МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Вятский государственный университет»**

**(ФГБОУ ВО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

«Базы данных»

Связывание приложения на C# с базой данных под управлением PostgreSQL

Выполнил студент группы ИВТб-31 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Кудяшев Я.Ю./

(подпись)

Проверил преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Клюкин В.Л/

(подпись)

Киров 2022

# Задание и цель лабораторной работы

Цель работы:

* Познакомиться со стандартным API в С# для связывания приложения с БД
* Изучить некоторые шаблоны проектирования, связанные с работой с БД
* Освоить на практике основы взаимодействия с БД под управлением PostgreSQL в приложении на С#

Задание на лабораторную работу:

Создать приложение с графическим приложением на языке С#. Приложение должно использовать БД, разработанную в предыдущих лабораторных работах.

Для любой одной таблицы, которая содержит внешний ключ на другую таблицу, приложение должно выполнять следующее:

* Выводить строки таблицы
* Предоставлять любой фильтр по значениям строк. (Например, «Дата с … по …» или «Имя содержит …»)
* Предоставлять возможность добавления новых строк
* Предоставлять возможность удаления строки

Требования к реализации:

* Заголовки должны быть осмысленными. Например, вместо «name» в таблице должен быть заголовок «Имя»
* При добавлении новой строки внешний ключ выбирается из списка
* Сохранение или удаление строки должно быть реализовано с помощью функции PL/pgSQL
* Фильтрация значений при поиске должна производиться через запрос, а не в полученной коллекции
* Разрешается использование любого фреймворка
* При разработке нужно использовать шаблоны проектирования, связанные с работой с БД

# Экранные формы

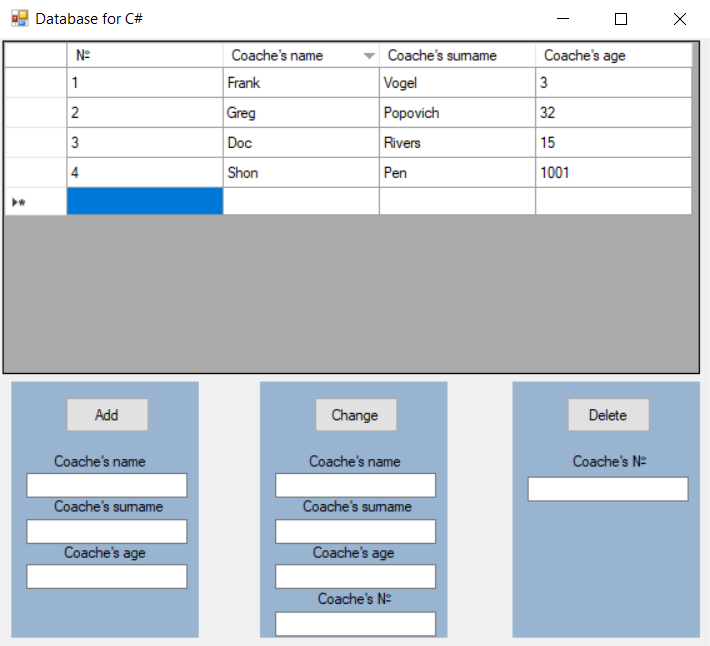


Рисунок 1 – Пример отображения приложения

# Код программы

**Form1.cs**

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 Database

{

public partial class Form1 : Form

{

int counter=4;

string[,] players = new string[8, 4]{

{"1", "Frank", "Vogel", "3"},

{ "2","Greg","Popovich","32"},

{ "3","Doc","Rivers","15"},

{ "4","Shon","Pen","1001"},

{ "","","",""},

{ "","","",""},

{ "","","",""},

{ "","","",""},

};

public Form1()

{

InitializeComponent();

}

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

{

dataGridView1.Rows.Add();

dataGridView1.Rows.Add();

dataGridView1.Rows.Add();

dataGridView1.Rows.Add();

for (int i = 0; i < 4; i++)

{

for(int j = 0; j < 4; j++)

{

dataGridView1.Rows[i].Cells[j].Value = players[i, j];

}

}

}

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

{

counter++;

dataGridView1.Rows.Add(counter, textBox1.Text, textBox2.Text, textBox3.Text);

}

private void label7\_Click(object sender, EventArgs e)

{

}

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

{

dataGridView1[1, int.Parse(textBox7.Text)+1].Value = textBox6.Text;

dataGridView1[ 2, int.Parse(textBox7.Text)+1].Value = textBox5.Text;

dataGridView1[ 3, int.Parse(textBox7.Text)+1].Value = textBox4.Text;

}

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

{

// int ind = dataGridView1.SelectedCells[0].RowIndex;

dataGridView1.Rows.RemoveAt(int.Parse(textBox9.Text)-1);

}

}

}

1. **Вывод**

В ходе выполнения лабораторной работы были изучены стандартные API в С# для связывания приложения с БД**,** изучены некоторые шаблоны проектирования, связанные с работой с БД и освоены на практике основы взаимодействия с БД под управлением PostgreSQL в приложении на C#.