**Министерство высшего образования и науки Российской Федерации**

Федеральное государственное бюджетное образовательное учреждение Высшего образования

«**Владимирский государственный университет**

**имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(ВлГУ)**

Колледж информационных технологий и предпринимательства (КИТП)

**Кафедра физики и прикладной математики (ФиПМ)**

Задание 13

Тема: «**Перегрузка методов и операций**»

Выполнил студент группы

Пксп-120

Родионов Ю. Р.

Принял Кабанова М. Ю.

Владимир 2022

**Постановка задачи:**

Требуется реализовать консольное приложение и приложение с графическим интерфейсом на языке программирования C#, выполнив следующие задания:

1) Создать абстрактный класс Телефонный\_справочник с методами, позволяющими вывести на экран информацию о записях в телефонном справочнике, а также определить соответствие записи критерию поиска.

2) Создать производные классы: Персона (фамилия, адрес, номер телефона), Организация (название, адрес, телефон, факс, контактное лицо), Друг (фамилия, адрес, номер телефона, дата рождения) со своими методами вывода информации на экран, и определения соответствия искомому типу.

3) Создать базу (массив) из n товаров, вывести полную информацию из базы на экран, а также организовать поиск в базе по фамилии.

**Практическая часть:**

**Листинг консольного приложения**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp\_13

{

internal class Program

{

private static void Main(string[] args)

{

Spravka[] spravkaDataBase = new Spravka[]

{

new Person("Попов", "г. Омск, 3-я ул. Белоконской, д. 4, кв. 70 ", "8 (900) 171-32-43"),

new Person("Иванов", "г. Кострома, ул. Гагарина, д. 12, кв. 80 ", "89301628241"),

new Organization("Кафе «Лосось и кофе»", "г. Владимир, ул. Большая Московская, д. 2", "+7 (4922) 45-17-05", 823052, "Денис Туркин"),

new Organization("Кинотеатр «Киномакс Буревестник»", "г. Владимир, Проспект Ленина, д. 29", "+7 (4922) 40-00-04", 7527492, "Андрей Семенов"),

new Friend("Заплаткин", "г. Владимир, ул. Горького, д.77, кв.22 ", "8 (915) 926-59-89", new DateTime(2004, 9, 9)),

new Friend("Рябов", "г. Владимир, ул. Заречная, д.2, кв.34 ", "8 (906) 325-71-43", new DateTime(2004, 5, 1))

};

Console.WriteLine("Записив в телефонном справочнике:\n");

Console.WriteLine("ПЕРСОНЫ:\n");

foreach (var s in FindSurname1(spravkaDataBase, ""))

s.PrintInfo();

Console.WriteLine("ОРГАНИЗАЦИИ:\n");

foreach (var s in FindSurname3(spravkaDataBase, ""))

s.PrintInfo();

Console.WriteLine("ДРУЗЬЯ:\n");

foreach (var s in FindSurname2(spravkaDataBase, ""))

s.PrintInfo();

Console.Write("ПОИСК (введите фамилию): ");

string familia = Console.ReadLine();

Console.WriteLine("\nПЕРСОНЫ:\n");

foreach (var s in FindSurname1(spravkaDataBase, familia))

s.PrintInfo();

Console.WriteLine("ОРГАНИЗАЦИИ:\n");

foreach (var s in FindSurname3(spravkaDataBase, familia))

s.PrintInfo();

Console.WriteLine("ДРУЗЬЯ:\n");

foreach (var s in FindSurname2(spravkaDataBase, familia))

s.PrintInfo();

Console.ReadLine();

}

private static IEnumerable<Person> FindSurname1(Spravka[] spravkaDataBase, string surnameSubstring)

{

foreach (var s in spravkaDataBase)

if (s is Person)

if ((s as Person).Surname.Contains(surnameSubstring))

yield return s as Person;

}

private static IEnumerable<Friend> FindSurname2(Spravka[] spravkaDataBase, string surnameSubstring)

{

foreach (var s in spravkaDataBase)

if (s is Friend)

if ((s as Friend).Surname.Contains(surnameSubstring))

yield return s as Friend;

}

private static IEnumerable<Organization> FindSurname3(Spravka[] spravkaDataBase, string surnameSubstring)

{

foreach (var s in spravkaDataBase)

if (s is Organization)

if ((s as Organization).Kontakt.Contains(surnameSubstring))

yield return s as Organization;

}

}

public abstract class Spravka

{

public abstract void PrintInfo();

}

public class Person : Spravka

{

public string Surname { get; set; }

public string Adres { get; set; }

public string Telephone { get; set; }

public Person(string surname, string adres, string telephone)

{

Surname = surname;

Adres = adres;

Telephone = telephone;

}

public override void PrintInfo()

{

Console.WriteLine("Фамилия: {0}", Surname);

Console.WriteLine("Адрес: {0}", Adres);

Console.WriteLine("Телефон: {0}\n", Telephone);

}

}

public class Organization : Spravka

{

public string Name { get; set; }

public string AdresOrg { get; set; }

public string Number { get; set; }

public int Faks { get; set; }

public string Kontakt { get; set; }

public Organization(string name, string adresOrg, string number, int faks, string kontakt)

{

Name = name;

AdresOrg = adresOrg;

Number = number;

Faks = faks;

Kontakt = kontakt;

}

public override void PrintInfo()

{

Console.WriteLine("Название организации: {0}", Name);

Console.WriteLine("Адрес: {0}", AdresOrg);

Console.WriteLine("Номер телефона: {0}", Number);

Console.WriteLine("Факс: {0}", Faks);

Console.WriteLine("Контактное лицо: {0}\n", Kontakt);

}

}

public class Friend : Spravka

{

public string Surname { get; set; }

public string Adres { get; set; }

public string Telephone { get; set; }

public DateTime BirthdayDate { get; set; }

public Friend(string surname, string adres, string telephone, DateTime birthdayDate)

{

Surname = surname;

Adres = adres;

Telephone = telephone;

BirthdayDate = birthdayDate;

}

public override void PrintInfo()

{

Console.WriteLine("Фамилия: {0}", Surname);

Console.WriteLine("Адрес: {0}", Adres);

Console.WriteLine("Телефон: {0}", Telephone);

Console.WriteLine("День рождения: {0}\n", BirthdayDate.ToLongDateString());

}

}

}

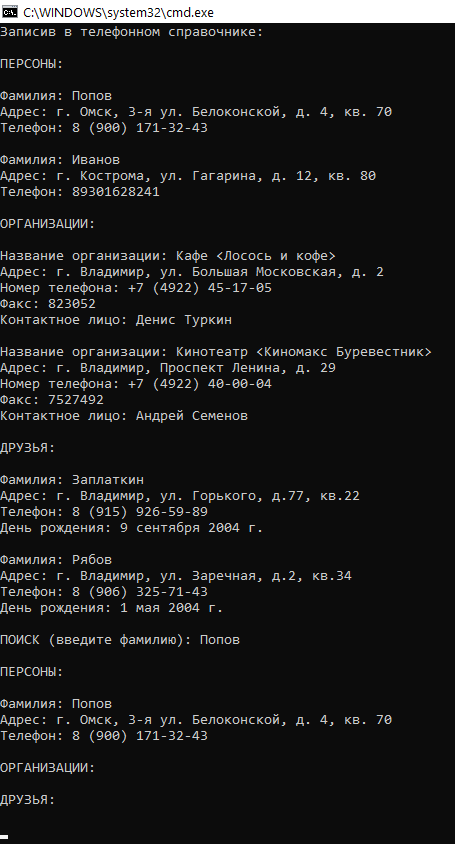


Рисунок 1 – Результат работы программы задания

**Листинг графического приложения**

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 ConsoleApp\_13\_form

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

public abstract class Spravka

{

public abstract void PrintInfo(TextBox textBox1);

}

public class Person : Spravka

{

public string Surname { get; set; }

public string Adres { get; set; }

public string Telephone { get; set; }

public Person(string surname, string adres, string telephone)

{

Surname = surname;

Adres = adres;

Telephone = telephone;

}

public override void PrintInfo(TextBox textBox1)

{

textBox1.Text += Environment.NewLine + $"Фамилия: {Surname}" + Environment.NewLine;

textBox1.Text += $"Адрес: {Adres}" + Environment.NewLine;

textBox1.Text += $"Телефон: {Telephone}" + Environment.NewLine;

}

}

public class Organization : Spravka

{

public string Name { get; set; }

public string AdresOrg { get; set; }

public string Number { get; set; }

public int Faks { get; set; }

public string Kontakt { get; set; }

public Organization(string name, string adresOrg, string number, int faks, string kontakt)

{

Name = name;

AdresOrg = adresOrg;

Number = number;

Faks = faks;

Kontakt = kontakt;

}

public override void PrintInfo(TextBox textBox1)

{

textBox1.Text += Environment.NewLine + $"Название организации: {Name}" + Environment.NewLine;

textBox1.Text += $"Адрес: {AdresOrg}" + Environment.NewLine;

textBox1.Text += $"Номер телефона: {Number}" + Environment.NewLine;

textBox1.Text += $"Факс: {Faks}" + Environment.NewLine;

textBox1.Text += $"Контактное лицо: {Kontakt}" + Environment.NewLine;

}

}

public class Friend : Spravka

{

public string Surname { get; set; }

public string Adres { get; set; }

public string Telephone { get; set; }

public DateTime BirthdayDate { get; set; }

public Friend(string surname, string adres, string telephone, DateTime birthdayDate)

{

Surname = surname;

Adres = adres;

Telephone = telephone;

BirthdayDate = birthdayDate;

}

public override void PrintInfo(TextBox textBox1)

{

textBox1.Text += Environment.NewLine + $"Фамилия: {Surname}" + Environment.NewLine;

textBox1.Text += $"Адрес: {Adres}" + Environment.NewLine;

textBox1.Text += $"Телефон: {Telephone}" + Environment.NewLine;

textBox1.Text += $"День рождения: {BirthdayDate.ToLongDateString()}" + Environment.NewLine + Environment.NewLine;

}

}

private static IEnumerable<Person> FindSurname1(Spravka[] spravkaDataBase, string surnameSubstring)

{

foreach (var s in spravkaDataBase)

if (s is Person)

if ((s as Person).Surname.Contains(surnameSubstring))

yield return s as Person;

}

private static IEnumerable<Friend> FindSurname2(Spravka[] spravkaDataBase, string surnameSubstring)

{

foreach (var s in spravkaDataBase)

if (s is Friend)

if ((s as Friend).Surname.Contains(surnameSubstring))

yield return s as Friend;

}

private static IEnumerable<Organization> FindSurname3(Spravka[] spravkaDataBase, string surnameSubstring)

{

foreach (var s in spravkaDataBase)

if (s is Organization)

if ((s as Organization).Kontakt.Contains(surnameSubstring))

yield return s as Organization;

}

private void Form1\_Load(object sender, EventArgs e)

{

Spravka[] spravkaDataBase = new Spravka[]

{

new Person("Попов", "г. Омск, 3-я ул. Белоконской, д. 4, кв. 70 ", "8 (900) 171-32-43"),

new Person("Иванов", "г. Кострома, ул. Гагарина, д. 12, кв. 80 ", "89301628241"),

new Organization("Кафе «Лосось и кофе»", "г. Владимир, ул. Большая Московская, д. 2", "+7 (4922) 45-17-05", 823052, "Денис Туркин"),

new Organization("Кинотеатр «Киномакс Буревестник»", "г. Владимир, Проспект Ленина, д. 29", "+7 (4922) 40-00-04", 7527492, "Андрей Семенов"),

new Friend("Заплаткин", "г. Владимир, ул. Горького, д.77, кв.22 ", "8 (915) 926-59-89", new DateTime(2004, 9, 9)),

new Friend("Рябов", "г. Владимир, ул. Заречная, д.2, кв.34 ", "8 (906) 325-71-43", new DateTime(2004, 5, 1))

};

textBox1.Text += "Записив в телефонном справочнике:" + Environment.NewLine;

textBox1.Text += Environment.NewLine + Environment.NewLine + "ПЕРСОНЫ:" + Environment.NewLine;

foreach (var s in FindSurname1(spravkaDataBase, ""))

s.PrintInfo(textBox1);

textBox1.Text += Environment.NewLine + "ОРГАНИЗАЦИИ:" + Environment.NewLine;

foreach (var s in FindSurname3(spravkaDataBase, ""))

s.PrintInfo(textBox1);

textBox1.Text += Environment.NewLine + "ДРУЗЬЯ:" + Environment.NewLine;

foreach (var s in FindSurname2(spravkaDataBase, ""))

s.PrintInfo(textBox1);

}

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

{

textBox1.Clear();

Spravka[] spravkaDataBase = new Spravka[]

{

new Person("Попов", "г. Омск, 3-я ул. Белоконской, д. 4, кв. 70 ", "8 (900) 171-32-43"),

new Person("Иванов", "г. Кострома, ул. Гагарина, д. 12, кв. 80 ", "89301628241"),

new Organization("Кафе «Лосось и кофе»", "г. Владимир, ул. Большая Московская, д. 2", "+7 (4922) 45-17-05", 823052, "Денис Туркин"),

new Organization("Кинотеатр «Киномакс Буревестник»", "г. Владимир, Проспект Ленина, д. 29", "+7 (4922) 40-00-04", 7527492, "Андрей Семенов"),

new Friend("Заплаткин", "г. Владимир, ул. Горького, д.77, кв.22 ", "8 (915) 926-59-89", new DateTime(2004, 9, 9)),

new Friend("Рябов", "г. Владимир, ул. Заречная, д.2, кв.34 ", "8 (906) 325-71-43", new DateTime(2004, 5, 1))

};

string familia = textBox2.Text;

textBox1.Text += Environment.NewLine + "ПЕРСОНЫ:" + Environment.NewLine;

foreach (var s in FindSurname1(spravkaDataBase, familia))

s.PrintInfo(textBox1);

textBox1.Text += Environment.NewLine + "ОРГАНИЗАЦИИ:" + Environment.NewLine;

foreach (var s in FindSurname3(spravkaDataBase, familia))

s.PrintInfo(textBox1);

textBox1.Text += Environment.NewLine + "ДРУЗЬЯ:" + Environment.NewLine;

foreach (var s in FindSurname2(spravkaDataBase, familia))

s.PrintInfo(textBox1);

}

}

}

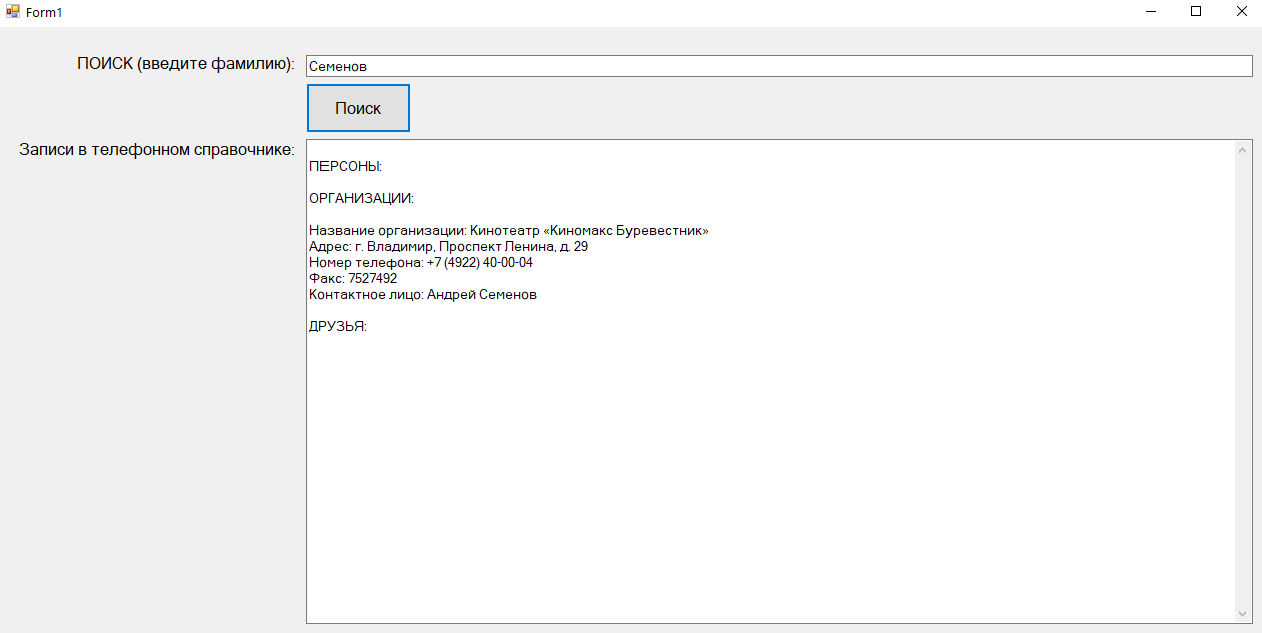


Рисунок 2 – Графический интерфейс первого задания

**Ссылка на github:**

Все работы будут храниться в следующем репозитории:

репозиторий со всеми консольными и графическими реализациями заданий:

https://github.com/Oxygen-182/Rodionov\_Week-1\_Task-13