using System;

using System.Collections;

using System.Collections.Generic;

using System.Diagnostics;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

namespace ConsoleApp2

{

#region Generic Collections

//public class Program

//{

// static void Main(string[] args)

// {

// //SortedList<string, string> mylist = new SortedList<string, string>();

// //mylist.Add("nar", "pomegranate");

// //mylist.Add("alma", "apple");

// //mylist.Add("ananas", "pineapple");

// //mylist.Add("banan", "banana");

// //mylist["nar"] = "Salam";

// //Console.WriteLine(mylist.ContainsKey("nar"));

// //Console.WriteLine(mylist.ContainsValue("apple"));

// //var key = Console.ReadLine();

// //bool hasValue=mylist.TryGetValue(key,out string value);

// //if (hasValue)

// //{

// // Console.WriteLine(value);

// //}

// //else

// //{

// // Console.WriteLine("We did not find key like this : "+key);

// //}

// //foreach (var item in mylist)

// //{

// // Console.WriteLine(item.Key + " " + item.Value);

// //}

// //Queue<string> vs = new Queue<string>();

// //Stack<int> vs1 = new Stack<int>();

// //List<int> vs2 =new List<int>();

// //Dictionary<string, double> mydictionary = new Dictionary<string, double>();

// //mydictionary["C#"] = 8.5;

// //mydictionary["C++"] = 11.2;

// //mydictionary["JS"] = 7.6;

// //foreach (var item in mydictionary)

// //{

// // Console.WriteLine(item.Key+" "+item.Value);

// //}

// //Dictionary<string, List<string>> ypx = new Dictionary<string, List<string>>();

// //ypx["99-CE-099"] = new List<string>

// //{

// // "Dayanma-Durma",

// // "Suret Heddi 90-100"

// //};

// //ypx.Add("10-LL-010", new List<string>

// //{

// // "Eks Istiqamet",

// // "Suret Heddi 120-130"

// //});

// //ypx.Add("11-TT-111", new List<string>

// //{

// // "Kemer",

// // "Qosha Xett"

// //});

// //Console.WriteLine("Enter car no : ");

// //var car\_no = Console.ReadLine();

// //var hasCar = ypx.TryGetValue(car\_no, out var penalties);

// //if (hasCar)

// //{

// // foreach (var item in penalties)

// // {

// // Console.WriteLine(item);

// // }

// //}

// //else

// //{

// // Console.WriteLine($"This car {car\_no} has no any penalties");

// //}

// //Dictionary<string, ArrayList> ypx = new Dictionary<string, ArrayList>();

// //ypx["99-CE-099"] = new ArrayList

// //{

// // "Dayanma-Durma",

// // 10,

// // "Suret Heddi 90-100",

// // 130

// //};

// //ypx["10-CE-099"] = new ArrayList

// //{

// // "Dayanma-Durma",

// // 10,

// // "Suret Heddi 90-100",

// // 130

// //};

// //ypx["11-CE-011"] = new ArrayList

// //{

// // "Dayanma-Durma",

// // 10,

// // "Suret Heddi 90-100",

// // 130

// //};

// //Console.WriteLine("Enter car no : ");

// //var car\_no = Console.ReadLine();

// //var hasCar = ypx.TryGetValue(car\_no, out var penalties);

// //if (hasCar)

// //{

// // int index = 0;

// // foreach (var item in penalties)

// // {

// // if(index%2!=0)

// // Console.WriteLine(item+" AZN");

// // else Console.Write(item+" ");

// // ++index;

// // }

// //}

// //else

// //{

// // Console.WriteLine($"This car {car\_no} has no any penalties");

// //}

// }

//}

#endregion

#region Extention Method

//class Human

//{

// public void Run()

// {

// }

// public void Eat()

// {

// }

// public void Speak()

// {

// }

//}

//class Human2 : Human

//{

//}

//class Human3 : Human2

//{

//}

//static class Extentions

//{

// public static int CountOfWord(this string text)

// {

// if (string.IsNullOrEmpty(text))

// {

// return 0;

// }

// return text.Split(' ').Length;

// }

// public static void AddDoubleItem(this List<string> items,string item1,string item2)

// {

// items.Add(item1);

// items.Add(item2);

// }

//}

//class Program

//{

// static void Main(string[] args)

// {

// List<string> list = new List<string>();

// list.AddDoubleItem("Salam","Necesen");

// foreach (var item in list)

// {

// Console.WriteLine(item);

// }

// // string text = "Salam millet necesiz";

// // Console.WriteLine(text.CountOfWord());

// }

//}

#endregion

#region Generic

//class Point<T> where T : class

//{

// public Point(T first, T second)

// {

// First = first;

// Second = second;

// }

// public Point()

// {

// First = default(T);

// Second = default(T);

// }

// public T First { get; set; }

// public T Second { get; set; }

// public void Show()

// {

// Console.WriteLine(First);

// Console.WriteLine(Second);

// }

//}

//class Point<T> where T : struct

//{

// public Point(T first, T second)

// {

// First = first;

// Second = second;

// }

// public Point()

// {

// First = default(T);

// Second = default(T);

// }

// public T First { get; set; }

// public T Second { get; set; }

// public void Show()

// {

// Console.WriteLine(First);

// Console.WriteLine(Second);

// }

//}

//class Program

//{

// static void Main(string[] args)

// {

// Point<DateTime> point = new Point<DateTime>();

// point.Show();

// }

//}

interface IReader

{

int Data { get; set; }

}

static class Student

{

}

class Point<T> where T : new()//IReader,IWriter

{

public Point(T first, T second)

{

First = first;

Second = second;

}

public Point()

{

First = default(T);

Second = default(T);

}

public T First { get; set; }

public T Second { get; set; }

public void Show()

{

Console.WriteLine(First);

Console.WriteLine(Second);

}

}

class Book : IReader,IWriter

{

public int Data { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

}

class Program

{

static void Main(string[] args)

{

Point<int> point = new Point<int>();

point.Show();

}

}

//class Test<T1, T2>

//{

// public Test(T1 data1, T2 data2)

// {

// Data1 = data1;

// Data2 = data2;

// }

// public T1 Data1 { get; set; }

// public T2 Data2 { get; set; }

// public void Show()

// {

// Console.WriteLine(Data1);

// Console.WriteLine(Data2);

// }

//}

//class Program

//{

// static void Main(string[] args)

// {

// Test<string, int> obj = new Test<string, int>("Apple", 100);

// obj.Show();

// }

//}

#endregion

}