Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №9**

**«Работа с типизированными файлами»**

**ПО «МДК 05.02 Разработка кода информационных систем»**

Выполнил: студент учебной группы

ИСПк-204-52-00

Титков Дмитрий Михайлович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

1. Цель работы: получить представление об организации хранения и использования информации посредством типизированных файлов, закрепить навыки создания пользовательского интерфейса.
2. Задание:
3. Сформировать структуру записи, с которой будет осуществляться работа, и согласовать ее с преподавателем.
4. Разработать приложение, позволяющее извлекать набор записей из типизированного файла, визуализировать данный набор, а также позволяющее изменять данных и выполнять сохранение проделанных изменений.

Описание алгоритма

Было реализовано приложение «Table» с помощью виртуальной среды разработки Visual Studio с использованием WPF и интеграцией базы данных на удаленном сервере PostgreSQL, взаимодействие с которой происходит через фреймворк EntityFrameworkCore.

Приложение работает следующим образом: перед пользователем предстает окно, в котором есть несколько интерактивных и 1 автономный элемент. Интерактивные элементы представляют из себя кнопки, которые взаимодействуют с интерфейсом программы, автономный – таблица.

Кнопки в программе выполняют несколько функций: удаление записи, добавление записи, сохранение всех записей.

Пользователь может вводить в таблицу любые данные, в соответствии с заголовками таблицы. Данные будут сохранятся в базу данных после нажатия на соответствующую кнопку на интерфейсе.

Описание используемых компонентов

В программе использовались только оригинальные компоненты WPF с разметкой XAML. Для реализации хранения данных использовалась PostgreSQL.



Рисунок 1 – Button

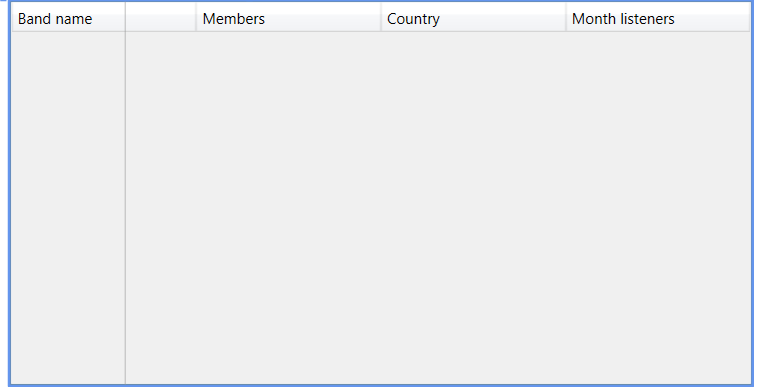


Рисунок 2 – DataGrid

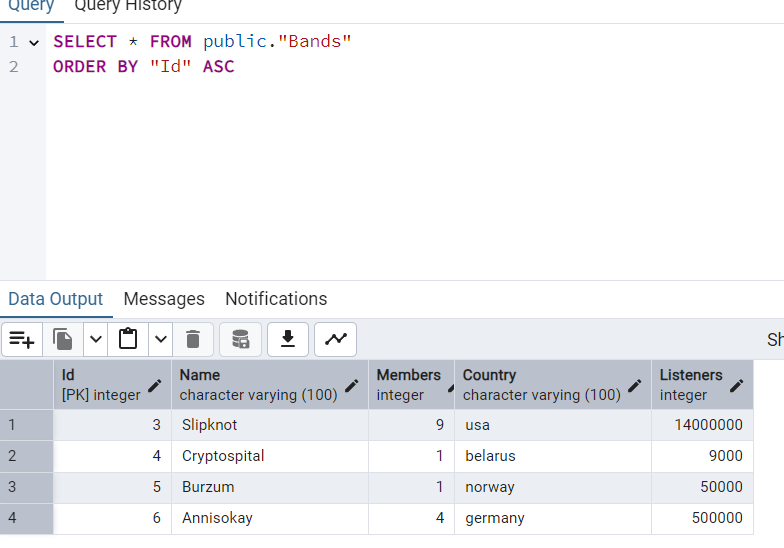


Рисунок 3 – База данных в PostgreSQL

Код программы

public partial class MainWindow : Window

{

private ObservableCollection<Band> bands { get; set; }

public MainWindow()

{

InitializeComponent();

LoadData();

Bands.ItemsSource = bands;

}

private void LoadData()

{

using (var db = new ApplicationContext())

{

bands = new ObservableCollection<Band>(db.Bands.ToList());

Bands.ItemsSource = bands;

}

}

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

{

var newBand = new Band

{

Name = "",

Members = 0,

Country = "",

Listeners = 0

};

using (var db = new ApplicationContext())

{

db.Bands.Add(newBand);

db.SaveChanges();

}

bands.Add(newBand);

}

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

{

using (var db = new ApplicationContext())

{

Console.WriteLine(Bands);

try

{

foreach (var band in bands)

{

var existing = db.Bands.FirstOrDefault(b => b.Id == band.Id);

if (existing != null)

{

existing.Name = band.Name;

existing.Members = band.Members;

existing.Country = band.Country;

existing.Listeners = band.Listeners;

}

else

{

db.Bands.Add(band);

}

}

db.SaveChanges();

}

catch

{

MessageBox.Show("Enter all necessary data!", "Save error",

MessageBoxButton.OK, MessageBoxImage.Error);

}

}

}

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

{

Band band = Bands.SelectedItem as Band;

if (band != null) bands.Remove(band);

using (var db = new ApplicationContext())

{

if (band != null)

{

db.Bands.Remove(band);

db.SaveChanges();

}

}

LoadData();

}

###

public class Band

{

[Key]

public int Id { get; set; }

public string Name { get; set; }

public int Members { get; set; }

public string Country { get; set; }

public int Listeners { get; set; }

}

###

public class ApplicationContext : DbContext

{

public ApplicationContext()

{

Database.EnsureCreated();

}

public DbSet<Band> Bands { get; set; }

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

optionsBuilder.UseNpgsql("Host=194.87.55.155;Port=5432;Database=vp25\_new;Username=sane5k;Password=JeckDog093");

}

}

Результат выполнения программы

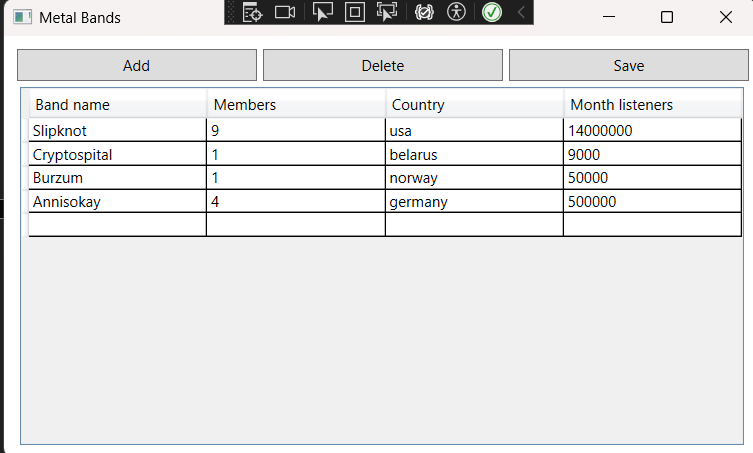


Рисунок 4 – Главное меню

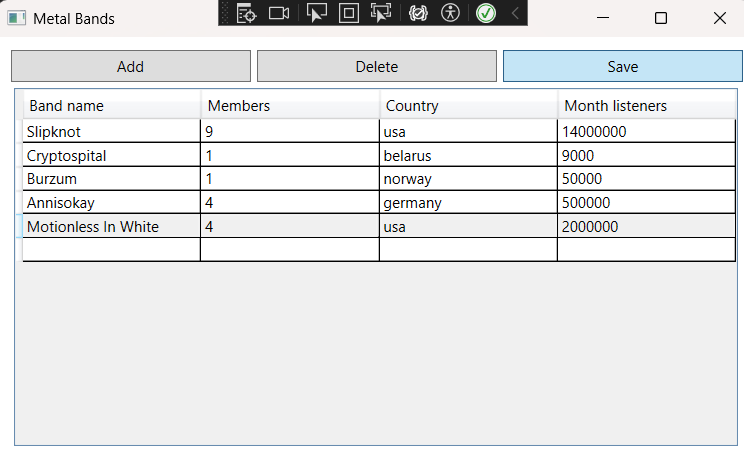


Рисунок 5 – Программа после добавления записи

Вывод

В ходе выполнения контрольной работы по реализации калькулятора объема фигур были изучены базовые навыки взаимодействия с WPF. Получены знания для работы с XAML разметкой.

Было разработано приложение таблицы, в котором предусмотрено добавление, редактирование и удаление записей. Также были получены навыки для взаимодействия с базой данных и интеграции ее в программу.