

using System;

using System.Windows.Forms;

using System.IO;

namespace JBStateCapSearchWinForm

{

public partial class StateCapitalForm : Form

{

// put here so it's scoped for the whole class. load and save can access

string filepath = @"C:\Users\benge\source\repos\JBStateCapSearch\JBStateCapSearchWinForm\bin\Debug\StateCapitals.txt";

public StateCapitalForm()

{

InitializeComponent();

}

/// <summary>

/// Runs on load of the application before the user sees anything

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void StateCapitalForm\_Load(object sender, EventArgs e)

{

StreamReader inputFile;

try

{

// inputFile = File.OpenText(@"C:\Users\benge\source\repos\JBStateCapSearch\JBStateCapSearchWinForm\bin\Debug\StateCapitals.txt");

// the filepath above is the default lookup location so I can just use the cmd below

// inputFile = File.OpenText("StateCapitals.txt");

inputFile = File.OpenText(filepath);

// reads every item out of the file

while (!inputFile.EndOfStream)

{

// adds the item to LstBoxStateCaps

LstBoxStateCaps.Items.Add(inputFile.ReadLine());

}

// closes the file

inputFile.Close();

}

catch (Exception ex)

{

// would show on read error

MessageBox.Show(ex.Message);

}

}

/// <summary>

/// Saves the user input to the .txt file

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

{

StreamWriter outputFile;

try

{

// obtain user input and format for save

string statecap = $"{TxtBoxState.Text}, {TxtBoxCapital.Text}";

// open

outputFile = File.AppendText(filepath);

// use

outputFile.WriteLine(statecap);

// close

outputFile.Close();

// clear results and reset cursor focus to first box

TxtBoxCapital.Clear();

TxtBoxState.Clear();

TxtBoxState.Focus();

// update lst box

LstBoxStateCaps.Items.Add(statecap);

// notify user of successful save

MessageBox.Show($"{statecap} was sucessfully saved");

}

catch (Exception ex)

{

// would show on write error

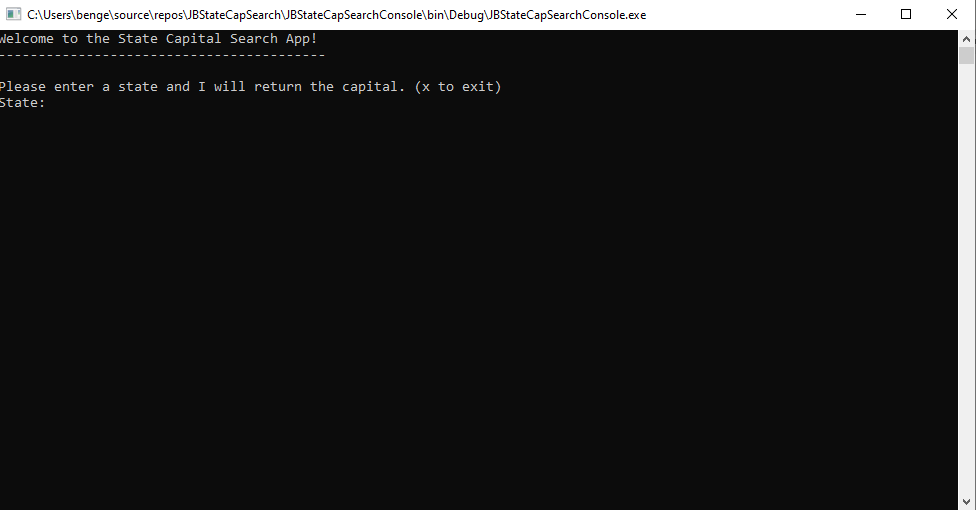
MessageBox.Show(ex.Message);

}

}

}

}



using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace JBStateCapSearchConsole

{

class Program

{

static void Main(string[] args)

{

//Console.WriteLine("Hello World");

//Console.Write("My Name is: ");

//Console.WriteLine("Jacob Benge");

// call the constructor method to open the file. see StateCapitals.cs

StateCapitals stateCapitals = new StateCapitals();

Console.WriteLine("Welcome to the State Capital Search App!");

Console.WriteLine("-----------------------------------------");

Console.WriteLine();

Console.WriteLine("Please enter a state and I will return the capital. (x to exit)");

Console.Write("State: ");

// waits for user input

string userInput = Console.ReadLine();

// Need an infinite loop to keep the console open. Otherwise it closes automatically once the code runs

while (userInput != "x")

{

// passes the userInput to SearchDict and returns the capital

string capital = stateCapitals.SearchDict(userInput.ToUpper());

// if a valid value is returned then return the string below

if (capital != null)

{

Console.WriteLine($"The capital of {userInput} is {capital}");

}

else

{

Console.WriteLine($"No results were found for {userInput}");

}

Console.WriteLine();

Console.WriteLine("Please enter a state and I will return the capital. (x to exit)");

Console.Write("State: ");

// waits for user input

userInput = Console.ReadLine();

}

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace JBStateCapSearchConsole

{

class StateCapitals

{

// holds the filepath for reference

string filepath = @"C:\Users\benge\source\repos\JBStateCapSearch\JBStateCapSearchWinForm\bin\Debug\StateCapitals.txt";

// calls the constructor method for a dictionary with a key value pair

Dictionary<string, string> stateCapDict = new Dictionary<string, string>();

/// <summary>

/// this is a constructor method because it has the same name as the class

/// </summary>

public StateCapitals()

{

// shows we are reading the file. Incoming, not outgoing

StreamReader inputFile;

// try catch is used in case of read errors due to permissions

try

{

// open

inputFile = File.OpenText(filepath);

// read

while (!inputFile.EndOfStream)

{

// reads the .txt file and saves the key and values into the stcp array

string[] stcp = inputFile.ReadLine().Split(',');

// takes the first item in the array and converts to uppercase

string state = stcp[0].ToUpper();

// saves the value in the capital string

string capital = stcp[1];

// adds the key value pair into the dictionary

stateCapDict.Add(state, capital);

}

// close

inputFile.Close();

}

catch (Exception ex)

{

// displays any error message

Console.WriteLine(ex.Message);

}

}

/// <summary>

/// Used to search the dictionary for the key (userInput) and returns the value

/// </summary>

/// <param name="userInput"></param>

/// <returns></returns>

public string SearchDict(string userInput)

{

// instantiates the string capital

string capital = null;

// TryGetValue() takes the key and returns the corrosponding value

if (stateCapDict.TryGetValue(userInput, out capital))

{

return capital;

}

return capital;

}

}

}