Одеський національний політехнічний університет

Кафедра «Комп'ютеризовані системи управління»

**КУРСОВА РОБОТА**

з дисципліни «Основи технології .Net»

на тему: «Створення інформаційної бази даних»

Варіант №1

Студентки 1 курсу, групи АТ–181

спеціальності «Автоматизація та комп'ютерно-інтегровані технології»

Прізвище: Баженова А. І.

Керівник: доц. Сперанський В.О.

Національна шкала:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оцінка: ECTS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Члени комісії \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

м. Одеса – 2019 рік

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc483951404)

[Постановка задачи варианта №1 3](#_Toc483951405)

[РАЗДЕЛ 1 ЭТАПЫ ПРОЕКТИРОВАНИЯ И ИХ РЕАЛИЗАЦИЯ 4](#_Toc483951406)

[1.1 Календарное планирование проекта 4](#_Toc483951407)

[1.1 Описание проектирования программы 4](#_Toc483951408)

[1.1.1 Создание базы данных. 4](#_Toc483951409)

[1.1.2 Функции программы 5](#_Toc483951410)

[1.1.3 Создание интерфейса пользователя 9](#_Toc483951411)

[1.1.4 Библиотеки 10](#_Toc483951413)

[РАЗДЕЛ 2 СПРАВОЧНАЯ ИНФОРМАЦИЯ ДЛЯ ПОЛЬЗОВАТЕЛЯ 11](#_Toc483951415)

[1.1 Краткое описание продукта 11](#_Toc483951416)

[1.1. Создание базы данных 11](#_Toc483951418)

[1.1 Импортирование базы данных 12](#_Toc483951419)

[1.2 Панель управления 13](#_Toc483951420)

[1.3 Управление информацией базы данных 14](#_Toc483951421)

[1.6.1 Добавление инвестора 14](#_Toc483951422)

[ВЫВОДЫ 15](#_Toc483951426)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 16](#_Toc483951427)

**ВВЕДЕНИЕ**

**База данных** — совокупность хранимых в памяти компьютера данных, относящихся к определенному объему или кругу деятельности, специально организованных, обновляемых и логически связанных между собой. Они представляют собой своеобразную информационную модель объекта (например, управляемого объекта в АСУ).

База данных, размещенная и работающая на одном компьютере, называется локальной, а на нескольких связанных между собой компьютерах — распределенной базой данных.

## Постановка задачи варианта №1

Создать базу данных, содержащую сведения об инвесторах трастовой компании «MMM». Каждая запись содержит следующие сведения:

* ФИО вкладчика;
* номер договора;
* домашний адрес;
* сумма вклада;
* на какой срок заключён договор.

Предусмотреть:

а) выдачу сведений о каждом из вкладчиков (по фамилии, введенной по запросу);

б) выдачу сведений о вкладчиках, вклад которых превышает заданную сумму;

в) выдачу сведений о вкладчиках, у которых срок договора свыше 12 месяцев.

г) удаление записей о вкладчиках с заданной фамилией.

# РАЗДЕЛ 1 ПРОЕКТИРОВАНИЕ И ИХ РЕАЛИЗАЦИЯ

## Календарное планирование проекта

Поэтапное календарное планирование для выполняемого курсового проекта представлено в табл. 1.1.

Таблица 1.1 – Календарный план курсового проектирования

|  |  |  |  |
| --- | --- | --- | --- |
| № | Наименование | Содержание | Дата окончания |
| 1 | Начало | Изучение постановочного задания, предметной области | 28.04.2019 |
| 2 | Теоретическая подготовка | Изучение литературы по дисциплины | 10.05.2019 |
| 3 | Проектирование модулей проекта | Разработка компонентов программы, их функций, свойств, а также связь между ними. | 20.05.2019 |
| 4 | Разработка программной реализации | Написание непосредственно самой программы с использованием изученных теоретических знаний | 21.05.2019 |
| 5 | Тестирование | Проверка созданного приложения на предмет некорректного поведения при вводе ошибочных значений, определение уровня эргономики, устойчивости разработки | 22.05.2019 |
| 6 | Написание документации | Написание документации с описанием выполненной работы и трудностями с которыми пришлось встретиться во время её выполнения | 30.05.2019 |

## Описание проектирования программы

### Проектирование архитектуры приложения начинается с формирования базы данных согласно пунктам задания. Экземпляром базы данных в коде является класс «Investor». Пример кода данного класса:

class Investor

{

// класс << Инвестор >>

// Поля класса

string investorName; // наименование

int contractNumber; // номер контракта

string homeAdress; // домашний адрес

int depositAmount; // депозитная сумма

int contractTerm; // срок контракта

// Конструктор по умолчанию

public Investor()

{

}

// Создание конструктора с входными параметрами

public Investor(string investorName, int contractNumber, string homeAdress, int depositAmount, int contractTerm)

{

InvestorName = investorName;

ContractNumber = contractNumber;

HomeAdress = homeAdress;

DepositAmount = depositAmount;

ContractTerm = contractTerm;

}

// Инкапсуляция полей

public string InvestorName { get => investorName; set => investorName = value; }

public int ContractNumber { get => contractNumber; set => contractNumber = value; }

public string HomeAdress { get => homeAdress; set => homeAdress = value; }

public int DepositAmount { get => depositAmount; set => depositAmount = value; }

public int ContractTerm { get => contractTerm; set => contractTerm = value; }

### Функции программы

При описании методов работы со списком были выполнены такие задачи как: создание списка, удаление списка, добавление записи в список, удаление записи из списка, редактирование списка, обновление таблицы, открытие базы, сохранение базы и загрузка базы. Пример кода предоставлен ниже.

//работа со списком

private void Button6\_Click(object sender, EventArgs e)

private void Button7\_Click(object sender, EventArgs e)

private void Button4\_Click(object sender, EventArgs e)

private void Button5\_Click(object sender, EventArgs e)

private void Button1\_Click(object sender, EventArgs e)

private void Button2\_Click(object sender, EventArgs e)

private void Button3\_Click(object sender, EventArgs e)

{

// Работа с Файлами

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

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

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

Также в программе были созданы функции, предусмотренные постановкой задачи:

* Функция, которая удаляет.Удаляет в базе все записи, где встречается такой номер.
* Функция вывода данных о сумме вклада и вкладчиков
* Функция вывода наименование и количество всех вкладчиков.
* Функция проверки корректности ввода.
* Функция проверки текстовых полей на наличие входных данных.
* Функция, которая выводит информацию об вкладчике по фамилии и сумме.
* Функция, которая выводит в таблицу записи по указанному номеру вклада

Создаём имя объекта excel:

Excel.Application excelapp = new Excel.Application();

Excel.Workbook workbook = excelapp.Workbooks.Add();

Excel.Worksheet worksheet = workbook.ActiveSheet;

excelapp.Visible = false;

for (int i = 0; i < dataGridView1.RowCount; i++)

{

worksheet.Rows[i + 1].Columns[1].Value = investors[i].InvestorName;

worksheet.Rows[i + 1].Columns[2].Value = investors[i].ContractNumber;

worksheet.Rows[i + 1].Columns[3].Value = investors[i].HomeAdress;

worksheet.Rows[i + 1].Columns[4].Value = investors[i].DepositAmount;

worksheet.Rows[i + 1].Columns[5].Value = investors[i].ContractTerm;

}

excelapp.AlertBeforeOverwriting = false;

workbook.SaveAs(path, ReadOnlyRecommended: true);

excelapp.Quit();

Чтение записей из файла:

investors.Clear();

saveFileDialog.Filter = "Excel (\*.xlsx)|\*.xlsx | Excel (\*.xls)|\*.xls";

if (openFileDialog.ShowDialog() == DialogResult.Cancel)

return;

string filename = openFileDialog.FileName;

Microsoft.Office.Interop.Excel.Application ExcelApp = new Microsoft.Office.Interop.Excel.Application();

Microsoft.Office.Interop.Excel.\_Workbook ExcelWorkBook;

Microsoft.Office.Interop.Excel.Worksheet ExcelWorkSheet;

//Opening

ExcelWorkBook = ExcelApp.Workbooks.Open(filename, 0, true, 5, "", "", true, Microsoft.Office.Interop.Excel.XlPlatform.xlWindows, "\t", false,

false, 0, true, 1, 0);

ExcelWorkSheet = (Microsoft.Office.Interop.Excel.Worksheet)ExcelWorkBook.Worksheets.get\_Item(1);

for (int i = 1; i < ExcelApp.Rows.Count; i++)

{

if (ExcelApp.Cells[i, 1].Value != null)

{

string a = ExcelWorkSheet.Rows[i].Columns[1].Value;

investors.Add(new Investor()

{

InvestorName = ExcelWorkSheet.Rows[i].Columns[1].Value,

ContractNumber = Convert.ToInt32(ExcelWorkSheet.Rows[i].Columns[2].Value),

HomeAdress = ExcelWorkSheet.Rows[i].Columns[3].Value,

DepositAmount = Convert.ToInt32(ExcelWorkSheet.Rows[i].Columns[4].Value),

ContractTerm = Convert.ToInt32(ExcelWorkSheet.Rows[i].Columns[5].Value)

});

dataGridView1.Rows.Add(ExcelWorkSheet.Rows[i].Columns[1].Value, ExcelWorkSheet.Rows[i].Columns[2].Value, ExcelWorkSheet.Rows[i].Columns[3].Value, ExcelWorkSheet.Rows[i].Columns[4].Value, ExcelWorkSheet.Rows[i].Columns[5].Value);

}

else

break;

}

ExcelApp.Quit();

UpdateNotes();

Запись списка в файл:

saveFileDialog.Filter = "Excel (\*.xlsx)|\*.xlsx | Excel (\*.xls)|\*.xls";

if (saveFileDialog.ShowDialog() == DialogResult.Cancel)

return;

string filename = saveFileDialog.FileName;

Write(filename);

MessageBox.Show("Файл сохранен");

### 1.1.3 Создание интерфейса пользователя

Интерфейс представляет собой рабочую область с таблицей (Рис.1.1), в которую выводятся записи из базы данных. Для работы с таблицей предусмотрены текстовые поля и кнопки (Рис. 1.2).

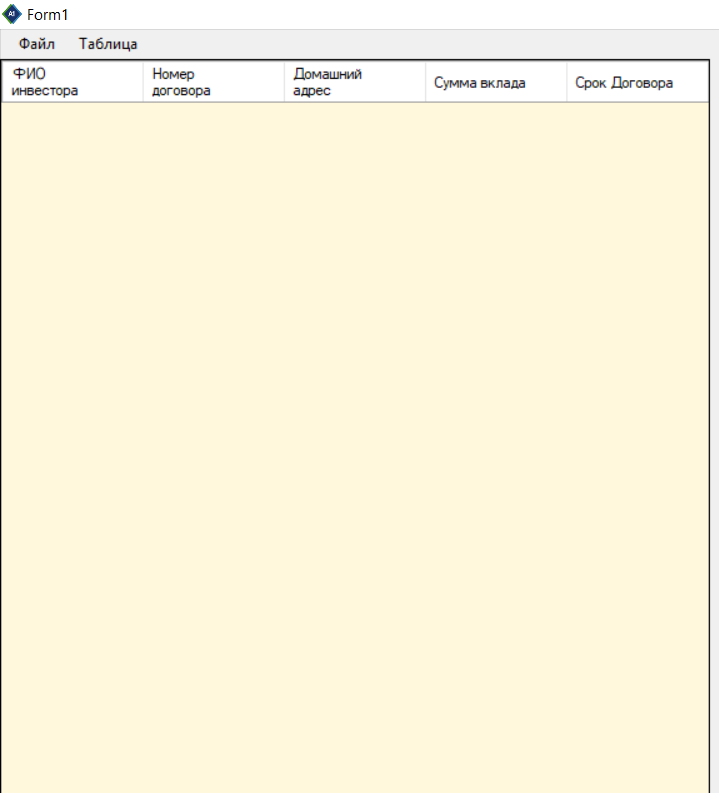


Рис. 1.1. - Элемент dataGridView

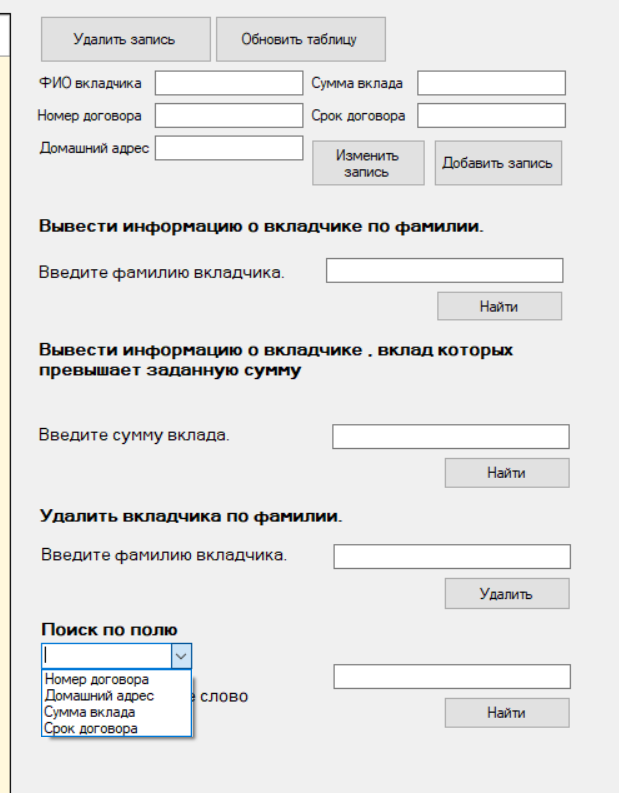


Рис. 1.2. - Элементы управления таблицей

# 1.1.4. Библиотеки

Использованные библиотеки:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using Excel = Microsoft.Office.Interop.Excel;

# РАЗДЕЛ 2 СПРАВОЧНАЯ ИНФОРМАЦИЯ ДЛЯ ПОЛЬЗОВАТЕЛЯ

## Краткое описание продукта

Приложение представляет собой ПО, в котором разработана база данных «Инвестор». Функциональность программы включает:

* управление индивидуальными данными инвестора: добавить и удалить из базы данных;
* вывод информации о вкладчике по фамилии и вкладе который превышает заданую сумму;
* удаление вкладчика из базы данных по фамилии;
* поиск по заданному полю;

**Как добавить Инвестора:**

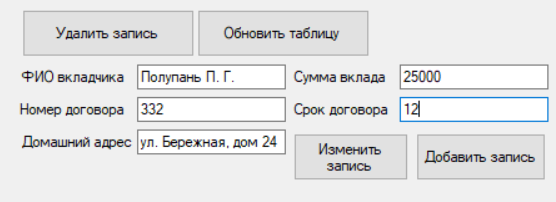
* 1. Заполняем поля нужней информацией (Рис. 2.1).

Рис. 2.1

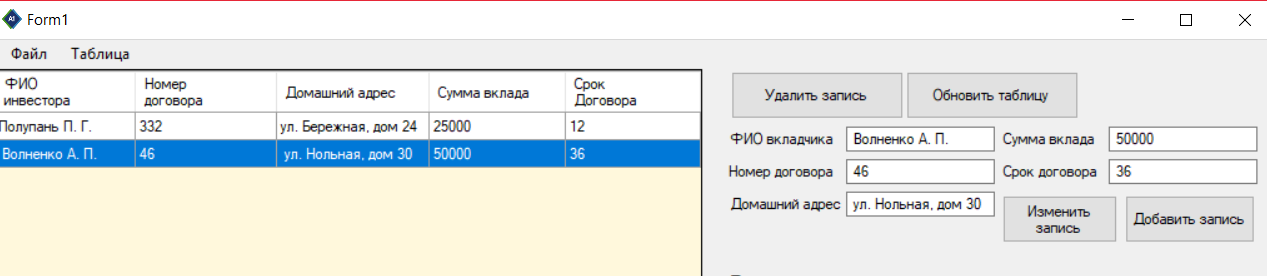
* 1. Далее надо нажать на кнопку “Добавить запись”.
  2. Теперь вкладчик добавлен в базу данных (Рис. 2.2).

Рис. 2.2

**Импорт данных:**

1. Экспорт данных из файла, в данной программе, производяться с расширением “xlsx” (Рис. 2.3).

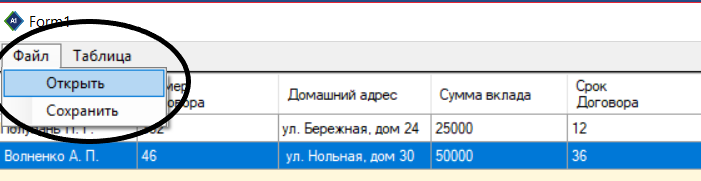


Рис. 2.3

1. В открывшемся окне, выберете файл и нажмите “Открыть” (Рис. 2.4).

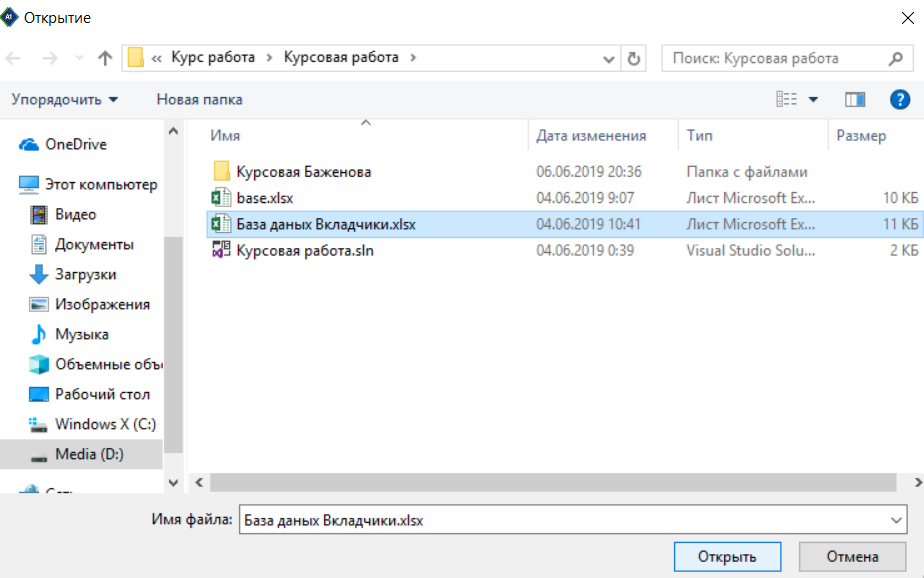


Рис. 2.4

**Экспорт данных:**

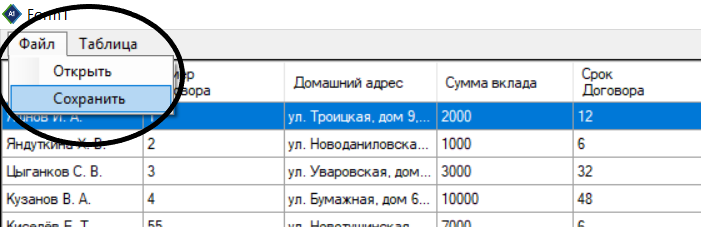
1. Сохраняем данные с расширением “xlsx”, выбераем пункт “Сохранить”. (Рис. 2.5).

Рис. 2.5

1. Далее выбираем путь хранения файла и даём ему название (Рис. 2.6).

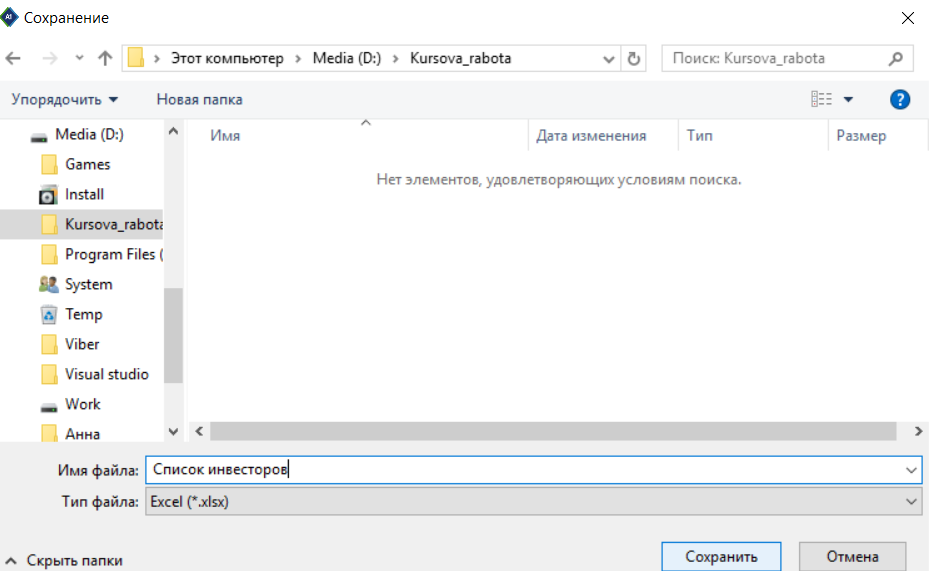


Рис. 2.6

**Поиск:**

В данной программе вы можем произвести поиск по таким параметрам:

* Вводим фамилию вкладчика и программа выдаёт всю информацию о нём;
* Вводим сумму вклада и программа выдаёт список вкладчиков вклад которых превышает данную сумму;
* Вводим фамилюю вкладчика и программа удаляет данного инвестора из базы данных;
* Также можно задать поиск по заданному полю (Номер договора, домашний адрес, сумма вклада, срок договора);

## Панель управления

В панели управления находяться такие элименты как: контекстное меню (1), таблица вывода (2), текстовые поля и кнопки для обработки записей в таблице (3).

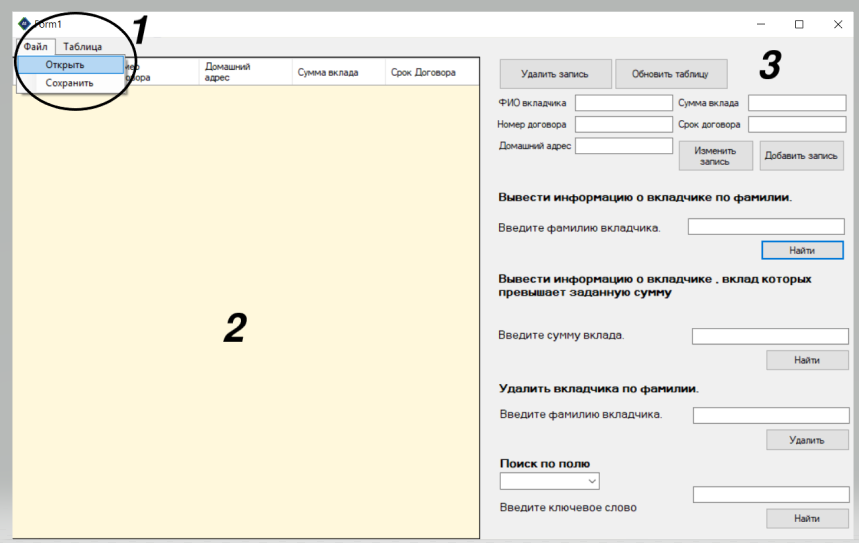


Рис. 2.1 – 1).Контекстное меню; 2). Таблица вывода; 3). Текстовые поля и кнопки для обработки зписей в таблице

## 1.3 Управление информацией базы данных

1.3.1 Добавление Инвестора

Функция записи данных включает в себя такую информацию об Инвесторе (рис 2.2): ФИО вкладчика, номер договора, домашний адрес, сумма договора, на сколько месяцев заключён договор.

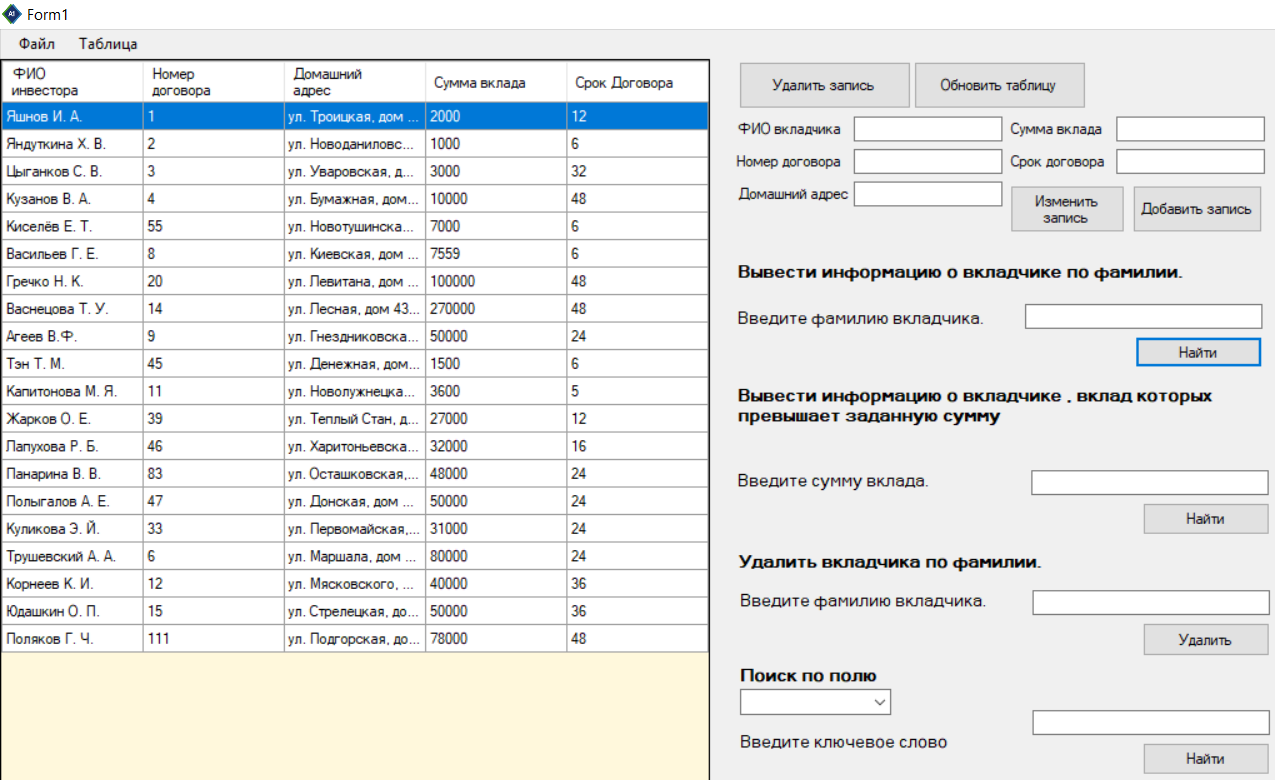


Рис. 2.2 – Вывод информации об инвесторах

# ВЫВОДЫ

На основании знаний, приобретенных на курсе “Основы технологии .NET”, было создано приложение, которое автоматизирует управление базы данных, упрощая поиск, редактирование и удаление информации, и экономя время на ее обработку. В ходе выполнения курсовой работы было изучено много новых инструментов, предоставлено практическое применение знаний что и повысило уровень профессиональных навыков.

# СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ:

* <https://ru.stackoverflow.com/>
* <http://www.cyberforum.ru/net-framework/>
* Герберт Шилдт - C# 4.0: полное руководство.
* <https://docs.microsoft.com/ru-ru/dotnet/csharp/>
* Работа с Windows Forms [Электронный ресурс] – Режим доступа к ресурсу: <https://bit.ly/2YvYvfA>.