МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования «Ижевский государственный технический университет

имени М. Т. Калашникова»

Факультет «Информатика и вычислительная техника»

Кафедра «Программное обеспечение»

Курсовая работа

по дисциплине «Объектно-ориентированное программирование»

Выполнил

студент группы Б19-191-2з А.Р.Закиров

Руководитель

Ст. преп. Кафедры ПО М. О.Еланцев

Рецензия

степень достижения поставленной цели работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

полнота разработки темы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

уровень самостоятельности работы обучающегося \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

недостатки работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ижевск 2021 СОДЕРЖАНИЕ

Введение3

ВВЕДЕНИЕ

В настоящее время ни одно предприятие, учреждение или организация не может функционировать достаточно эффективно без применения компьютерной техники. Непрекращающееся развитие любого предприятия, учреждения или организации требует расширения автоматизированных информационных систем.

Рынок гостиничных услуг, прежде всего, ориентирован на работу с клиентами, на создание безопасных и комфортных условий проживания и предоставление полного перечня современных информационных услуг. Уровень сервиса напрямую влияет на классность гостиничного объекта и определяет его конкурентные преимущества на рынке. Одну из главных функций в организации высококлассного сервиса в гостиничном комплексе выполняют системы автоматизации, отвечающие не только за инженерные и телекоммуникационные процессы в здании, но и за все этапы работы с клиентами, начиная бронированием номера и заканчивая созданием базы данных.

Организовать работу гостиничного комплекса с максимальной эффективностью и в соответствии с современными мировыми требованиями сегодня возможно с использованием современных компьютерных технологий. Вопрос комплексной автоматизации процессов бронирования, регистрации и размещения гостей, расчетов с ними является весьма актуальным для гостиниц.

1. ОПИСАНИЕ ЗАДАНИЯ

Задача данной курсовой работы - составить систему управления местами гостиницы.

Данная система должна отвечать следующим требованиям:

1. Ведение базы данных (номера, услуги, клиенты).
2. Выполнение функций (поселение и выселение клиентов, бронирование мест, учёт оказанных услуг).
3. Получение данных (счет за проживание и услуги, список проживающих на момент времени, список номеров, прейскурант услуг).

2. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Гостиница - это дом с меблированными комнатами для кратковременного проживания туристов, предназначенный и оборудованный для предоставления гостям комплекса услуг, связанных с путешествием, размещения, проживания, информационного и бытового обслуживания, активного отдыха и др.

Услуги размещения, питания и прочие услуги, предоставляемые гостиничным предприятием, дополняют друг друга, в большинстве случаев являются взаимозависимыми и воспринимаются гостем как единое целое. С учётом того, как они оформлены и скомбинированы в единый комплекс, складывается определённый тип предприятия.

Задачей администратора гостиницы является отслеживание финансовой стороны работы гостиницы. Его деятельность организована следующим образом: гостиница предоставляет номера клиентам на определённый срок. Гостиница хранит информацию о номерах в гостинице, которые заняты клиентами. Клиентами гостиницы являются различные лица, о которых администратор собирает определённую информацию (фамилия, имя). Сдача номера клиенту производится при наличии свободных мест в номерах. При выезде из гостиницы для каждого клиента формируется счёт за оказанные ему услуги.

1. СИСТЕМА КЛАССОВ

В данном разделе представлена структура классов проекта и их взаимодействие.

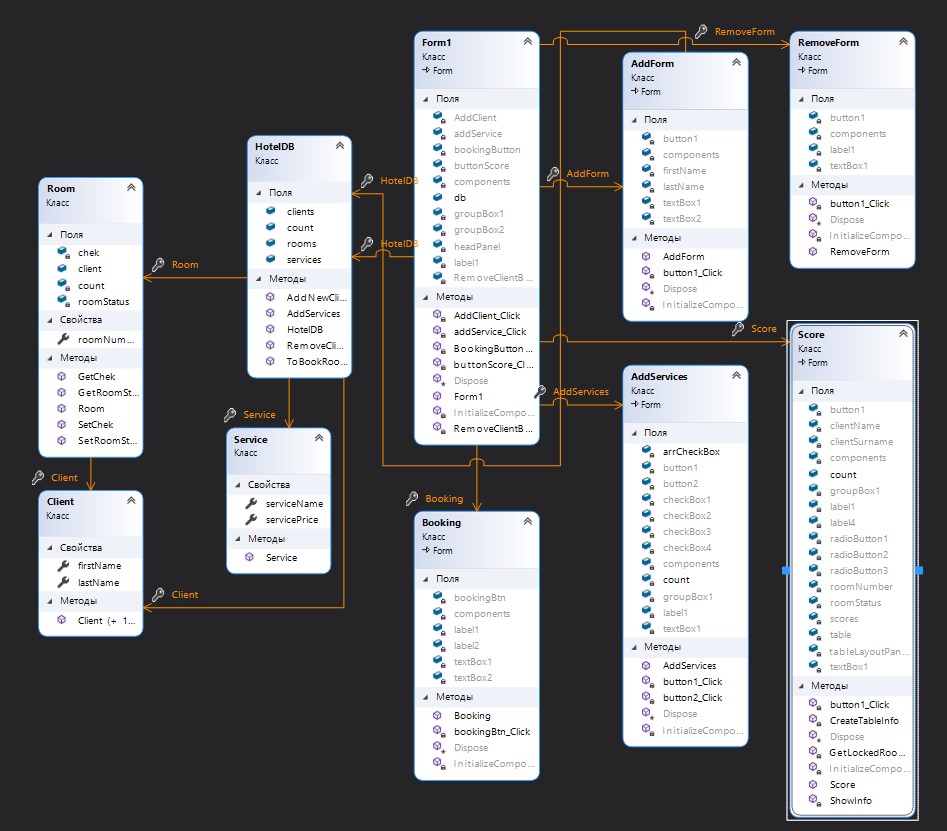
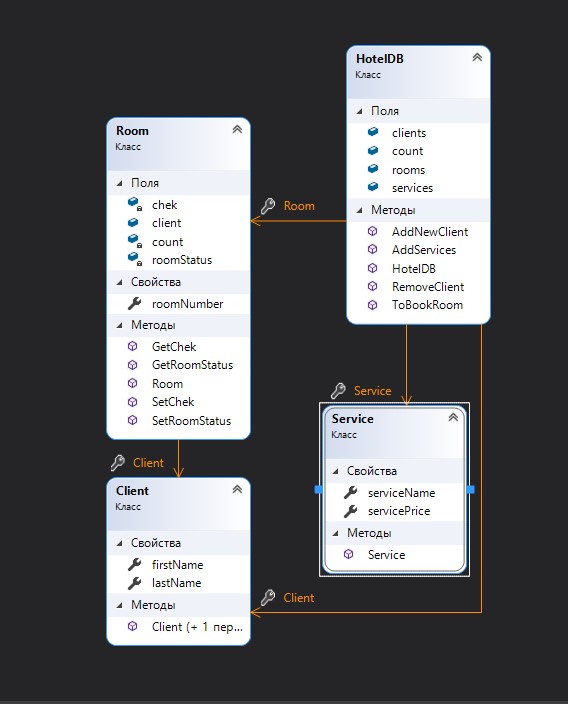


Диаграмма классов программы.



(рис. 1) Диаграмма классов структуры базы данных

1. ВЫБОР И ОБОСНОВАНИЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ

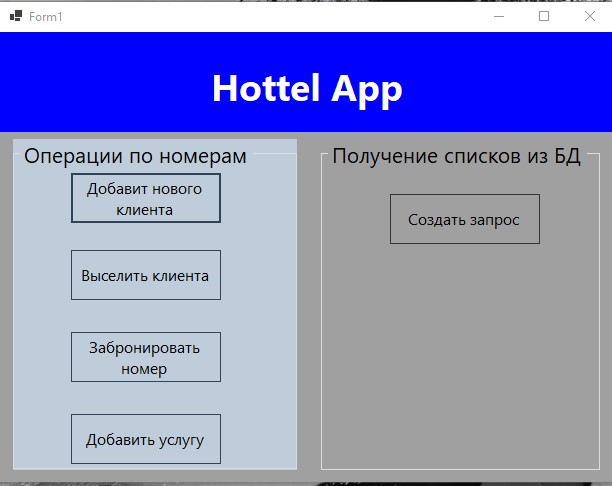
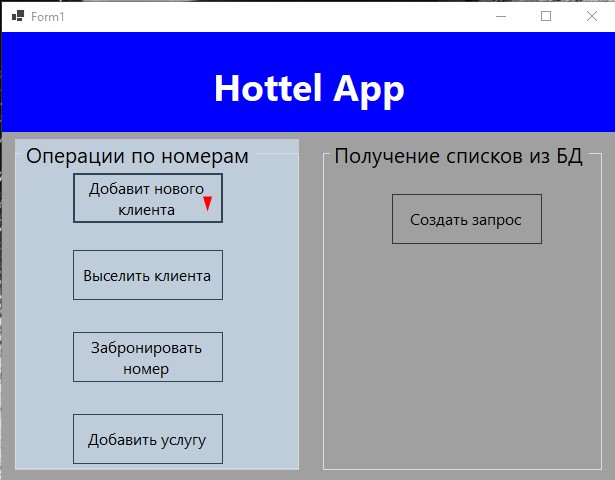
Для выполнения данной работы мною был выбран язык программирования C# (си шарп), так как на мой взгляд на сегодняшний день язык программирования C# один из самых мощных, быстро развивающихся и востребованных языков в ИТ-отрасли. В настоящий момент на нем пишутся самые различные приложения: от небольших десктопных программ до крупных веб-порталов и веб-сервисов, обслуживающих ежедневно миллионы пользователей.

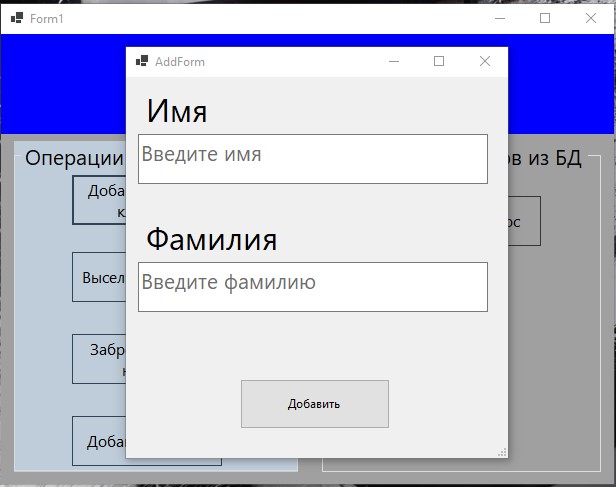
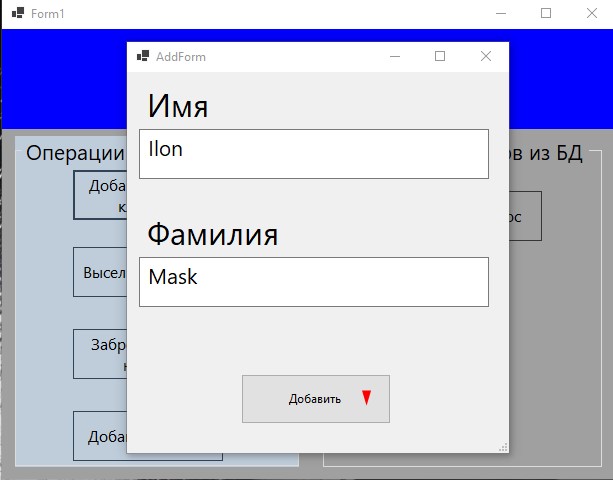
C# уже не молодой язык и как и вся платформа .NET уже прошел большой путь. Первая версия языка вышла вместе с релизом Microsoft Visual Studio .NET в феврале 2002 года. Текущей версией языка является версия C# 9.0, которая вышла 10 ноября 2020 года вместе с релизом .NET 5.

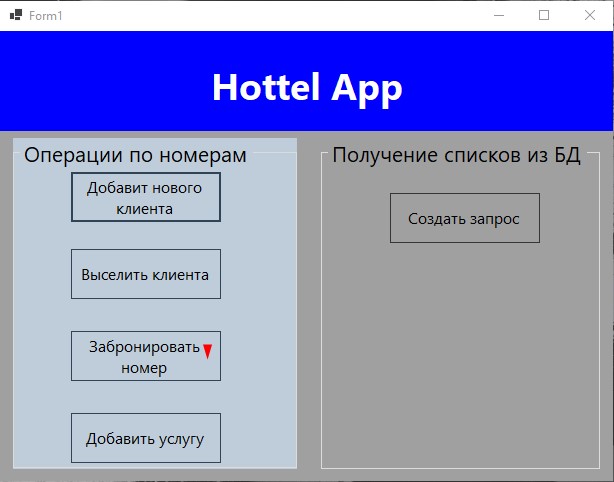
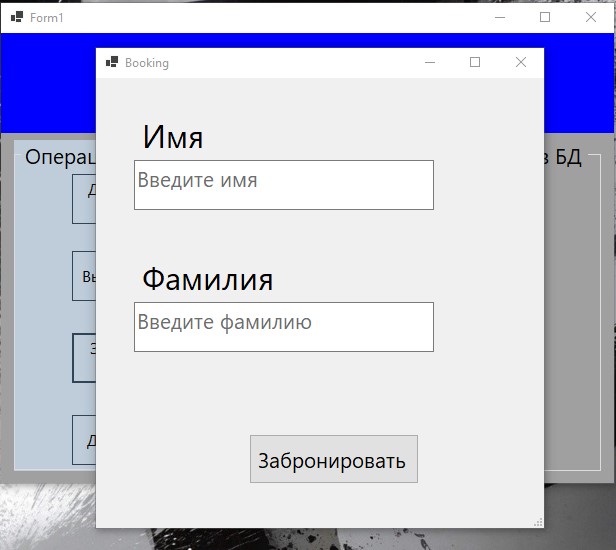
C# является объектно-ориентированным и поддерживает полиморфизм, наследование, перегрузку операторов, статическую типизацию. Объектно-ориентированный подход позволяет решить задачи по построению крупных, но в тоже время гибких, масштабируемых и расширяемых приложений. И C# продолжает активно развиваться, и с каждой новой версией появляется все больше интересных функциональностей.

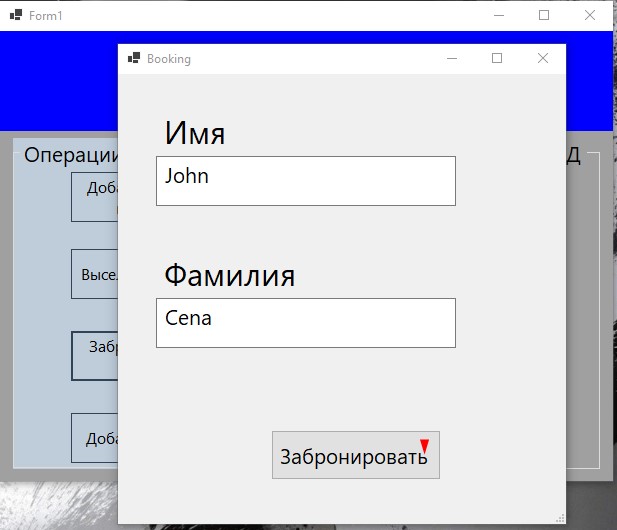
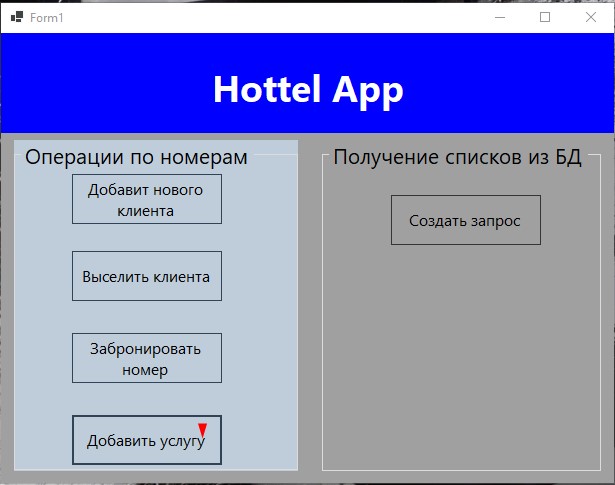
Задание было решено выполнить в виде проекта Windows Forms. Windows Forms — [интерфейс программирования приложений](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9" \o "Интерфейс программирования приложений) (API), отвечающий за [графический интерфейс пользователя](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F" \o "Графический интерфейс пользователя) и являющийся частью [Microsoft](https://ru.wikipedia.org/wiki/Microsoft" \o "Microsoft) [.NET Framework](https://ru.wikipedia.org/wiki/.NET_Framework" \o ".NET Framework). Данный интерфейс упрощает доступ к элементам интерфейса [Microsoft Windows](https://ru.wikipedia.org/wiki/Microsoft_Windows" \o "Microsoft Windows) за счет создания обёртки для существующего [Win32 API](https://ru.wikipedia.org/wiki/Windows_API" \o "Windows API) в [управляемом коде](https://ru.wikipedia.org/wiki/%D0%A3%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D1%8F%D0%B5%D0%BC%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4" \o "Управляемый код). Причём управляемый код — классы, реализующие [API](https://ru.wikipedia.org/wiki/API" \o "API) для Windows Forms, не зависят от языка разработки.

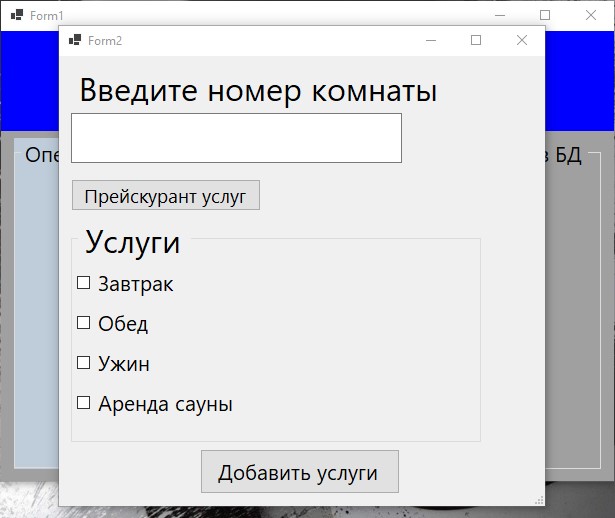
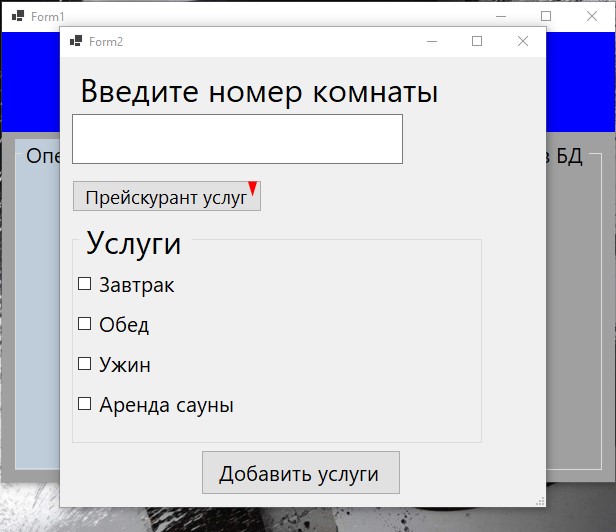
6. КОНТРОЛЬНЫЙ ПРИМЕР

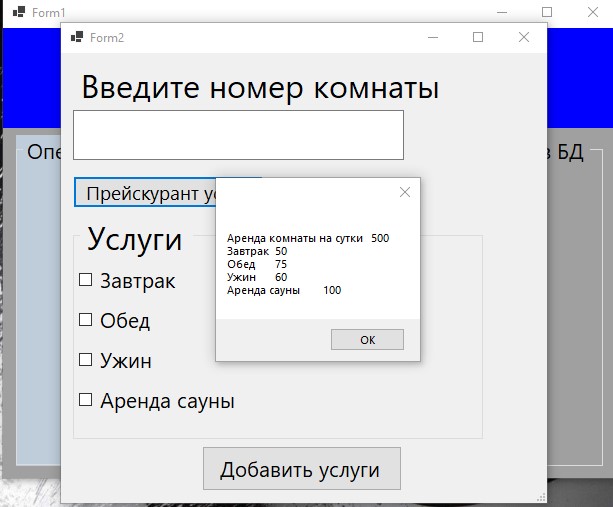
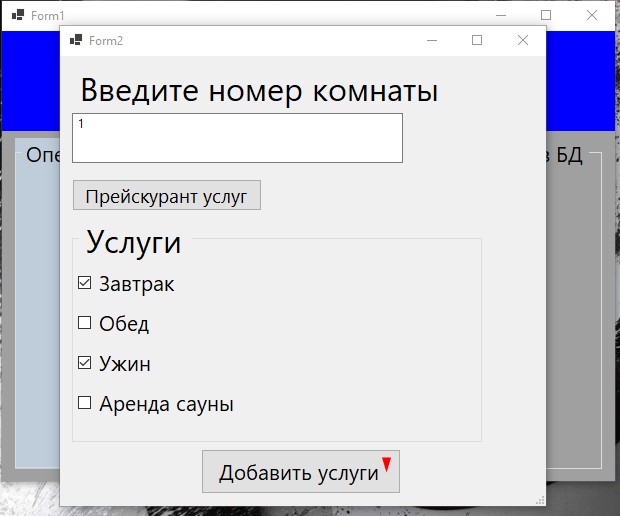
 

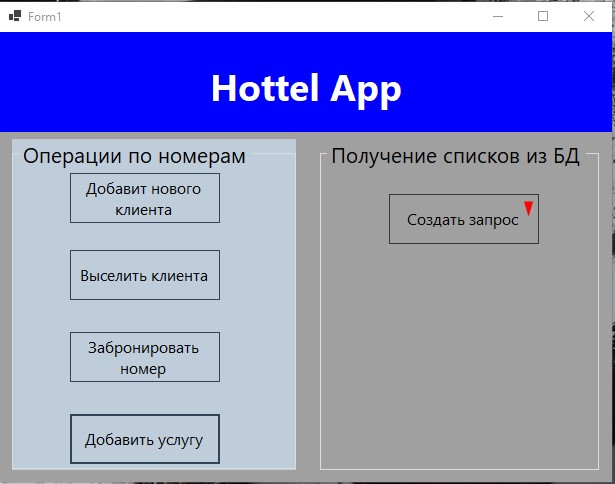
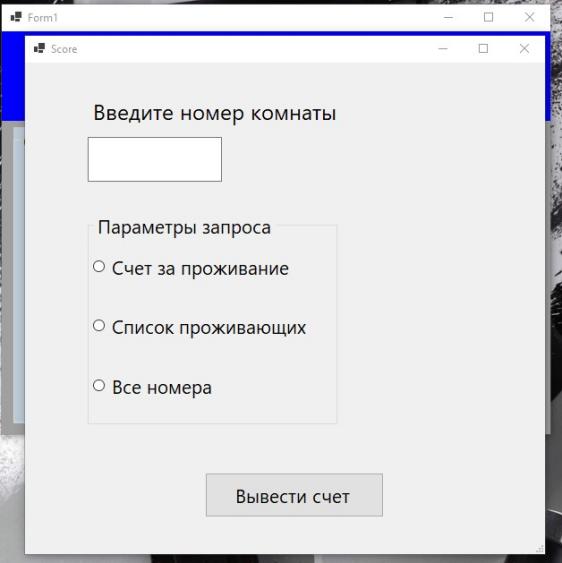
 

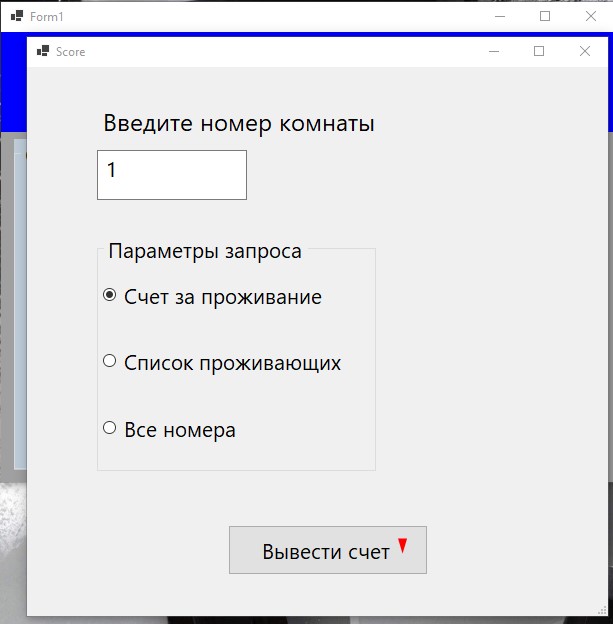
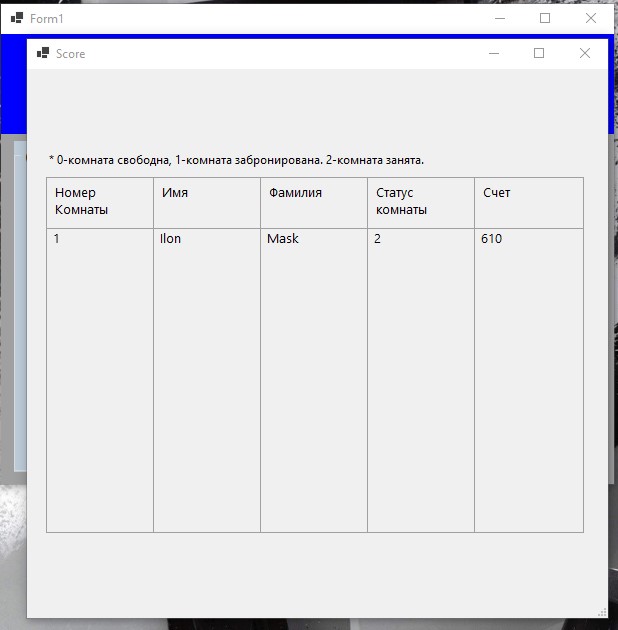
 

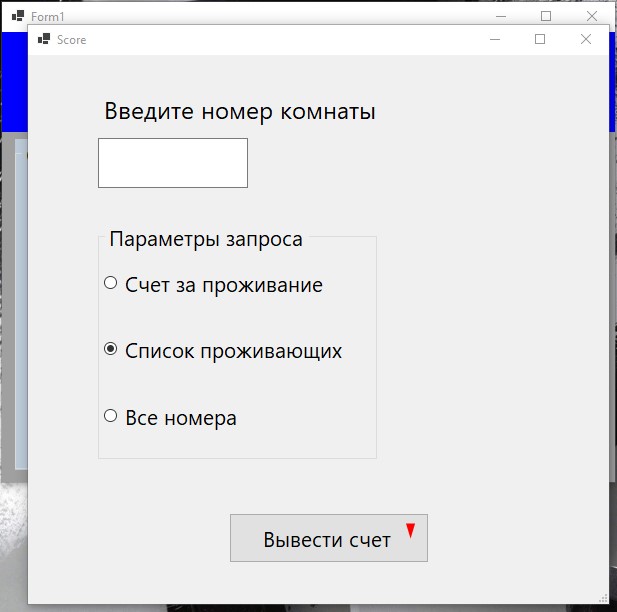
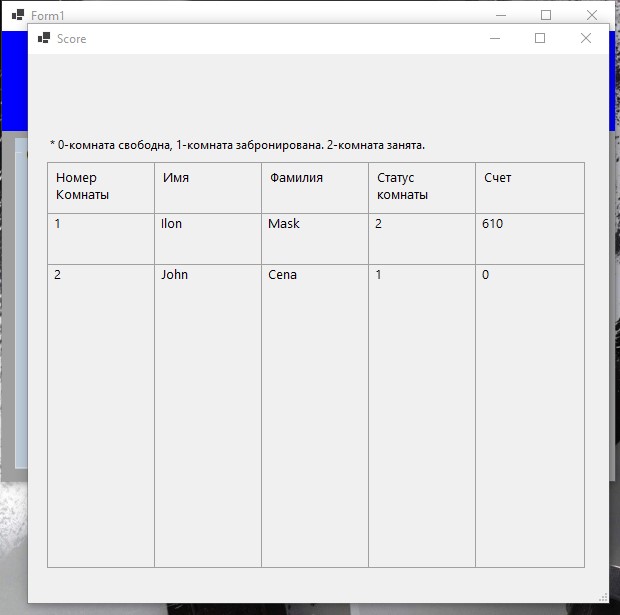
 

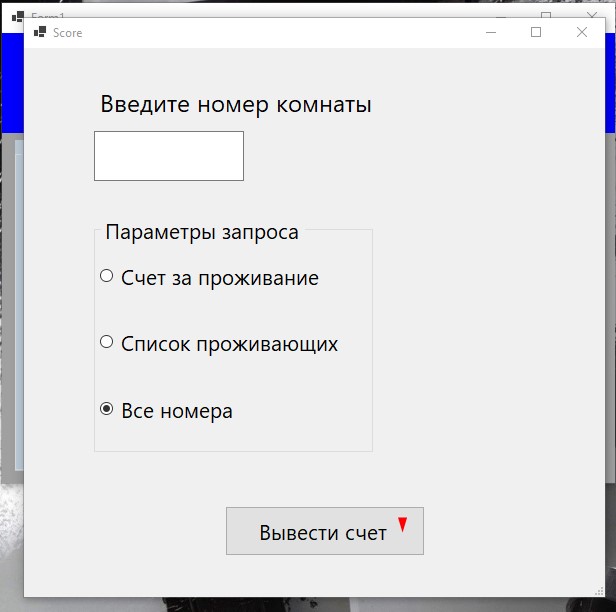
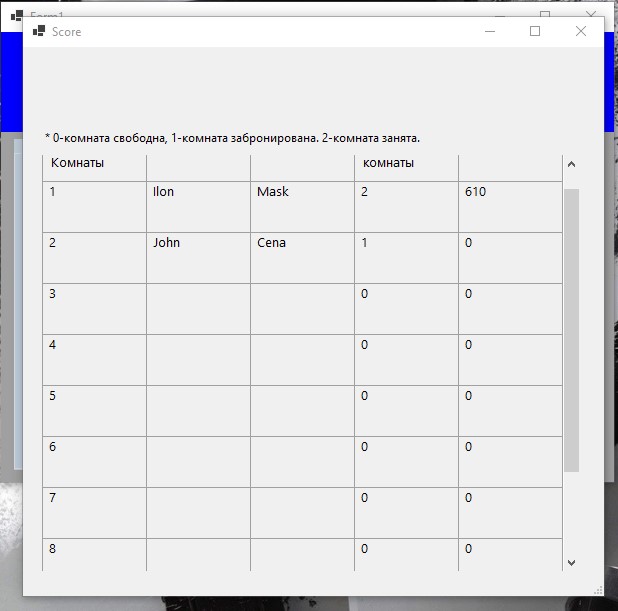
 

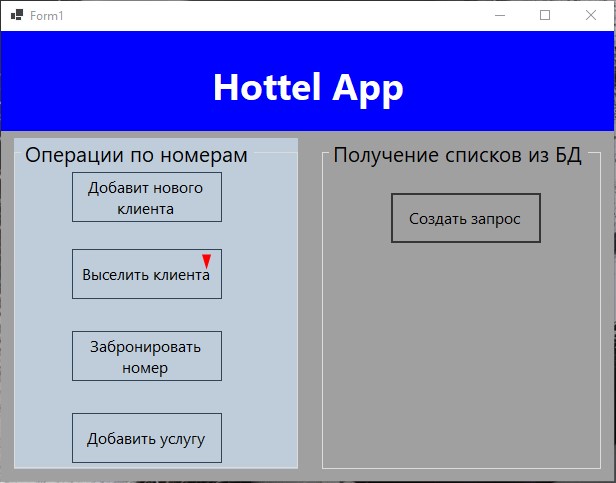
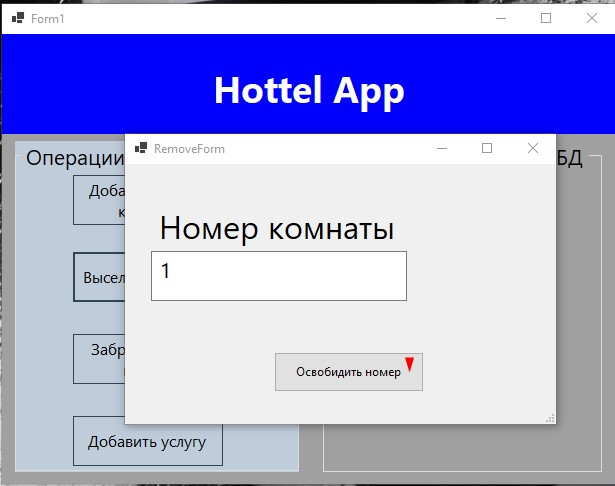


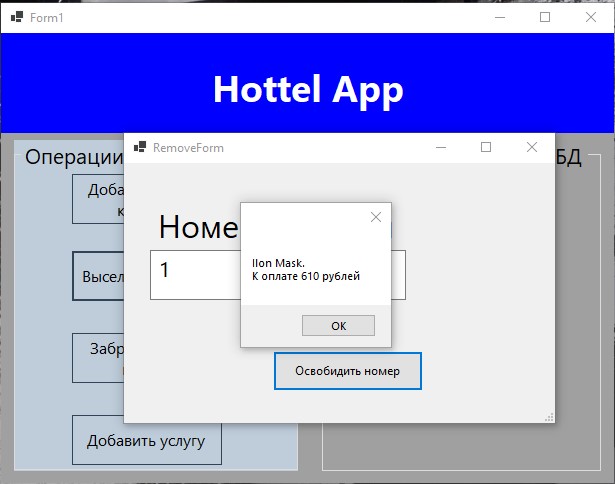
 



7. КОД ПРОГРАММЫ

Класс Client

using System;

using System.Collections.Generic;

using System.Text;

namespace HotelApp

{

public class Client

{

public string firstName { get; private set; }

public string lastName { get; private set; }

public Client() { }

public Client(string firstName = null, string lastName = null)

{

this.firstName = firstName;

this.lastName = lastName;

}

}

}

Класс Room

using System;

using System.Collections.Generic;

using System.Text;

namespace HotelApp

{

public class Room

{

public int roomNumber { get; private set; }

private int roomStatus;

private int chek;

public Client client;

static int count = 1;

public Room()

{

this.roomNumber = count;

this.roomStatus = 0;

this.client = null;

this.chek = 0;

count++;

}

public int GetRoomStatus()

{

return roomStatus;

}

public void SetRoomStatus(int status)

{

this.roomStatus = status;

}

public int GetChek()

{

return chek;

}

public void SetChek(int chek)

{

this.chek = chek;

}

}

}

Класс Service

using System;

using System.Collections.Generic;

using System.Text;

namespace HotelApp

{

public class Service

{

public string serviceName { get; private set; }

public int servicePrice { get; private set; }

public Service(string serviceName, int servicePrice)

{

this.serviceName = serviceName;

this.servicePrice = servicePrice;

}

}

}

Класс HotelDB

using System;

using System.Collections.Generic;

using System.Text;

namespace HotelApp

{

public class HotelDB

{

public Client[] clients;

public Room[] rooms;

public Service[] services;

public int count;

public HotelDB()

{

clients = new Client[1];

rooms = new Room[10];

count = 0;

for (int i = 0; i < rooms.Length; i++)

{

rooms[i] = new Room();

}

services = new Service[]

{

new Service("Аренда комнаты на сутки", 500),

new Service("Завтрак", 50),

new Service("Обед", 75),

new Service("Ужин", 60),

new Service("Аренда сауны", 100)

};

}

public void AddNewClient(string firstName, string lastName)

{

Client newClient = new Client(firstName, lastName);

Array.Resize(ref this.clients, this.clients.Length + 1);

this.clients[this.clients.Length -1] = newClient;

foreach(var room in rooms)

{

if(room.GetRoomStatus() != 2)

{

if(room.client != null && firstName == room.client.firstName && lastName == room.client.lastName)

{

room.SetRoomStatus(2);

int chek = room.GetChek();

room.SetChek(chek += services[0].servicePrice);

}

else if (room.GetRoomStatus() == 0)

{

room.client = newClient;

room.SetRoomStatus(2);

int chek = room.GetChek();

room.SetChek(chek += services[0].servicePrice);

count++;

}

break;

}

}

}

public void RemoveClient(int roomNum)

{

foreach (var room in rooms)

{

if (room.roomNumber == roomNum)

{

room.client = null;

room.SetRoomStatus(0);

room.SetChek(0);

count--;

break;

}

}

}

public void ToBookRoom(string firstName, string lastName)

{

Client newClient = new Client(firstName, lastName);

Array.Resize(ref this.clients, this.clients.Length + 1);

this.clients[this.clients.Length - 1] = newClient;

foreach (var room in rooms)

{

if (room.GetRoomStatus() == 0)

{

room.client = newClient;

room.SetRoomStatus(1);

count++;

break;

}

}

}

public void AddServices(string[] arr, int room)

{

for(int i = 0; i < services.Length; i++)

{

for(int j = 0; j < arr.Length; j++)

{

if(arr[j] == services[i].serviceName)

{

int chek = rooms[room - 1].GetChek();

rooms[room-1].SetChek(chek + services[i].servicePrice);

}

}

}

}

}

}

Класс Form1

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 HotelApp

{

public partial class Form1 : Form

{

public static HotelDB db;

public Form1()

{

InitializeComponent();

db = new HotelDB();

}

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

{

AddForm addForm = new AddForm();

addForm.ShowDialog();

}

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

{

Booking bookingForm = new Booking();

bookingForm.ShowDialog();

}

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

{

RemoveForm removeForm = new RemoveForm();

removeForm.ShowDialog();

}

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

{

AddServices addServices = new AddServices();

addServices.ShowDialog();

}

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

{

Score score = new Score();

score.ShowDialog();

}

}

}

Класс AddForm

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;

namespace HotelApp

{

public partial class AddForm : Form

{

public AddForm()

{

InitializeComponent();

}

public HotelDB HotelDB

{

get => default;

set

{

}

}

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

{

if(textBox1 == null || textBox2 == null)

{

throw new NullReferenceException();

}

if (textBox1.Text.Length != 0 && textBox2.Text.Length != 0)

{

if(Form1.db.count < 10)

{

Form1.db.AddNewClient(textBox1.Text, textBox2.Text);

this.Close();

}

else

{

MessageBox.Show("Все номера заняты!");

this.Close();

}

}

else

{

firstName.BackColor = Color.Red;

lastName.BackColor = Color.Red;

}

}

}

}