Министерство науки и высшего образования Российской Федерации

федеральное государственное бюджетное образовательное учреждение

высшего образования

КУЗБАССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ИМЕНИ Т.Ф. ГОРБАЧЕВА

(КузГТУ)

ОТЧЕТ

по учебной практике

по получению профессиональных умений и опыта профессиональной деятельности

Индивидуальное задание

Разработка ИС с использованием принципов ООП.

|  |  |
| --- | --- |
| Выполнил: | ст. гр. ИТб-202  Исомадинов А.Х. |
| Руководитель от кафедры ИиАПС: | Доцент(к.н)  Ванеев Олег Николаевич |
|  |  |

Кемерово 2022

**Оглавление**

[Исходное задание 2](#_Toc298876627)

[Описание функциональности разрабатываемой системы 2](#_Toc2087980260)

[Разработка информационной системы 3](#_Toc130022315)

[1. Описание классов разрабатываемой системы 3](#_Toc997326441)

[2. Разработка классов на языке C# 4](#_Toc2119172944)

[3. Добавление структуры и статического класса 7](#_Toc679885023)

[4. Реализация абстрактных классов и полиморфизма 7](#_Toc363734965)

[5. Реализация интерфейса 8](#_Toc799509210)

[6. Реализация пользовательского интерфейса 8](#_Toc1742118356)

# Исходное задание

Разработка информационной системы, состоящей из базовых классов по 7 варианту: «Банк», «Банковская услуга», «Кредит», «Депозит», «Клиент». Классы содержат атрибуты и функции, указанные в таблице 1. Вместе с этим требуется разработка графического интерфейса, отображающего всю информацию в существующих экземплярах класса, позволяющий выполнять весь функционал классов.

**Таблица 1**

**Атрибуты и функции базовых классов**

|  |  |  |
| --- | --- | --- |
| **Класс** | **Атрибуты** | **Функции** |
| Банковская услуга | * Дата * Сумма * Процент | - |
| Кредит | * Вид * Срок погашения | - |
| Депозит | * Вид * Номер счёта | - |
| Банк | * Название * Телефон * Адрес | * Добавить кредит * Удалить кредит * Добавить депозит * Удалить депозит * Добавить клиента * Удалить клиента |
| Клиент | * ФИО * Паспортные данные * Телефон | - |

# Описание функциональности разрабатываемой системы

В разрабатываемой программе мы создаём новый банк, после пользователь имеет возможность добавить клиента, кредит, депозит, удалить выбранного клиента, кредита, депозита или просмотреть подробную информацию.

Диаграмма использования представлена на рисунке 1.

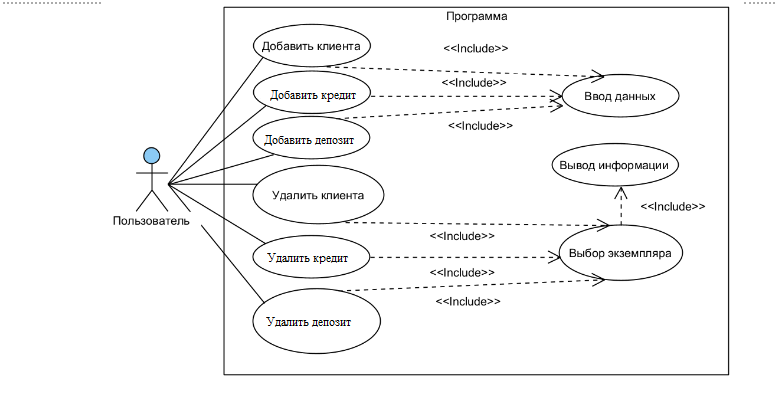


Рисунок 1 – Диаграмма вариантов использования

# Разработка информационной системы

## Описание классов разрабатываемой системы

Для подробного описания каждого из классов разрабатываемой системы была сделана диаграмма классов на языке UML. Она представлена на рисунке 2.

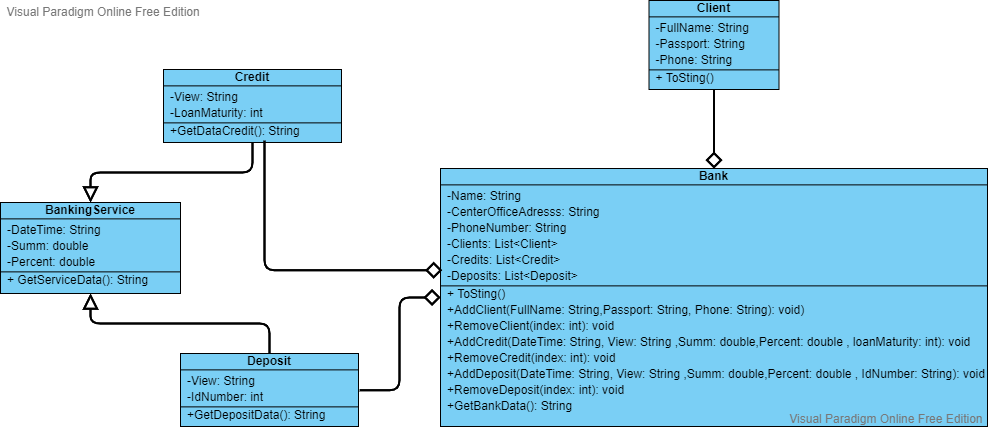
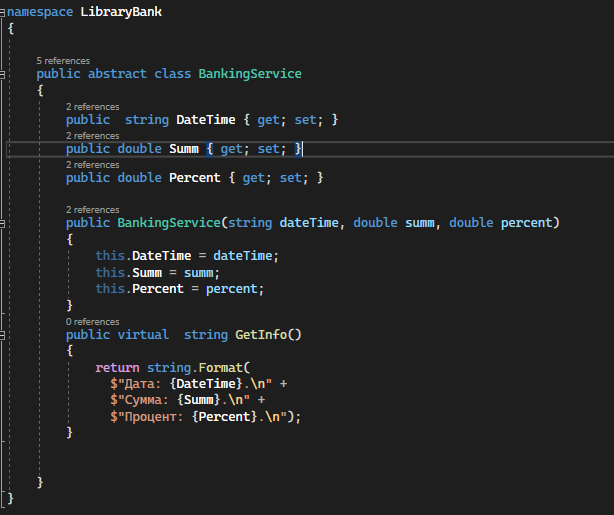


Рисунок 2 – Диаграмма классов

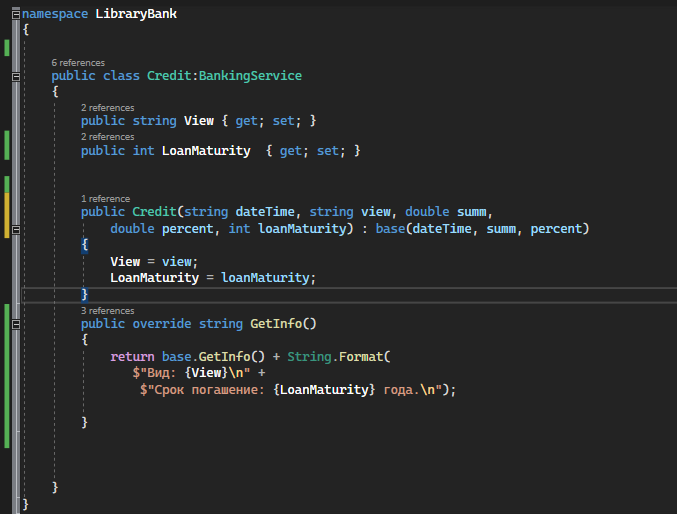
## Разработка классов на языке C#

На основе диаграммы классов написана библиотека классов на языке C#. Вначале написан класс BankingService – Банковская услуга с атрибутами, дата, сумма и процент. Код класса в листинге 1.



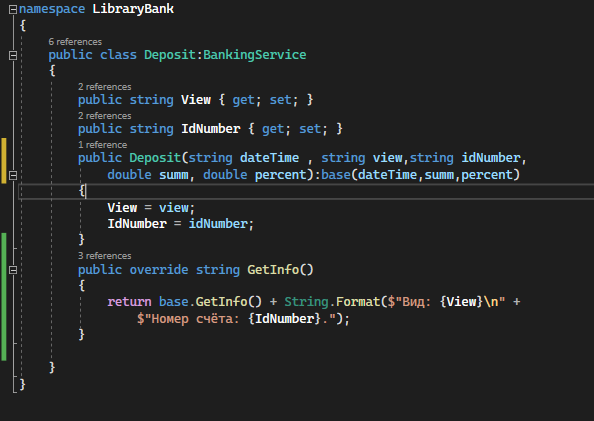
Листинг 1 – Код класса BankingService

Далее написан класс Credit - кредит, наследует класс BankingService и имеет атрибуты, вид и срок погашение. Код класса в листинге 2.



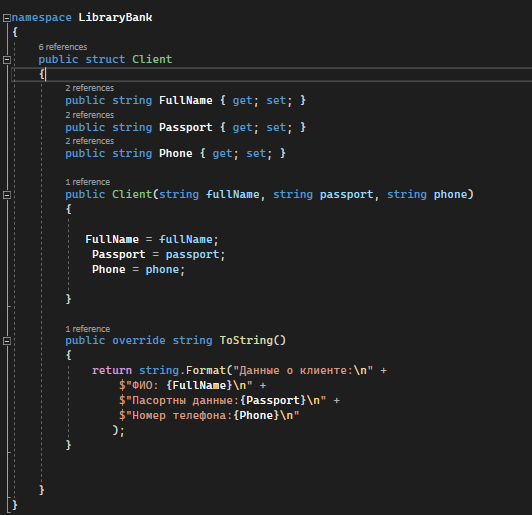
Листинг 2 – Код класса Credit

Далее написан класс Deposit – депозит, наследует класс BankingService и имеет атрибуты вид и номер счёта. Код класса в листинге 3.



Листинг 3 – Код класса Deposit

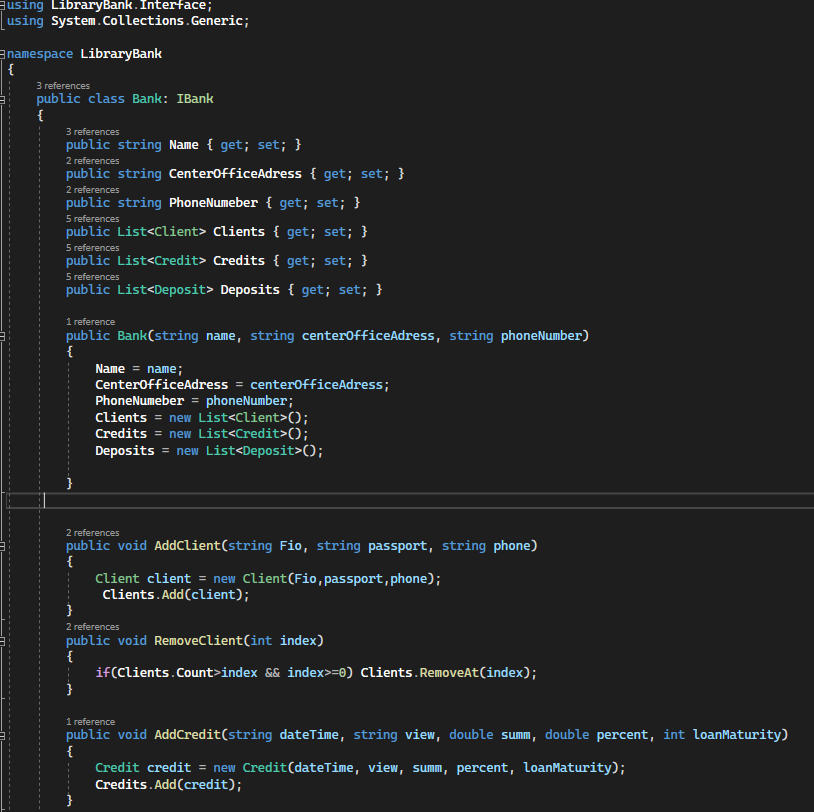
Далее написан класс Client – клиент, позже переделанный в структуру, атрибуты: ФИО, паспорт, номер телефона. Код класса в листинге 4.



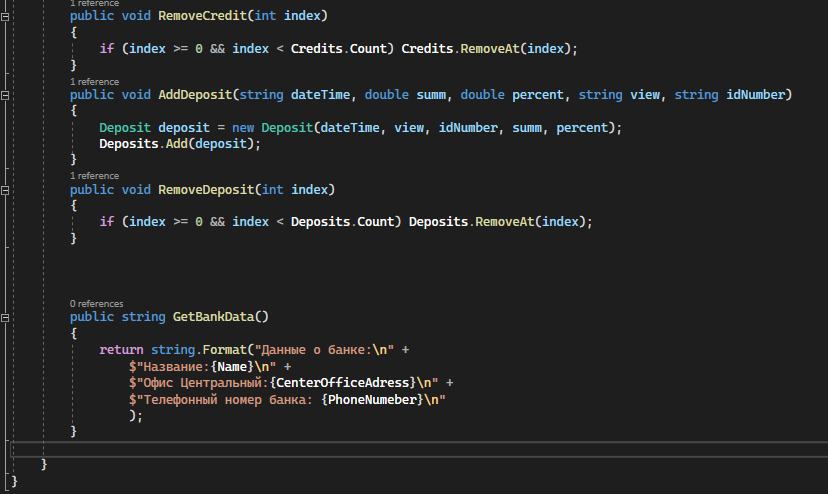
Листинг 4 – Код класса Client

Далее написан класс Bank – Банк с атрибутами адреса; номера телефона; название: списком клиентов, кредитов, депозитов и методами добавления клиента, кредита, депозита методы их удаления.

Код класса в листингах 5.1 и 5.2 .



Листинг 5.1 – Код класса Bank

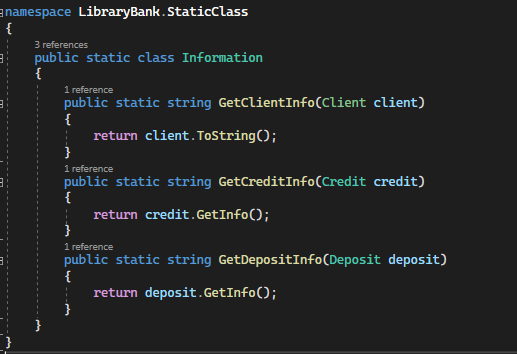


Листинг 5.2– Код класса Bank

## Добавление структуры и статического класса

Для реализации структуры класс Client был сделан структурой, его код представлен в листинге 4.

Для реализации статического класса был написан класс Information, содержащий в себе методы для возврата информации о выбранном клиенте, кредите и депозите. Код класса в листинге 6.



Листинг 6 – Код класса Information

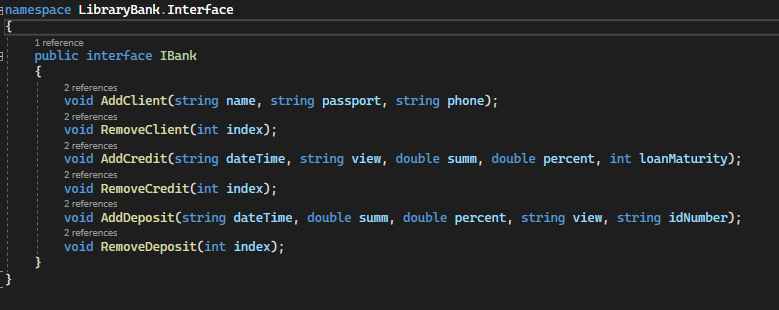
## Реализация абстрактных классов и полиморфизма

Для реализации абстрактного класса класс BankingService был сделан абстрактным, так как его экземпляры не создаются, а используется он только для дальнейшего наследования. Код класса в листинге 1.

Для реализации полиморфизма в классе BankingService – Банковская услуга был написан виртуальный метод GetInfo(), он служит для получения информации об экземпляре объекта. Класс наследуется в Credit и Deposit – кредит и депозите, в этих классах метод переопределяется с помощью ключевого слова override, в котором добавляется информация о экземпляре объекта класса кредита или депозита. Код BankingService – листинг 1, Credit – листинг 2, Deposit – листинг 3.

## Реализация интерфейса

Был реализован интерфейс IBank, в нём перечислены методы для класса Bank, а именно: добавление и удаление клиента, кредита и депозита. Код интерфейса в листинге 7.



Листинг 7 – Код интерфейса IBank

## Реализация пользовательского интерфейса

Для работы со всей вышеописанной библиотекой классов была разработана программа Windows Form. При её запуске перед пользователем появляется форма, представленная на рисунке 3. На ней предлагается ввести данные Банка. Код формы в листингах 8.1-8.4 .

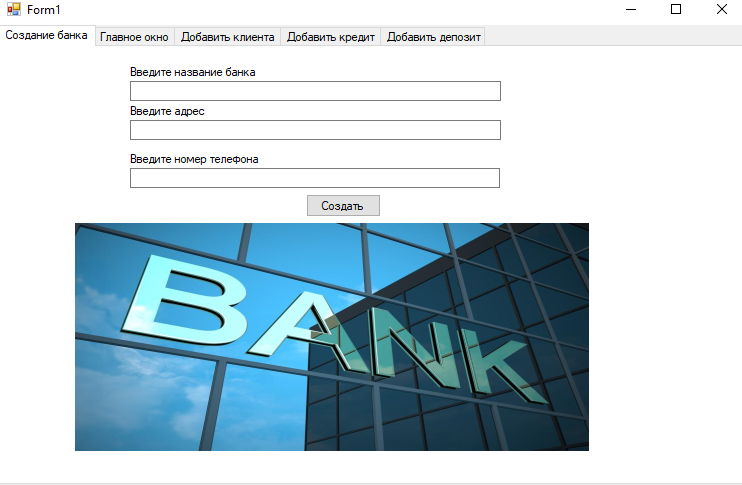
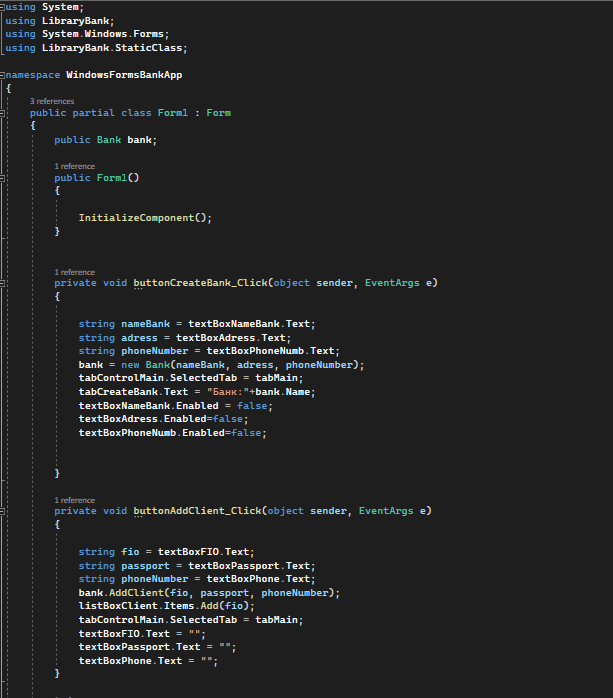
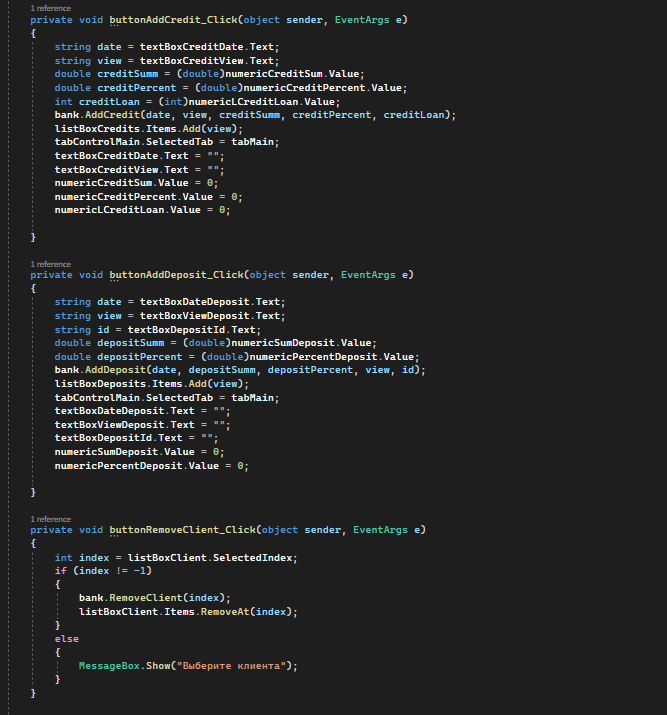


Рисунок 3 – Форма при запуске программы



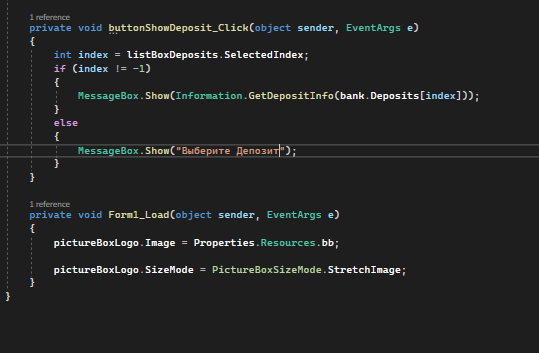
Листинг 8.1



Листинг 8.2



Листинг 8.3



Листинг 8.4

Листинг 8.1 - 8.4 – Код формы

После заполнения данных Банка и нажатия кнопки «Создать» создаётся новый Банк и пользователя перенаправляет на главную вкладку (рисунок 4).

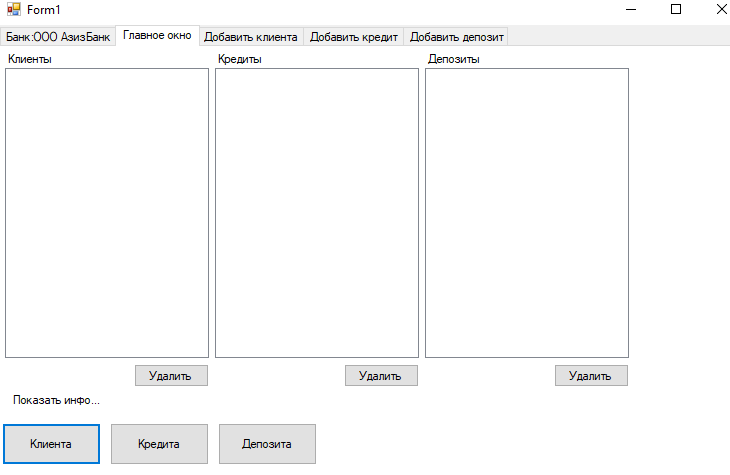


Рисунок 4 – Главная вкладка

Для добавления нового клиента, кредита или депозита существуют отдельные вкладки, их вид и заполненная в них информация представлена на рисунке 5, 6 и 7 соответственно.

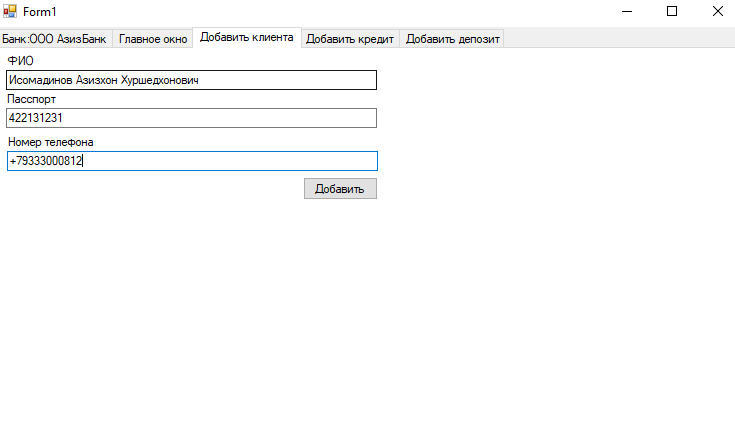


Рисунок 5 – Вкладка добавления клиента

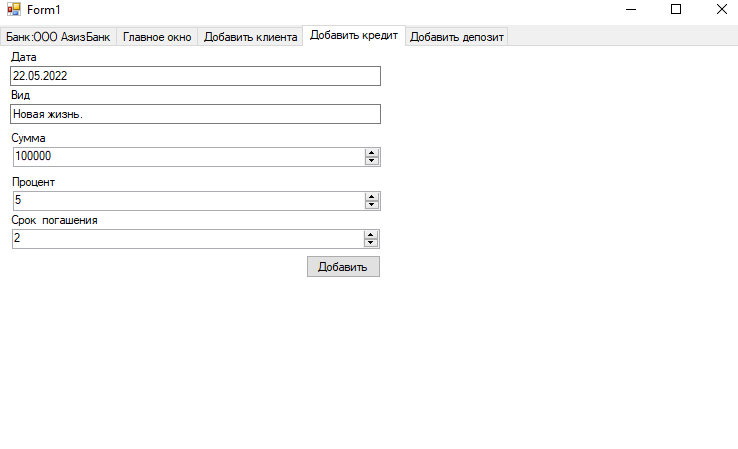


Рисунок 6 – Вкладка добавления кредита

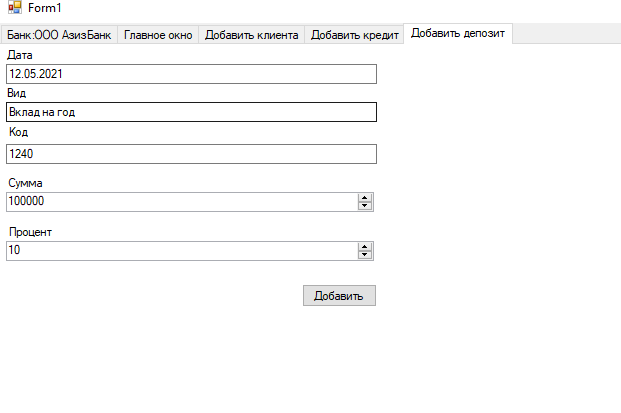


Рисунок 7 - Вкладка добавления депозита

Заполним форму некоторой информацией, после которой списки на главной вкладке пополнятся новыми пунктами, вид формы после проделанных манипуляций на рисунке 8.

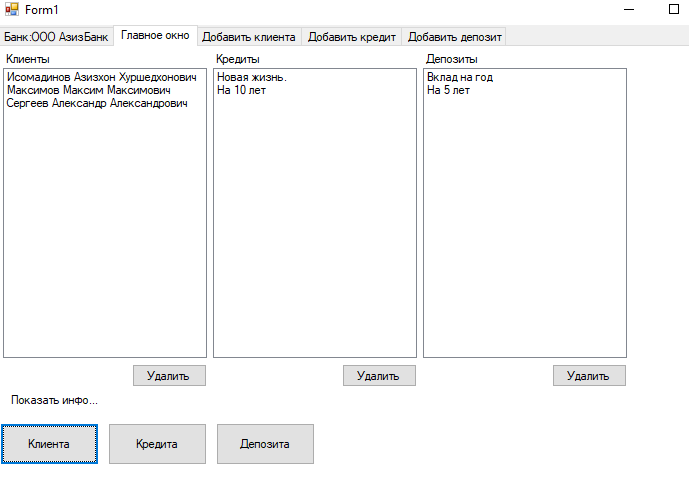


Рисунок 8 – Заполненная форма

Существует возможность показа информации выбранного в списке клиента (рисунок 9), кредита (рисунок 10), депозита (рисунок 11) или сообщение об ошибке, если не выбран конкретный экземпляр (пример на клиенте – рисунок 12).

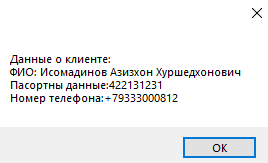


Рисунок 9 – Информация о клиенте

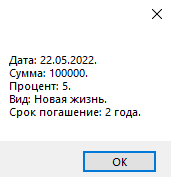


Рисунок 10 – Информация о кредите

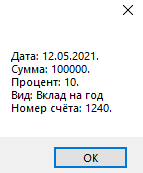


Рисунок 11 – Информация о депозите

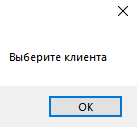


Рисунок 12 – Сообщение об ошибке

Для удаления выбранного клиента, кредита или депозита нужно в главной вкладке нажать на соответствующую кнопку «Удалить» под нужным списком. Произведём удаление некоторых случайных экземпляров, вид формы после удаления – рисунок 13.

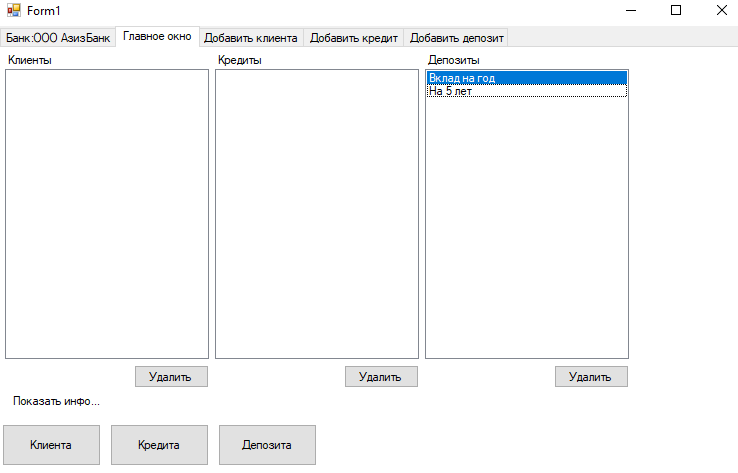


Рисунок 13 – Форма после удаления некоторых данных