

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования

**ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Институт высоких технологий  
наименование института

Вычислительная техника  
наименование кафедры

Допускаю к защите

Руководитель:

Столбов А.Б.  
И.О. Фамилия



Обработка файлов на языке C#  
Наименование темы

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**  
к курсовому проекту по дисциплине  
Программирование на языке высокого уровня

1.016.00.00 ПЗ  
обозначение документа

Выполнил студент группы Истб-17-1  
шифр

  
подпись


Мускатин И.А.  
Фамилия И.О.

Нормоконтроль

  
подпись

Столбов А.Б.  
Фамилия И.О.

Курсовой проект защищен с оценкой:

Отлично 

Иркутск 2018 г.

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ЗАДАНИЕ

НА КУРСОВОЕ ПРОЕКТИРОВАНИЕ

По курсу Программирование на языке высокого уровня

Студенту Мускату Игорю Алексеевичу

Тема проекта Обработка файлов на языке C#

Исходные данные Вариант 2,1,2,1

Рекомендуемая литература

1. Сосинская С.С. «Обработка файлов»: Методические указания по выполнению курсового проекта» Электронный каталог кафедры.

Графическая часть на   -   листах.

Дата выдачи задания 09 / 11 /2018 г.

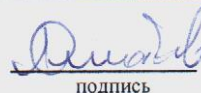
Задание получил студент

  
подпись

Мускатин И.А.  
Фамилия И.О.

Дата представления проекта руководителю 19 / 12 /2018 г.

Руководитель курсового проектирования

  
подпись

Столбов А.Б.  
Фамилия И.О.

## Содержание

Введение .....	4
1 Индивидуальное задание .....	5
2 Иерархия форм и функций с описанием их назначения .....	6
3 Алгоритмы основных событий .....	8
4 Формы .....	13
5 Коды основных событий .....	17
6 Таблица тестов .....	25
7 Результаты тестирования .....	27
Заключение .....	40
Список использованных источников .....	41

## Введение

Требуется на основании индивидуального задания разработать Windows-приложение на языке C#, таблицу тестов и протестировать программу для создания файла, добавления записи, удаления записи, изменения записи, просмотра файла и выполнения двух запросов. Удаление и изменение записи происходит по ее номеру. Способы выполнения действий – согласно индивидуальному заданию. При ошибочных ситуациях (отсутствие записи, отсутствие файла и т.д.) необходимо выдавать сообщения.

C# – простой, современный объектно-ориентированный и типобезопасный язык программирования. C# относится к широко известному семейству языков C, что делает его хорошим языком для обучения программированию.

Реализация данного задания происходит с помощью программы Visual Studio корпорации Майкрософт. Visual Studio поддерживает Visual C# с полнофункциональным редактором кода, компилятором, шаблонами проектов, конструкторами, мастерами кода, мощным и простым в использовании отладчиком и многими другими средствами.

На основе индивидуального задания необходимо разработать Windows-приложение на языке C#, которое соответствует требованиям работы.

## **1 Индивидуальное задание**

### Вариант 2-1-2-1

2- Файл **“Справочное агентство”** с информацией о наличии мест на рейсы (название города, N рейса, тип самолета, цена билета, время вылета, время прибытия, количество свободных мест).

1 - По заданному городу указать номера рейсов;

2 - По заданному городу указать количество свободных мест;

1 - Добавление в конец файла

## 2 Иерархия форм и функций с описанием их назначения

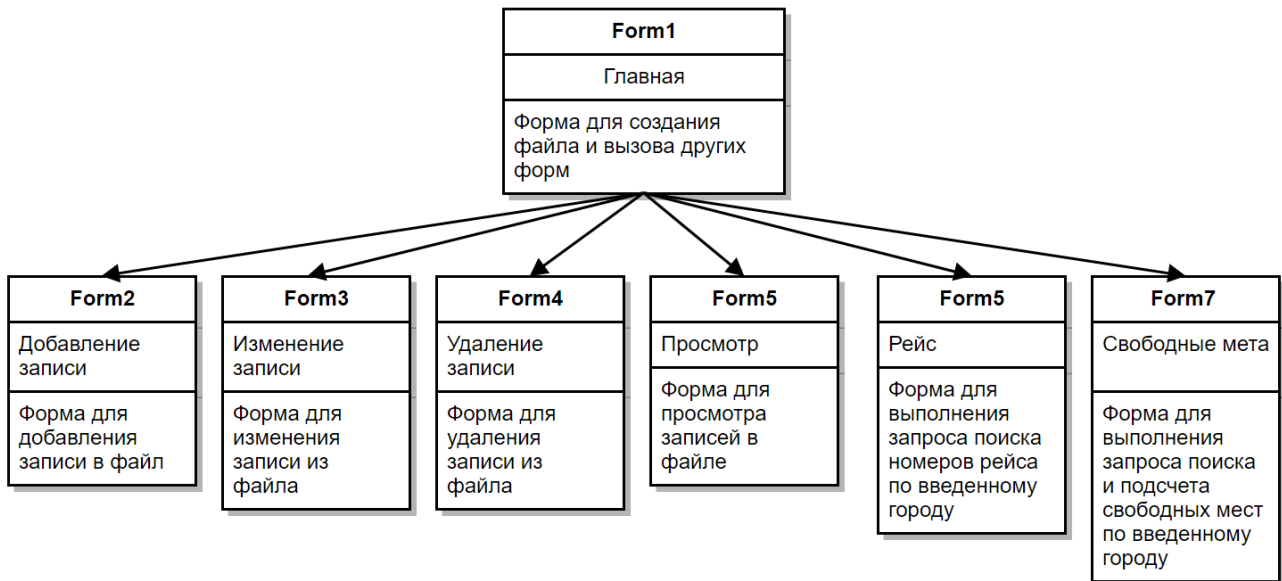


Схема 1 – Иерархия форм

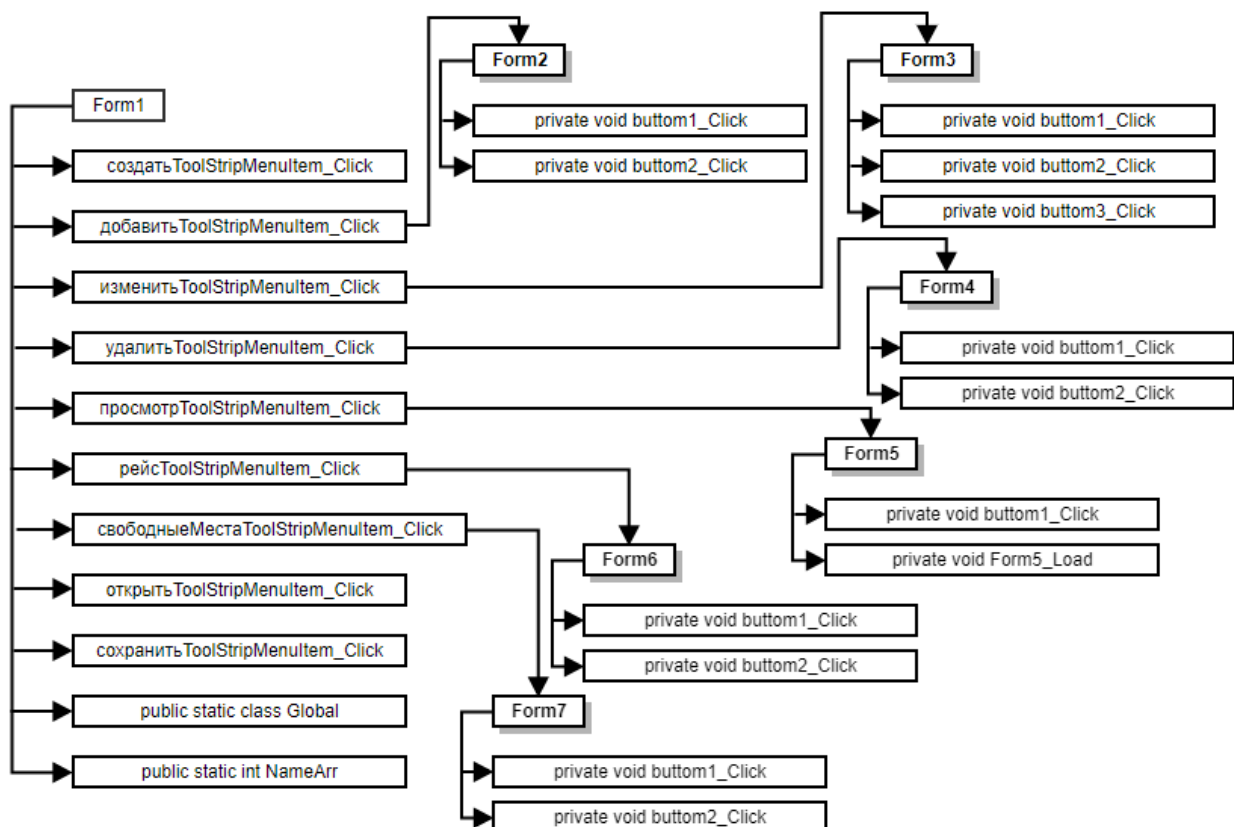


Схема 2 – Иерархия функций

Таблица 1 – Название и назначение функций

Название функции	Назначение
Form 1 (Главная форма)	
создатьToolStripMenuItem_Click	Создание нового файла
открытьToolStripMenuItem_Click	Открыть файл
сохранитьToolStripMenuItem_Click	Сохранить файла
добавитьToolStripMenuItem_Click	Вызов Form 2 для добавления записи к файлу
удалитьToolStripMenuItem_Click	Вызов Form 4 для удаления записи из файла
изменитьToolStripMenuItem_Click	Вызов Form 3 для изменения записи файла
просмотрToolStripMenuItem_Click	Вызов Form 5 для просмотра записей
рейсToolStripMenuItem_Click	Вызов Form 6 для выполнения 1 запроса
свободныеМестаToolStripMenuItem_Click	Вызов Form 7 для выполнения 2 запроса
public static class Global	Создание глобальных переменных
public static int NameArr	Чтение файла в массиве строк
Form 2 (Форма для добавления записей в файл)	
private void button1_Click	Добавление записи в файл
private void button2_Click	Заккрыть форму
Form 3(Форма для изменения записей в файл)	
private void button3_Click	Находит запись по введенному номеру
private void button1_Click	Изменение данных записи после ее корректировки
private void button2_Click	Заккрыть форму
Form 4 (Форма для удаления записей из файл)	
private void button1_Click	Удалить запись из файла по номеру
private void button2_Click	Заккрыть форму
Form 5 (Форма для просмотра записей из файл)	
private void button1_Click	Заккрыть форму
private void Form5_Load	Загрузка записей из файла в таблицу приложения
Form 6 (Форма для выполнения запроса: поиска номера рейса)	
private void button1_Click	Вывод списка рейсов по названию города
private void button2_Click	Заккрыть форму
Form 7 (Форма для выполнения запроса: поиска свободных мест)	
private void button1_Click	Вывод списка свободных номеров по названию города
private void button2_Click	Заккрыть форму

### 3 Алгоритмы основных событий

Функция NameArr

1. Открываем файл "Global.fileName" для чтения
2. N=0
3. ПОКА не дошли до конца файла ВЫПОЛНИТЬ
  - 3.1. NovArr[N]=чтение строки из файла
  - 3.2. Записываем в массив Arr данные строки NovArr[N]
  - 3.3. reis[N].City = Arr[0];
  - 3.4. student[N].Nreisa = Arr[1];
  - 3.5. Если (Arr[3] == «Ty-134») reis[N].Aircraft = Ty-134;
  - 3.6. Если (Arr[3] == «Ty-154») reis[N].Aircraft = Ty-154;
  - 3.7. Если (Arr[3] == «SSJ 100») reis[N].Aircraft = SSJ 100;
  - 3.8. Если (Arr[3] == «Airbus A310») reis[N].Aircraft = Airbus A310;
  - 3.9. Если (Arr[3] == «Airbus A330») reis[N].Aircraft = Airbus A330;
  - 3.10. Если (Arr[3] == «Boeing-747») reis[N].Aircraft = Boeing-747;
  - 3.11. reis[N].freep = Convert.ToInt32(Arr[6]);
  - 3.12. reis[N].Price = Convert.ToInt32(Arr[5]);
  - 3.13. reis[N].Tup = Arr[2];
  - 3.14. reis[N].Tdown = Arr[4];
  - 3.15. reis[N].Tupmin = Arr[7];
  - 3.16. reis[N].Tdownmin = Arr[8];
  - 3.17. N=N+1
- ПОКА ВСЕ
4. Закрываем файл "Global.fileName"
5. Возврат N

Выбор пункта меню создатьФайлToolStripMenuItem\_Click на форме Form1

1. ЕСЛИ файл "Reis.txt" существует, ТО
  - 1.1. Сообщение "Файл уже существует!"
- ИНАЧЕ
  - 1.2. Создаем новый файл "Reis.txt"
  - 1.3. Закрываем файл "Reis.txt"
  - 1.4. Сообщение "Файл создан!"
- ЕСЛИ ВСЕ

Событие нажатия кнопки Button1 на форме Form2(Добавление записи)

1. ЕСЛИ файл "Global.fileName" существует, ТО
  - 1.1. Вызов функции NameArr для получения R, NovArr[] и reis[]
  - 1.2. ЕСЛИ хотя бы одно поле ввода пустое, ТО
    - 1.2.1. Сообщение "Не все поля заполнены!"
  - ИНАЧЕ
    - 1.2.2. Создаем новый файл "Global.fileName", с добавлением новых записей в конец файла
    - 1.2.3. string text = "";



- 1.2.4. Если (radioButton1.Checked) text = radioButton1.Text;
- 1.2.5. Если (radioButton2.Checked) text = radioButton2.Text;
- 1.2.6. Если (radioButton3.Checked) text = radioButton3.Text;
- 1.2.7. Если (radioButton4.Checked) text = radioButton4.Text;
- 1.2.8. Если (radioButton5.Checked) text = radioButton5.Text;
- 1.2.9. Если (radioButton6.Checked) text = radioButton6.Text;
- 1.2.10. Записываем строку, содержащую значения глобальных переменных, в конец файла
- 1.2.11. Закрываем файл
- 1.2.12. Сообщение “Запись добавлена!”

ЕСЛИ ВСЕ

ИНАЧЕ

- 1.3. Сообщение “Файл не найден!”

ЕСЛИ ВСЕ

Событие нажатие кнопки Button3 на форме Form3(Поиск записи)

- 1. ЕСЛИ файл “Global.fileName” существует, ТО

- 1.1. ЕСЛИ поле ввода номера записи принимает пустое значение, ТО

- 1.1.1. Сообщение “Введите номер записи”

ИНАЧЕ

- 1.1.2. N = преобразованное в целое значение поля ввода для номера записи

- 1.1.3. Вызов функции NameArr для получения R, RArr[] и reis[]

- 1.1.4. ЕСЛИ N>0 и N-1<R, ТО

- 1.1.4.1. Делаем доступными элементы управления

- 1.1.4.2. Записываем строки из структуры reis в поля ввода

ИНАЧЕ

- 1.1.4.3. Сообщение “Запись не найдена!”

- 1.1.4.4. Делаем недоступными элементы управления

ЕСЛИ ВСЕ

ЕСЛИ ВСЕ

ИНАЧЕ

- 1.2. Сообщение “Файл не найден!”

ЕСЛИ ВСЕ

Событие нажатия кнопки Button1 на форме Form3(Изменение записи)

- 1. ЕСЛИ хотя бы одно поле ввода пустое, ТО

- 1.1. Сообщение “Не все поля заполнены!”

ИНАЧЕ

- 1.2. Создаем новый файл “Global.fileName”

- 1.3. string text = "";

- 1.4. Если (radioButton1.Checked) text = radioButton1.Text;

- 1.4. Если (radioButton2.Checked) text = radioButton2.Text;

- 1.5. Если (radioButton3.Checked) text = radioButton3.Text;

- 1.6. Если (radioButton4.Checked) text = radioButton4.Text;
- 1.7. Если (radioButton5.Checked) text = radioButton5.Text;
- 1.8. Если (radioButton6.Checked) text = radioButton6.Text;
- 1.9. i=0
- 1.10. ПОКА i<R ВЫПОЛНИТЬ
  - 1.10.1. ЕСЛИ i=N-1, ТО
    - 1.10.1.1. Записываем измененные значения глобальных переменных  
ИНАЧЕ
    - 1.10.1.2. Записываем без изменений в файл строку RArr[i]  
ЕСЛИ ВСЕ
  - 1.10.2. i=i+1  
ПОКА ВСЕ
- 1.11. Закрываем файл
- 1.12. Сообщение “Запись изменена”  
ЕСЛИ ВСЕ

Событие нажатия кнопки Button1 на форме Form4(Удаление записи)

1. ЕСЛИ файл “Global.fileName” существует, ТО
  - 1.1. ЕСЛИ поле ввода номера записи принимает пустое значение, ТО
    - 1.1.1. Сообщение “Введите номер записи”  
ИНАЧЕ
    - 1.1.2. N= преобразованное в целое значение поля ввода для номера записи
    - 1.1.3. Вызов функции NameArr для получения R, DelArr[] и reis[]
    - 1.1.4. ЕСЛИ N>0 и N-1<R, ТО
      - 1.1.4.1. Создаем новый файл “Global.fileName”
      - 1.1.4.2. i=0
      - 1.1.4.3. ПОКА i <R ВЫПОЛНИТЬ
        - 1.1.4.3.1.ЕСЛИ i ≠ N-1, ТО
          - 1.1.4.3.1.1. Записываем DelArr[i] в следующую строку файла  
ЕСЛИ ВСЕ
        - 1.1.4.3.2.i=i+1  
ПОКА ВСЕ
      - 1.1.4.4. Закрываем файл “Global.fileName”
      - 1.1.4.5. Сообщение “Запись удалена!”  
ИНАЧЕ
      - 1.1.4.6. Сообщение “Запись не найдена!”  
ЕСЛИ ВСЕ  
ЕСЛИ ВСЕ
  - ИНАЧЕ
  - 1.2. Сообщение “Файл не найден!”  
ЕСЛИ ВСЕ

Событие Load на форме Form5

1. ЕСЛИ файл “Global.fileName” существует, ТО

- 1.1. Вызов функции NameArr для получения R, Array[] и student[]
- 1.2. ЕСЛИ R=0, ТО
  - 1.2.1. Сообщение “Записей не найдено!”
  - 1.2.2. Заккрыть форму
  - ИНАЧЕ
  - 1.2.3. i=0
  - 1.2.4. ПОКА i<R ВЫПОЛНИТЬ
    - 1.2.4.1. Добавляем строку в таблицу
    - 1.2.4.2. Заполняем строку таблицы соответствующими данными структуры reis
    - 1.2.4.3. i=i+1
  - ПОКА ВСЕ
  - ЕСЛИ ВСЕ
  - ИНАЧЕ
- 1.3. Сообщение “Файл не найден!”
- 1.4. Заккрыть форму
- ЕСЛИ ВСЕ

Событие нажатия кнопки Button1 на форме Form6(Запрос 1)

1. ЕСЛИ файл “Global.fileName” существует, ТО
  - 1.1. ЕСЛИ поле ввода maskedTextBox1 принимает пустое значение, ТО
    - 1.1.1. Сообщение “Введите название города!”
    - ИНАЧЕ
    - 1.1.2. Очищаем listBox1
    - 1.1.3. Вызов функции Z1Arr для получения R, Z1Arr [] и reis[]
    - 1.1.4. i=0
    - 1.1.5. ПОКА i<R ВЫПОЛНИТЬ
      - 1.1.5.1. ЕСЛИ reis[i].City = maskedTextBox1 .Text, ТО
        - 1.1.5.1.1. Добавляем reis[i].Nreisa в listBox1
      - ЕСЛИ ВСЕ
      - 1.1.5.2. i=i+1
    - ПОКА ВСЕ
    - 1.1.6. ЕСЛИ listBox1 пустой, ТО
      - 1.1.6.1. Сообщение “Записей по запросу не найдено”
      - ЕСЛИ ВСЕ
      - ЕСЛИ ВСЕ
  - ЕСЛИ ВСЕ

Событие нажатия кнопки Button1 на форме Form7(Запрос 2)

1. ЕСЛИ файл “Global.fileName” существует, ТО
  - 1.1. ЕСЛИ поле ввода maskedTextBox1 принимает пустое значение, ТО
    - 1.1.1. Сообщение “Введите название города!”
    - ИНАЧЕ
    - 1.1.2. Очищаем listBox1
    - 1.1.3. Вызов функции Z2Arr для получения R, Z2Arr [] и reis[]
    - 1.1.4. i=0, FR=0
    - 1.1.5. ПОКА i<R ВЫПОЛНИТЬ

1.1.5.1. ЕСЛИ  $\text{reis}[i].\text{City} = \text{maskedTextBox1}.\text{Text}$ , ТО  
1.1.5.1.1. Выполняем  $\text{FR} = \text{reis}[i].\text{freep} + \text{FP}$   
ЕСЛИ ВСЕ  
1.1.5.2.  $i = i + 1$   
ПОКА ВСЕ  
1.1.6. Добавляем FR в listBox1  
1.1.7. ЕСЛИ listBox1 пустой, ТО  
1.1.7.1. Сообщение “Записей по запросу не найдено”  
ЕСЛИ ВСЕ  
ЕСЛИ ВСЕ  
ЕСЛИ ВСЕ

## 4 Формы

В этом разделе показан внешний вид всех форм.

### 1. Главная форма (Рис.1)

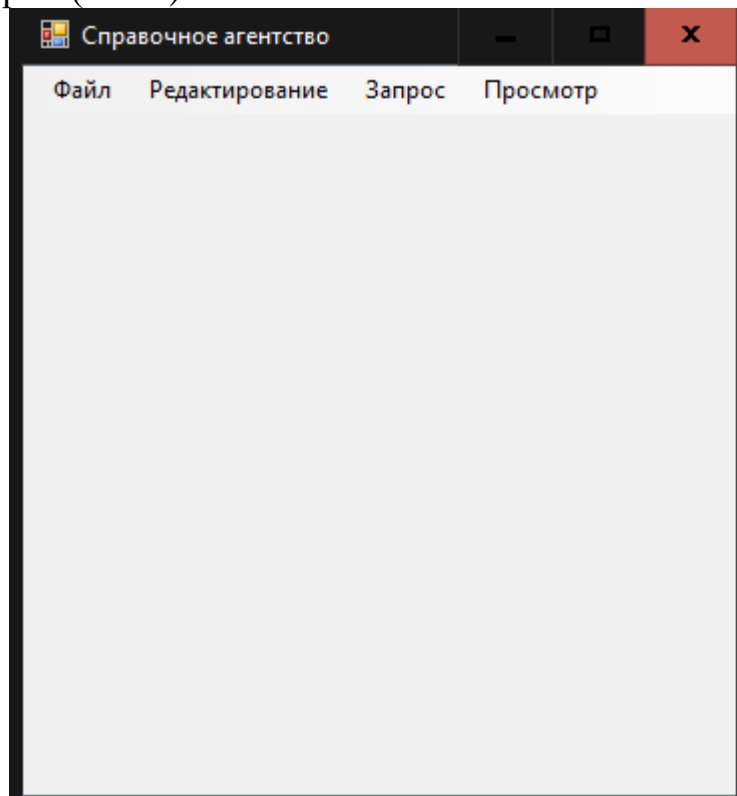


Рисунок 1 – Главная форма

### 2. Форма “Добавление записи” (Рис.2)

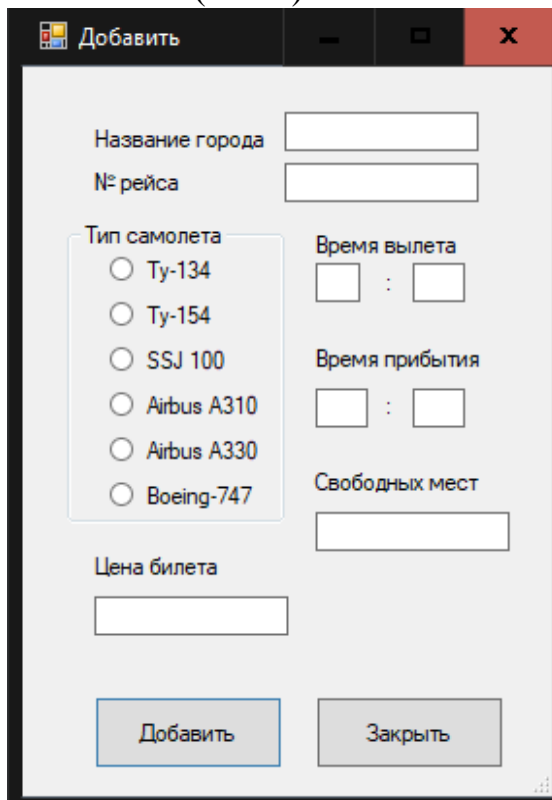


Рисунок 2 – Форма добавления записи

### 3. Форма “Form2” (Рис.3)

The screenshot shows a Windows-style window titled "Изменить" (Edit). It contains several input fields and buttons. At the top, there is a "№ записи" (Record number) field and a "Поиск" (Search) button. Below this are fields for "№ рейса" (Flight number) and "Название города" (City name). A group box labeled "Тип самолета" (Airplane type) contains six radio button options: "Ту-134", "Ту-154", "SSJ 100", "Airbus A310", "Airbus A330", and "Boeing-747". To the right of this group are two time selection fields: "Время вылета" (Departure time) and "Время прибытия" (Arrival time), each consisting of two boxes separated by a colon. Below these is a "Свободных мест" (Available seats) field. At the bottom left is a "Цена билета" (Ticket price) field. At the bottom right are two buttons: "Изменить" (Edit) and "Отмена" (Cancel).

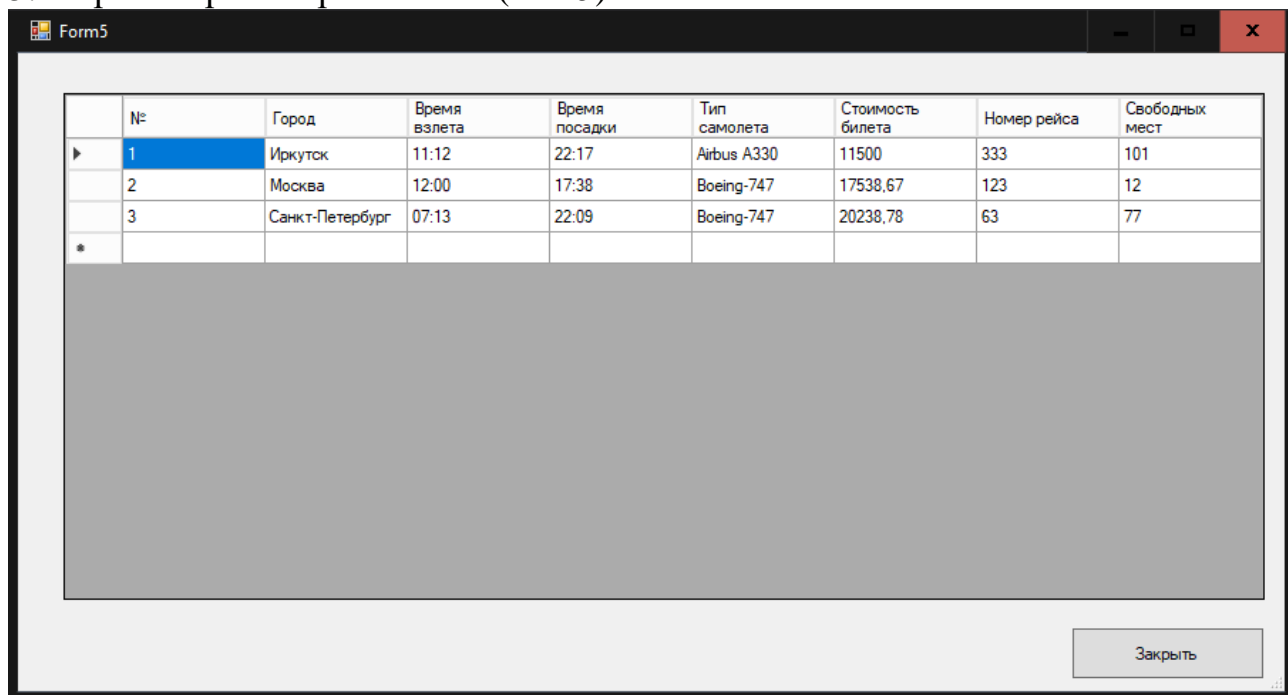
Рисунок 3 – Форма редактирование записи

### 4. Форма “Удаление записи” (Рис.4)

The screenshot shows a Windows-style window titled "Справочное агентство" (Reference Agency) with a menu bar containing "Файл", "Редактирование", "Запрос", and "Просмотр". Inside the window is a smaller window titled "Удаление" (Delete). This inner window has a label "Введите № записи" (Enter record number) followed by an input field. At the bottom of the inner window are two buttons: "Удалить" (Delete) and "Отмена" (Cancel).

Рисунок 4 – Форма удаления записи

## 5. Форма “Просмотр записей” (Рис.5)

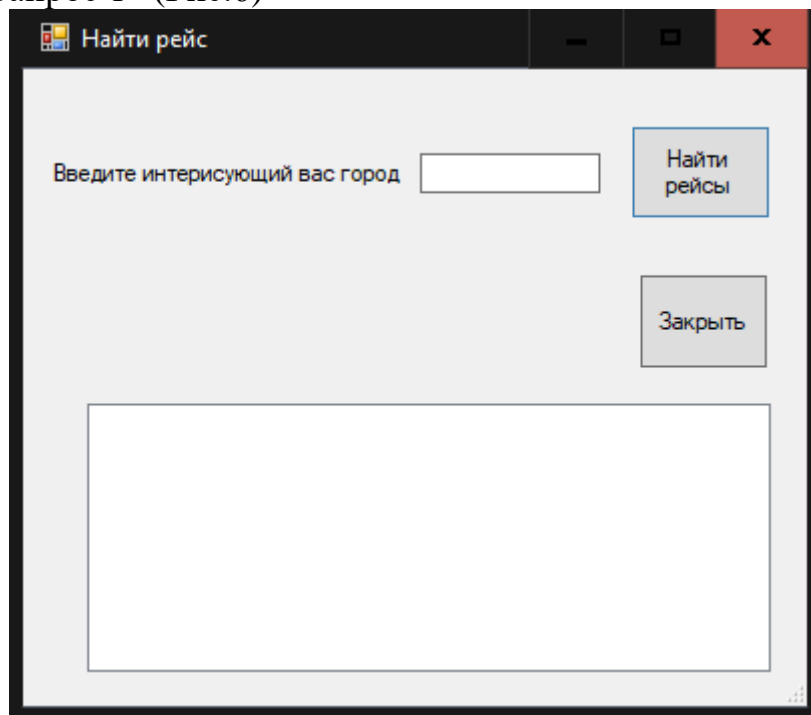


The screenshot shows a window titled "Form5" with a table of flight records. The table has 9 columns: an empty column, "№", "Город", "Время взлета", "Время посадки", "Тип самолета", "Стоимость билета", "Номер рейса", and "Свободных мест". The first three rows are populated with data, and the fourth row is marked with an asterisk. Below the table is a large gray rectangular area, and a "Заккрыть" button is in the bottom right corner.

	№	Город	Время взлета	Время посадки	Тип самолета	Стоимость билета	Номер рейса	Свободных мест
▶	1	Иркутск	11:12	22:17	Airbus A330	11500	333	101
	2	Москва	12:00	17:38	Boeing-747	17538,67	123	12
	3	Санкт-Петербург	07:13	22:09	Boeing-747	20238,78	63	77
*								

Рисунок 5 – Форма просмотра записи

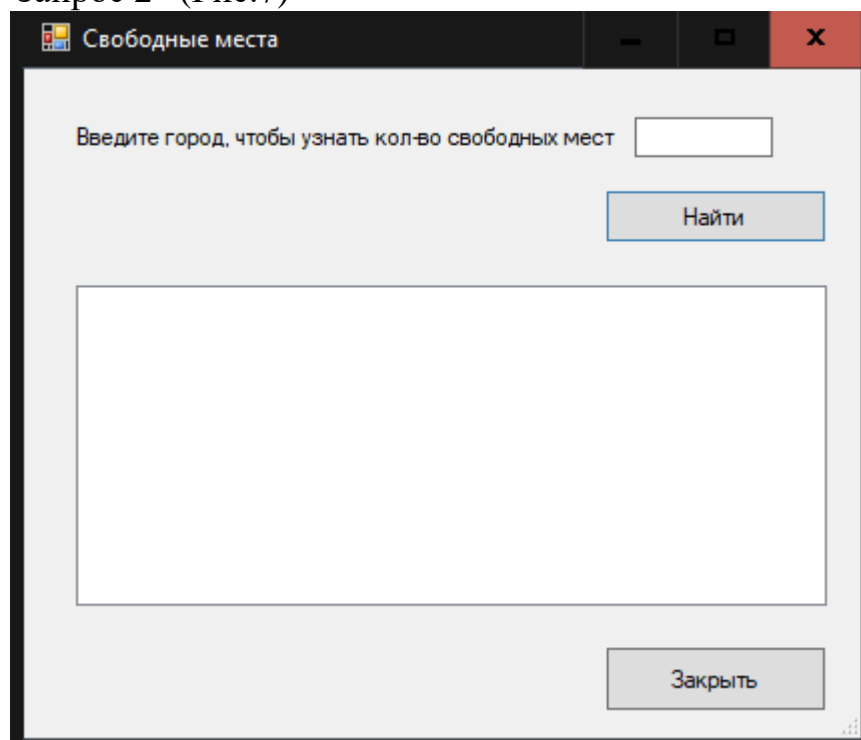
## 6. Форма “Запрос 1” (Рис.6)



The screenshot shows a window titled "Найти рейс". It contains a text label "Введите интересующий вас город" followed by a text input field. To the right of the input field is a blue button labeled "Найти рейсы". Below these is a gray button labeled "Заккрыть". At the bottom of the window is a large empty rectangular box.

Рисунок 6 – Форма запроса 1

7. Форма “Запрос 2” (Рис.7)



Свободные места

Введите город, чтобы узнать кол-во свободных мест

Найти

Закрыть

Рисунок 7 – Форма запроса 2



## 5 Коды основных событий

### Функция NameArr

```
public static int NameArr(out string[] NovArr, out Reis[] reis)
{
    NovArr = new string[100];
    reis = new Reis[100];
    file = new FileStream(Global.fileName, FileMode.Open);
    Global.SR = new StreamReader(Global.file);
    int N = 0; // проверка кол-во строк в файле
    string[] Arr;
    while (!SR.EndOfStream)
    {
        NovArr[N] = SR.ReadLine();
        Arr = NovArr[N].Split('|');

        reis[N].City = Arr[0];
        reis[N].Nreisa = Arr[1];
        if (Arr[3] == "Ty-134") reis[N].Aircraft = "Ty-134";
        if (Arr[3] == "Ty-154") reis[N].Aircraft = "Ty-154";
        if (Arr[3] == "SSJ 100") reis[N].Aircraft = "SSJ 100";
        if (Arr[3] == "Airbus A310") reis[N].Aircraft = "Airbus A310";
        if (Arr[3] == "Airbus A330") reis[N].Aircraft = "Airbus A330";
        if (Arr[3] == "Boeing-747") reis[N].Aircraft = "Boeing-747";
        reis[N].freep = Convert.ToInt32(Arr[6]);
        reis[N].Price = Convert.ToDouble(Arr[5]);
        reis[N].Tup = Arr[2];
        reis[N].Tupmin = Arr[7];
        reis[N].Tdown = Arr[4];
        reis[N].Tdownmin = Arr[8];
        N++;
    }
    SR.Close();
    return N;
}
```

### Функция OpenFile

```
private void OpenFile()
{
    try
    {
        textBox1.Clear();
        textBox1.Text = File.ReadAllText(Global.fileName);
    }
    catch (IOException ex)
    {
        MessageBox.Show(ex.Message, "Simple Editor",
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
    }
}
```

### Функция SaveFile

```
private void SaveFile()
{
    try
    { File.WriteAllText(Global.fileName, textBox1.Text); }
    catch (IOException ex)
    {
        MessageBox.Show(ex.Message, "Simple Editor",
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
    }
}
```

Выбор пункта меню создать ToolStripMenuItem\_Click на форме Form1

```
private void создатьToolStripMenuItem_Click(object sender, EventArgs e)
{
    {
        {
            if (File.Exists("Reis.txt"))
            {
                MessageBox.Show("Файл уже существует!");
            }
            else
            {
                Global.file = new FileStream("Reis.txt", FileMode.Create);
                Global.fileName = "Reis.txt";
                Global.file.Close();
                MessageBox.Show("Файл создан!");
            }
        }
    }
}
```

Событие нажатия кнопки сохранить ToolStripMenuItem\_Click на форме Form1

```
private void сохранитьToolStripMenuItem_Click(object sender, EventArgs e)
{
    {
        if (dlgSaveFile.ShowDialog() == DialogResult.OK)
        {
            Global.fileName = dlgSaveFile.FileName;
            SaveFile();
        }
    }
}
```

Событие нажатия кнопки создать ToolStripMenuItem\_Click на форме Form1

```
private void создатьToolStripMenuItem_Click(object sender, EventArgs e)
{
    {
        {
            if (File.Exists("Reis.txt"))
            {
                MessageBox.Show("Файл уже существует!");
            }
            else
            {
                Global.file = new FileStream("Reis.txt", FileMode.Create);
                Global.fileName = "Reis.txt";
                Global.file.Close();
                MessageBox.Show("Файл создан!");
            }
        }
    }
}
```

Все события нажатия кнопки “Заккрыть” выполняются аналогично форме Form2

```
private void button2_Click(object sender, EventArgs e)
{this.Close();}
```

## Событие нажатия кнопки Button1 на форме Form2

```
private void button1_Click(object sender, EventArgs e)
{
    {
        var NovArr = new string[100];
        var reis = new Global.Reis[100];
        if (File.Exists(Global.fileName))
        {
            int R = Global.NameArr(out NovArr, out reis);
            if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "" ||
textBox7.Text == "" || textBox8.Text == "" || textBox4.Text == "" || textBox5.Text == "" ||
textBox6.Text == "" || (!radioButton1.Checked && !radioButton2.Checked &&
!radioButton3.Checked && !radioButton4.Checked && !radioButton5.Checked &&
!radioButton6.Checked))
            {
                MessageBox.Show("Не все поля заполнены!");
                this.DialogResult = DialogResult.None;
                return;
            }

            else
            {
                Global.file = new FileStream(Global.fileName, FileMode.Append);

                Global.SW = new StreamWriter(Global.file);
                string text = "";
                if (radioButton1.Checked) text = radioButton1.Text;
                if (radioButton2.Checked) text = radioButton2.Text;
                if (radioButton3.Checked) text = radioButton3.Text;
                if (radioButton4.Checked) text = radioButton4.Text;
                if (radioButton5.Checked) text = radioButton5.Text;
                if (radioButton6.Checked) text = radioButton6.Text;
                Global.SW.WriteLine("{0}|{1}|{2}|{3}|{4}|{5}|{6}|{7}|{8}",
                    textBox1.Text, //город
                    textBox2.Text, //номер рейса
                    textBox3.Text, //время вылета (час)
                    text, //самолет
                    textBox4.Text, //время посадки (час)
                    textBox5.Text, //цена
                    textBox6.Text, //свободные места
                    textBox7.Text, //время вылета (мин)
                    textBox8.Text); //время посадки (мин)
                Global.SW.Close();
                MessageBox.Show("Запись добавлена!");
                this.DialogResult = DialogResult.None;
            }
        }
        else
        {
            MessageBox.Show("Файл не найден!");
        }
    }
}
```

## Событие нажатие кнопки Button1 на форме Form3

```
private void button1_Click(object sender, EventArgs e)
{
    var RArr = new string[100];
    var reis = new Global.Reis[100];
    if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "" ||
    textBox4.Text == "" || textBox5.Text == "" || textBox6.Text == "" || (!radioButton1.Checked
    && !radioButton2.Checked && !radioButton3.Checked && !radioButton4.Checked &&
    !radioButton5.Checked && !radioButton6.Checked))
    {
        MessageBox.Show("Не все поля заполнены!");
        this.DialogResult = DialogResult.None;
        return;
    }
    else
    {
        int N = Convert.ToInt32(maskedTextBox1.Text);
        int R = Global.NameArr(out RArr, out reis);
        Global.file = new FileStream(Global.fileName, FileMode.Create);
        Global.SW = new StreamWriter(Global.file);
        string text = "";
        if (radioButton1.Checked) text = radioButton1.Text;
        if (radioButton2.Checked) text = radioButton2.Text;
        if (radioButton3.Checked) text = radioButton3.Text;
        if (radioButton4.Checked) text = radioButton4.Text;
        if (radioButton5.Checked) text = radioButton5.Text;
        if (radioButton6.Checked) text = radioButton6.Text;
        for (int i = 0; i < R; i++)
        {
            if (i == N - 1)
            {
                Global.SW.WriteLine("{0}|{1}|{2}|{3}|{4}|{5}|{6}|{7}|{8}",
                textBox1.Text, //город
                textBox2.Text, //номер рейса
                textBox3.Text, //время вылета (час)
                text, //самолет
                textBox4.Text, //время посадки (час)
                textBox5.Text, //цена
                textBox6.Text, //свободные места
                textBox7.Text, //время вылета (мин)
                textBox8.Text); //время посадки (мин)
            }
            else
            {
                Global.SW.WriteLine(RArr[i]); // запись без изменений
            }
        }
        Global.SW.Close();
        MessageBox.Show("Запись изменена!");
    }
}
```

## Событие нажатия кнопки Button3 на форме Form3

```
private void button3_Click(object sender, EventArgs e)
{
    {
        var RArr = new string[100];
        var reis = new Global.Reis[100];
        if (File.Exists(Global.fileName))
        {
            if (maskedTextBox1.Text == "")
            {
                MessageBox.Show("Введите номер записи!");
            }
            else
            {
                int N = Convert.ToInt32(maskedTextBox1.Text);
                int R = Global.NameArr(out RArr, out reis);
                if (N > 0 && (N - 1) < R)
                {
                    textBox1.Enabled = true; textBox2.Enabled = true;
                    textBox3.Enabled = true; textBox4.Enabled = true;
                    textBox5.Enabled = true; textBox6.Enabled = true;
                    textBox7.Enabled = true; textBox8.Enabled = true;
                    radioButton1.Enabled = true; radioButton2.Enabled = true;
                    radioButton3.Enabled = true; radioButton4.Enabled = true;
                    radioButton5.Enabled = true; radioButton6.Enabled = true;
                    button1.Enabled = true; button2.Enabled = true;
                    textBox1.Text = reis[N - 1].City;
                    textBox2.Text = reis[N - 1].Nreisa;
                    textBox3.Text = reis[N - 1].Tup;
                    textBox7.Text = reis[N - 1].Tupmin;
                    textBox4.Text = reis[N - 1].Tdown;
                    textBox8.Text = reis[N - 1].Tdownmin;
                    textBox5.Text = Convert.ToString(reis[N - 1].Price);
                    textBox6.Text = Convert.ToString(reis[N - 1].freep);
                    if (reis[N - 1].Aircraft == "Ty-134") radioButton1.Checked = true;
                    if (reis[N - 1].Aircraft == "Ty-154") radioButton2.Checked = true;
                    if (reis[N - 1].Aircraft == "SSJ 100") radioButton3.Checked = true;
                    if (reis[N - 1].Aircraft == "Airbus A310") radioButton4.Checked = true;
                    if (reis[N - 1].Aircraft == "Airbus A330") radioButton5.Checked = true;
                    if (reis[N - 1].Aircraft == "Boeing-747") radioButton6.Checked = true;
                }
                else
                {
                    MessageBox.Show("Запись не найдена!");
                    textBox1.Enabled = false; textBox2.Enabled = false;
                    textBox3.Enabled = false; textBox4.Enabled = false;
                    textBox5.Enabled = false; textBox6.Enabled = false;
                    textBox7.Enabled = false; textBox8.Enabled = false;
                    radioButton1.Enabled = false;
                    radioButton2.Enabled = false;
                    radioButton3.Enabled = false;
                    radioButton4.Enabled = false;
                    radioButton5.Enabled = false;
                    radioButton6.Enabled = false;
                    button1.Enabled = false;
                }
            }
        }
    }
    else
    {
        MessageBox.Show("Файл не найден!");
    }
}
```

## Событие нажатия кнопки Button1 на форме Form4

```
private void button1_Click(object sender, EventArgs e)
{
    if (File.Exists(Global.fileName))
    {
        if (maskedTextBox1.Text == "")
        {
            MessageBox.Show("Введите номер записи!");
        }
        else
        {
            string[] DelArr;
            Global.Reis[] reis;
            int R = Global.NameArr(out DelArr, out reis);
            int N = Convert.ToInt32(maskedTextBox1.Text);
            if (N > 0 && N - 1 < R)
            {
                Global.file = new FileStream(Global.fileName, FileMode.Create);
                Global.SW = new StreamWriter(Global.file);
                for (int i = 0; i < R; i++)
                {
                    if (i != N - 1)
                        Global.SW.WriteLine(DelArr[i]);
                }
                Global.SW.Close();
                MessageBox.Show("Запись удалена!");
            }
            else
            {
                MessageBox.Show("Запись не найдена!");
            }
        }
    }
    else
    {
        MessageBox.Show("Файл не найден!");
    }
}
```

## Событие Load на форме Form5

```
private void Form5_Load(object sender, EventArgs e)
{
    {
        if (File.Exists(Global.fileName))
        {
            string[] Array;
            Global.Reis[] reis;
            int R = Global.NameArr(out Array, out reis);
            if (R == 0)
            {
                MessageBox.Show("Записей не найдено!");
                Close();
            }
            else
            {
                string line;
                Global.file = new FileStream(Global.fileName, FileMode.Open);
                Global.SR = new StreamReader(Global.file);
                int N = 1;
                int i = 0;
```

```

while (!Global.SR.EndOfStream)
{
    line = Global.SR.ReadLine();
    string[] Arr = line.Split('|');
    Global.City = Arr[0];
    Global.Tup = Arr[2];
    Global.Tupmin = Arr[7];
    Global.Tdown = Arr[4];
    Global.Tdownmin = Arr[8];
    Global.Aircraft = Arr[3];
    Global.Price = Convert.ToSingle(Arr[5]);
    Global.Nreisa = Convert.ToInt32(Arr[1]);
    Global.freep = Convert.ToInt32(Arr[6]);
    dataGridView1.Rows.Add();
    dataGridView1[0, i].Value = Convert.ToString(N);
    dataGridView1[1, i].Value = Global.City;
    dataGridView1[2, i].Value = Global.Tup+":"+Global.Tupmin;
    dataGridView1[3, i].Value = Global.Tdown+":"+Global.Tdownmin;
    dataGridView1[4, i].Value = Global.Aircraft;
    dataGridView1[5, i].Value = Convert.ToString(Global.Price);
    dataGridView1[6, i].Value = Convert.ToString(Global.Nreisa);
    dataGridView1[7, i].Value = Convert.ToString(Global.freep);
    i++; N++;
}
}
}
else
{
    MessageBox.Show("Файл не найден!");
    Close();
}
}
}

```

## Событие нажатия кнопки Button1 на форме Form6

```

private void button1_Click(object sender, EventArgs e)
{
    {
        if (File.Exists(Global.fileName))
        {
            if (maskedTextBox1.Text == "")
            {
                MessageBox.Show("Введите название города!");
            }
            else
            {
                listBox1.Items.Clear();
                string[] Z1Arr;
                Global.Reis[] reis;
                int R = Global.NameArr(out Z1Arr, out reis);
                for (int i = 0; i < R; i++)
                {
                    if ((reis[i].City == maskedTextBox1.Text))
                    {
                        listBox1.Items.Add("Номер рейса из города"+" "+
reis[i].City+" :"+reis[i].Nreisa);
                    }
                }
                if (listBox1.Items.Count == 0) MessageBox.Show("Записей по запросу
не найдено!");
            }
        }
    }
}

```

```

    }
    else
    {
        MessageBox.Show("Файл не найден!");
    }
}
}

```

## Событие нажатия кнопки Button1 на форме Form7

```

private void button1_Click(object sender, EventArgs e)
{
    {
        if (File.Exists(Global.fileName))
        {
            if (maskedTextBox1.Text == "")
            {
                MessageBox.Show("Введите название города!");
            }
            else
            {
                listBox1.Items.Clear();
                string[] Z2Arr; int FP = 0;
                Global.Reis[] reis;
                int R = Global.NameArr(out Z2Arr, out reis);
                for (int i = 0; i < R; i++)
                {
                    if ((reis[i].City == maskedTextBox1.Text))
                    {
                        FP = Convert.ToInt32(reis[i].freep)+FP;
                    }
                }
                listBox1.Items.Add("На рейсе(ах) из города "+ maskedTextBox1.Text+" Кол-во свободных мест " + Convert.ToString(FP));
                if (listBox1.Items.Count == 0) MessageBox.Show("Записей по запросу не найдено!");
            }
        }
        else
        {
            MessageBox.Show("Файл не найден!");
        }
    }
}

```

## Событие нажатия кнопки открытьToolStripMenuItem\_Click на форме Form1

```

private void открытьToolStripMenuItem_Click_1(object sender, EventArgs e)
{
    {
        if (dlgOpenFile.ShowDialog() == DialogResult.OK)
        {
            string directory =
Environment.GetFolderPath(Environment.SpecialFolder.Desktop);
            dlgOpenFile.InitialDirectory = directory;
            Global.fileName = dlgOpenFile.FileName;
            OpenFile();
        }
    }
}

```



## 6 Таблица тестов

В этом разделе описаны примеры тестов для проверки работоспособности программы.

Таблица 2 –Тесты для проверки работоспособности программы

<b>N теста</b>	<b>Назначение теста</b>	<b>Входные данные для теста</b>	<b>Выходные данные для теста</b>
1	Создание файла		Сообщение “Файл создан” (Рис.8)
2	Повторное создание файла		Сообщение “Файл уже существует” (Рис.9)
3	Добавление записи (успех)	Иркутск 333 11:12 22:17 101 Airbus A330 11500	Сообщение “Запись добавлена” (Рис.10)
4	Ошибка при добавлении записи (заполнены не все поля)	Москва 29 SSJ 100	Сообщение “Не все поля заполнены” (Рис.11)
5	Ошибка при добавлении записи (файл не найден)		Сообщение “Файл не найден” (Рис.12)
6	Удаление записи (успех)	2	Сообщение “Запись удалена” (Рис.13)
7	Ошибка при удалении записи (не введен номер записи)		Сообщение “Введите номер записи” (Рис.14)
8	Ошибка при удалении записи (запись не найдена)	99	Сообщение “Запись не найдена” (Рис.15)
9	Ошибка при удалении записи (файл не найден)	1	Сообщение “Файл не найден” (Рис.16)
10	Поиск записи для редактирования (успех)	1	Иркутск 333 11:12 22:17 101 Airbus A330 11500
11	Поиск записи для редактирования (не введен номер записи)		Сообщение “Введите номер записи” (Рис.18)

Продолжение таблицы 2

12	Поиск записи для редактирования (запись не найдена)	777	Сообщение “Запись не найдена” (Рис.19)
13	Поиск записи для редактирования (файл не найден)	1	Сообщение “Записей по запросу не найдено” (Рис.20)
14	Редактирование записи (успех)	1	Кёльн 067 11:12 22:17 101 Boeing-747 11500 Сообщение “Запись изменена” (Рис.21)
15	Редактирование записи (не все поля заполнены)	1	Кёльн 067 11:12 22:17 101 Boeing-747 Сообщение “ Не все поля заполнены” (Рис.22)
16	Просмотр записей (успех)		Рис.23
17	Просмотр записей (нет записей)		Сообщение “Нет ни одной записи” (Рис.24)
18	Просмотр записей (нет файла)		Сообщение “Файл не найден” (Рис.25)
19	Запрос 1 (успех)	Кёльн	Номер рейса из города Кёльн: 067 Номер рейса из города Кёльн: 3
20	Запрос 1 (нет записей)	2	Сообщение “Записей по запросу не найдено” (Рис.27)
21	Запрос 1(пустые поля ввода)		Сообщение “Введите № комнаты” (Рис.28)
22	Запрос 1 (нет файла)	8	Сообщение “Файл не найден” (Рис.29)
23	Запрос 2 (успех)	Кёльн	На рейсе(ах) из города Кёльн. Кол-во свободных мест 202.
24	Запрос 2 (пустое поле ввода)		Сообщение “Введите первую букву фамилии” (Рис.31)
25	Запрос 2(Нет файла)		Сообщение “Файл не найден”(Рис.32)
26	Запрос 2 (Нет записей)	Москва	Сообщение “Записей по запросу не найдено ”(Рис.33)

## 7 Результаты тестирования

В этом разделе показан внешний вид и выполненное событие во всех формах при выполнении тестов.

Результат тестирования создания файла (тест 1) показан на рисунке 8.

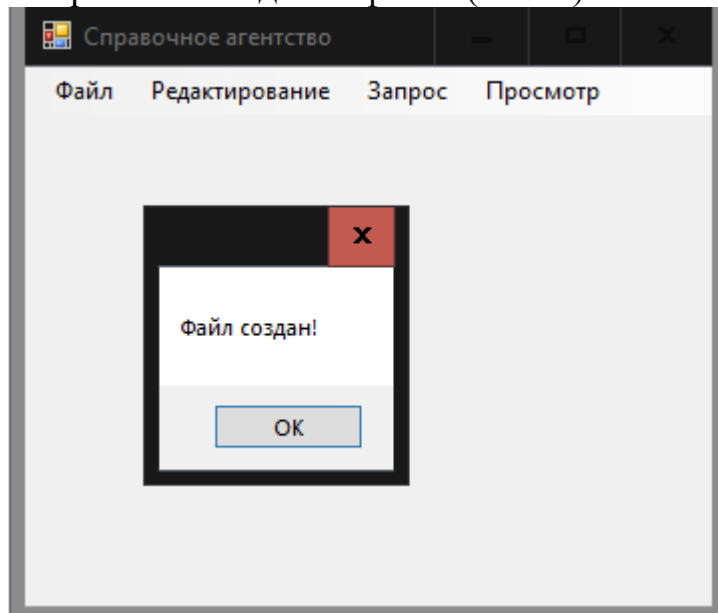


Рисунок 8 – Результат выполнения теста 1

Результат тестирования создания файла, когда файл уже существует (тест 2) показан на рисунке 9

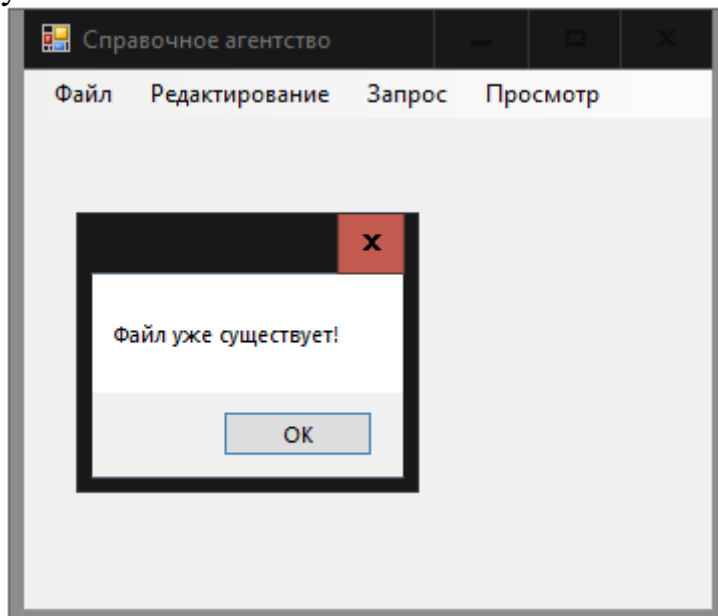


Рисунок 9 – Результат выполнения теста 2

Результат тестирования добавления записи (тест 3) показан на рисунке 10

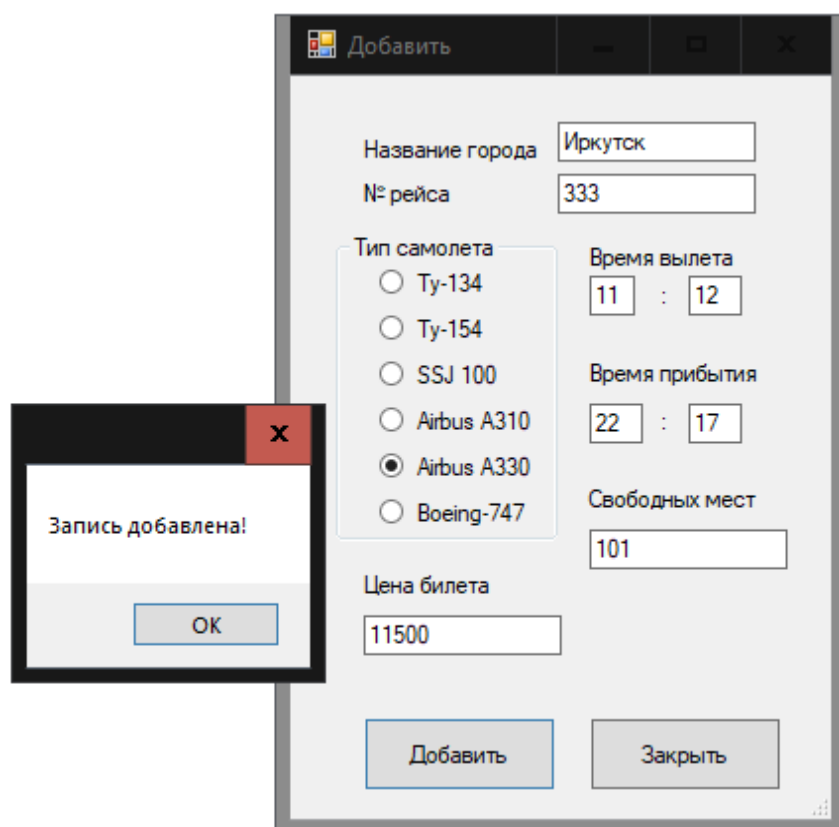


Рисунок 10 – Результат выполнения теста 3

Результат тестирования добавления записи, когда не все поля заполнены (тест 4) показан на рисунке 11

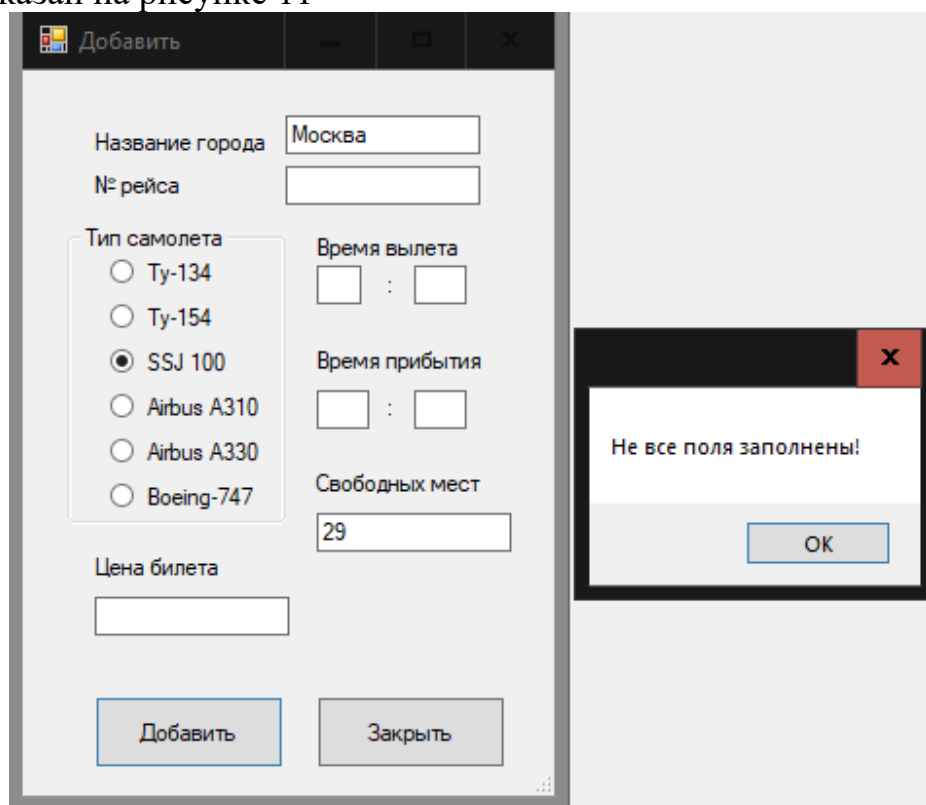


Рисунок 11 – Результат выполнения теста 4

Результат тестирования добавления записи, когда файл не создан (тест 5) показан на рисунке 12

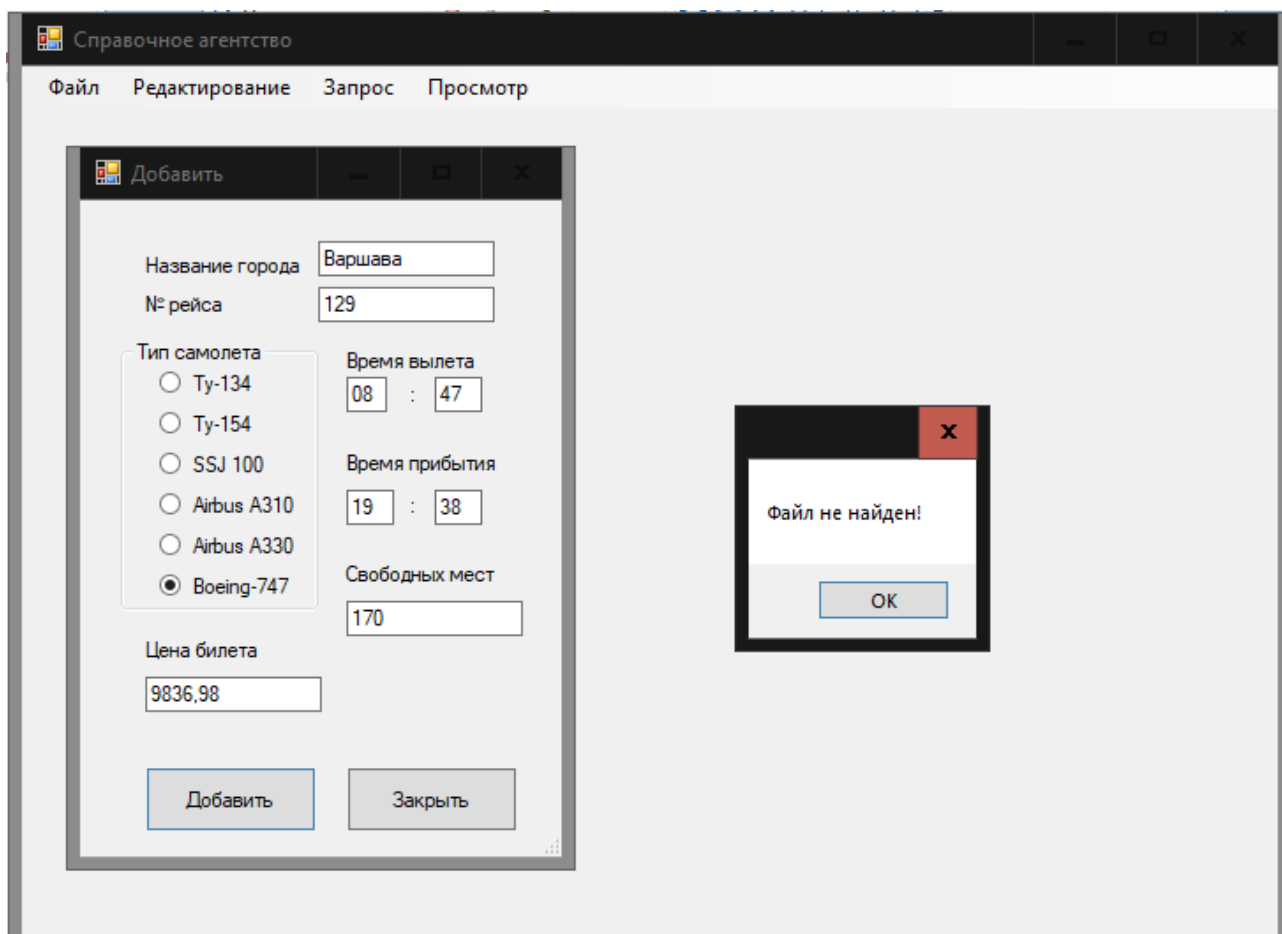


Рисунок 12 – Результат выполнения теста 5

Результат тестирования удаления записи (тест 6) показан на рисунке 13

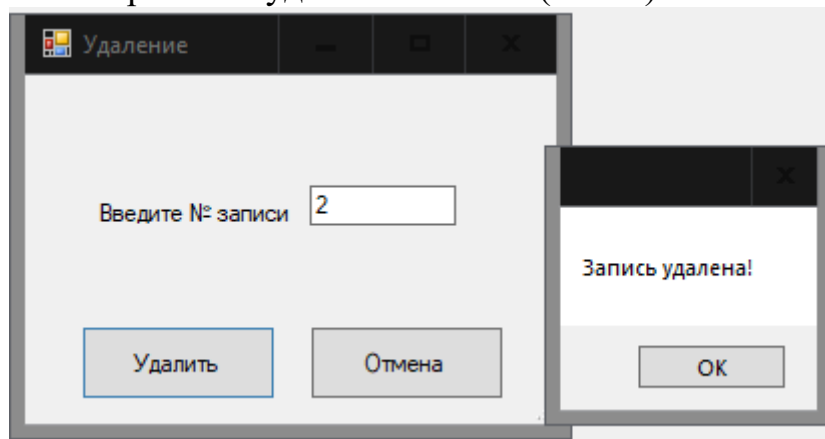


Рисунок 13 – Результат выполнения теста 6

Результат тестирования удаления записи, когда не введен номер записи (тест 7) показан на рисунке 14

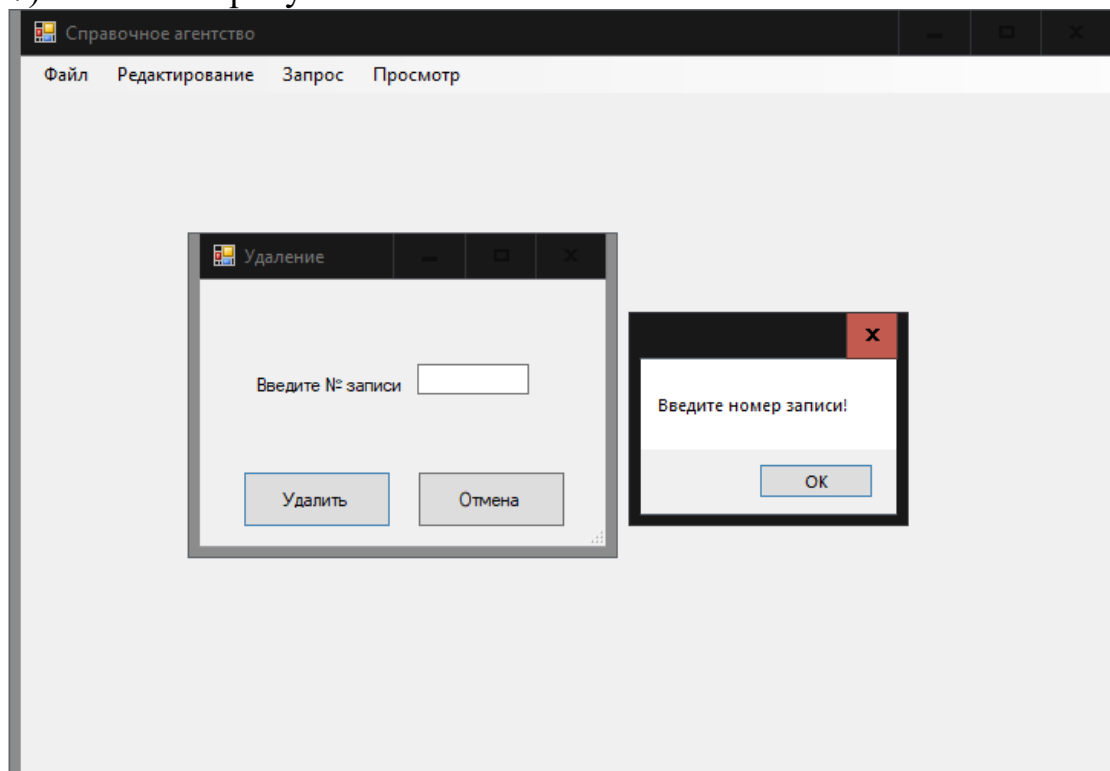


Рисунок 14 – Результат выполнения теста 7

Результат тестирования удаления записи, когда запись с таким номером не существует (тест 8) показан на рисунке 15

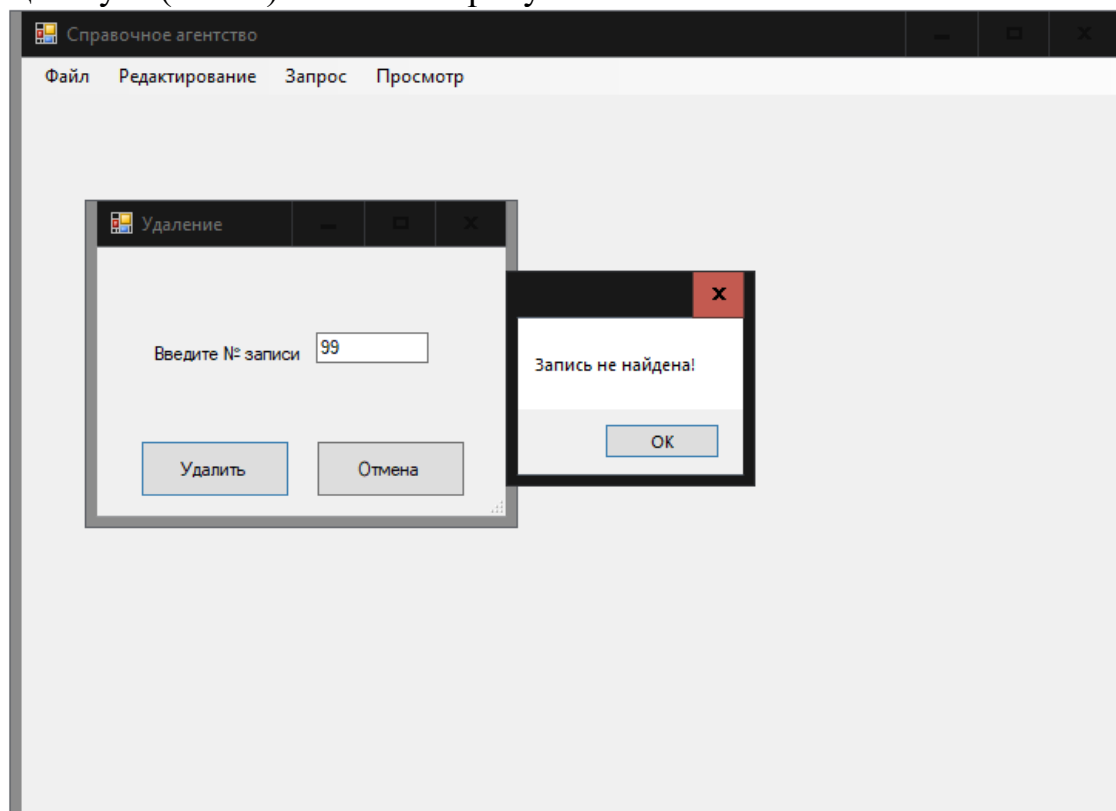


Рисунок 15 – Результат выполнения теста 8

Результат тестирования удаления записи, когда файл не создан (тест 9) показан на рисунке 16

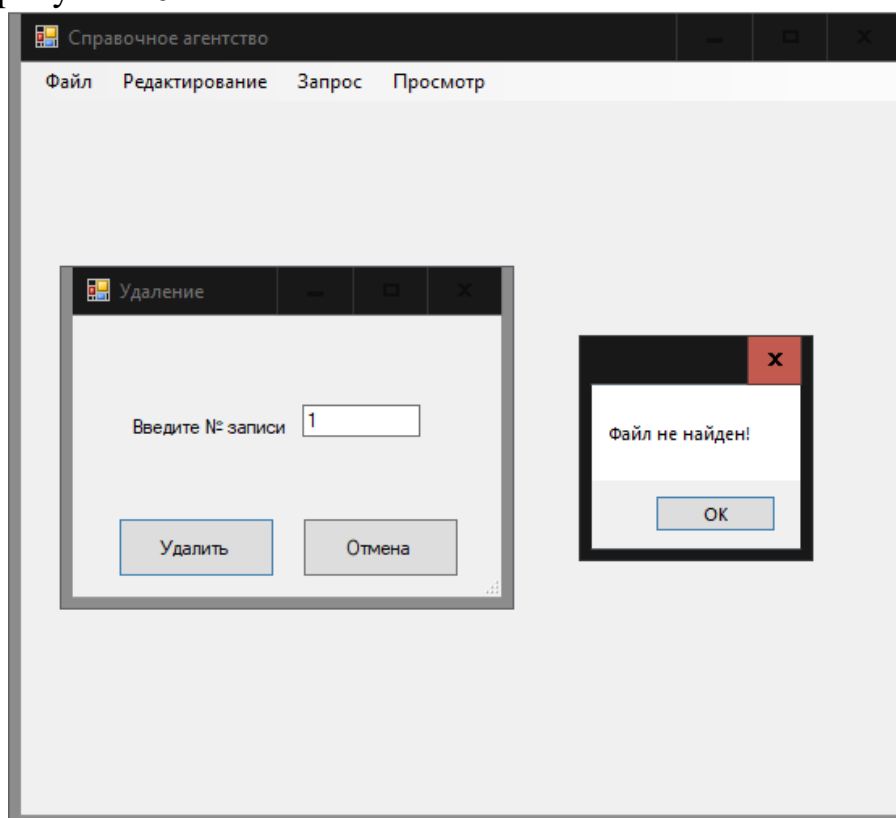


Рисунок 16 – Результат выполнения теста 9

Результат тестирования изменения записи, нажатие на кнопку “Найти” (тест 10) показан на рисунке 17

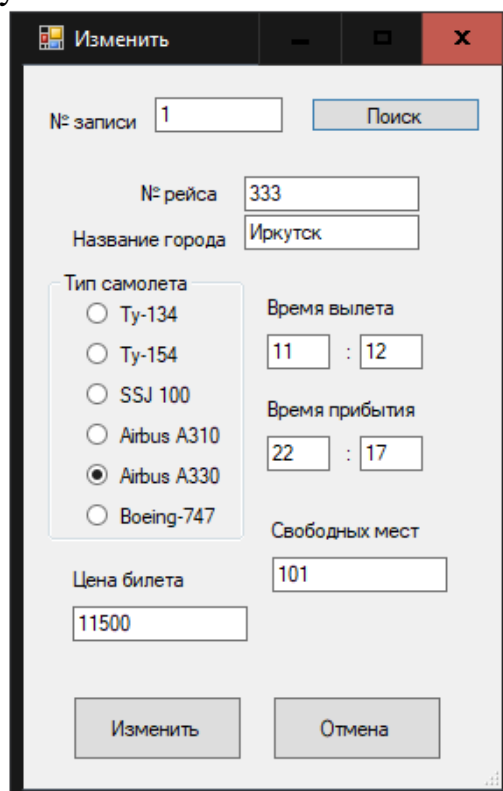


Рисунок 17 – Результат выполнения теста 10

Результат тестирования изменения записи, нажатие на кнопку “Найти” когда не введен номер записи (тест 11) показан на рисунке 18

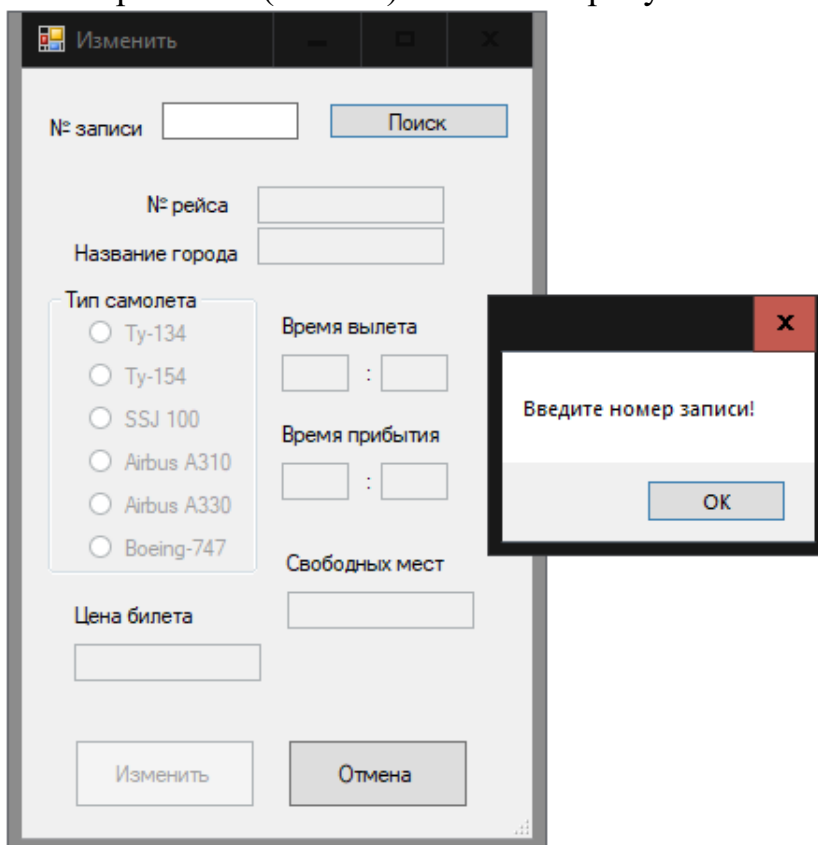


Рисунок 18 – Результат выполнения теста 11

Результат тестирования изменения записи, нажатие на кнопку “Найти” когда запись с таким номером не существует (тест 12) показан на рисунке 19

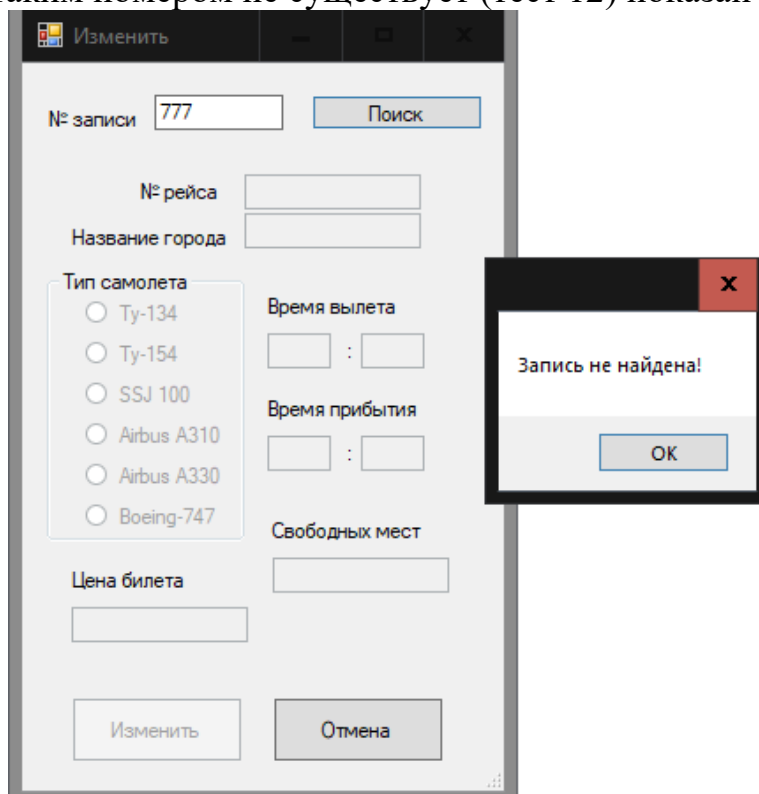


Рисунок 19 – Результат выполнения теста 12



Результат тестирования изменения записи, нажатие на кнопку “Найти” когда файл не создан (тест 13) показан на рисунке 20

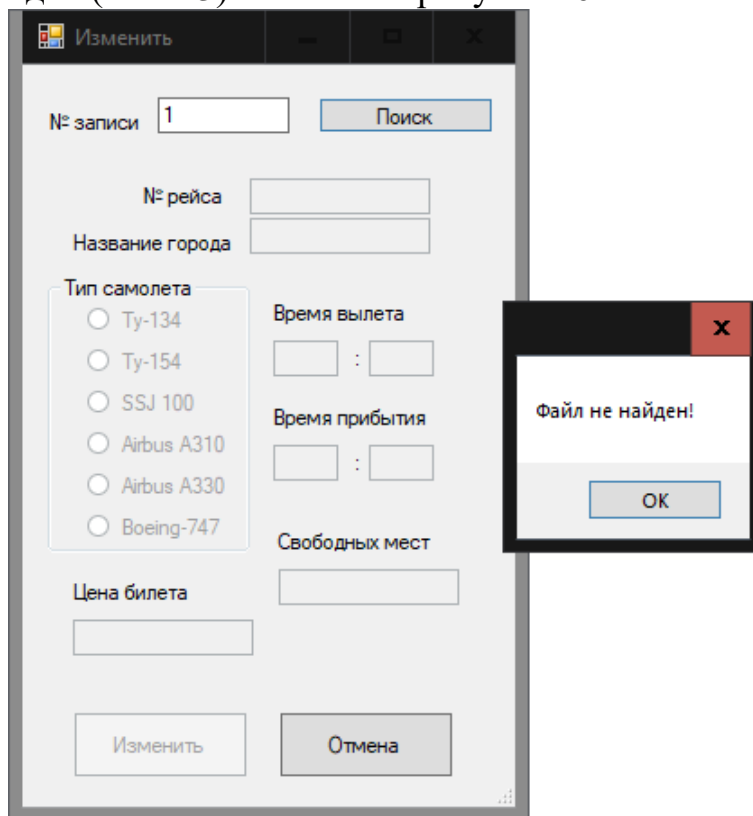


Рисунок 20 – Результат выполнения теста 13

Результат тестирования изменения записи, нажатие на кнопку “Изменить” (тест 14) показан на рисунке 21

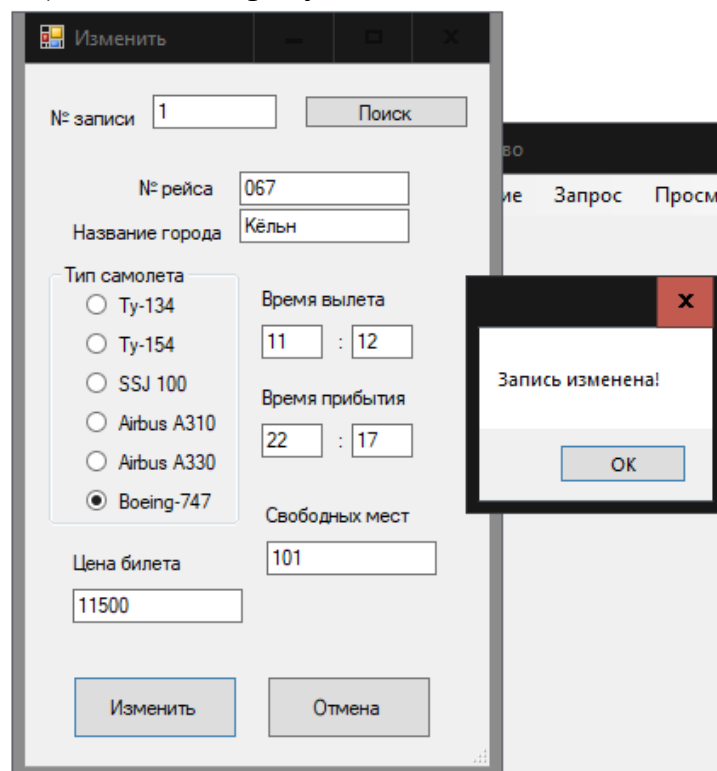


Рисунок 21 – Результат выполнения теста 14

Результат тестирования изменения записи, нажатие на кнопку “Изменить”, когда не все поля заполнены (тест 15) показан на рисунке 22

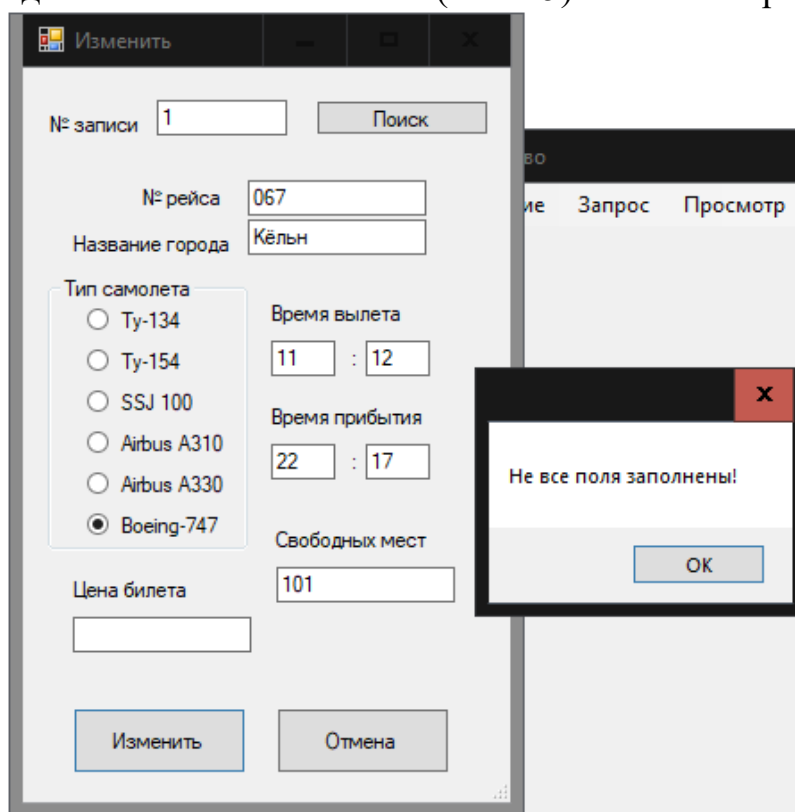


Рисунок 22 – Результат выполнения теста 15

Результат тестирования формы просмотра записей, (тест 16) показан на рисунке 23

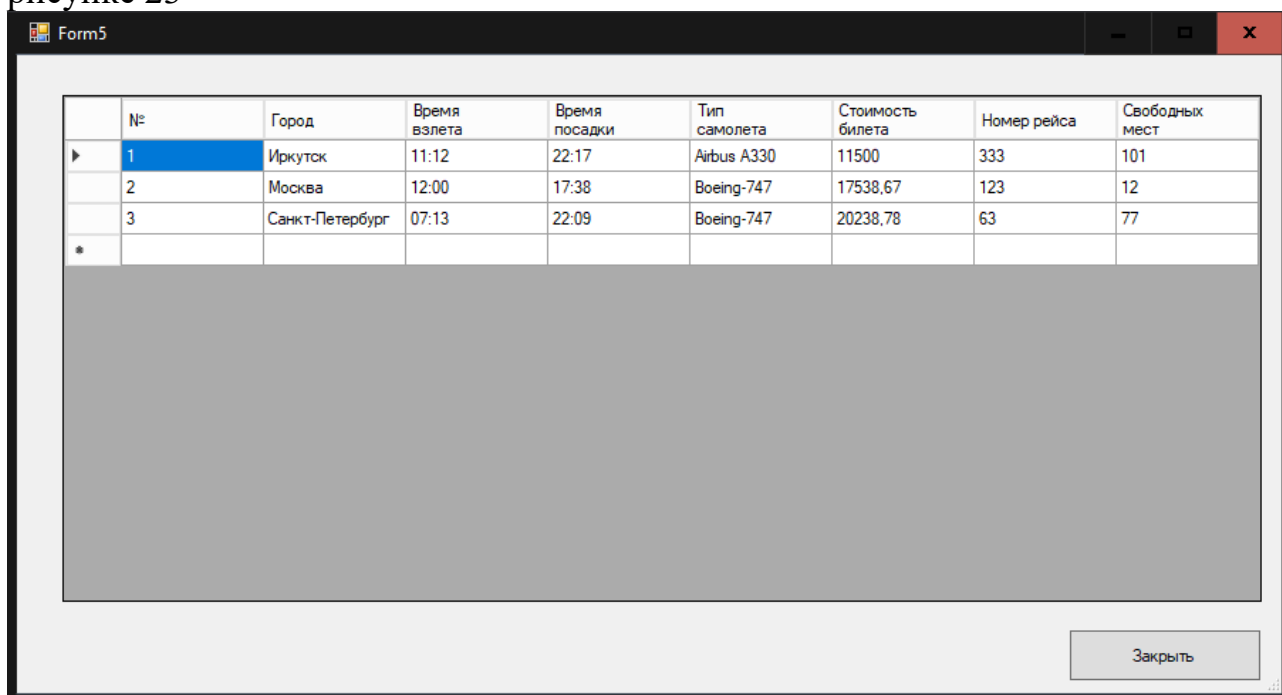


Рисунок 23 – Результат выполнения теста 16

Результат тестирования формы просмотра записей, когда в файле нет записей (тест 17) показан на рисунке 24

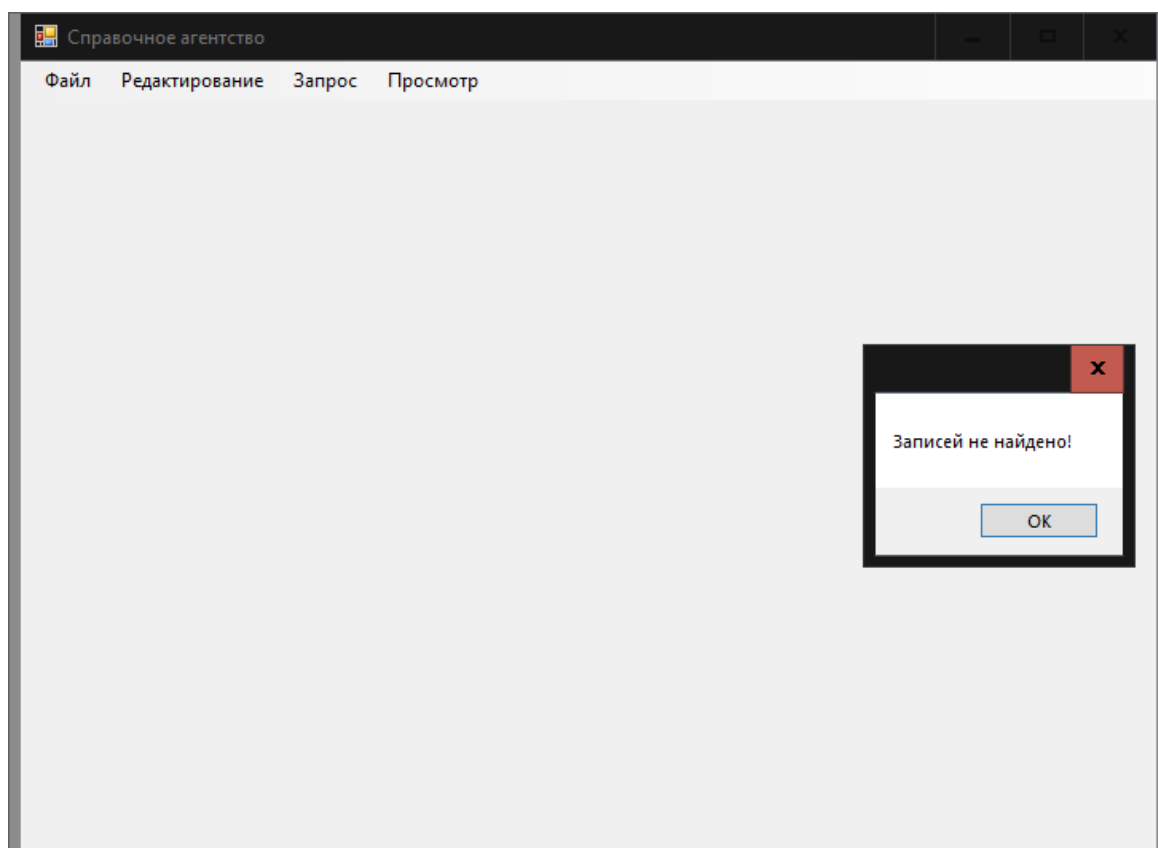


Рисунок 24 – Результат выполнения теста 17

Результат тестирования формы просмотра записей, когда файл не создан (тест 18) показан на рисунке 25

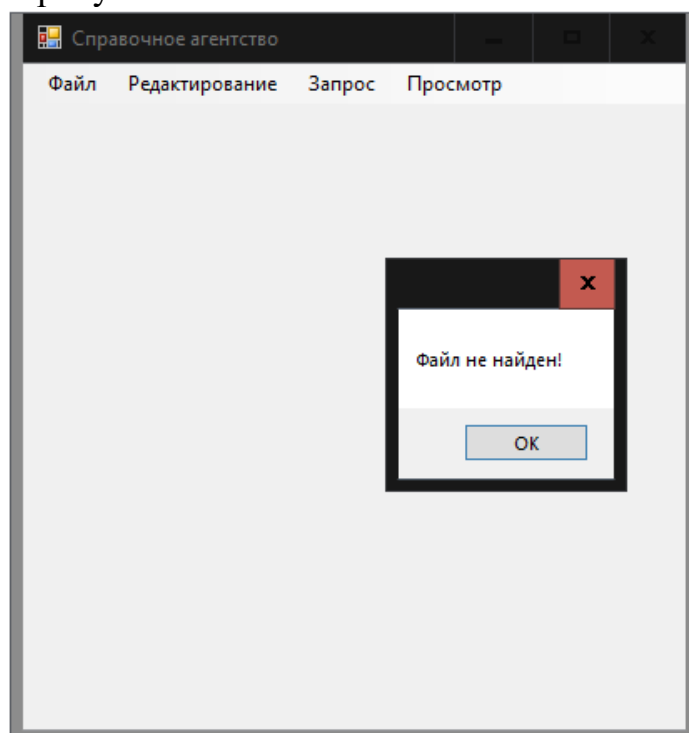


Рисунок 25 – Результат выполнения теста 18

Результат тестирования запроса 1, (тест 19) показан на рисунке 26

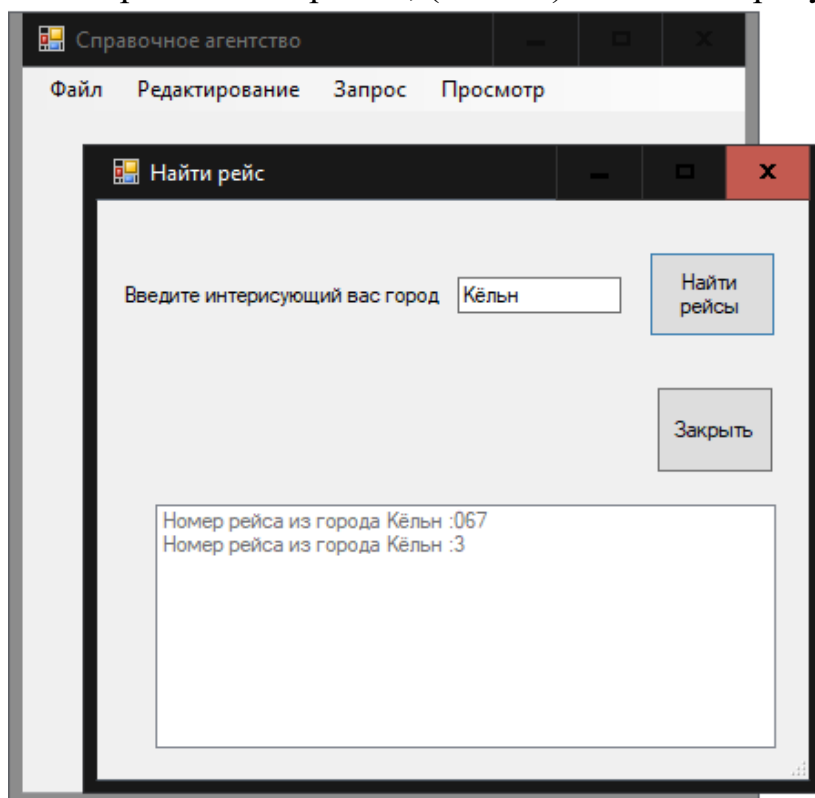


Рисунок 26 – Результат выполнения теста 19

Результат тестирования запроса 1, когда не найдено записей по запросу (тест 20) показан на рисунке 27

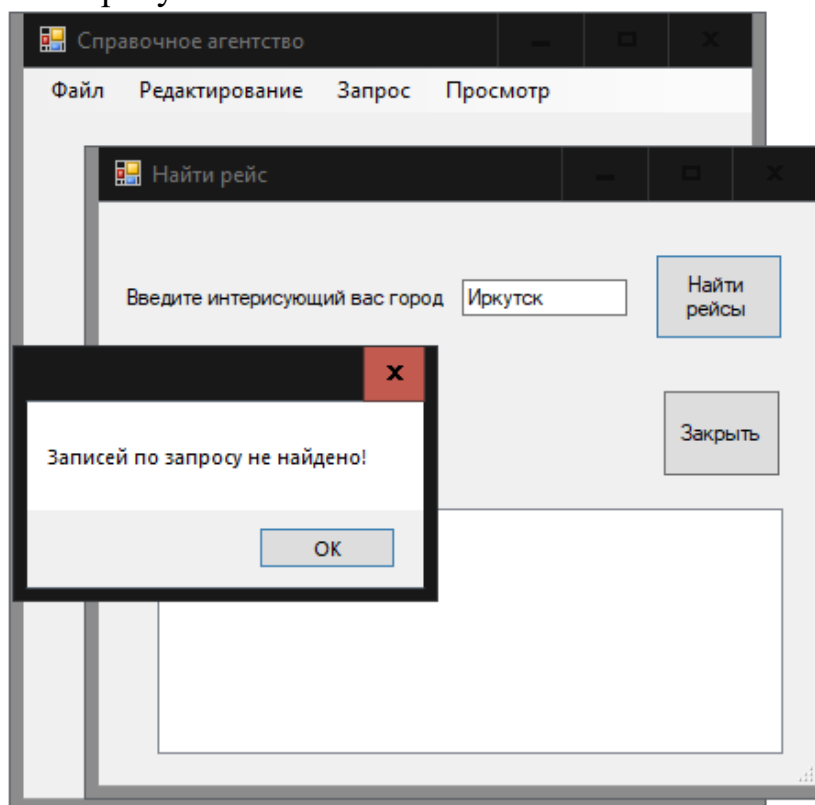


Рисунок 27 – Результат выполнения теста 20

Результат тестирования запроса 1, когда не введено название города (тест 21) показан на рисунке 28

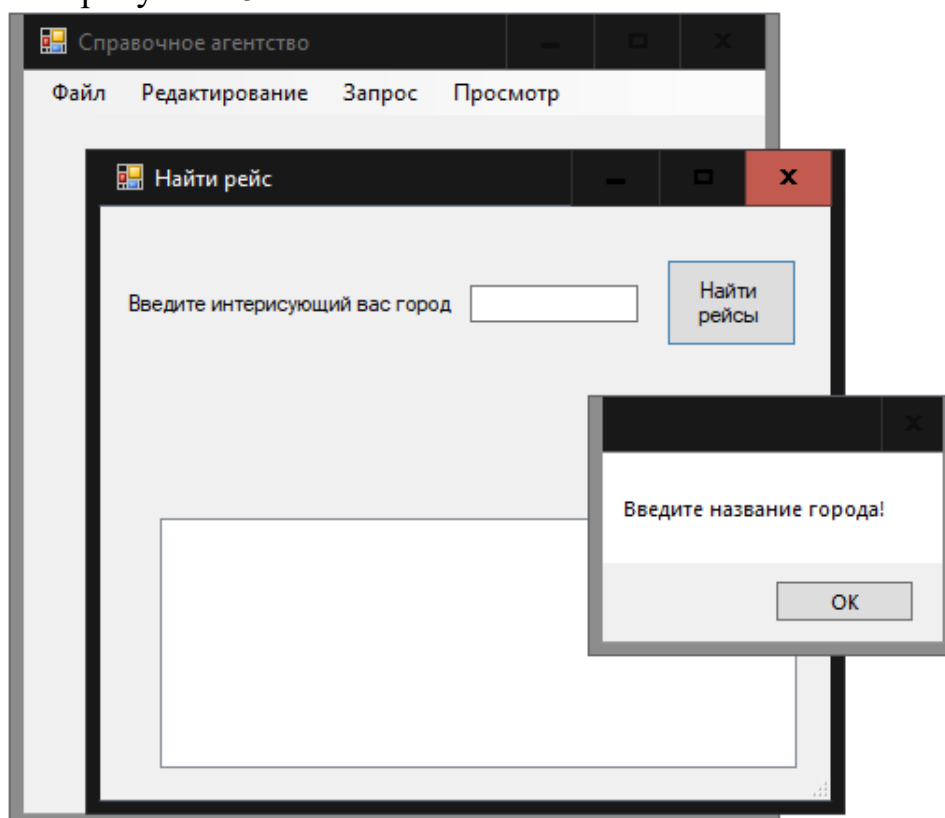


Рисунок 28 – Результат выполнения теста 21

Результат тестирования запроса 1, когда файл не создан (тест 22) показан на рисунке 29

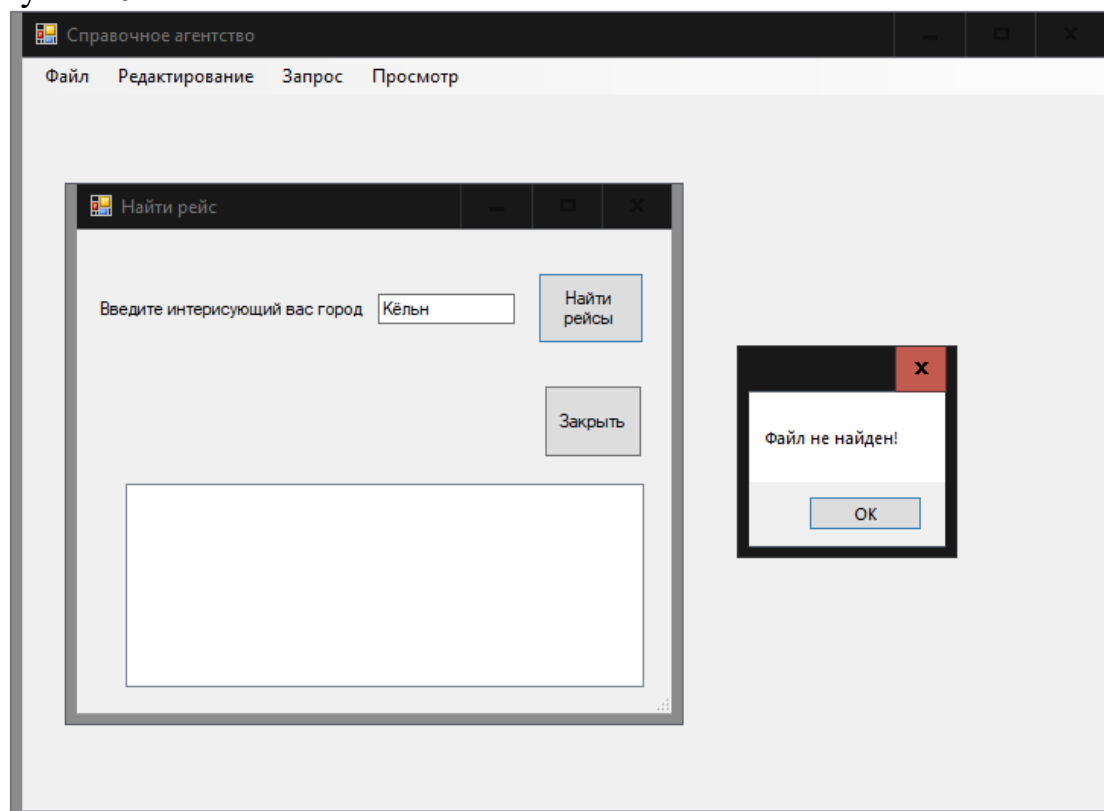


Рисунок 29 – Результат выполнения теста 22

Результат тестирования запроса 2 (тест 23) показан на рисунке 30

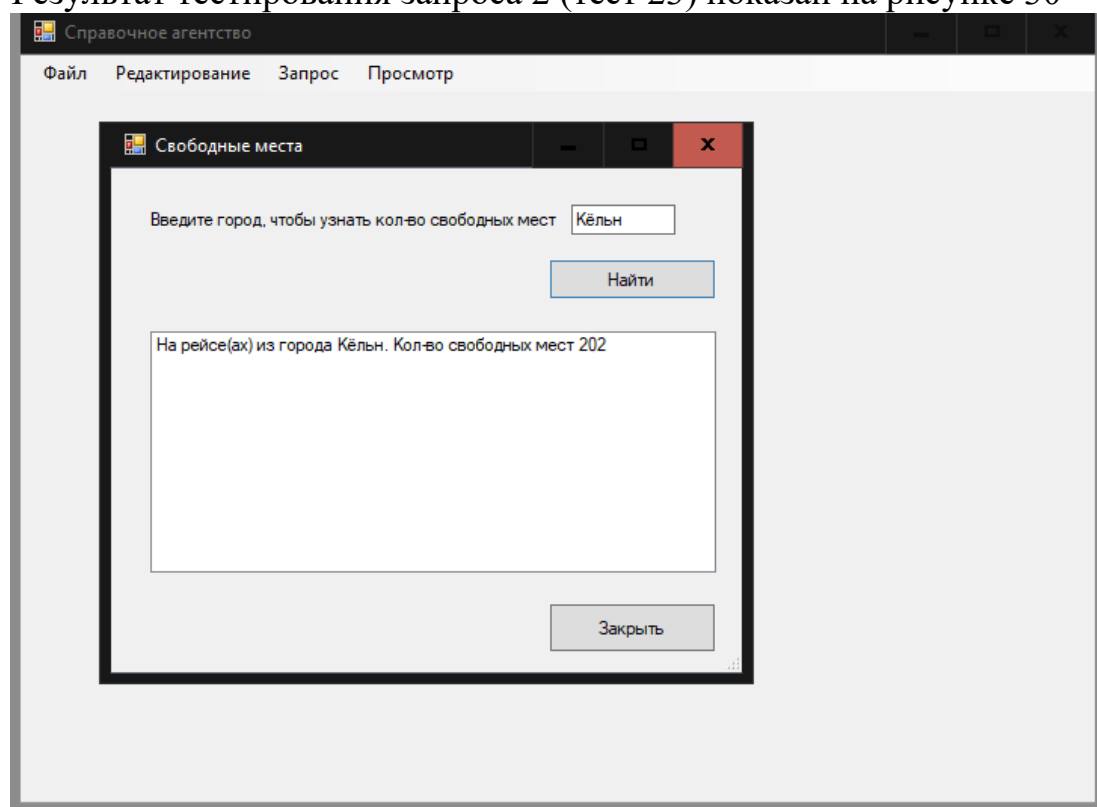


Рисунок 30 – Результат выполнения теста 23

Результат тестирования запроса 2, когда не введено название города (тест 24) показан на рисунке 31

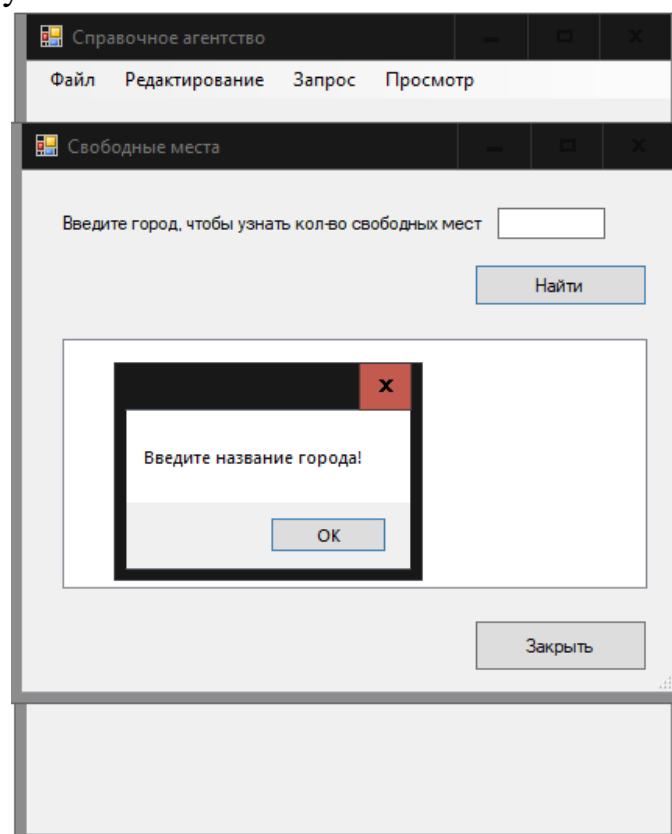


Рисунок 31 – Результат выполнения теста 24

Результат тестирования запроса 2, когда файл не создан (тест 25) показан на рисунке 32

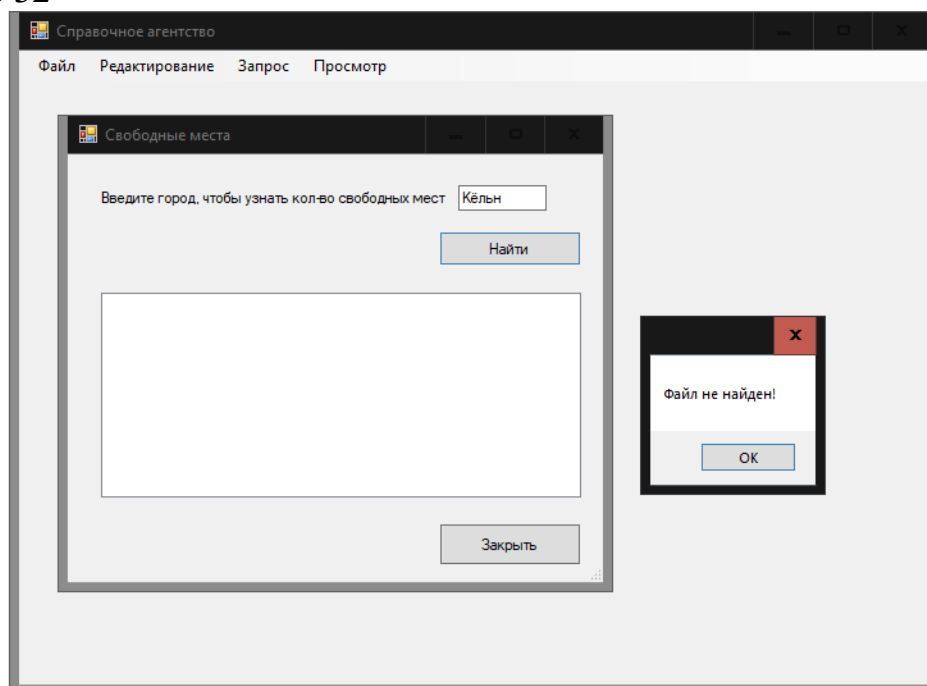


Рисунок 32 – Результат выполнения теста 25

Результат тестирования запроса 2, когда не существует записей с данным городом (тест 26) показан на рисунке 33

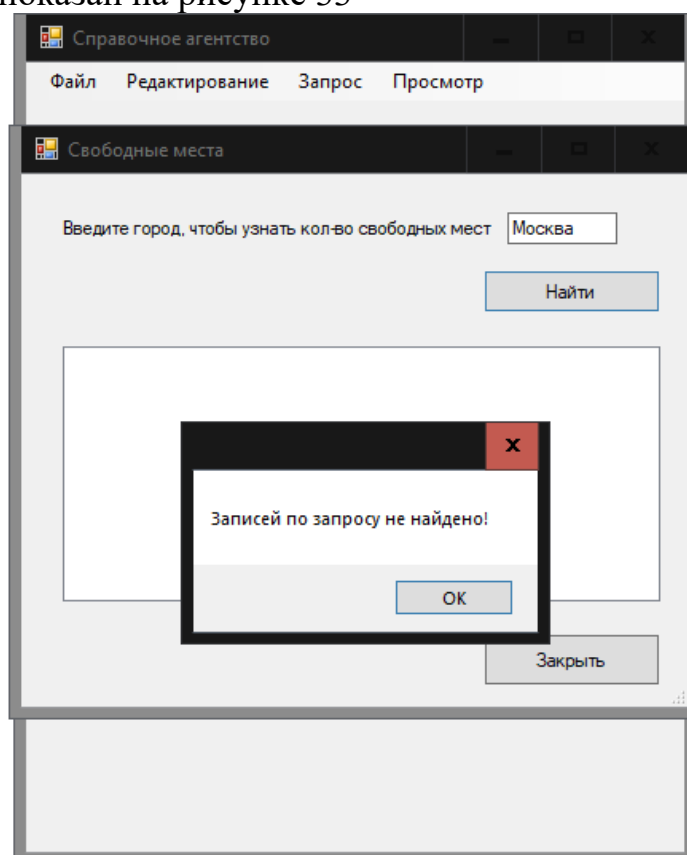


Рисунок 33 – Результат выполнения теста 26

Все тесты программы сошлись с предполагаемой таблицей тестов, из этого следует, что программа работает без ошибок.

## **Заключение**

Вышеизложенной работой было продемонстрировано что в результате курсового проекта было разработано Windows-приложение соответствующее варианту индивидуального задания. Данное приложение было протестировано на самые разнообразные ошибки и благополучно прошло их все. Следуя из этого можно заявлять, что цель работы была достигнута, было разработано корректно функционирующее Windows-приложение. Автором проекта были усвоены полученные в ходу работы навыки и знания работы с языком C#, в частности в создании оконных приложений на данном языке.



### **Список использованных источников**

- 1) Сосинская С.С. «Обработка файлов»: Методические указания по выполнению курсового проекта» Электронный каталог кафедры.
- 2) Гуннерсон Э. Введение в С#. СПб: Питер, 2001.
- 3) Вагнер, Билл С# Эффективное программирование / Билл Вагнер. - М.: ЛОРИ, 2013. - 320 с.
- 4) Ишкова, Э. А. Самоучитель С#. Начала программирования / Э.А. Ишкова. - М.: Наука и техника, 2013. - 496 с.
- 5) Троелсен, Эндрю Язык программирования С# 5.0 и платформа .NET 4.5 / Эндрю Троелсен. - М.: Вильямс, 2015. - 486 с.
- 6) Биллинг В.А. Основы программирования на С#. - Т.: Интернет-университет информационных технологий, Бином, 2012. - 488 с.
- 7) Евдокимов П.В. С# на примерах. - М.: Наука и техника, 2016. - 304 с.
- 8) Нейгел К., Ивьен Б., Глинн Д., Уотсон К., Скиннер М. С# 4.0 и платформа .NET 4, 2011.