Учреждение образования

«Гродненский государственный политехнический колледж»

**ОТЧЁТ  
ПО УЧЕБНОЙ ПРАКТИКЕ ПО ПРОГРАММИРОВАНИЮ**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Учащегося |  | 3 | | | | |  | курса, группы |  | ПЗТ-40 |
| специальности | | |  | 2 - 40 01 01 «Программное обеспечение информационных | | | | | | |
|  | | |  | технологий» | | | | | | |
| Место прохождения практики | | | | |  | УО «Гродненский государственный | | | | |
|  | | | | |  | политехнический колледж» | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Тема проекта: |  | «Разработка программного обеспечения «Песни Великой | | |
|  |  | Отечественной Войны» | | |
| Ссылка на проект: | | |  | https://github.com/MatveyBaulin/- |
|  | | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Выполнил |  |  |  | М.О. Баулин |
|  |  |  |  | (инициалы, фамилия) |
| Руководитель практики от колледжа |  |  |  | А.Г. Бабуль |
|  |  |  | (инициалы, фамилия) |

Содержание

[Введение. Описание структуры предприятия 3](#_Toc194324354)

[1 Анализ предметной области и формулировка требований к программе 4](#_Toc194324355)

[1.1 Исследование предметной области 4](#_Toc194324356)

[1.2 Инструменты разработки 4](#_Toc194324357)

[2 Проектирование 5](#_Toc194324358)

[2.1 Диаграмма объектов 5](#_Toc194324359)

[2.2 Диаграмма деятельности 5](#_Toc194324360)

[2.3 Описание тестов](#_Toc194324361) 6

[3 Построение программы 7](#_Toc194324362)

3.1 Руководство программиста 7

3.2 Создание вопросов диктанта 7

3.3 Создание главного меню и сборника песен 9

[4 Тестирование](#_Toc194324363) 11

4.1 Тесты на использование 11

4.2 Отчёт о результатах тестирования 11

[5 Применение 12](#_Toc194324364)

[5.1 Назначение и условия применения программы](#_Toc194324365) 12

[5.2 Инсталляция](#_Toc194324366) 12

[5.3 Выполнение программы](#_Toc194324367) 12

[Заключение](#_Toc194324368) 15

[Список использованных источников 16](#_Toc194324369)6

[Приложение A Листинг программы 17](#_Toc194324370)7

Приложение Б Тест-кейсы 29

Введение. Описание структуры предприятия

Была поставлена задача по разработке программного обеспечения «Песни Великой Отечественной Войны».

Суть данного проекта заключается в демонстрации легендарных песен времён Великой Отечественной Войны, а также тренировке памяти при помощи музыкального диктанта.

В разделе анализа предметной области и формулировки требований к программе описана сама предметная область, по которой было создано программное обеспечение и применимые инструменты для разработки программного продукта.

В разделе проектирования описаны диаграммы объектов и последовательности, а также описание тестов выполнения программы.

В разделе построения программы описана инструкция по созданию проекта.

В разделе тестирования описаны результаты тестирования программного продукта.

В разделе применения описаны назначение программы, условия работы программы и инструкция по работе с программой.

«Заключение» будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств.

В разделе «Список использованных источников» будет приведён список используемых при разработке источников.

В приложении А будут изображён листинг программы.

В приложении Б будут изображены тест-кейсы программы.

1. Анализ предметной области и формулировка требований к программе
   1. Исследование предметной области

Данный программный продукт был разработан для демонстрации легендарных песен Великой Отечественной Войны. Данная тема важна для каждого, т.к. Великая Отечественная Война – одно из важнейших исторических событий, которое нужно помнить. Также, нужно помнить и песни этих времён, т.к. это тоже является вкладом в историю войны в области музыки.

* 1. Инструменты разработки

Данный программный продукт был разработан в среде Visual Studio 2022.

Visual Studio — это интегрированная среда разработки (IDE), созданная компанией Microsoft. Она предназначена для разработки программного обеспечения на различных языках программирования, таких как C#, VB.NET, C++, Python и других. Visual Studio предоставляет разработчикам мощные инструменты для написания, отладки и тестирования кода, что делает процесс разработки более эффективным и удобным.

1. Проектирование
   1. Диаграмма объектов

Диаграмма объектов содержит в себе информацию о последовательности объектов при прохождении музыкального диктанта.

Диаграмма объектов изображена на рисунке 2.1.

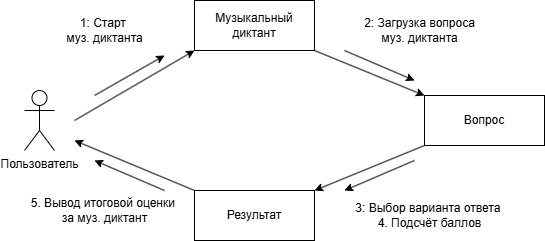


Рисунок 2.1 – Диаграмма объектов

* 1. Диаграмма деятельности

Диаграмма деятельности содержит информацию о действиях пользователя в программном продукте.

Диаграмма деятельности изображена на рисунке 2.2.

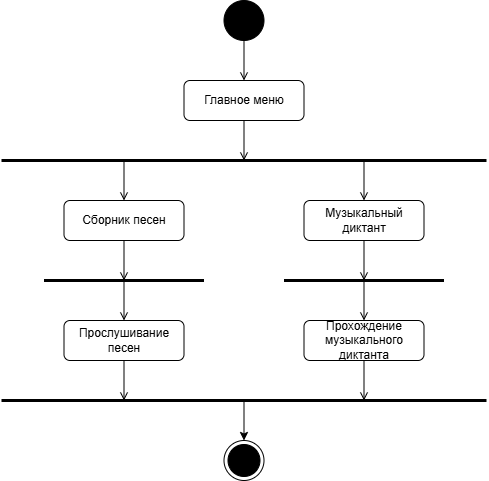


Рисунок 2.2 – Диаграмма деятельности

* 1. Описание тестов

Тест-кейсы изображены в приложении Б.

1. Построение программы
   1. Руководство программиста

Данный программный продукт разрабатывался в программной среде Visual Studio.

Для того, чтобы создать проект, нужно запустить среду. Далее нужно нажать на кнопку «Создание проекта». Потом, выбрать шаблон проекта «Приложение Windows Forms». Выбор шаблона изображён на рисунке 3.1.

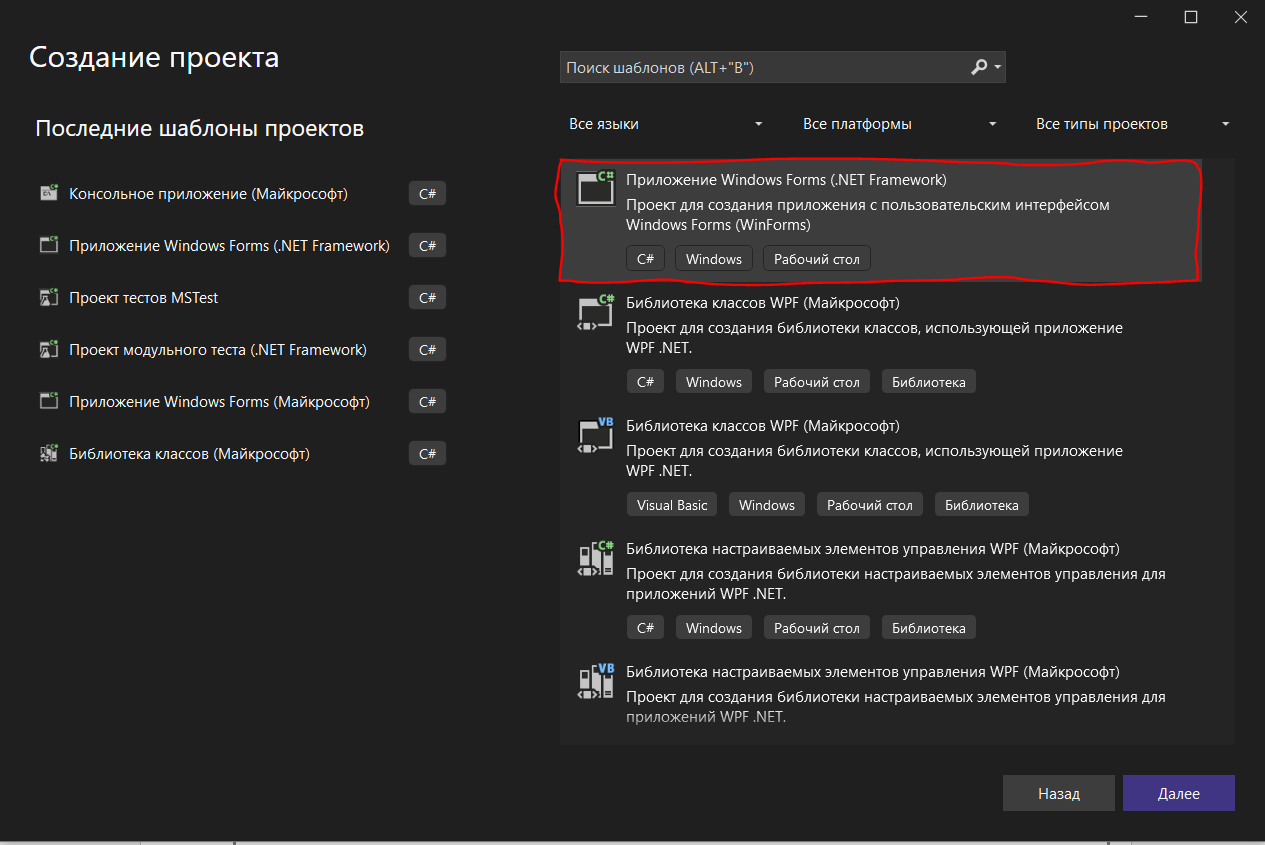


Рисунок 3.1 – Выбор шаблона

Далее, нужно выбрать название проекта и место хранения, а затем нажать на «Создать».

Теперь, нужно создать 12 форм для создания страниц проекта.

10 форм создаются для вопросов, а другие 2 – для главного меню и сборника песен.

* 1. Создание вопросов диктанта

Для того, чтобы создать вопрос в каждой из 10 форм, нужно при помощи панели элементов добавить на форму следующие элементы:

- Кнопка(Button) – для перехода на следующую форму;

- 2 Label – для меток текста;

- 3 кнопки выбора(RadioButton) – для выбора варианта ответа;

- Windows Media Player – для воспроизведения аудиозаписей песен.

- Таймер(timer) – для ограничения выполнения диктанта по времени.

Для того, чтобы выбрать название кнопки, радиокнопки или метки текста, нужно выделить нужный объект и в разделе «Свойства» указать название объекта в параметре «Text».

Создание названия объекта изображено на рисунке 3.2.

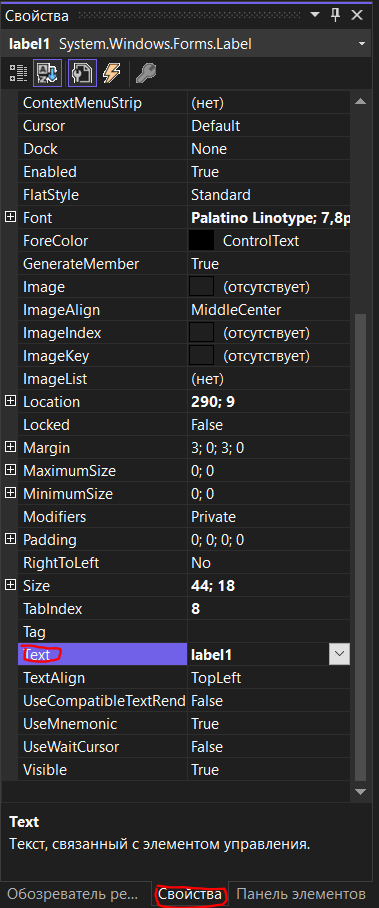


Рисунок 3.2 – Создание названия объекта

Для того, чтобы создать функционал кнопки «Далее», нужно дважды нажать на кнопку при создании формы. Откроется код формы, где сразу создастся метод работы при нажатии на кнопку(button1\_Click).

Для того, чтобы добавить медиаплеер на форму, нужно перейти в панель инструментов, нажать в любом разделе на правую кнопку мыши, нажать на «Выбрать элементы» и во вкладке «Компоненты СОМ» выбрать «Windows Media Player».

Добавление медиаплеера изображено на рисунке 3.3.

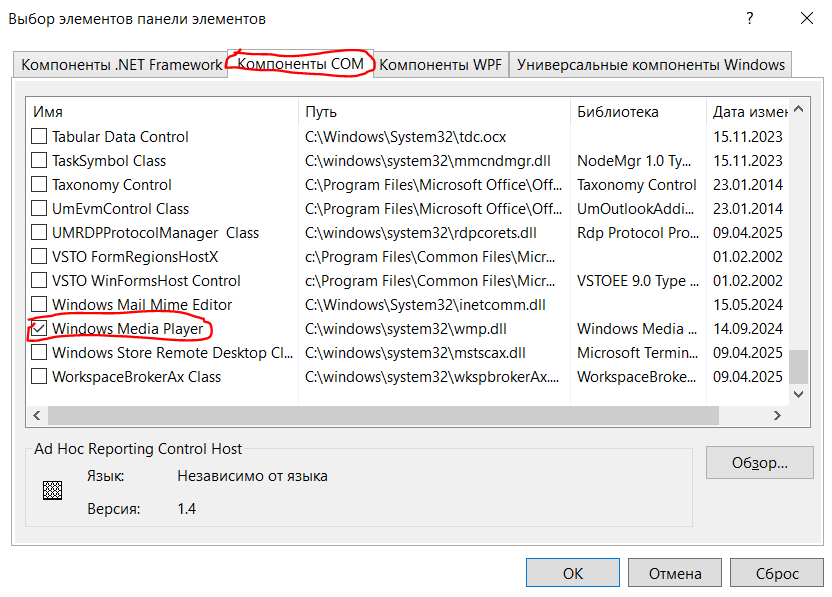


Рисунок 3.3 – Добавление медиаплеера

Для ограничения времени выполнения диктанта, нужно добавить на форму таймер и поставить ограничения при помощи свойства «Interval», которое будет иметь значение 100 – секунда, и при помощи вложенной переменной.

Форма с вопросом готова.

* 1. Создание главного меню и сборника песен

Для создания главного меню понадобятся компоненты label и 3 button.

Две кнопки предназначены для перехода на форму сборника песен и на форму с вопросами диктанта, а третья – для закрытия программы.

Метка текста предназначена для обозначения названия приложения.

Для создания сборника песен понадобятся 10 label, 10 медиаплееров Windows и button.

Метки текста предназначены для обозначения названия печен.

Медиаплееры Windows предназначены для воспроизведения аудиофайлов песен.

Кнопка предназначена для перехода обратно в главное меню.

1. Тестирование

**4.1 Тесты на использование**

При разработке данного программного продукта многие недоработки были исправлены на этапе реализации проекта. Было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов форм в автономном режиме.

Расписание работ над проектом представлено в таблице 4.1.

Таблица 4.1 – Расписание работ над проектом

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Дата | Деятельность | Продолжительность, ч |
| Баулин Матвей | 07.05.2025 | Разработка тестов | 3 |
| Баулин Матвей | 07.05.2025 | Тестирование приложения | 3 |
| Баулин Матвей | 07.05.2025 | Составление отчётов о найденных дефектах | 2 |
| Баулин Матвей | 08.05.2025 | Исправление найденных ошибок | 1 |
| Баулин Матвей | 08.05.2025 | Проведение регрессивного тестирования | 3 |
| Баулин Матвей | 09.05.2025 | Составление отчёта о результатах тестирования | 2 |

**4.2 Отчёт о результатах тестирования**

Статистика по всем дефектам представлена в таблице 4.2.

Таблица 4.2 – Статистика по всем дефектам

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Статус | Количество | Важность | | | |
| Низкая | Средняя | Высокая | Критическая |
| Найдено | 1 | 0 | 0 | 0 | 0 |
| Исправлено | 0 | 0 | 0 | 0 | 0 |
| Проверено |  |  |  |  |  |
| Открыто заново | 0 | 0 | 0 | 0 | 0 |
| Отклонено | 1 | 1 | 0 | 0 | 0 |

По результатам тестирования все элементы были проверены и был найден 1 дефект, который невозможно было исправить – разногласие размера окна на форме шестого вопроса. Остальные компоненты программного продукта работают как следует.

1. Применение
   1. Назначение и условия применения программы

Данный программный продукт предназначен для ознакомления с песнями времён Великой Отечественной Войны, а также для тренировки памяти по началу песен. Продукт может быть полезен для различных презентаций по теме Великой Отечественной Войны, ведь музыка тоже является частью истории

* 1. Инсталляция

Для запуска данного продукта должен быть файл с данным проектом и Visual Studio.

* 1. Выполнение программы

Начинается работа программы с главного меню.

Главное меню изображено на рисунке 5.1.

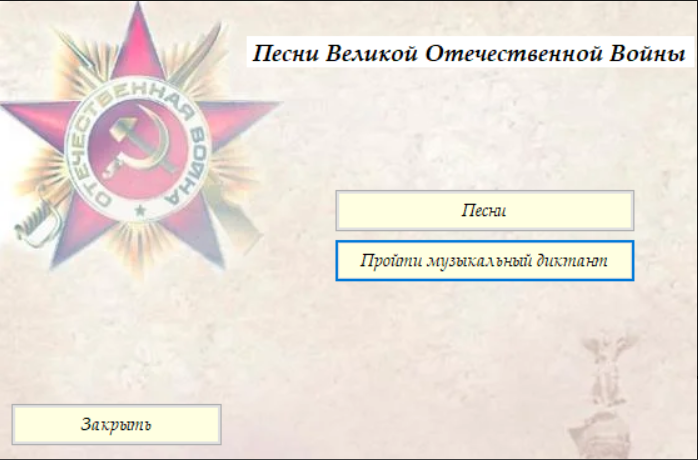


Рисунок 5.1 – Главное меню

На ней можно перейти по кнопкам на различные страницы.

При нажатии на «Песни» откроется сборник песен, где можно будет послушать каждую песню времён Великой Отечественной Войны.

Сборник песен изображён на рисунке 5.2.

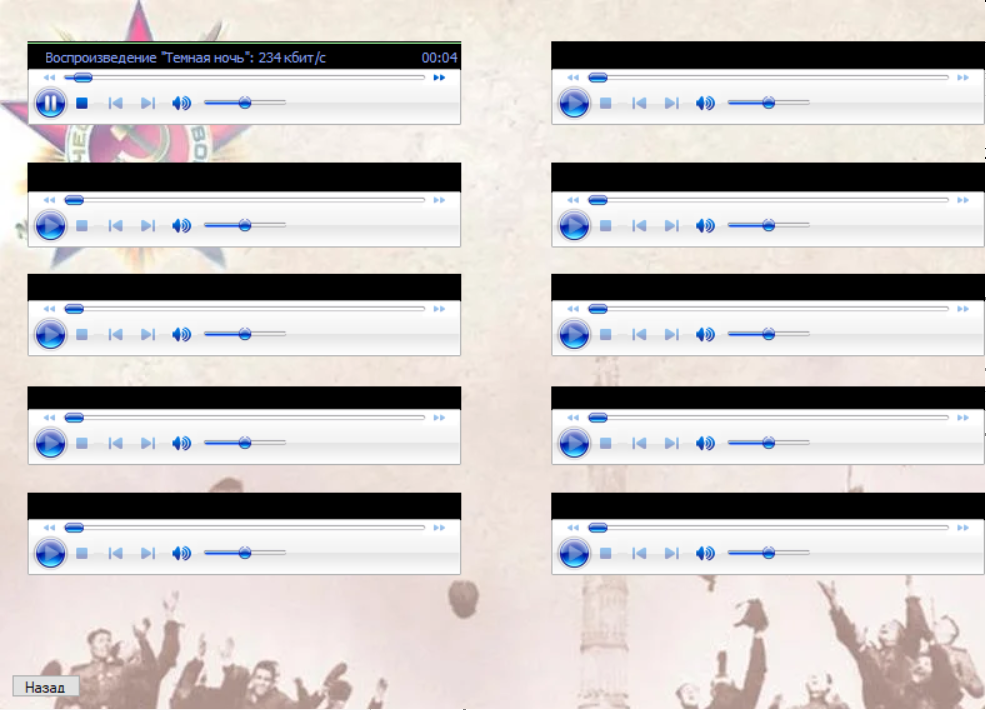


Рисунок 5.2 – Сборник песен

Также, там имеется кнопка «Назад», чтобы вернуться на главную форму.

В главном меню есть ещё кнопки «Закрыть» и «Пройти музыкальный диктант».

Если нажать на кнопку «Закрыть», то программа закроется.

Если нажать на кнопку «Пройти музыкальный диктант», то откроется окно с первым вопросом музыкального диктанта – теста для проверки знаний, где проигрывается часть музыки и за 20 секунд нужно выбрать вариант ответа.

Вопрос диктанта изображён на рисунке 5.3

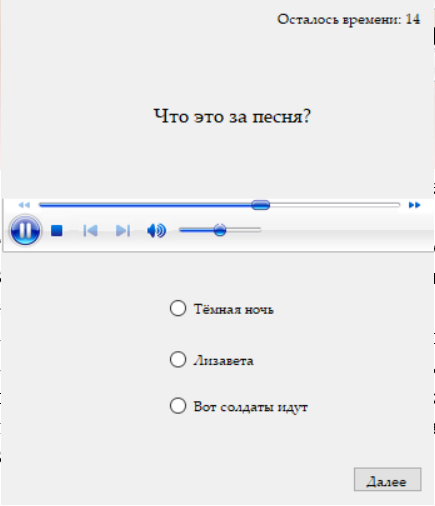


Рисунок 5.3 – Вопрос диктанта

Если ответить или вообще не отвечать на вопросы, после этого откроется главное меню и откроется окошко с результатом диктанта.

Заключение

В результате учебной практики был создан программный продукт для ознакомления с песнями времён Великой Отечественной Войны, а также для напоминания, что музыка – тоже часть истории.

Все элементы приложения работают нормально, хоть и с одним неважным дефектом.

Этим приложением можно пользоваться в любой момент, т.к. найдётся желающий узнать о музыке времён столь важного события в истории как Великой Отечественной Войны.

Список использованных источников

Лучшие песни Великой Отечественной войны / Антон Поспелов [Электронный ресурс]. – Режим доступа: <https://pravoslavie.ru/53349.html> - Дата доступа: 07.05.2025.

Приложение Б  
Тест-кейсы

Form1:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form1 : Form

{

private int time = 20;

public Form1()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Тёмная\_ночь(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

timer1.Stop();

if (radioButton1.Checked)

{

TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form2 nextForm = new Form2();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton1.Checked)

{

TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form2 nextForm = new Form2();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form2:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form2 : Form

{

private int time = 20;

public Form2()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Лизавета(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton3.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form3 nextForm = new Form3();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton3.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form3 nextForm = new Form3();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form 3:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form3 : Form

{

private int time = 20;

public Form3()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Священная\_война(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton1.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form4 nextForm = new Form4();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton1.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form3 nextForm = new Form3();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form4:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form4 : Form

{

private int time = 20;

public Form4()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Жди\_меня(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton2.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form5 nextForm = new Form5();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton2.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form5 nextForm = new Form5();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form 5:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form5 : Form

{

private int time = 20;

public Form5()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Эх\_дороги(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton1.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form6 nextForm = new Form6();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton1.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form6 nextForm = new Form6();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form 6:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form6 : Form

{

private int time = 20;

public Form6()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Огонёк(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton3.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form7 nextForm = new Form7();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton3.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form7 nextForm = new Form7();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form 7:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form7 : Form

{

private int time = 20;

public Form7()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Вечер\_на\_рейде(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton1.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form8 nextForm = new Form8();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton1.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form8 nextForm = new Form8();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form 8:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form8 : Form

{

private int time = 20;

public Form8()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Вот\_солдаты\_идут(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton2.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form9 nextForm = new Form9();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton2.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form9 nextForm = new Form9();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form 9:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form9 : Form

{

private int time = 20;

public Form9()

{ //Создаём таймер

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

//Создаём класс для хранения результата диктанта

public static class TestResult

{

public static int Score { get; set; } = 0;

}

//Метод привязки к медиаплееру файла

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Враги\_сожгли\_родную\_хату(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

//Метод для работы таймера

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton3.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form10 nextForm = new Form10();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

//Метод работы кнопки

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

{

timer1.Stop();

if (radioButton3.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form10 nextForm = new Form10();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form 10

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form10 : Form

{

private int time = 20; // Сделать переменную полем класса

public Form10()

{

InitializeComponent();

label1.Text = "Осталось времени: " + time.ToString();

timer1 = new System.Windows.Forms.Timer();

timer1.Interval = 1000;

timer1.Tick += timer1\_Tick;

timer1.Start();

}

public static class TestResult

{

public static int Score { get; set; } = 0;

}

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Казаки\_в\_Берлине(Диктант).mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

private void timer1\_Tick(object sender, EventArgs e)

{

if (time > 0)

{

time--;

label1.Text = "Осталось времени: " + time.ToString();

}

else

{

if (radioButton2.Checked)

{

Form1.TestResult.Score += 1;

}

timer1.Stop();

Form11 nextForm = new Form11();

nextForm.Show();

//Создаём оператор выбора для вывода разных результатов

switch (Form1.TestResult.Score)

{

case int i when (i >= 0 && i <= 3):

MessageBox.Show("Ваш результат: " + Convert.ToString(Form1.TestResult.Score) + "\nВам стоит попробовать запомнить песни ещё раз. У Вас всё получится!");

break;

case int i when (i > 3 && i <= 6):

MessageBox.Show("Ваш результат: " + Convert.ToString(Form1.TestResult.Score) + "\nСтоит ещё раз послушать песни для более высокого результата. У Вас всё получится!");

break;

case int i when (i > 6 && i <= 9):

MessageBox.Show("Ваш результат: " + Convert.ToString(Form1.TestResult.Score) + "\nНеплохой результат! Попробуйте послушать песни ещё раз для наивысшего результата. У Вас всё получится!");

break;

case 10:

MessageBox.Show("Ваш результат: " + Convert.ToString(Form1.TestResult.Score) + "\nВы запомнили все песни Великой Отечественной Войны! Так держать!");

break;

}

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

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

{

timer1.Stop();

if (radioButton2.Checked)

{

Form1.TestResult.Score += 1;

}

// Создаем и показываем следующую форму

Form11 nextForm = new Form11();

nextForm.Show();

switch(Form1.TestResult.Score)

{

case int i when(i >= 0 && i <= 3):

MessageBox.Show("Ваш результат: " + Convert.ToString(Form1.TestResult.Score) + "\nВам стоит попробовать запомнить песни ещё раз. У Вас всё получится!");

break;

case int i when (i > 3 && i <= 6):

MessageBox.Show("Ваш результат: " + Convert.ToString(Form1.TestResult.Score) + "\nСтоит ещё раз послушать песни для более высокого результата. У Вас всё получится!");

break;

case int i when (i > 6 && i <= 9):

MessageBox.Show("Ваш результат: " + Convert.ToString(Form1.TestResult.Score) + "\nНеплохой результат! Попробуйте послушать песни ещё раз для наивысшего результата. У Вас всё получится!");

break;

case 10:

MessageBox.Show("Ваш результат: " + Convert.ToString(Form1.TestResult.Score) + "\nВы запомнили все песни Великой Отечественной Войны! Так держать!");

break;

}

axWindowsMediaPlayer1.Ctlcontrols.stop();

this.Hide();

}

}

}

Form 11:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form11 : Form

{

public Form11()

{

InitializeComponent();

}

//Метод работы кнопки старта диктанта

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

{

Form1 nextForm = new Form1();

nextForm.Show();

this.Hide();

}

//Метод работы кнопки перехода к сборнику песен

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

{

Form12 nextForm = new Form12();

nextForm.Show();

this.Hide();

}

//Метод работы кнопки закрытия программы

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

{

Close();

}

}

}

Form 12:

using System;

using System.Windows.Forms;

namespace Курсовой\_проект

{

public partial class Form12 : Form

{

public Form12()

{

InitializeComponent();

}

//Методы работы медиаплееров

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Тёмная\_ночь.mp3";

axWindowsMediaPlayer1.URL = audioFilePath;

}

private void axWindowsMediaPlayer5\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Эх\_дороги.mp3";

axWindowsMediaPlayer5.URL = audioFilePath;

}

private void axWindowsMediaPlayer2\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Жди\_меня.mp3";

axWindowsMediaPlayer2.URL = audioFilePath;

}

private void axWindowsMediaPlayer6\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Враги\_сожгли\_родную\_хату.mp3";

axWindowsMediaPlayer6.URL = audioFilePath;

}

private void axWindowsMediaPlayer3\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Вот\_солдаты\_идут.mp3";

axWindowsMediaPlayer3.URL = audioFilePath;

}

private void axWindowsMediaPlayer7\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Вечер\_на\_рейде.mp3";

axWindowsMediaPlayer7.URL = audioFilePath;

}

private void axWindowsMediaPlayer4\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Огонёк.mp3";

axWindowsMediaPlayer4.URL = audioFilePath;

}

private void axWindowsMediaPlayer8\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Священная\_война.mp3";

axWindowsMediaPlayer8.URL = audioFilePath;

}

private void axWindowsMediaPlayer9\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Казаки\_в\_Берлине.mp3";

axWindowsMediaPlayer9.URL = audioFilePath;

}

private void axWindowsMediaPlayer10\_Enter(object sender, EventArgs e)

{

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer1.Ctlcontrols.stop();

string audioFilePath = "C:/Работы 2 Курса/C#/Курсовой проект/Аудио/Лизавета.mp3";

axWindowsMediaPlayer10.URL = audioFilePath;

}

//Метод работы кнопки возвращения в главное меню

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

{

Form11 nextForm = new Form11();

nextForm.Show();

axWindowsMediaPlayer1.Ctlcontrols.stop();

axWindowsMediaPlayer2.Ctlcontrols.stop();

axWindowsMediaPlayer3.Ctlcontrols.stop();

axWindowsMediaPlayer4.Ctlcontrols.stop();

axWindowsMediaPlayer5.Ctlcontrols.stop();

axWindowsMediaPlayer6.Ctlcontrols.stop();

axWindowsMediaPlayer7.Ctlcontrols.stop();

axWindowsMediaPlayer8.Ctlcontrols.stop();

axWindowsMediaPlayer9.Ctlcontrols.stop();

axWindowsMediaPlayer10.Ctlcontrols.stop();

this.Hide();

}

}

}