**Course Project   
DeVry University  
College of Engineering and Information Sciences**

***Screenshot of program running:***

A screenshot of a computer

Description automatically generated

***Form code (only the code for the form, not program.cs):***

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Runtime.Remoting.Messaging;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using Microsoft.Web.WebView2.Core;

namespace Rodriguez\_CourseProject\_Part1

{

public partial class MainForm : Form

{

// class level references

string[] titleArray = new string[5];

string[] artistArray = new string[5];

string[] genreArray = new string[5];

int[] yearArray = new int[5];

string[] urlArray = new string[5];

int songCount = 0;

public MainForm()

{

InitializeComponent();

}

public bool ValidInput()

{

// Return true if all fields are non-empty

bool isValid = true;

if (string.IsNullOrEmpty(titleText.Text))

{

// Title is blank

MessageBox.Show("The song title cannot be blank.");

isValid = false;

}

else if (string.IsNullOrEmpty(artistText.Text))

{

// Artist is blank

MessageBox.Show("The artist cannot be blank.");

isValid = false;

}

else if (string.IsNullOrEmpty(genreText.Text))

{

// Genre is blank

MessageBox.Show("The genre cannot be blank.");

isValid = false;

}

else if (string.IsNullOrEmpty(yearText.Text))

{

// Year is blank

MessageBox.Show("The year cannot be blank.");

isValid = false;

}

else if (string.IsNullOrEmpty(yearText.Text))

{

// URL is blank

MessageBox.Show("The url cannot be blank.");

isValid = false;

}

return isValid;

}

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

{

StringBuilder sb = new StringBuilder(outputText.Text);

string nl = "\r\n";

if (songCount == titleArray.Length)

MessageBox.Show("Song list is full!");

else if (ValidInput())

{

// Add title to ListBox and song list

songList.Items.Add(titleText.Text);

titleArray[songCount] = titleText.Text;

artistArray[songCount] = artistText.Text;

genreArray[songCount] = (genreText.Text);

yearArray[songCount] = int.Parse(yearText.Text);

urlArray[songCount] = urlText.Text;

// Increment the song counter

songCount++;

// Build the output text

sb.Append(titleText.Text);

sb.Append(nl);

sb.Append(artistText.Text);

sb.Append(nl);

sb.Append(genreText.Text);

sb.Append(nl);

sb.Append(yearText.Text);

sb.Append(nl);

sb.Append(urlText.Text);

sb.Append(nl);

sb.Append(nl);

outputText.Text = sb.ToString();

}

}

private bool SongInList(string songTitle)

{

bool found = false;

foreach (var item in songList.Items)

{

string currentSong = item as string;

// lowercase comparison so not case sensitive

if (songTitle.ToLower() == currentSong.ToLower())

{

found = true;

}

}

return found;

}

private int GetSongIndex(string songTitle)

{

int songIndex = songList.Items.IndexOf(songTitle);

return songIndex;

}

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

{

StringBuilder sb = new StringBuilder(String.Empty);

string nl = "\r\n";

// Build output text

foreach (var item in songList.Items)

{

sb.Append(item.ToString());

sb.Append(nl);

}

// Put the output text into the output textbox

outputText.Text = sb.ToString();

}

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

{

if( SongInList(titleText.Text) )

{

StringBuilder sb = new StringBuilder(String.Empty);

string nl = "\r\n";

int songIndex = GetSongIndex(titleText.Text);

// Build the output text

sb.Append(titleArray[songIndex]);

sb.Append(titleArray[songIndex]);

sb.Append(nl);

sb.Append(artistArray[songIndex]);

sb.Append(nl);

sb.Append(genreArray[songIndex]);

sb.Append(nl);

sb.Append(yearArray[songIndex]);

sb.Append(nl);

sb.Append(urlArray[songIndex]);

sb.Append(nl);

outputText.Text = sb.ToString();

}

}

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

{

titleText.Text = ""; // one to clear textbox

artistText.Text = String.Empty; // second way to clear

genreText.Clear();

urlText.Clear();

}

private void songList\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void webView21\_Click(object sender, EventArgs e)

{

}

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

{

int songIndex = songList.SelectedIndex;

string url = urlArray[songIndex];

webViewDisplay.CoreWebView2.Navigate(url);

}

}

}