## Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Практическое занятие13.3**»

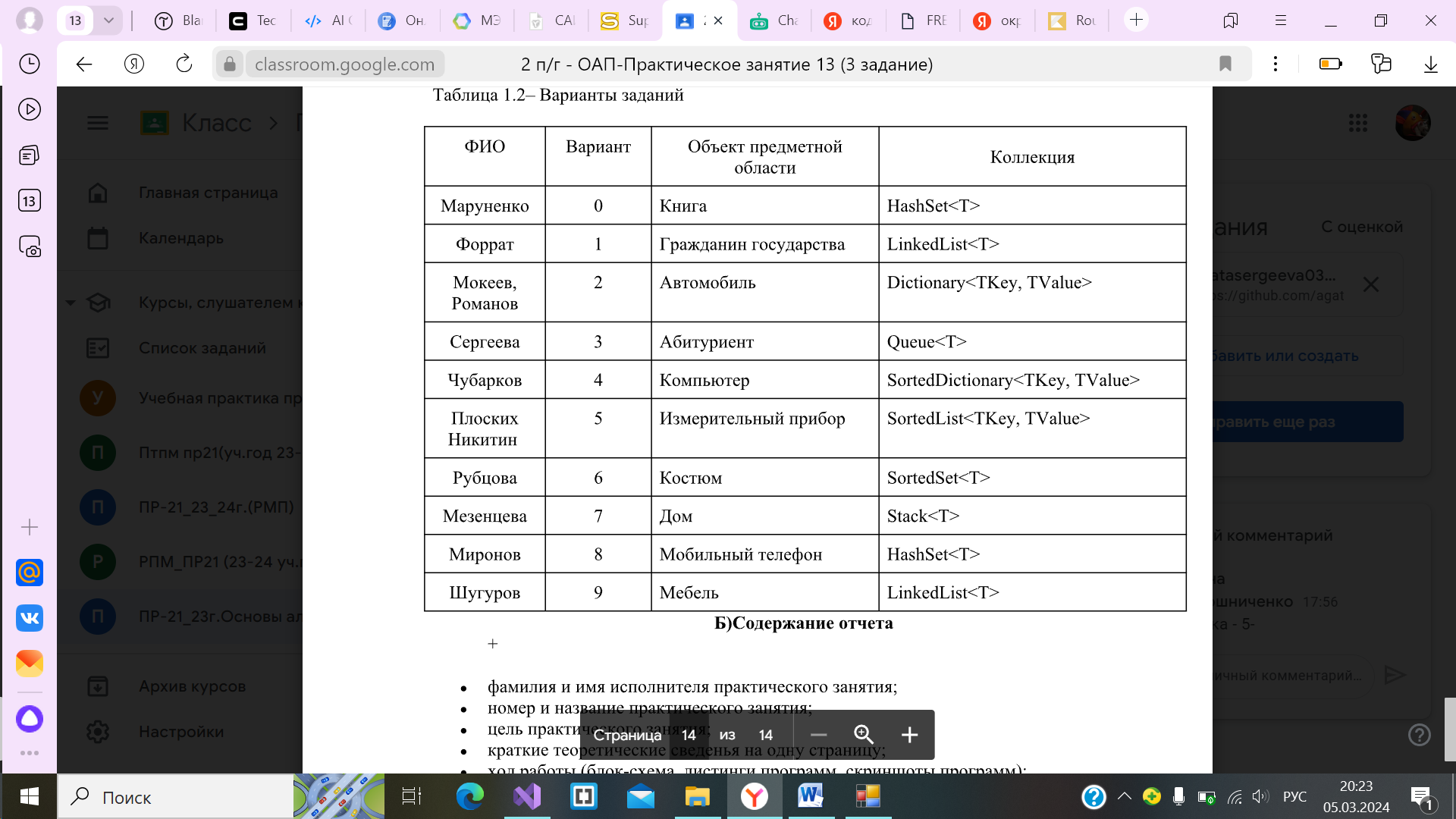
Выполнил: Сергеева Агата Андреевна

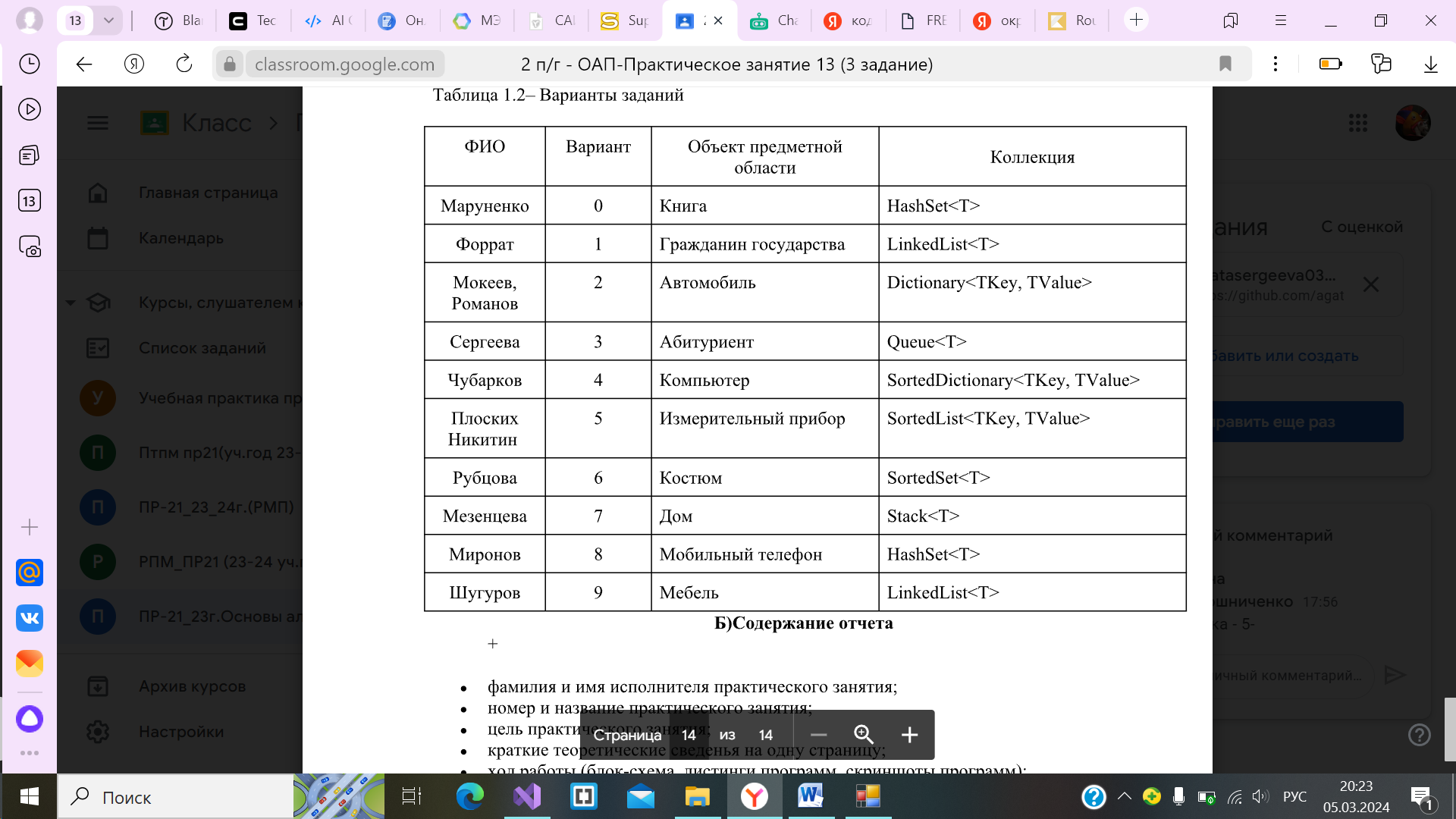
Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2024 г.

**Вариант 3:**





**Входные данные:** name – имя абитуриента, string; lastmane – его фамилия, string; otfather – его отчество ,string; medium – средний балл, double.

**Выходные данные:** заполненная отсортированная по алфавиту таблица по колонке «Фамилия», поиск позиции заданной фамилии

**Блок-схема:**

**Листинг программы:**

Class:

class TheApplicant

{

private string name;

private string lastname;

private string otfather;

private double medium;

public TheApplicant(string name, string lastname, string otfather, double medium)

{

this.name = name;

this.lastname = lastname;

this.otfather = otfather;

this.medium = medium;

}

public string get\_name()

{

return this.name;

}

public void set\_name(string n)

{

this.name = n;

}

public string get\_lastname()

{

return this.lastname;

}

public void set\_lastname(string n)

{

this.lastname = n;

}

public string get\_otfather()

{

return this.otfather;

}

public void set\_otfather(string n)

{

this.otfather = n;

}

public double get\_medium()

{

return this.medium;

}

public void set\_medium(double n)

{

this.medium = n;

}

}

Форма:

public partial class Form1 : Form

{

private DataGridViewColumn dataGridViewColumn1 = null;

private DataGridViewColumn dataGridViewColumn2 = null;

private DataGridViewColumn dataGridViewColumn3 = null;

private DataGridViewColumn dataGridViewColumn4 = null;

private void initDataGridView()

{

dataGridView1.DataSource = null;

dataGridView1.Columns.Add(getdataGridViewColumn1());

dataGridView1.Columns.Add(getdataGridViewColumn2());

dataGridView1.Columns.Add(getdataGridViewColumn3());

dataGridView1.Columns.Add(getdataGridViewColumn4());

dataGridView1.AutoResizeColumns();

}

private DataGridViewColumn getdataGridViewColumn1()

{

if (dataGridViewColumn1 == null)

{

dataGridViewColumn1 = new DataGridViewTextBoxColumn();

dataGridViewColumn1.Name = "";

dataGridViewColumn1.HeaderText = "Имя";

dataGridViewColumn1.ValueType = typeof(string);

dataGridViewColumn1.Width = dataGridView1.Width / 6;

}

return dataGridViewColumn1;

}

private DataGridViewColumn getdataGridViewColumn2()

{

if (dataGridViewColumn2 == null)

{

dataGridViewColumn2 = new DataGridViewTextBoxColumn();

dataGridViewColumn2.Name = "";

dataGridViewColumn2.HeaderText = "Фамилия";

dataGridViewColumn2.ValueType = typeof(string);

dataGridViewColumn2.Width = dataGridView1.Width / 4;

}

return dataGridViewColumn2;

}

private DataGridViewColumn getdataGridViewColumn3()

{

if (dataGridViewColumn3 == null)

{

dataGridViewColumn3 = new DataGridViewTextBoxColumn();

dataGridViewColumn3.Name = "";

dataGridViewColumn3.HeaderText = "Отчество";

dataGridViewColumn3.ValueType = typeof(string);

dataGridViewColumn3.Width = dataGridView1.Width / 4;

}

return dataGridViewColumn3;

}

private DataGridViewColumn getdataGridViewColumn4()

{

if (dataGridViewColumn4 == null)

{

dataGridViewColumn4 = new DataGridViewTextBoxColumn();

dataGridViewColumn4.Name = "";

dataGridViewColumn4.HeaderText = "Средний балл";

dataGridViewColumn4.ValueType = typeof(double);

dataGridViewColumn4.Width = dataGridView1.Width / 8;

}

return dataGridViewColumn4;

}

private Queue<TheApplicant> studentQueue = new Queue<TheApplicant>();

static Queue<string> fam = new Queue<string>();

string[] arr = new string[fam.Count()];

private void showListInGrid()

{

dataGridView1.Rows.Clear();

foreach (TheApplicant s in studentQueue)

{

DataGridViewRow row = new DataGridViewRow();

DataGridViewTextBoxCell cell1 = new DataGridViewTextBoxCell();

DataGridViewTextBoxCell cell2 = new DataGridViewTextBoxCell();

DataGridViewTextBoxCell cell3 = new DataGridViewTextBoxCell();

DataGridViewTextBoxCell cell4 = new DataGridViewTextBoxCell();

cell1.ValueType = typeof(string);

cell1.Value = s.get\_name();

cell2.ValueType = typeof(string);

cell2.Value = s.get\_lastname();

cell3.ValueType = typeof(string);

cell3.Value = s.get\_otfather();

cell4.ValueType = typeof(double);

cell4.Value = s.get\_medium();

row.Cells.Add(cell1);

row.Cells.Add(cell2);

row.Cells.Add(cell3);

row.Cells.Add(cell4);

dataGridView1.Rows.Add(row);

}

}

private void addTheApplication(string name, string lastname, string otfather, double medium)

{

int k = 0;

foreach (TheApplicant prov in studentQueue)

{

if (prov.get\_otfather() == otfather && prov.get\_lastname() == lastname && prov.get\_name() == name) k++;

}

if (k == 0)

{

TheApplicant info = new TheApplicant(name, lastname, otfather, medium);

studentQueue.Enqueue(info);

fam.Enqueue(lastname);

showListInGrid();

}

else MessageBox.Show("Такой абитуриент уже существует", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

private void correctTheApplicant(string name, string lastname, string otfather, int a)

{

int k = 0;

foreach (TheApplicant prov in studentQueue)

{

if (prov.get\_otfather() == otfather && prov.get\_lastname() == lastname && prov.get\_name() == name) k++;

}

if (k == 0)

{

TheApplicant add = new TheApplicant(name, lastname, otfather, a + 1);

studentQueue.Enqueue(add);

fam.Enqueue(lastname);

showListInGrid();

}

else MessageBox.Show("Такой абитуриент уже существует", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

public Form1()

{

InitializeComponent();

initDataGridView();

}

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

{

TheApplicant student = new TheApplicant("", "","",0);

string name = textBox2.Text;

if (name != "")

{

student.set\_name(name);

string lastname = textBox1.Text;

if (lastname != "")

{

student.set\_lastname(lastname);

string otfather = textBox3.Text;

if (otfather != "")

{

student.set\_otfather(otfather);

double number = Convert.ToDouble(comboBox1.Text);

student.set\_medium(number);

addTheApplication(student.get\_name(), student.get\_lastname(), student.get\_otfather(), student.get\_medium());

}

else MessageBox.Show($"Поле 'Отчество' пустое", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else MessageBox.Show($"Поле 'Фамилия' пустое", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else MessageBox.Show($"Поле 'Имя' пустое", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

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

{

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

DialogResult b = MessageBox.Show("Вы действительно хотите редактировать данную строку?", "Информация", MessageBoxButtons.YesNo);

if (b == DialogResult.Yes)

{

studentQueue.Dequeue();

TheApplicant student = new TheApplicant("", "", "", 0);

string name = textBox2.Text;

student.set\_name(name);

string lastname = textBox1.Text;

student.set\_lastname(lastname);

string otfather = textBox3.Text;

student.set\_otfather(otfather);

double medium = Convert.ToDouble(comboBox1.Text);

student.set\_medium(medium);

correctTheApplicant(name, lastname, otfather, a);

}

}

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

{

button1.Enabled = false;

groupBox1.Enabled = false;

label5.Visible = true;

textBox4.Visible = true;

button3.Visible = true;

dataGridView1.Sort(dataGridView1.Columns[1], ListSortDirection.Ascending);

}

private void button3\_Click(object sender, EventArgs e)

{

arr = fam.ToArray();

Array.Sort(arr);

button2.Enabled = false;

string familia = textBox4.Text;

if (fam.Contains($"{familia}"))

{

//button3.Enabled = false;

//textBox4.Enabled = false;

MessageBox.Show($"Позиция фамилии, которую вы ищите {Array.IndexOf(arr,familia)+1}", "Информация", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else MessageBox.Show($"Такой фамилии нет в списке!", "Информация", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void Form1\_Load(object sender, EventArgs e)

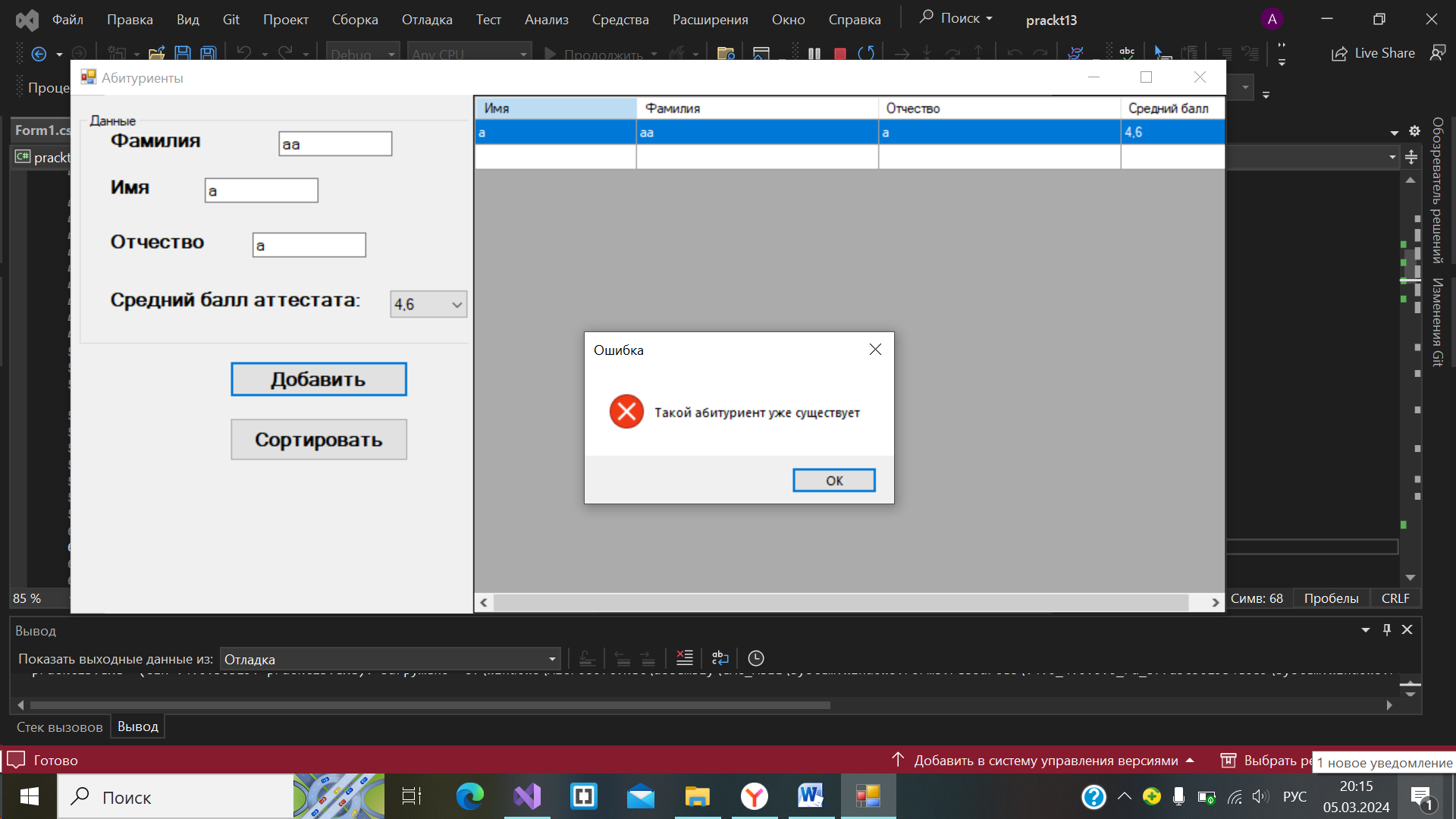
{

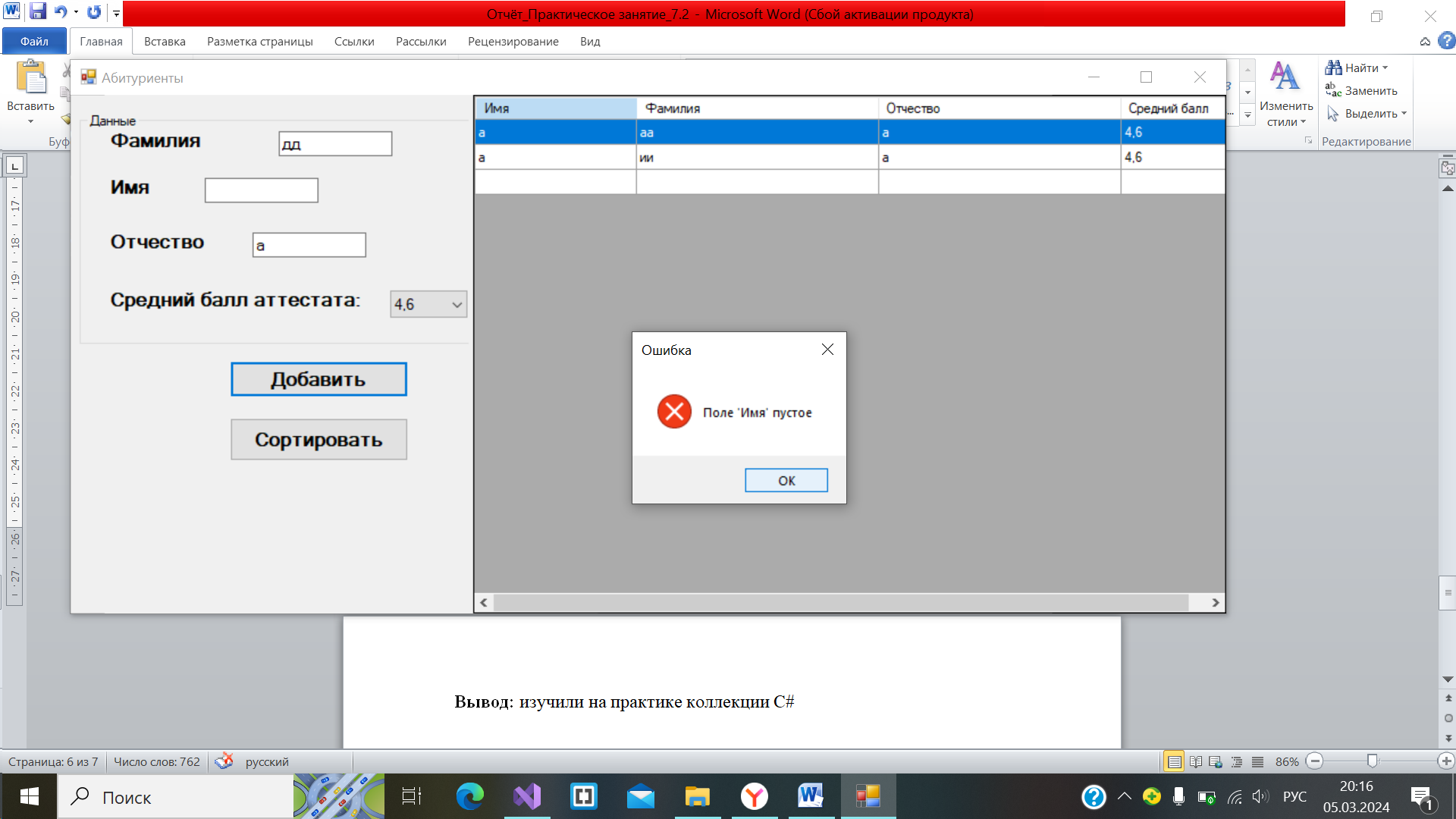
comboBox1.DropDownStyle = ComboBoxStyle.DropDownList;

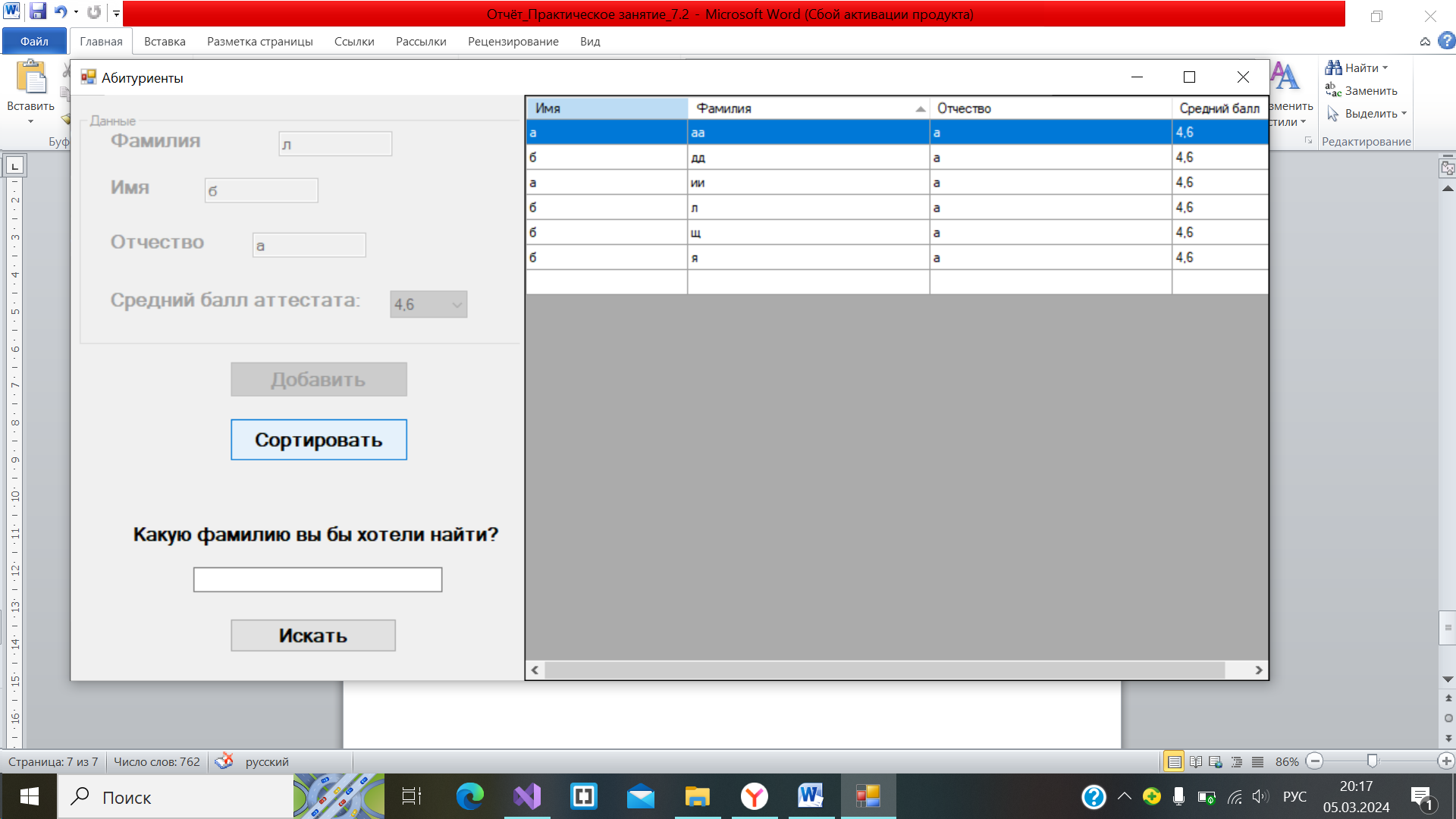
}

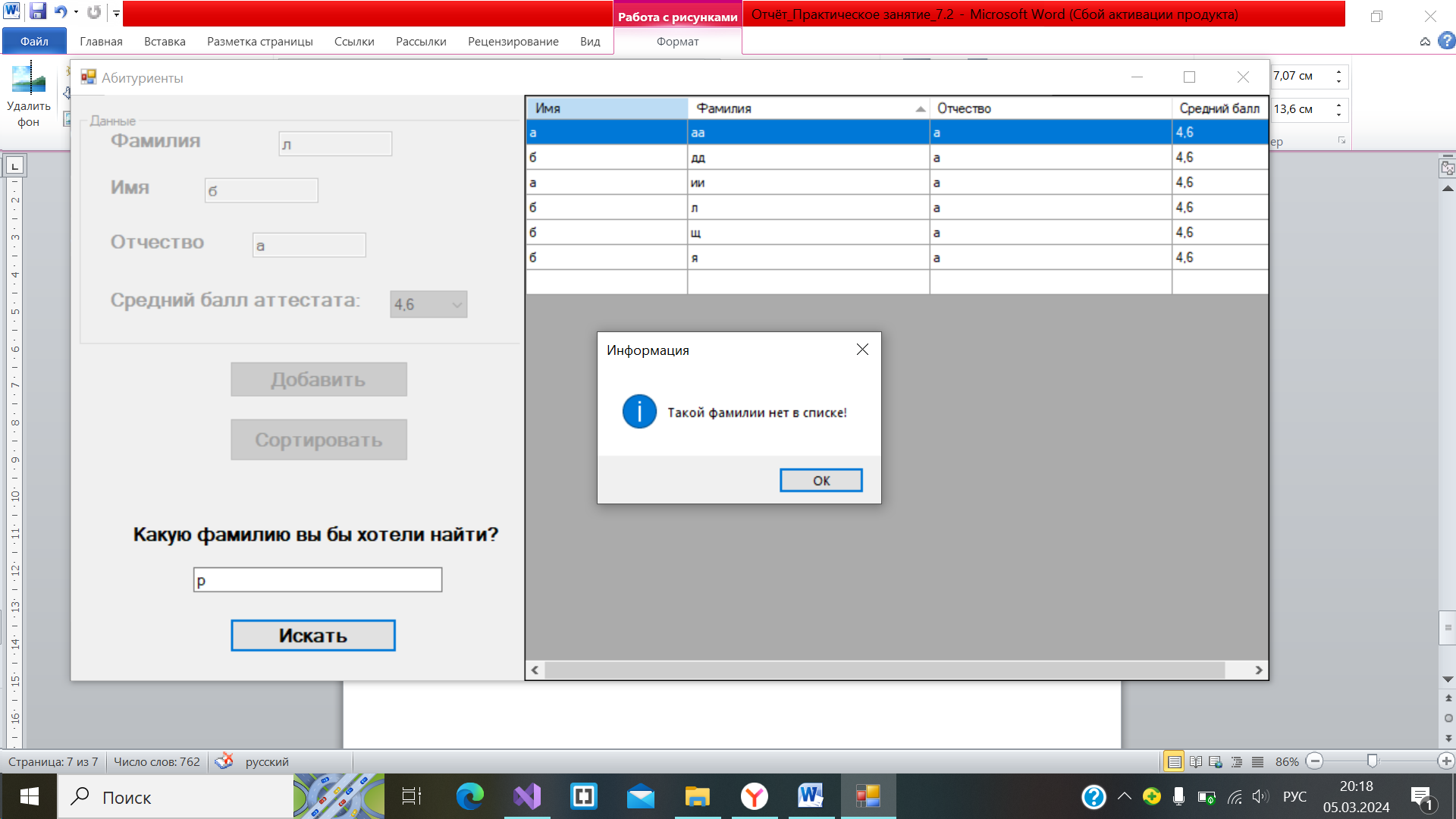
}

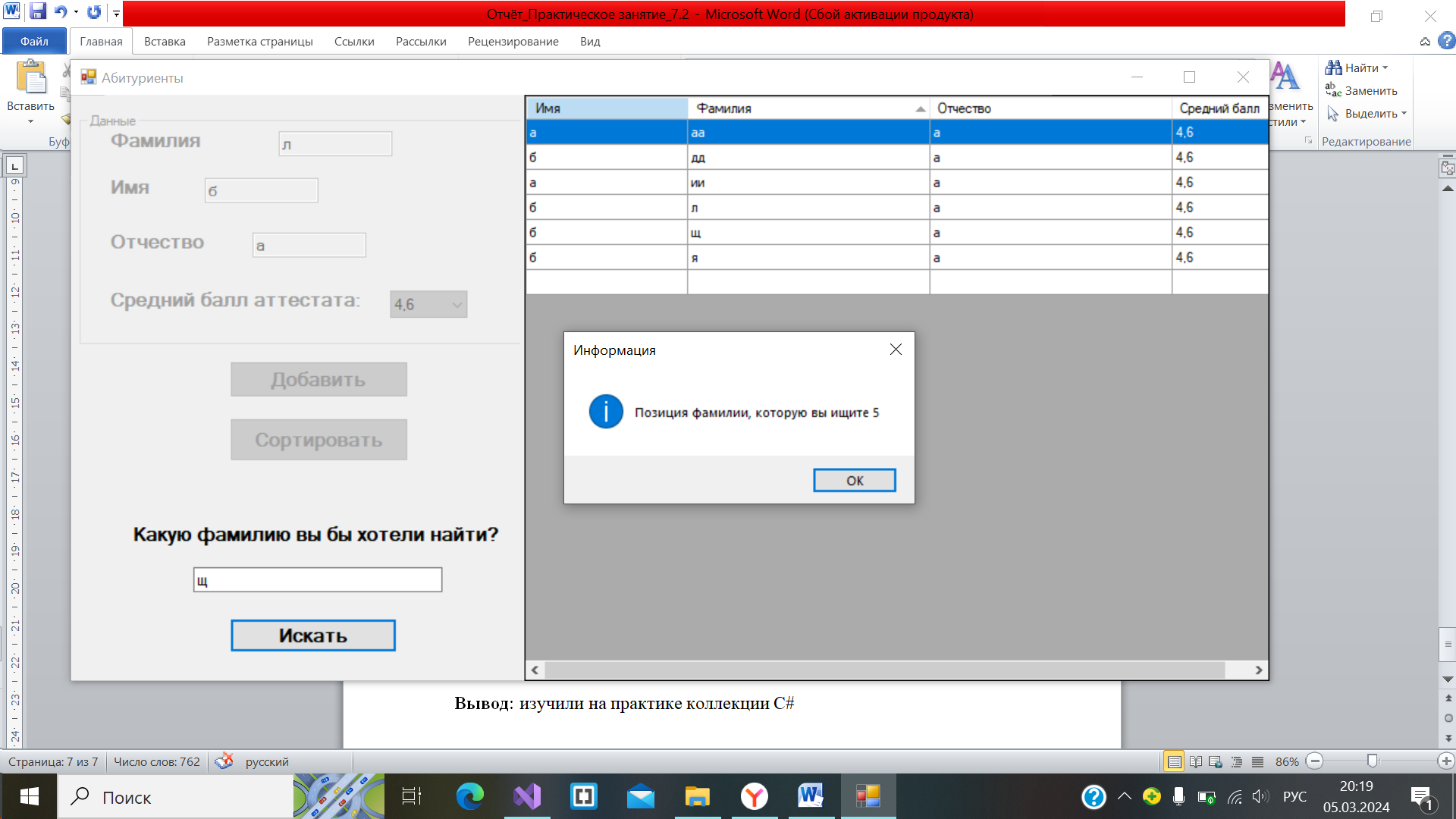
**Тестовые ситуации:**











**Вывод:** изучили на практике коллекции C#