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;

using System.IO;

namespace Name\_Search

{

public partial class nameSearchForm : Form

{

public nameSearchForm()

{

InitializeComponent();

}

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

{

this.Close();

}

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

{

userInputTextBox.Text = "";

maleRadioButton.Checked = false;

femaleRadioButton.Checked = false;

allRadioButton.Checked = true;

wasFoundLabel.Visible = false;

notFoundLabel.Visible = false;

userInputTextBox.Focus();

}

//create the Array of male names

private bool SearchMaleNameArray(string name)

{

const int ARRAY\_SIZE = 200;

string[] maleNameArray = new string[ARRAY\_SIZE];

int index=0;

try

{

//create an array from the file by creating a streamreader variable

StreamReader maleInputFile = File.OpenText("BoyNames.txt");

//read the names into the boyNamesList

while (index<maleNameArray.Length && !maleInputFile.EndOfStream )

{

maleNameArray[index] = maleInputFile.ReadLine();

index++;

}

//close the file

maleInputFile.Close();

//return the value from search names as a bool

return SearchNames(name,maleNameArray);

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

return false;

}

}

//create the Array of female names

private bool SearchFemaleNameArray(string name)

{

const int ARRAY\_SIZE = 200;

string[] femaleNameArray = new string[ARRAY\_SIZE];

int index = 0;

try

{

//create an array from the file by creating a streamreader variable

StreamReader femaleInputFile = File.OpenText("GirlNames.txt");

//read the names into the boyNamesList

while (index < femaleNameArray.Length && !femaleInputFile.EndOfStream)

{

femaleNameArray[index] = femaleInputFile.ReadLine();

index++;

}

//close the file

femaleInputFile.Close();

//return the value from search names as a bool

return SearchNames(name, femaleNameArray);

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

return false;

}

}

//perform the sequential search for the array

private bool SearchNames(string name, string[] nameArray)

{

bool found = false; //found the name was in the list

int index = 0; //start the index at 0

//ensure that it wasn't already found AND within bounds of the array

while(!found && index< nameArray.Length )

{

// compare each value against the searched name

if(nameArray[index]== name)

{

found = true;

}

// increment by one

index++;

}

return found;

}

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

{

//had persistence issues so Manually clearing out the field on search

wasFoundLabel.Visible = false;

notFoundLabel.Visible = false;

//read userinput as a new variable

string nameToSearch = userInputTextBox.Text;

try

{

//make sure it isn't blank because people suck

if (nameToSearch != "")

{

//if male radio button is checked then search the male list

if (maleRadioButton.Checked == true)

{

if (SearchMaleNameArray(nameToSearch))

{

//let the user know that it was found

wasFoundLabel.Visible = true;

}

else

{

//if it wasn't found then show not found

notFoundLabel.Visible = true;

}

}

//if female radio button is checked then search the female list

else if (femaleRadioButton.Checked == true)

{

if (SearchFemaleNameArray(nameToSearch))

{

//let the user know that it was found

wasFoundLabel.Visible = true;

}

else

{

//if it wasn't found then show not found

notFoundLabel.Visible = true;

}

}

//when the user doesn't change the radio button perform the search

//we will check both

else if (allRadioButton.Checked == true)

{

//call one search first

if (SearchMaleNameArray(nameToSearch))

{

wasFoundLabel.Visible = true;

}

//if not found above continue to the next search

else if (SearchFemaleNameArray(nameToSearch))

{

wasFoundLabel.Visible = true;

}

//if it wasn't found in the first two then show the not visible

else

{

notFoundLabel.Visible = true;

}

}

}

else

{

MessageBox.Show("Please enter a name or else this wont work");

}

}

catch (Exception ex)

{

//just in case

MessageBox.Show(ex.Message);

}

}

}

}

