**Лабораторная работа № 3**: Работа с коллекциями в C#

Выполнил студент группы ИСП-224 Шевцов Иван

**Цель**:

Научиться работать с различными типами коллекций в C# и понимать их преимущества и недостатки.

**Задание**:

**Разработать приложение, которое будет иметь следующие функции:**

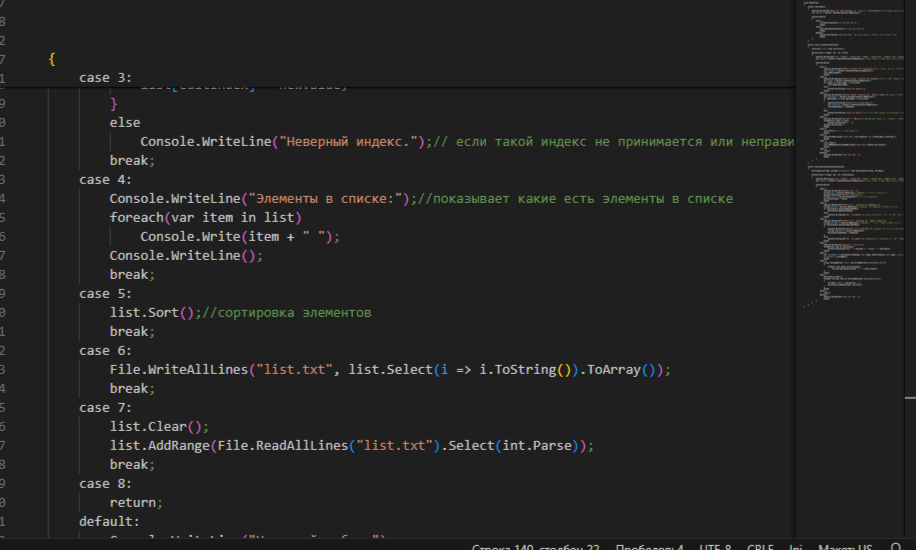
Позволяет создавать коллекции List или Dictionary.

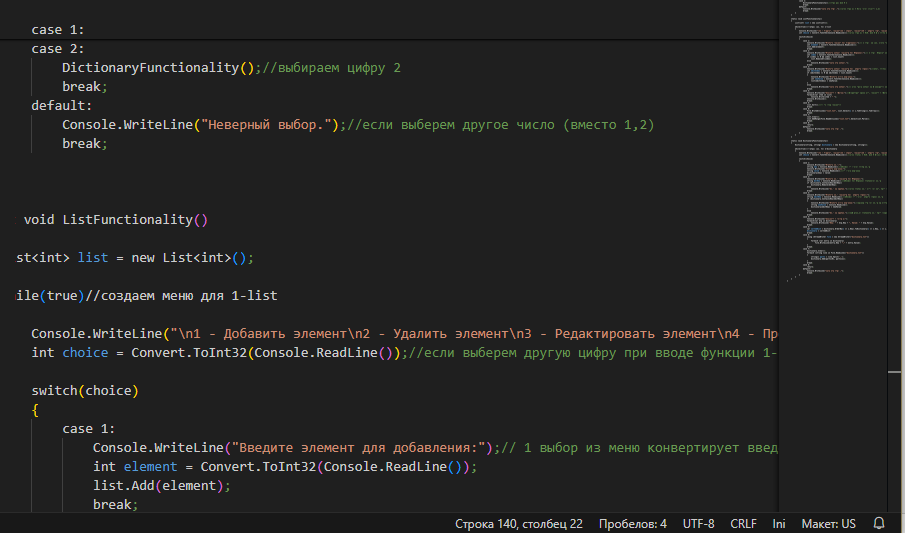
Позволять пользователю добавлять, удалять, редактировать и просматривать элементы в коллекциях List или Dictionary.

Возможность выбора типа коллекции в приложении.

Реализовать не менее трех различных алгоритмов сортировки для коллекций List или Dictionary.

Реализовать возможность сохранения и загрузки коллекций в файл.



****

**Программа:**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

class MainClass

{

static void Main()

{

Console.WriteLine("Выберите тип коллекции (1 - List, 2 - Dictionary):");//создаем главное меню, которое будет высвечиваться в самом начале программы

int choice = Convert.ToInt32(Console.ReadLine());

switch(choice)

{

case 1:

ListFunctionality();//выбираем цифру 1

break;

case 2:

DictionaryFunctionality();//выбираем цифру 2

break;

default:

Console.WriteLine("Неверный выбор.");//если выберем другое число (вместо 1,2)

break;

}

}

static void ListFunctionality()

{

List<int> list = new List<int>();

while(true)//создаем меню для 1-list

{

Console.WriteLine("\n1 - Добавить элемент\n2 - Удалить элемент\n3 - Редактировать элемент\n4 - Просмотреть элементы\n5 - Сортировка\n6 - Сохранить в файл\n7 - Загрузить из файла\n8 - Выход");

int choice = Convert.ToInt32(Console.ReadLine());//если выберем другую цифру при вводе функции 1-list (не 1-8) то меню повториться заново

switch(choice)

{

case 1:

Console.WriteLine("Введите элемент для добавления:");// 1 выбор из меню конвертирует введенный текст или файл и показыает его свойства

int element = Convert.ToInt32(Console.ReadLine());

list.Add(element);

break;

case 2:

Console.WriteLine("Введите индекс элемента для удаления:");// 2 выбор удаляет индекс, если ввести его неправильно то выдаст что индекс неправильный

int index = Convert.ToInt32(Console.ReadLine());

if (index >= 0 && index < list.Count)

list.RemoveAt(index);

else

Console.WriteLine("Неверный индекс.");

break;

case 3:

Console.WriteLine("Введите индекс элемента для редактирования:");//здесь вводим правильный индекс и редактируем или записываем новый

int editIndex = Convert.ToInt32(Console.ReadLine());

if (editIndex >= 0 && editIndex < list.Count)

{

Console.WriteLine("Введите новое значение:");

int newValue = Convert.ToInt32(Console.ReadLine());

list[editIndex] = newValue;

}

else

Console.WriteLine("Неверный индекс.");// если такой индекс не принимается или неправильный то выдаст ошибку

break;

case 4:

Console.WriteLine("Элементы в списке:");//показывает какие есть элементы в списке

foreach(var item in list)

Console.Write(item + " ");

Console.WriteLine();

break;

case 5:

list.Sort();//сортировка элементов

break;

case 6:

File.WriteAllLines("list.txt", list.Select(i => i.ToString()).ToArray());

break;

case 7:

list.Clear();

list.AddRange(File.ReadAllLines("list.txt").Select(int.Parse));

break;

case 8:

return;

default:

Console.WriteLine("Неверный выбор.");

break;

}

}

}

static void DictionaryFunctionality()//создание dictionary

{

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

while(true)//создаем меню для 2-dictionary

{

Console.WriteLine("\n1 - Добавить элемент\n2 - Удалить элемент\n3 - Редактировать элемент\n4 - Просмотреть элементы\n5 - Сортировка\n6 - Сохранить в файл\n7 - Загрузить из файла\n8 - Выход");

int choice = Convert.ToInt32(Console.ReadLine());//если введем другую цифру помимо (1-8) то меню повториться заново

switch(choice)

{

case 1:

Console.WriteLine("Введите ключ:");

string key = Console.ReadLine();//функция строчного ввода ключа

Console.WriteLine("Введите значение:");

string value = Console.ReadLine();//строчное значение

dictionary[key] = value;

break;

case 2:

Console.WriteLine("Введите ключ элемента для удаления:");

string delKey = Console.ReadLine();//функция для удаления введенного ключа

if (dictionary.ContainsKey(delKey))

dictionary.Remove(delKey);

else

Console.WriteLine("Ключ не найден.");//если введем ключ которого нет, даст ошибку

break;

case 3:

Console.WriteLine("Введите ключ элемента для редактирования:");

string editKey = Console.ReadLine();//функция строчного редактирования ключа

if (dictionary.ContainsKey(editKey))

{

Console.WriteLine("Введите новое значение:");//замена старого ключа на новый

string newValue = Console.ReadLine();

dictionary[editKey] = newValue;

}

else

Console.WriteLine("Ключ не найден.");//неправильно введенный ключ даст ошибку

break;

case 4:

Console.WriteLine("Элементы в словаре:");

foreach(var kvp in dictionary)

Console.WriteLine("Key: " + kvp.Key + ", Value: " + kvp.Value);

break;

case 5:

var sortedDict = dictionary.OrderBy(x => x.Key).ToDictionary(x => x.Key, x => x.Value);

dictionary = sortedDict;

break;

case 6:

using (StreamWriter file = new StreamWriter("dictionary.txt"))

{

foreach (var entry in dictionary)

file.WriteLine(entry.Key + "," + entry.Value);

}

break;

case 7:

dictionary.Clear();

foreach (string line in File.ReadLines("dictionary.txt"))

{

string[] parts = line.Split(',');

dictionary.Add(parts[0], parts[1]);

}

break;

case 8:

return;

default:

Console.WriteLine("Неверный выбор.");

break;

}

}

}

}

