МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ»

Кафедра вычислительной техники

**Расчётно-графическое задание**

**Вариант №4**

По дисциплине: «Информационные системы»

|  |  |
| --- | --- |
| Факультет: АВТФ  Группа: АВТ-018  Студентка: Наумова Д.К. | Преподаватель:  Дубков И.С. |

**Цель работы:**

Разработать информационную систему, которая автоматизирует учебный процесс и обрабатывает данные о преподавателях, кафедрах, дисциплинах, читаемых преподавателями кафедры.

**Задание:**

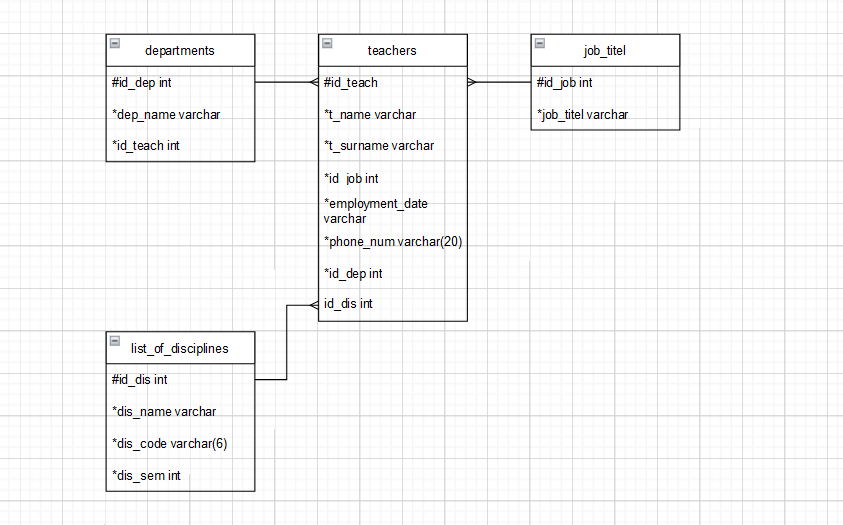
Система должна содержать:

1. Списки кафедр (название, номер, заведующий, факультет);
2. Списки преподавателей (ФИО, дата приема на работу, должность, номер преподавателя, кафедра);
3. Список дисциплин (название, код, семестр, специальность).

Система должна:

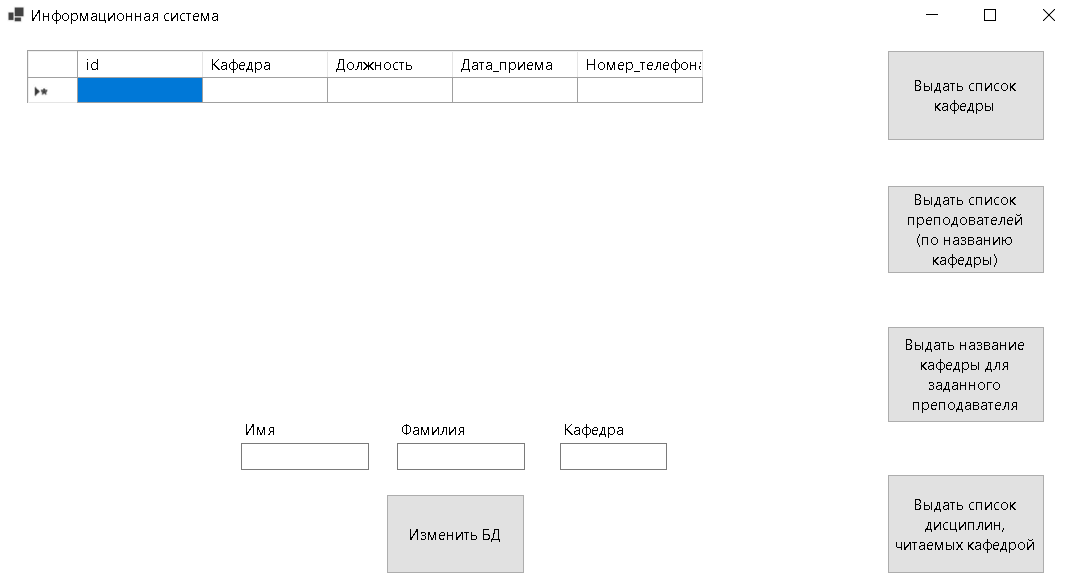
1. Выдавать списки преподавателей заданной кафедры;
2. Выдать название кафедры для заданного преподавателя и все сведения о нем;
3. Выводить список дисциплин, читаемых кафедрой;
4. Выводить список кафедр университета;
5. Иметь средства для ввода и модификации всех перечисленных данных.

**Схема базы данных:**

****

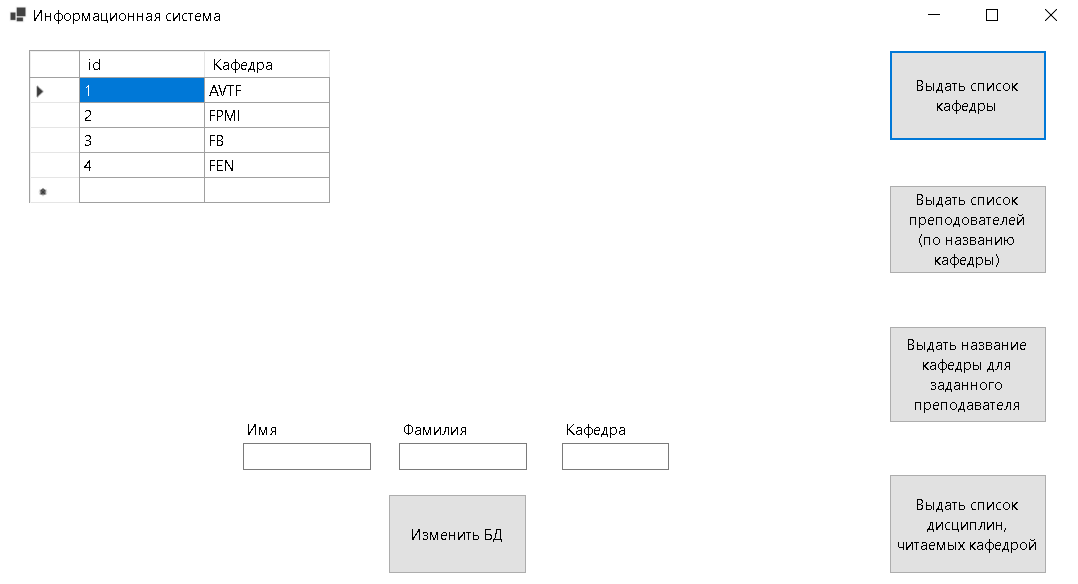
*Рис. 1. Схема базы данных isrgr.*

**Демонстрация работы приложения:**



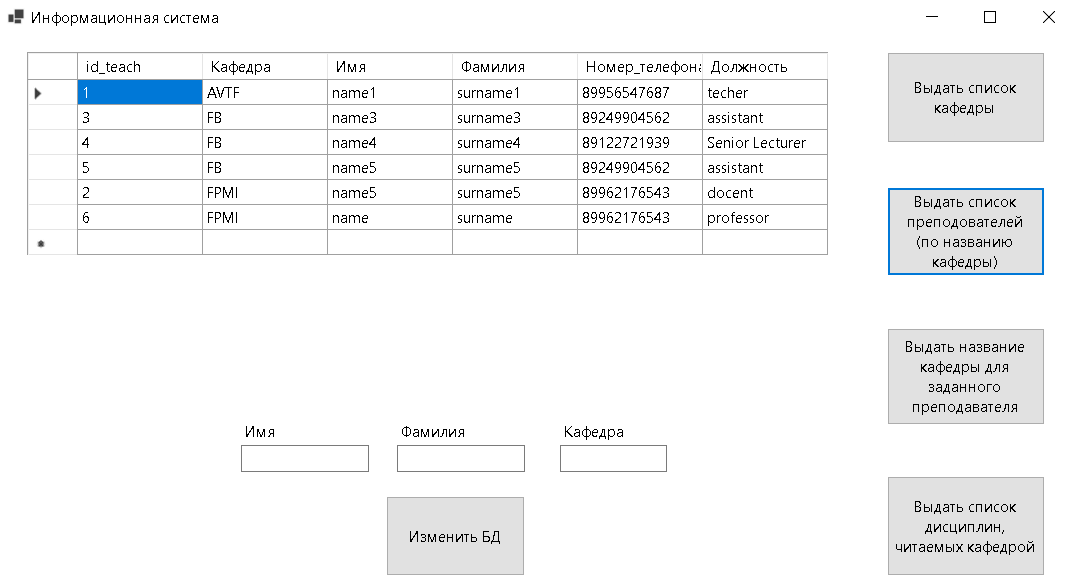
*Рис. 2. Начальный экран приложения.*

При нажатии на кнопку «Выдать список кафедры» в элемент DataGridView выводится таблица БД с кафедрами.

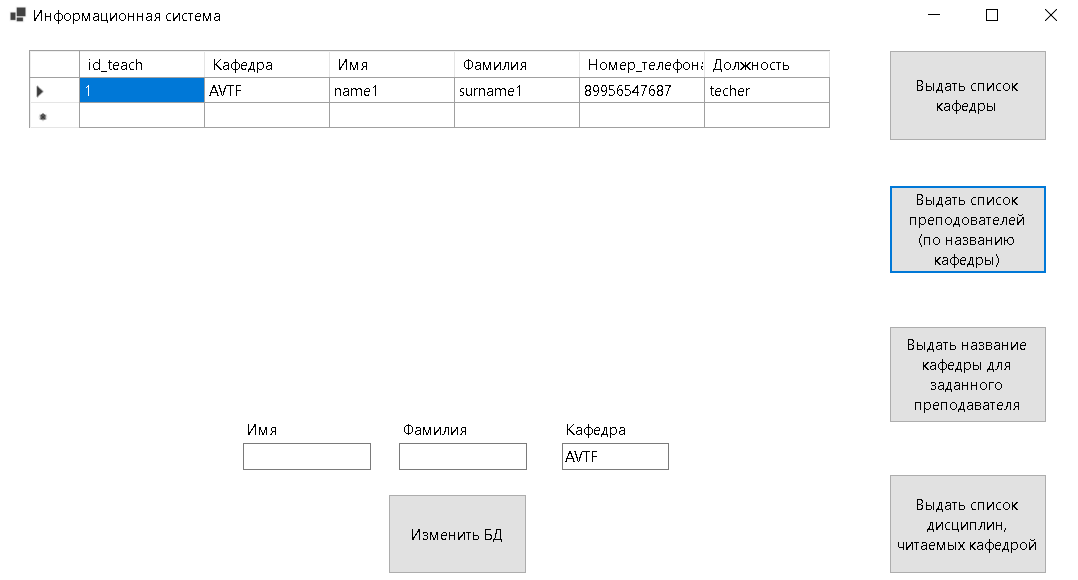


*Рис. 3. Полученный список кафедр.*

При нажатии на кнопку «Выдать список преподавателей» в элемент DataGridView выводится таблица с преподавателями, если ввести название кафедры выводится список преподавателей с данной кафедры.

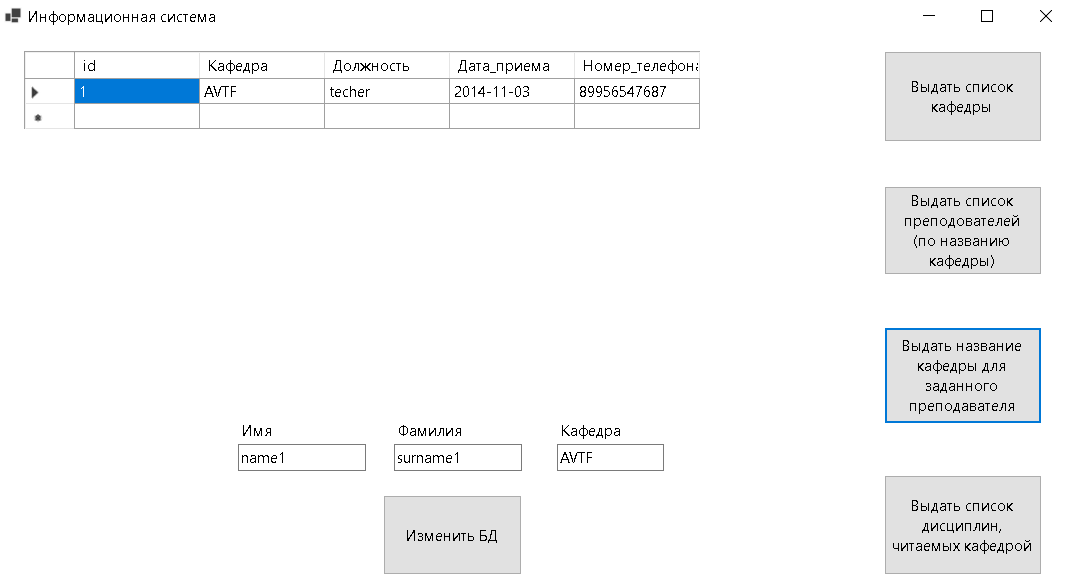


*Рис. 4. Полученный список преподавателей.*



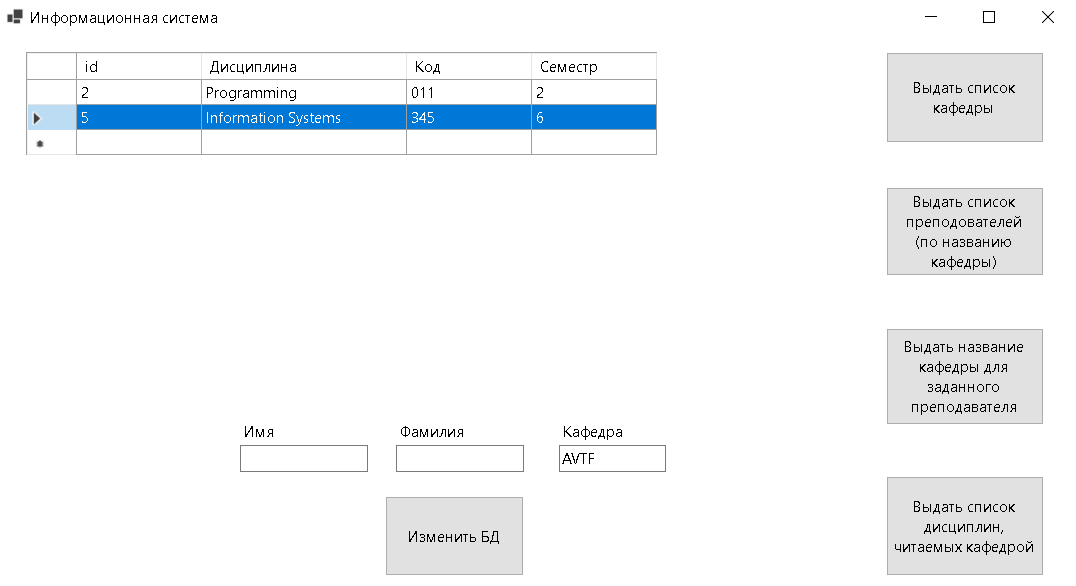
*Рис. 5. Полученный список преподавателей с кафедры “AVTF”.*

Если ввести имя и фамилию, при нажатии на кнопку «Выдать название кафедры для заданного преподавателя» в элемент DataGridView выводятся данные об этом преподавателе.



*Рис. 6. Данные о преподавателе “name1 surname1”.*

При нажатии на кнопку «Выдать список дисциплин, читаемых кафедрой» в элемент DataGridView выводятся дисциплины, читаемые этой кафедрой.



*Рис. 7. Данные о дисциплинах читаемых кафедрой.*

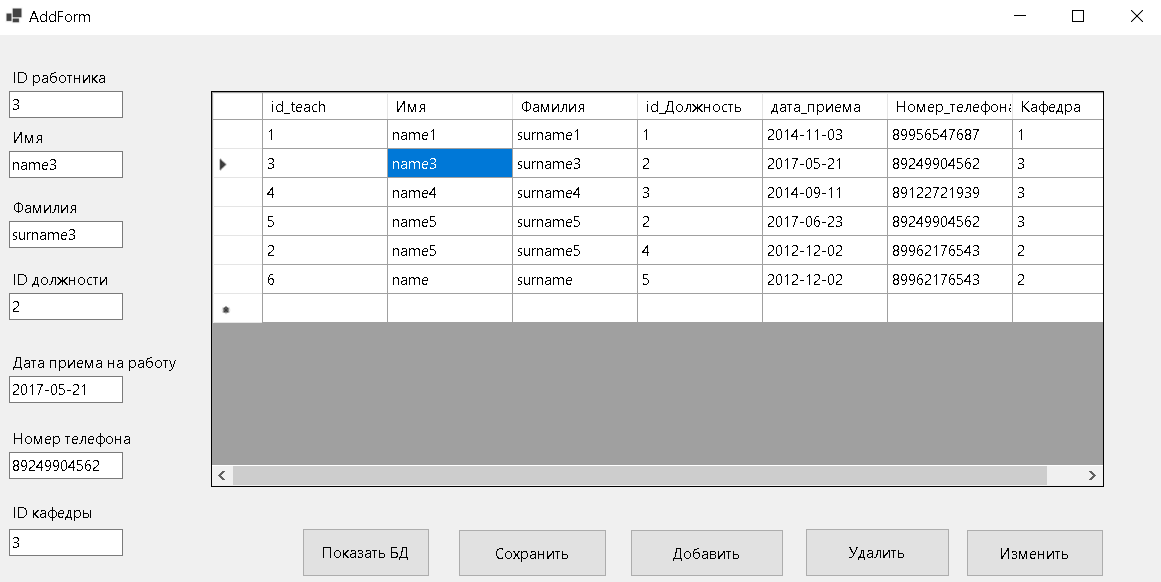
При нажатии на кнопку «Изменить БД» открывается новое окно – AddForm. В данную форму вносятся данные пассажира и информация о рейсе.

При нажатии на кнопку «Зарезервировать» окно закрывается и в элемент DataGridView на главной форме выводится список резерваций.



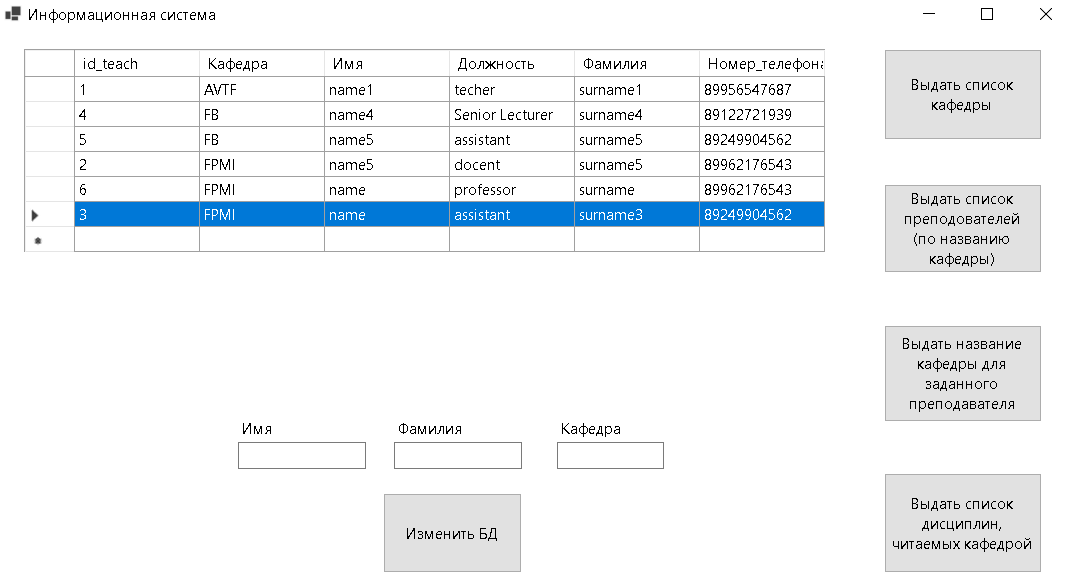
*Рис.8. Форма редактирования таблицы с преподавателями.*

При нажатии на кнопку «Показать БД» в элемент DataGridView выводится список преподавателей и их данных. При нажатии на преподавателя вводятся данные о нем в текст боксах.



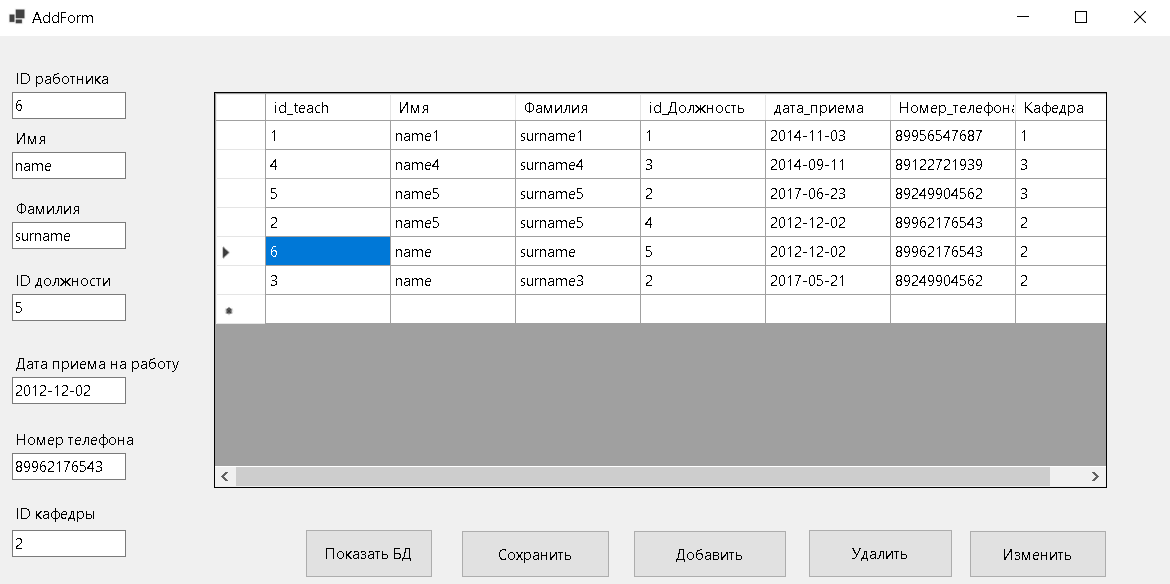
*Рис. 9. Список преподавателей.*

При изменении данных в TextBox и нажатии на кнопку «Изменить», затем «Сохранить » в таблице преподавателей сохраняются данные.

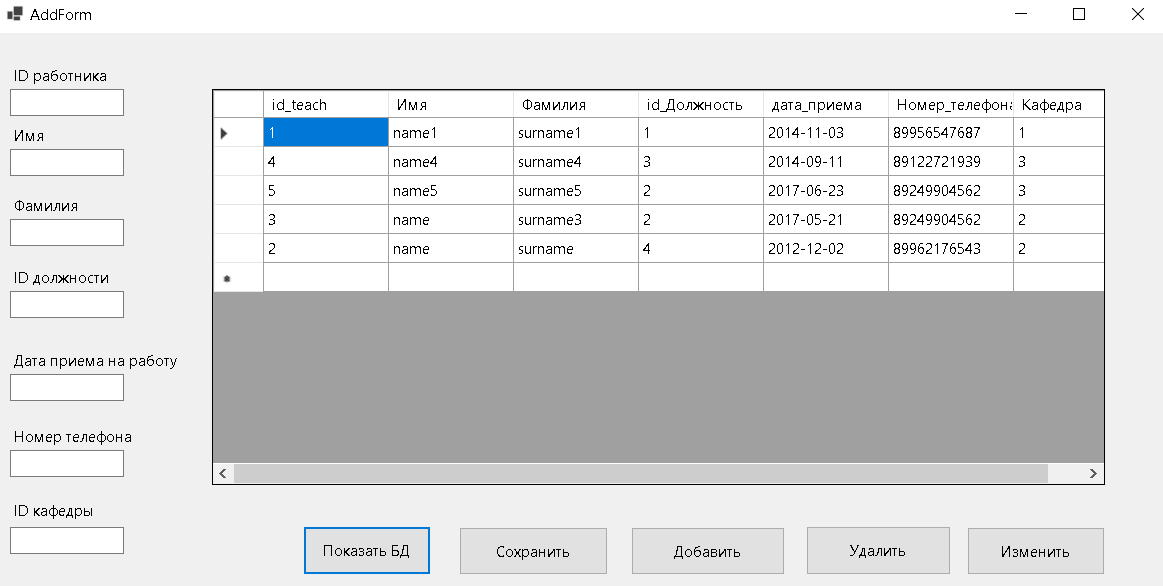


*Рис. 10. Измененный список преподавателей.*

При нажатии на кнопку «Удалить» и «Сохранить» из списка преподавателей удаляется данные о преподавателе, заданные в поле TextBox.

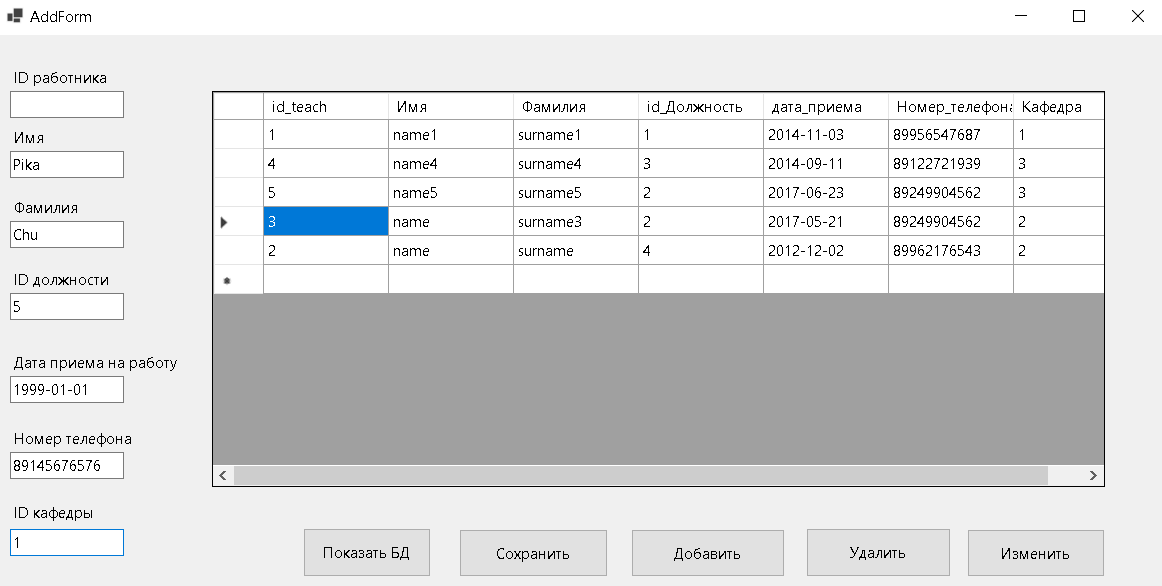


*Рис. 11. Удаление преподавателя с id=6.*

**

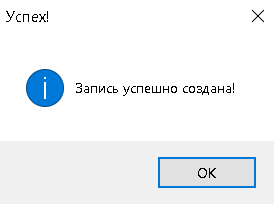
*Рис. 12 Результат удаление преподавателя с id=6.*

При нажатии на кнопку «Добавить» данные из TextBox будут добавлены, если данные введены корректно.

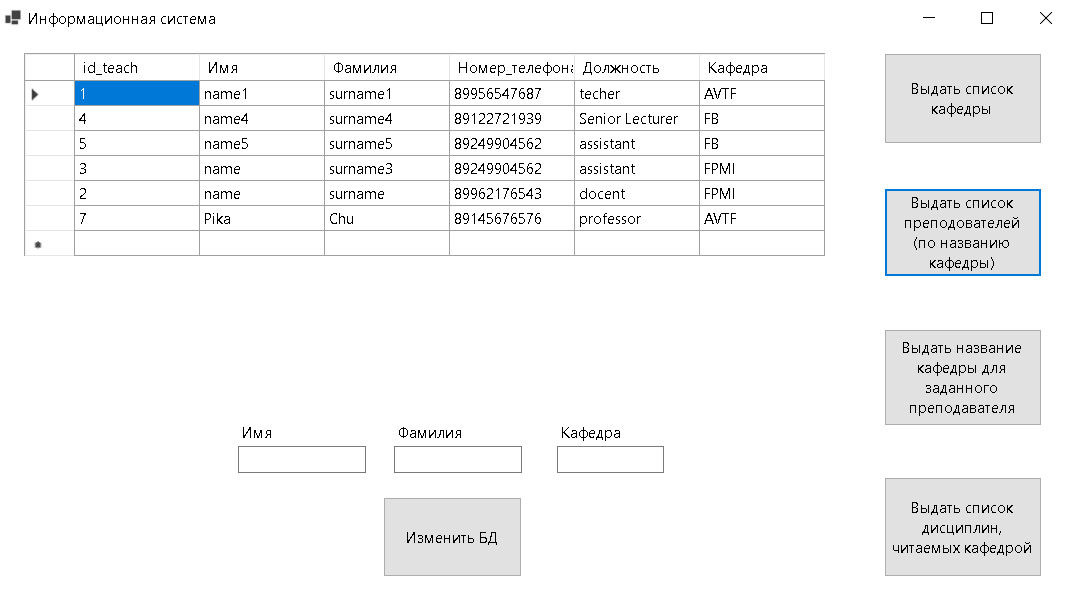


*Рис. 13 Добавление нового преподавателя Pika Chu.*

Если данные ведены корректно и запись добавлена в таблицу, то выведется окно «Успех».

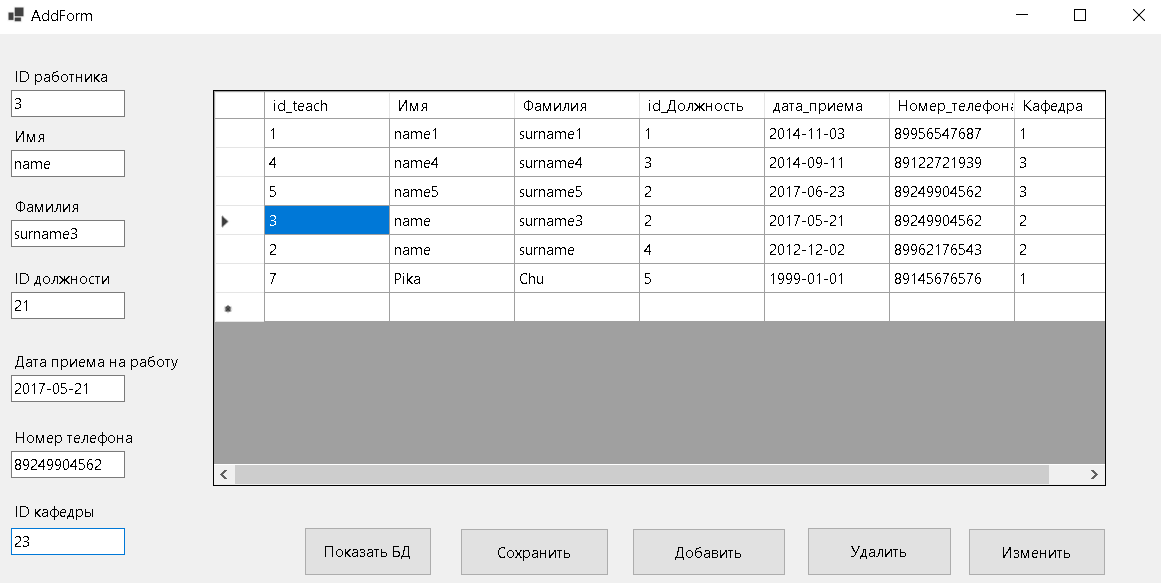


*Рис. 14 Окно о подтверждении добавления данных.*

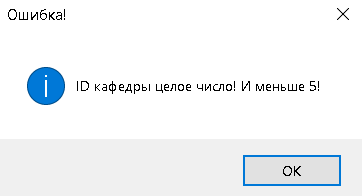
**

*Рис. 15. Обновленный список преподавателей.*

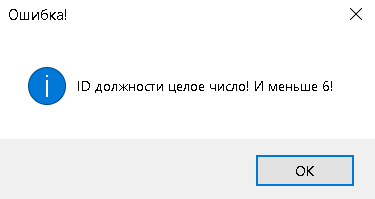
Если ввести не корректные значения, например, в поле ID кафедры или ID должности выведутся окна об ошибке.



*Рис. 16 Добавление нового преподавателя с некорректными данными.*



*Рис. 17 Окно - предупреждение, об ошибке.*



*Рис. 18 Окно - предупреждение, об ошибке.*

**Вывод:**

В результате выполнения расчётно-графического задания была разработана информационная система, которая автоматизирует учебный процесс и обрабатывает данные о преподавателях, кафедрах, дисциплинах, читаемых преподавателями кафедры.

**Листинг программы:**

**База данных:**

Create database isrgr;

//кафедры

CREATE TABLE departments(

id\_dep serial not null primary key,

dep\_name varchar

);

INSERT INTO departments(dep\_name)

VALUES('AVTF'),

('FPMI'),

('FB'),

('FEN');

//должности

CREATE TABLE job\_titel(

id\_job serial not null primary key,

job\_titel varchar

);

INSERT INTO job\_titel(job\_titel)

VALUES('techer'),

('assistant'),

('Senior Lecturer'),

('docent'),

('professor');

//список дисциплин

CREATE TABLE list\_of\_dis(

id\_dis serial not null primary key,

dis\_name varchar,

dis\_code varchar(6),

dis\_sem int,

id\_dep int references departments on delete restrict

);

INSERT INTO list\_of\_dis(dis\_name, dis\_code, dis\_sem, id\_dep)

VALUES('Math','093', 1, 2),

('Programming','011', 2, 1),

('Philosophy','023', 4, 4),

('History','145',1,3),

('Information Systems','345', 6, 1);

//список преподавателей

CREATE TABLE teachers(

id\_teach serial not null primary key,

t\_name varchar,

t\_surname varchar,

id\_job int references job\_titel on delete restrict,

employment\_date varchar,

phone\_number varchar,

id\_dep int references departments on delete restrict,

id\_dis int references list\_of\_dis on delete restrict

);

INSERT INTO teachers(t\_name,t\_surname,id\_job,employment\_date,phone\_number, id\_dep, id\_dis)

VALUES ('name1','surname1',1,'2014-11-03',89956547687,1,2),

('name2','surname2',4,'2015-02-16',89841041424,2,4),

('name3','surname3',2,'2017-05-21',89249904562, 3, 2),

('name4','surname4',3,'2014-09-11',89122721939, 3, 3),

('name5','surname5',2,'2017-06-23',89249904562, 3, 4);

**MainForm.cs**

using System;

using System.Data;

using System.Drawing;

using System.Drawing.Printing;

using System.Windows.Forms;

using Npgsql;

namespace rgz

{

public partial class РГЗ : Form

{

private readonly NpgsqlConnection connection = new(ServerInfo.GetConnectionString());

public РГЗ()

{

InitializeComponent();

Button3\_Click(this, new EventArgs());

}

private void Button3\_Click(object sender, EventArgs e)

{

//5)кнопка, которая, выдает названия кафедры для заданного преподавателя и все его данные.

connection.Open();

var queryString =

@"SELECT id\_teach AS id, dep\_name as Кафедра, job\_titel AS Должность, employment\_date AS Дата\_приема,

phone\_number AS Номер\_телефона from teachers

inner join departments ON teachers.id\_dep=departments.id\_dep

inner join job\_titel ON teachers.id\_job=job\_titel.id\_job where t\_name='" + textBox5.Text + "'and t\_surname='" +

textBox6.Text + "'";

var query = new NpgsqlCommand(queryString, connection);

var reader = query.ExecuteReader();

var table = new DataTable();

table.Load(reader);

dataGridView1.DataSource = table;

connection.Close();

}

private void Button1\_Click(object sender, EventArgs e)

{

//4)выдает список преподавателей всех или заданной кафедры.

// label1.Text = "Преподаватели кафедры " + textBox1.Text;

connection.Open();

string queryString;

if (textBox1.Text == "")

{

queryString =

@"SELECT id\_teach ,t\_name as Имя,t\_surname AS Фамилия, phone\_number as Номер\_телефона,

job\_titel AS Должность, dep\_name as Кафедра FROM teachers

inner join departments ON teachers.id\_dep=departments.id\_dep

inner join job\_titel ON teachers.id\_job=job\_titel.id\_job";

// label1.Text = "Все преподаватели";

}

else

{

queryString =

@"SELECT id\_teach ,t\_name as Имя,t\_surname AS Фамилия, phone\_number as Номер\_телефона,

job\_titel AS Должность, dep\_name as Кафедра FROM teachers

inner join departments ON teachers.id\_dep=departments.id\_dep

inner join job\_titel ON teachers.id\_job=job\_titel.id\_job

where dep\_name='" + textBox1.Text+"'";

}

var query = new NpgsqlCommand(queryString, connection);

var reader = query.ExecuteReader();

var table = new DataTable();

table.Load(reader);

dataGridView1.DataSource = table;

connection.Close();

}

//переход на addforms

private void Button2\_Click(object sender, EventArgs e)

{

var addForm = new AddForm();

addForm.Show();

}

private void Button4\_Click(object sender, EventArgs e)

{

connection.Open();

var queryString =

@"SELECT id\_dis as id, dis\_name as Дисциплина, dis\_code as Код, dis\_sem as Семестр from list\_of\_dis

inner join departments ON departments.id\_dep=list\_of\_dis.id\_dep

where dep\_name='" + textBox1.Text + "';";

var query = new NpgsqlCommand(queryString, connection);

var reader = query.ExecuteReader();

var table = new DataTable();

table.Load(reader);

dataGridView1.DataSource = table;

connection.Close();

}

private void Button6\_Click(object sender, EventArgs e)

{

connection.Open();

var queryString =

@"SELECT id\_dep as id, dep\_name as Кафедра from departments;";

var query = new NpgsqlCommand(queryString, connection);

var reader = query.ExecuteReader();

var table = new DataTable();

table.Load(reader);

dataGridView1.DataSource = table;

connection.Close();

}

private void label6\_Click(object sender, EventArgs e)

{

}

private void label2\_Click(object sender, EventArgs e)

{

}

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

{

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

}

}

**AddForm.cs**

using System;

using System.Data;

using System.Linq;

using System.Text;

using System.Data.SqlClient;

using System.Drawing;

using System.Drawing.Printing;

using System.Windows.Forms;

using Npgsql;

namespace rgz

{

enum RoWState

{

Existed,

New,

Modified,

ModifiedNew,

Deleted

}

//форма добавления

public partial class AddForm : Form

{

private readonly NpgsqlConnection connection = new(ServerInfo.GetConnectionString());

public AddForm()

{

InitializeComponent();

}

private void Change()

{

var selectedRowIndex = dataGridView1.CurrentCell.RowIndex;

var id\_teach= textBox1.Text;

var t\_name = textBox2.Text;

var t\_surname = textBox3.Text;

int id\_job;

var employment\_date = textBox5.Text;

var phone\_num = textBox6.Text;

int id\_dep;

if (dataGridView1.Rows[selectedRowIndex].Cells[0].Value.ToString() != string.Empty)

{

if ((int.TryParse(textBox4.Text, out id\_job) && (id\_job < 6)) && ((int.TryParse(textBox7.Text, out id\_dep)) && (id\_dep < 5)))

{

dataGridView1.Rows[selectedRowIndex].SetValues(id\_teach, t\_name, t\_surname, id\_job, employment\_date, phone\_num, id\_dep);

dataGridView1.Rows[selectedRowIndex].Cells[6].Value = RoWState.Modified;

}

else

{

MessageBox.Show("Ошибка формата записи!");

}

}

}

private void Button1\_Click(object sender, EventArgs e)

{

connection.Open();

var t\_name = textBox2.Text;

var t\_surname = textBox3.Text;

int id\_job;

var employment\_date = textBox5.Text;

var phone\_num = textBox6.Text;

int id\_dep;

if ((int.TryParse(textBox4.Text, out id\_job) && (id\_job < 6)) && ((int.TryParse(textBox7.Text, out id\_dep)) && (id\_dep < 5)))

{

var queryString =

$"INSERT INTO teachers(t\_name,t\_surname,id\_job,employment\_date,phone\_number,id\_dep)" +

$" values('{t\_name}','{t\_surname}',{id\_job},'{employment\_date}','{phone\_num}',{id\_dep})";

var query = new NpgsqlCommand(queryString, connection);

query.ExecuteReader();

MessageBox.Show("Запись успешно создана!", "Успех!", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

{

if (int.TryParse(textBox7.Text, out id\_dep) && (id\_dep < 5))

{

MessageBox.Show("ID кафедры целое число! И меньше 5!", "Ошибка!", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

if (int.TryParse(textBox4.Text, out id\_job) && (id\_job < 6))

{

MessageBox.Show("ID должности целое число! И меньше 6!", "Ошибка!", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

connection.Close();

Close();

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void label5\_Click(object sender, EventArgs e)

{

}

int indexRow;

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

indexRow = e.RowIndex; // get the selected Row Index

DataGridViewRow row = dataGridView1.Rows[indexRow];

textBox1.Text = row.Cells[0].Value.ToString();

textBox2.Text = row.Cells[1].Value.ToString();

textBox3.Text = row.Cells[2].Value.ToString();

textBox4.Text = row.Cells[3].Value.ToString();

textBox5.Text = row.Cells[4].Value.ToString();

textBox6.Text = row.Cells[5].Value.ToString();

textBox7.Text = row.Cells[6].Value.ToString();

}

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

{

connection.Open();

string queryString;

queryString =

@"SELECT id\_teach ,t\_name as Имя,t\_surname AS Фамилия,

id\_job AS ID\_Должность,

employment\_date as дата\_приема, phone\_number as Номер\_телефона ,

id\_dep as Кафедра from teachers;

";

var query = new NpgsqlCommand(queryString, connection);

var reader = query.ExecuteReader();

var table = new DataTable();

table.Load(reader);

dataGridView1.DataSource = table;

connection.Close();

}

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

{

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

private void label4\_Click(object sender, EventArgs e)

{

}

private void deleteRow()

{

int index = dataGridView1.CurrentCell.RowIndex;

if (dataGridView1.Rows[index].Cells[0].Value.ToString() == string.Empty)

{

dataGridView1.Rows[index].Cells[6].Value = RoWState.Deleted;

return;

}

dataGridView1.Rows[index].Cells[6].Value = RoWState.Deleted;

}

private void update()

{

connection.Open();

for (int index = 0; index < dataGridView1.Rows.Count; index++)

{

var rowState = (RoWState)dataGridView1.Rows[index].Cells[6].Value;

if (rowState == RoWState.Existed)

continue;

if (rowState == RoWState.Deleted)

{

var id = Convert.ToInt32(dataGridView1.Rows[index].Cells[0].Value);

var deleteQuery = $"delete from teachers where id\_teach={id}";

var query = new NpgsqlCommand(deleteQuery, connection);

var reader = query.ExecuteReader();

var table = new DataTable();

table.Load(reader);

dataGridView1.DataSource = table;

}

if (rowState == RoWState.Modified)

{

var id\_teach = Convert.ToInt32(dataGridView1.Rows[index].Cells[0].Value);

var t\_name = textBox2.Text;

var t\_surname = textBox3.Text;

var id\_job = Convert.ToInt32(dataGridView1.Rows[index].Cells[3].Value);

var employment\_date = textBox5.Text;

var phone\_number = textBox6.Text;

var id\_dep = Convert.ToInt32(dataGridView1.Rows[index].Cells[6].Value);

var updateQuery = $"update teachers set t\_name='{t\_name}'," +

$"t\_surname='{t\_surname}',id\_job={id\_job},employment\_date='{employment\_date}'," +

$"phone\_number='{phone\_number}' , id\_dep='{id\_dep}' where id\_teach={id\_teach};";

var query = new NpgsqlCommand(updateQuery, connection);

var reader = query.ExecuteReader();

var table = new DataTable();

table.Load(reader);

dataGridView1.DataSource = table;

}

}

connection.Close();

Close();

}

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

{

deleteRow();

}

private void button4\_Click(object sender, EventArgs e)

{

update();

}

private void button5\_Click(object sender, EventArgs e)

{

Change();

}

private void label3\_Click(object sender, EventArgs e)

{

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

}

}