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;

namespace CarDealer

{

public partial class Form1 : Form

{

private CarManagment AutoCar = new CarManagment();

public Form1()

{

InitializeComponent();

UpdateCarList();

AddValueToComboBox();

AddValueToSearchPriceComboBox();

AddComboBoxSearchBay();

}

private void ListCar\_AddComboBox(object sender, EventArgs e)

{

AddValueToComboBox();

}

public void AddValueToComboBox()

{

// add value to combobox

for (int bay = 1; bay <= 20; bay++)

{

comboBoxBay.Items.Add(bay);

}

}

public void UpdateCarList()

{

ListCar.Items.Clear();

foreach (Car car in AutoCar.GetCars())

{

if (car != null)

{

ListCar.Items.Add(car);

}

}

}

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

{

//add to array

int bay = int.Parse(comboBoxBay.SelectedItem.ToString());

string regnumber = textRegNumber.Text;

string make = textMake.Text;

string model = textModel.Text;

int year = int.Parse(textYear.Text);

int price = int.Parse(textPrice.Text);

string filename = textPhoto.Text;

Car newCar = new Car(bay, regnumber, make, model, year, price, filename);

AutoCar.AddCar(newCar);

UpdateCarList();

}

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

{

// create parthway to file

OpenFileDialog photo = new OpenFileDialog();

photo.Filter = "Text files (All files (\*.\*)|\*.\*";

if (photo.ShowDialog() == DialogResult.OK)

{

textPhoto.Text = photo.FileName;

}

}

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

{

if (ListCar.SelectedItem != null)

{

//remove from array

AutoCar.RemoveCar((Car)ListCar.SelectedItem);

//add to sold car display list

ListSoldCar.Items.Add(ListCar.SelectedItem);

//remove from display listbox

ListCar.Items.Remove(ListCar.SelectedItem);

}

}

private void ListCar\_AddComboBoxSearchPrice(object sender, EventArgs e)

{

AddValueToSearchPriceComboBox();

}

public void AddValueToSearchPriceComboBox()

{

// create data for combobox to search car in price range

SearchPriceComboBox.Items.Add("0 - 10000");

SearchPriceComboBox.Items.Add("10000 - 20000");

SearchPriceComboBox.Items.Add("20000 - 30000");

SearchPriceComboBox.Items.Add("30000 - 40000");

SearchPriceComboBox.Items.Add("40000 - 50000");

}

private void ListCar\_AddComboBoxSearchBay(object sender, EventArgs e)

{

AddComboBoxSearchBay();

}

public void AddComboBoxSearchBay()

{

// add bays to combobox

for (int bay = 1; bay <= 20; bay++)

{

SearchBayComboBox.Items.Add(bay);

}

}

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

{

// convert string in combobox into int

string value = SearchPriceComboBox.Text.ToString();

string valueBay = SearchBayComboBox.Text.ToString();

//ListCar.Items.Clear();

if (value != "") {

ListCar.Items.Clear();

string[] splitValueprice = value.Split(' ');

int min = int.Parse(splitValueprice[0]);

int max = int.Parse(splitValueprice[2]);

// sequential search price in array

foreach (Car car in AutoCar.GetCars())

{

if (car != null)

{

if (car.GetPrice() >= min && car.GetPrice() <= max)

{

ListCar.Items.Add(car);

Console.WriteLine($" car {car}");

}

}

}

}

// random access to the position using bay

if (valueBay != "")

{

ListCar.Items.Clear();

Console.WriteLine($"bay {valueBay}");

Car car = AutoCar.GetCar(int.Parse(valueBay));

ListCar.Items.Add(car);

}

}

}

}