using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lesson\_03

{

class Program

{

static void Main(string[] args)

{

List<Car> carList = new List<Car>();

string choice = "";

do

{

Console.WriteLine("WHAT WOULD YOU LIKE TO DO?");

Console.WriteLine("[A] Add a Car [B] View All Cars [C] View Number of Cars [D] Delete a car");

choice = (Console.ReadLine()).ToUpper();

switch (choice)

{

case "A":

Car.ADDCAR(carList);

break;

case "B":

Car.VIEWALL(carList);

break;

case "C":

Car.COUNTALL();

break;

case "D":

Car.DELETE(carList);

break;

}

Console.WriteLine("Would you like to exit? Y/N");

if ((Console.ReadLine()).ToUpper() == "Y")

{

Console.WriteLine("Shutting down..");

System.Threading.Thread.Sleep(1000);

Environment.Exit(0);

}

else { Console.WriteLine("Go back to main menu? Y/N"); }

} while ((Console.ReadLine()).ToUpper() == "Y");

}

}

}

class Car

{

public string make;

public string model;

public float price;

public static int numCars;

public Car(string make, string model, float price)

{

this.make = make;

this.model = model;

this.price = price;

numCars++;

}

public static void TotalNumcar()

{

Console.WriteLine(numCars);

}

public static void DisplayCars(List<Car> carList)

{

for (int f =0; f<numCars; f++)

{

Console.WriteLine(" Make : {0} Model : {1} Price : £{2:N0}", carList[f].make, carList[f].model, carList[f].price);

}

}

public static void ADDCAR(List<Car> carList)

{

string again = "";

string makeChoice = "";

string modelChoice = "";

float priceChoice = 0;

do

{

Console.WriteLine("Enter your Make");

makeChoice = Console.ReadLine();

Console.WriteLine("Enter your Model");

modelChoice = Console.ReadLine();

try

{

Console.WriteLine("Enter your Price");

priceChoice = float.Parse(Console.ReadLine());

}

catch (FormatException e)

{

Console.WriteLine(e.Message);

}

Car obj = new Car(makeChoice, modelChoice, priceChoice);

Console.WriteLine(" Make : {0} | Model : {1} | Price : £{2:N0}", obj.make, obj.model, obj.price);

carList.Add(obj);

Console.WriteLine();

Console.WriteLine("Would you like to add another car ? Y/N");

again = (Console.ReadLine()).ToUpper();

} while (again == "Y");

}

public static void VIEWALL(List<Car> carList)

{

Console.WriteLine("Would you like to see the cars Y/N");

if ((Console.ReadLine()).ToUpper() == "Y")

{

Car.DisplayCars(carList);

}

else { Console.WriteLine("ERROR"); }

}

public static void COUNTALL()

{

Console.WriteLine("Would you like to see the total number of cars Y/N");

if ((Console.ReadLine()).ToUpper() == "Y")

{

Car.TotalNumcar();

}

else { Console.WriteLine("ERROR"); }

}

public static void DELETE(List<Car> carList)

{

string make = "";

Console.WriteLine("Which car would you like to rmeove ENTER THE MAKE:");

make = Console.ReadLine();

for (int i = 0; i < carList.Count; i++)

{

if (make == carList[i].make)

{

carList.Remove(carList[i]);

Car.numCars--;

}

else

{

Console.WriteLine("Wrong choice, please enter a make to delete");

}

}

}

}