студента

private string group; //группа студента

public string Name

{

get { return name; }

set { name = value; }

}

public string Surname

{

get { return surname; }

set { surname = value; }

}

public int BookNumber

{

get { return bookNumber; }

set { bookNumber = value; }

}

public int Age

{

get { return age; }

set { age = value; }

}

public string Group

{

get { return group; }

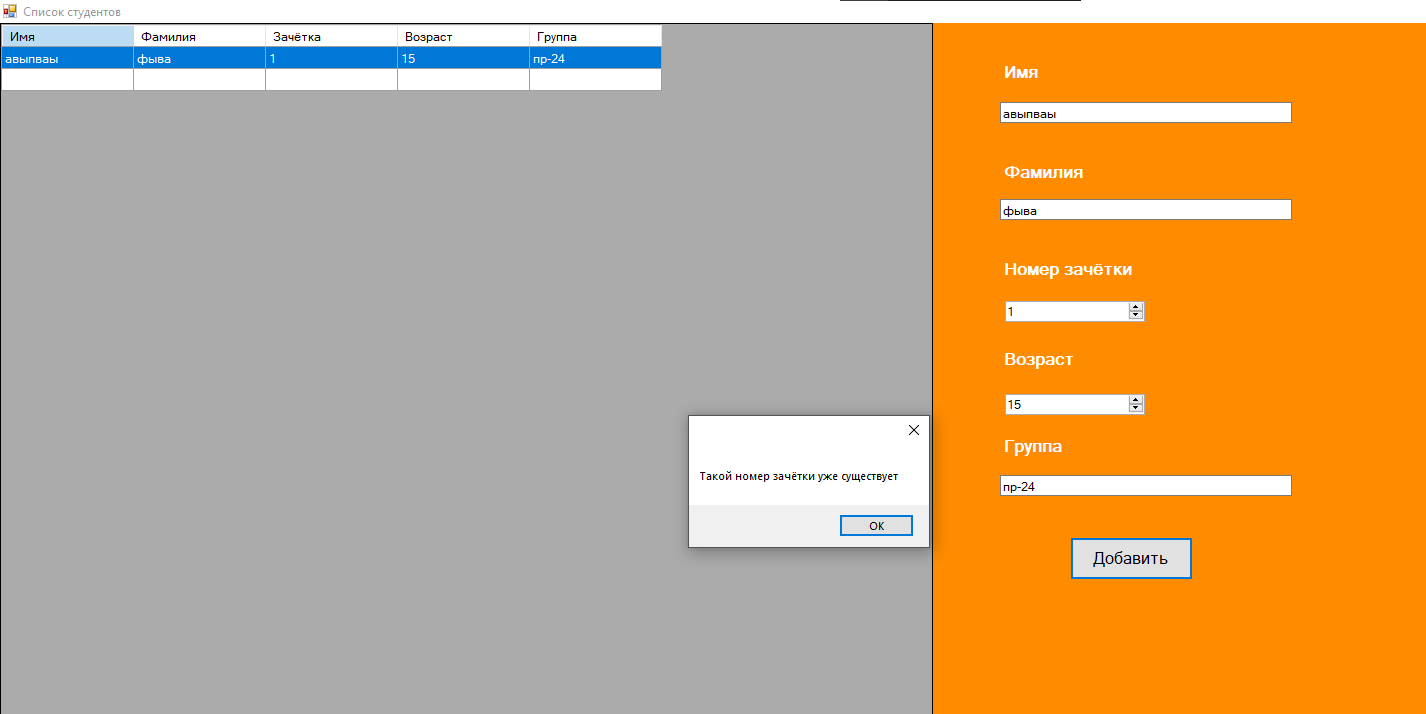
set { group = value; }

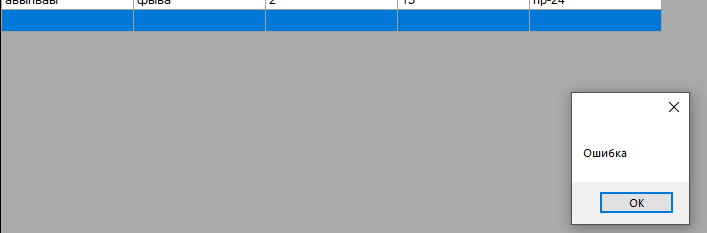
}

}

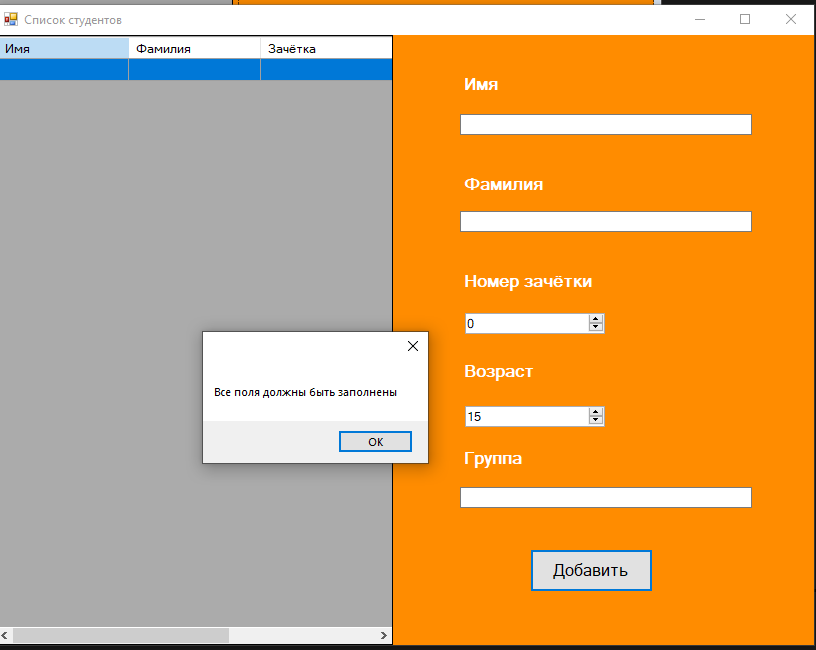
}

**Тестовые ситуации**

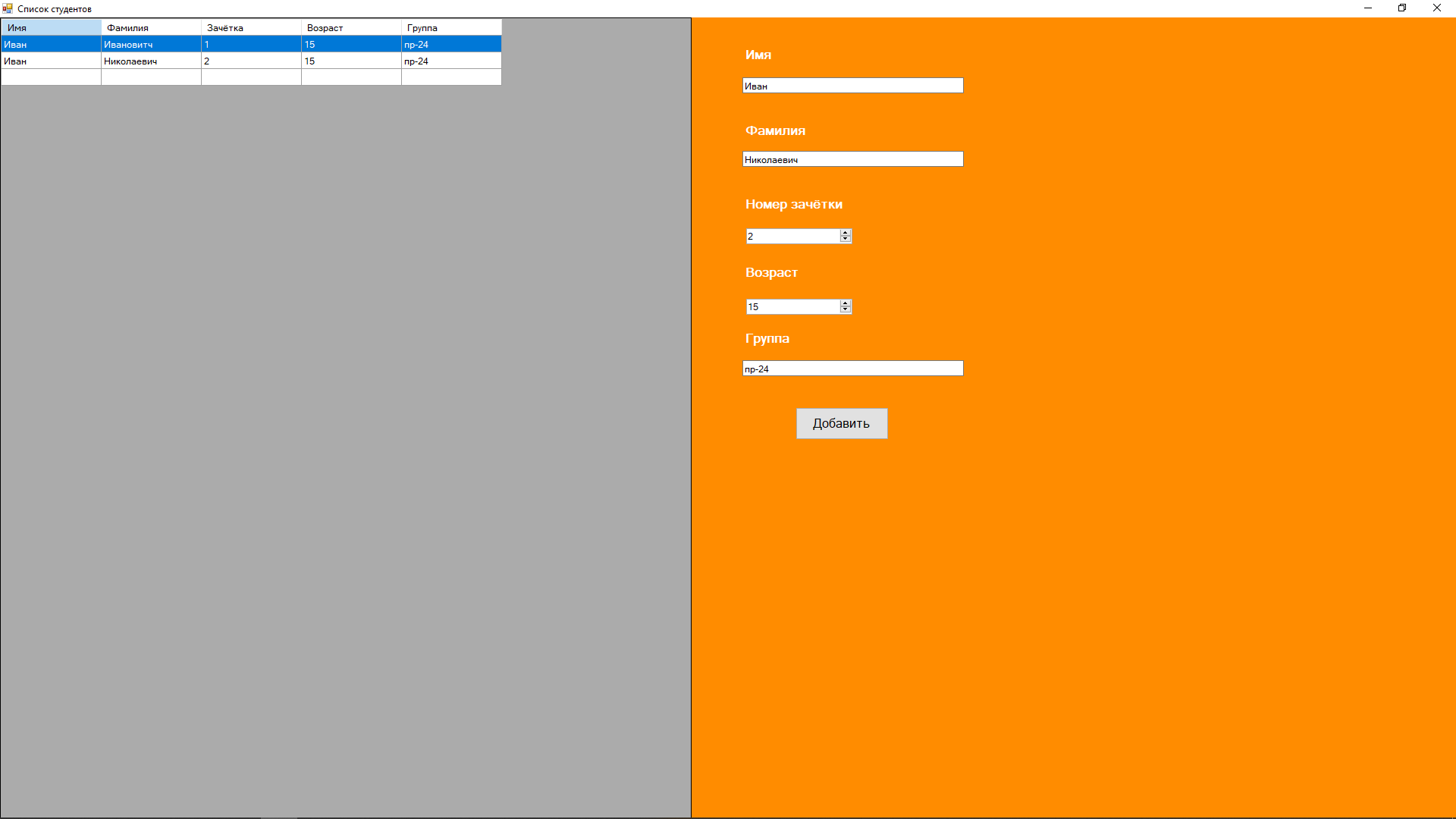
****

****

**(удаление студента в пустой строке)**

****

**Результат**



**Вывод:**

**Я научился создавать список студентов**