using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace OOPhw

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

private class Person

{

public int Id { get; set; }

public string Name { get; set; }

public int Age { get; set; }

}

private Person[] peopleArray = new Person[100];

private int currentIndex = 0;

public MainWindow()

{

InitializeComponent();

}

private void AddPersonButton\_Click(object sender, RoutedEventArgs e)

{

if (currentIndex < peopleArray.Length)

{

var person = new Person

{

Id = int.Parse(IdTextBox.Text),

Name = NameTextBox.Text,

Age = int.Parse(AgeTextBox.Text)

};

peopleArray[currentIndex++] = person;

ClearInputFields();

}

else

{

MessageBox.Show("Data storage is full.");

}

}

private void DisplayButton\_Click(object sender, RoutedEventArgs e)

{

DisplayData(peopleArray.Take(currentIndex).ToArray());

}

private void SortByAgeButton\_Click(object sender, RoutedEventArgs e)

{

DisplayData(peopleArray.Take(currentIndex).OrderBy(p => p.Age).ToArray());

}

private void SortByNameButton\_Click(object sender, RoutedEventArgs e)

{

DisplayData(peopleArray.Take(currentIndex).OrderBy(p => p.Name).ToArray());

}

private void SearchByAgeButton\_Click(object sender, RoutedEventArgs e)

{

int targetAge = int.Parse(SearchAgeTextBox.Text);

var result = peopleArray.Take(currentIndex).Where(p => p.Age == targetAge).ToArray();

DisplayData(result);

}

private void SearchByNameButton\_Click(object sender, RoutedEventArgs e)

{

string targetName = SearchNameTextBox.Text;

var result = peopleArray.Take(currentIndex).Where(p => p.Name.Equals(targetName, StringComparison.OrdinalIgnoreCase)).ToArray();

DisplayData(result);

}

private void RemoveByIdButton\_Click(object sender, RoutedEventArgs e)

{

int targetId = int.Parse(RemoveIdTextBox.Text);

for (int i =0; i< currentIndex; i++)

{

if(peopleArray[i] != null && peopleArray[i].Id == targetId) {

for (int j=i; j<currentIndex - 1; j++)

{

peopleArray[j] = peopleArray[j + 1];

}

peopleArray[currentIndex - 1] = null;

currentIndex--;

break;

}

}

DisplayData(peopleArray.Take(currentIndex).ToArray());

}

private void ClearInputFields()

{

IdTextBox.Clear();

NameTextBox.Clear();

AgeTextBox.Clear();

}

private void DisplayData(Person[] data)

{

OutputListBox.Items.Clear();

foreach (var person in data)

{

OutputListBox.Items.Add($"{person.Id} - {person.Name} - {person.Age} years old");

}

}

}

}