Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

# Отчёт по программе «**Индивидуальные задания по теме «Коллекции в с#» (4 задание)**»

Выполнила: Зиновьева Кристина Дмитриевна

Группа: ПР-24

Преподаватель: Мирошниченко Г.В

2025

**Задача 4**

Решить задачу, используя класс HashTable:реализовать

простейший каталог музыкальных компакт-дисков (музыкальных

подборок), который позволяет:

\* Добавлять и удалять диски.

\* Добавлять и удалять песни.

\* Просматривать содержимое целого каталога и каждого диска в

отдельности.

\* Осуществлять поиск всех записей заданного исполнителя по

всему каталогу.

К методам добавить комментарии.

**Программа**

internal class Katalog

{

public List<string> Albom { get; set; } = new List<string>();

public List<string> Pesni { get; set; } = new List<string>();

private string fail = "Песни.txt"; //Файл куда сохраняются альбомы и песни

public void Sawe()//метод для сохранения

{

List<string> lines = new List<string>();

foreach (string album in Albom)

lines.Add($"Альбом:{album}");

foreach (string song in Pesni)

lines.Add($"Песня:{song}");

File.WriteAllLines(fail, lines);

}

public void DobavitAlbom(string artist, string title)//доступ для добавления

{

string album = $"{artist} - {title}";

if (!Albom.Contains(album))

{

Albom.Add(album);

Sawe();

}

}

public void UdalitAlbom(string album)//доступ для удаления

{

Albom.Remove(album);

Pesni.RemoveAll(p => p.StartsWith(album + "|"));

Sawe();

}

public void DobavitPesny(string album, string artist, string song)//добавление песни

{

Pesni.Add($"{album}|{artist} - {song}");

Sawe();

}

public void UdalitPesny(string song)//доступ для удаления

{

Pesni.Remove(song);

Pesni.RemoveAll(p => p.StartsWith(song + "|"));

Sawe();

}

public List<string> PoluchitPesni(string album)

{

List<string> result = new List<string>();

foreach (string song in Pesni)//перебор каждой буквы и знака строки

{

if (song.StartsWith(album + "|")) //если песня начинается с "Альбом|..."

result.Add(song.Substring(album.Length + 1));

}

return result;

}

}

public partial class Form1 : Form

{

private string file = "Песни.txt";//тексстовый файл

public Form1()

{

InitializeComponent();

}

private Katalog katalog = new Katalog();

// Обновить список альбомов на форме

private void Restart()//обновление листбокса

{

listBox1.Items.Clear();

foreach (string album in katalog.Albom)//проходим по всем элементам

listBox1.Items.Add(album);

}

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

{

string artist = textBox3.Text;

string title = textBox4.Text;

if (!string.IsNullOrEmpty(artist) && !string.IsNullOrEmpty(title))//проверка что все введено

{

katalog.DobavitAlbom(artist, title);

Restart();

textBox3.Clear();

textBox4.Clear();

}

}

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

{

if (listBox1.SelectedItem == null)

{

MessageBox.Show("Выберите альбом для удаления");

return;

}

string albumToDelete = listBox1.SelectedItem.ToString();

katalog.UdalitAlbom(albumToDelete);

Restart();

textBox1.Clear();//очищение полей

textBox2.Clear();

}

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

{

if (listBox1.SelectedItem == null)

{

MessageBox.Show("Сначала выберите альбом!");

return;

}

string artist = textBox2.Text;

string song = textBox1.Text;

string album = listBox1.SelectedItem.ToString();

if (!string.IsNullOrEmpty(artist) && !string.IsNullOrEmpty(song))//проверка что все введено

{

katalog.DobavitPesny(album, artist, song);

textBox2.Clear();//очищение полей

textBox1.Clear();

}

}

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

{

if (listBox1.SelectedItem != null)

{

string songToDelete = listBox1.SelectedItem.ToString();

katalog.UdalitPesny(songToDelete);

Restart();

textBox1.Clear();//очищение полей

textBox2.Clear();

}

else

{

MessageBox.Show("Введите название и исполнитея песни");

return;

}

}

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

{

}

private void button5\_Click\_1(object sender, EventArgs e)//<<

{

if (listBox1.SelectedIndex != -1)

{

// Если выделенный элемент не последний перемещаем выделение вниз

if (listBox1.SelectedIndex < listBox1.Items.Count - 1)

{

listBox1.SelectedIndex++;

}

else

{

// Если выделенный элемент последний перемещаем выделение на первый элемент

listBox1.SelectedIndex = 0;

}

}

else

{

MessageBox.Show("Пожалуйста, выберите элемент", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

private void button6\_Click(object sender, EventArgs e)//>>

{

// Проверяем выбран ли элемент в ListBox

// свойство SelectedIndex указывает индекс текущего выбранного элемента

if (listBox1.SelectedIndex != -1)

{

// Если выделенный элемент не первый перемещаем вверх

if (listBox1.SelectedIndex > 0)

{

listBox1.SelectedIndex--;

}

else

{

// Если выделенный элемент первы перемещаем на последний элемент

listBox1.SelectedIndex = listBox1.Items.Count - 1;

}

}

else

{

MessageBox.Show("Пожалуйста, выберите элемент", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

private void button7\_Click(object sender, EventArgs e)

{

if (listBox1.SelectedItem == null)

{

MessageBox.Show("Выберите альбом");

return;

}

string album = listBox1.SelectedItem.ToString();

var pesni = katalog.PoluchitPesni(album);

string info = $"Альбом: {album}\n\nПесни:\n";

if (pesni.Count > 0)

info += string.Join("\n", pesni);

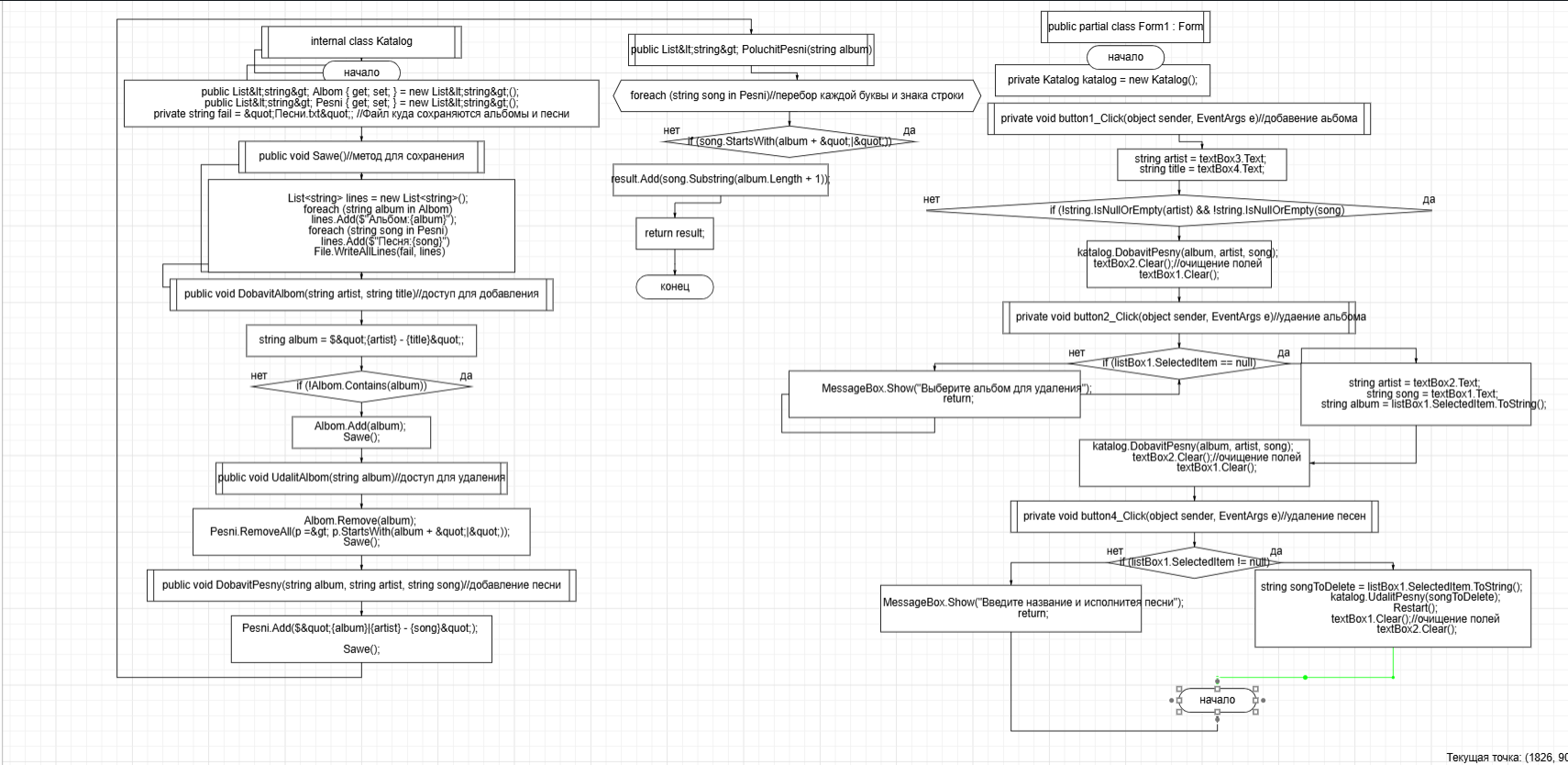
else

info += " Нет добавленных песен";

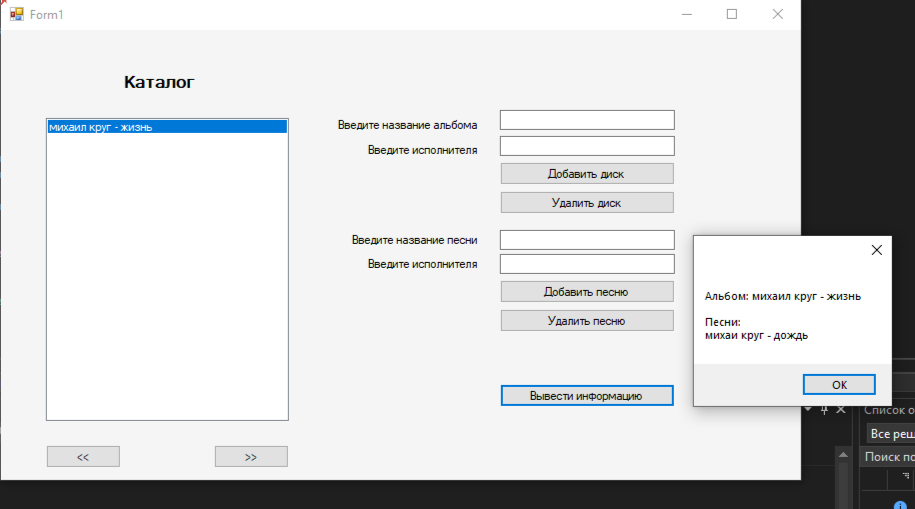
MessageBox.Show(info);

}

}

**Блоксхема**

**Листинг**

****

**Вывод:**

Мы сделали простейший каталог музыкальных компакт-дисков (музыкальных

подборок)