МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕНГО ОБРАЗОВАНИЯ

«ТЮМЕНСКИЙ ГОСУДАРСВЕННЫЙ УНИВЕРСТИТЕТ»

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК

Кафедра информационных систем

ОТЧЕТ ОБ ИНДИВИДУАЛЬНОЙ РАБОТЕ

ДИСЦИПЛИНЫ «Информатика и Программирование»

«АВТОМАТИЗАЦИЯ УЧЕТА ОТДЕЛА КАДРОВ»

Выполнил:

студент группы ИСиТ-21.01-2

Алексеев Алексей Вячеславович

Руководитель:

к. ф.- м. н., доцент

Моор Павел Климентьевич

Тюмень

2022

Оглавление

[1. Описание предметной области постановка задачи 2](#_Toc105813929)

[1.1. Описание предметной области 2](#_Toc105813930)

[1.2. Постановка задач 2](#_Toc105813931)

[1.3 Требования к программе 2](#_Toc105813932)

[1.4. Среда разработки 2](#_Toc105813933)

[1.5. Структура данных 2](#_Toc105813934)

[2. Описание приложения, алгоритмов, классов, методов 3](#_Toc105813935)

[2.1. Главная форма 3](#_Toc105813936)

[2.2. Форма "Сотрудники" 5](#_Toc105813937)

[2.3. Форма "Специальности" 10](#_Toc105813938)

[2.4. Форма "Назначения" 10](#_Toc105813939)

[2.4. Форма "Итоги по специальности" 11](#_Toc105813940)

[Заключение 12](#_Toc105813941)

# 1. Описание предметной области постановка задачи

## 1.1. Описание предметной области

Цель: разработать Windows-приложение, реализующее основные функции СУБД для автоматизации учета отдела кадров (добавление, удаление, редактирование данных, выборка).

## 1.2. Постановка задач

Разработать форму для отображения обобщенной информации;

Разработать форму «с подчиненной» для просмотра информации.

## 1.3 Требования к программе

Обеспечение информации, а также доступ к ней в случае необходимости

Добавление, изменение и удаление данных

Удобный, интуитивно-понятный пользовательский интерфейс

## 1.4. Среда разработки

Microsoft Visual Studio 2022.

## 1.5. Структура данных

Исходные данные для программы хранятся в текстовых файлах формата .txt (примеры представлены ниже), которые расположены в одной папке с программой. При запуске программы информация, содержащаяся в файлах, прочитывается и используется в программе.

Файл "Employees.txt"

|  |  |  |
| --- | --- | --- |
| **Наименование** | **Назначение** | **Тип** |
| id\_person | Код сотрудника | int |
| full\_name | ФИО сотрудника | string |
| gender | Гендер | string |
| date\_of\_birth | Дата рождения | datetime |

Примеры:

1~Иванов Иван Иванович~мужской~04.06.1998

2~Васильев Петр Александрович~мужской~11.01.1991

3~Сидорова Алена Витальевна~женский~22.11.1980

Файл "Post.txt"

|  |  |  |
| --- | --- | --- |
| **Наименование** | **Назначение** | **Тип** |
| id\_post | Код Должности | int |
| post | Должность | string |
| salary | Зарплата | int |

Примеры:

1~Инженер~60000

2~Программист~200000

3~Юрист~50000

Файл "Appointment.txt"

|  |  |  |
| --- | --- | --- |
| **Наименование** | **Назначение** | **Тип** |
| id\_case | Код назначения | int |
| data\_appointment | Дата назначения | string |
| id\_person | Код сотрудника | int |
| id\_post | Код должности | int |

Примеры:

1~29.01.2022~2~1

2~18.05.2000~3~3

# 2. Описание приложения, алгоритмов, классов, методов

## 2.1. Главная форма

Основная форма "Отдел кадров" появляется при запуске программы.

На рисунке 1 приведена основная форма программы. С её помощью осуществляется переход к другим разделам программы.

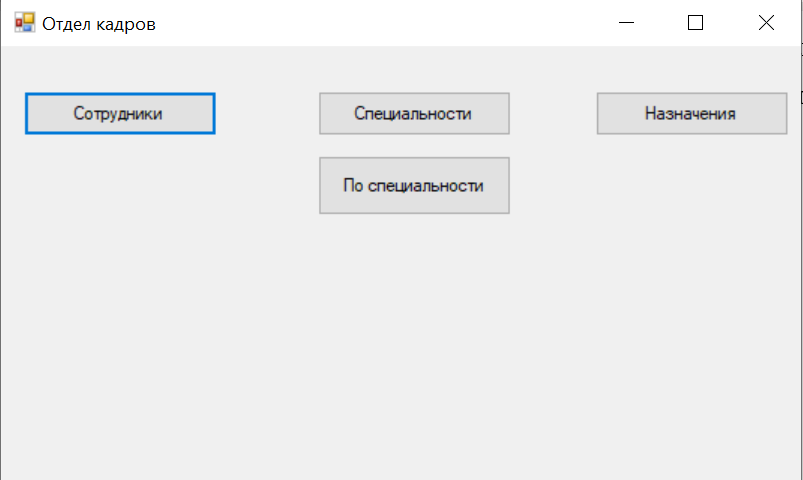


Рисунок 1. Внешний вид программы.

На рисунке 2 мы видим, как программным кодом реализован переход от основной формы в разделы программы.

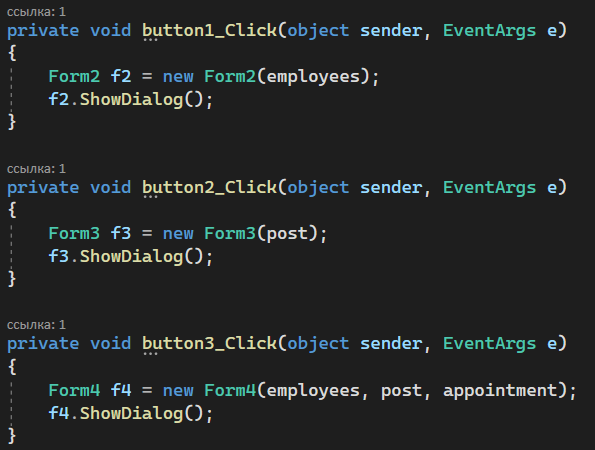


Рисунок 2. Программный код основной формы.

На рисунки 2.1 изображен программный код всех классов приложения.

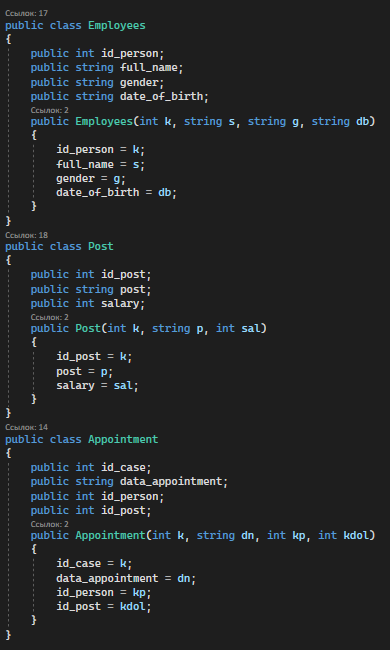


Рисунок 2.1 Программный код классов приложения

## 2.2. Форма "Сотрудники"

Форма " Сотрудники" появляется при нажатии на кнопку " Сотрудники".

На рисунке 3 представлен внешний вид раздела "Сотрудники", в котором можно работать с информацией, полученной из текстового файла под названием " Employees.txt".

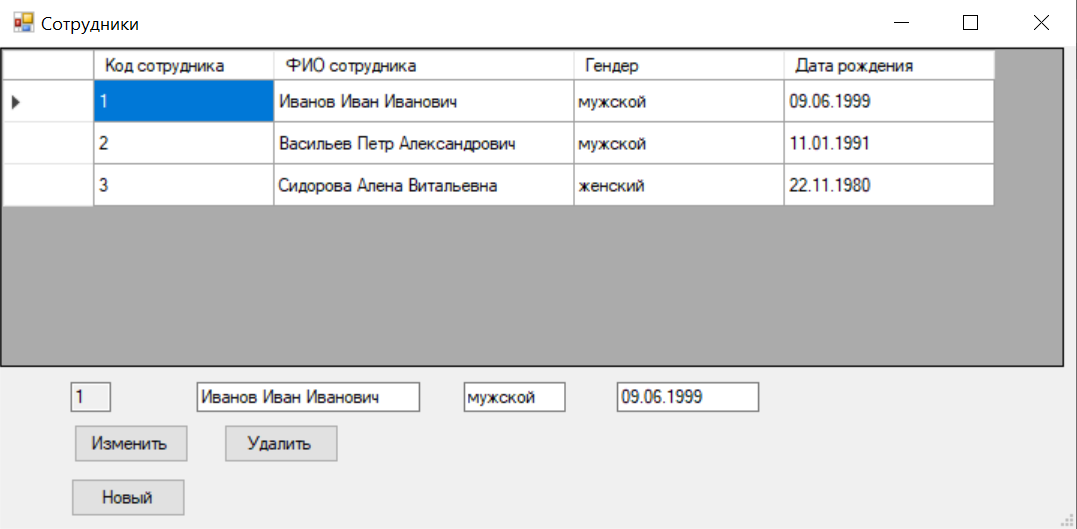


Рисунок 3. Интерфейс формы "Сотрудники".

На рисунке 4 показан программный код для осуществления чтения информации из файла. Для этого созданы методы Read\_e, Read\_p, Read\_a, с помощью которых и производится чтение из файлов для всех форм.

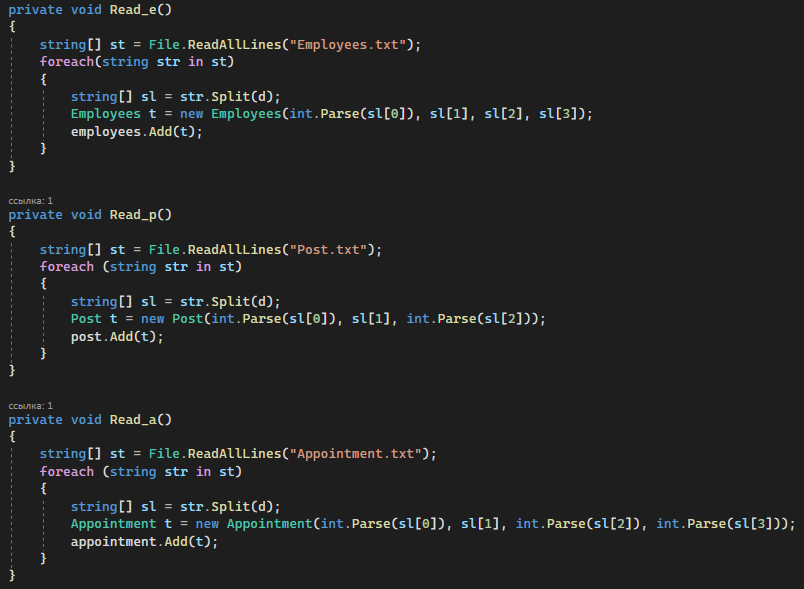


Рисунок 4. Фрагмент кода, содержащий методы для чтения из файла.

В методе Form2\_Load() содержится код для заполнения Datagridwiew1, с помощью метода Read\_e().

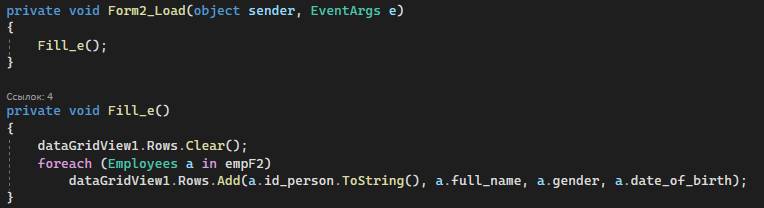


Рисунок 5. Программный код заполнения Datagridwiew1.

На рисунке 6 изображена форма "Сотрудники", после изменения в текстовом поле информации о выбранном сотруднике (изменено ФИО, дата рождения) и нажатия кнопки "Изменить". Информация о сотруднике была изменена.

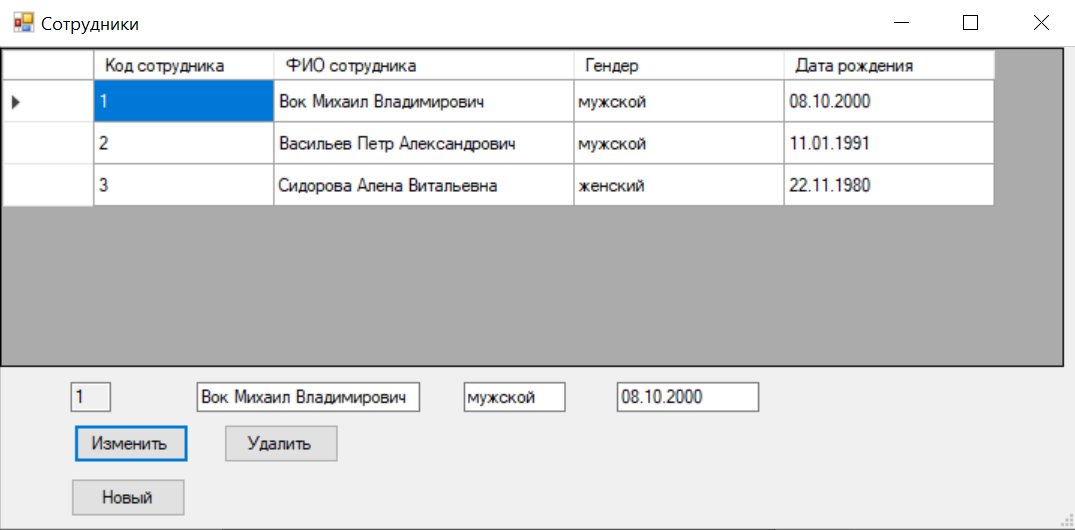


Рисунок 6. Кнопка "Изменить".

На рисунке 7 представлен код кнопки " Изменить", в котором осуществляется изменение данных в таблице.

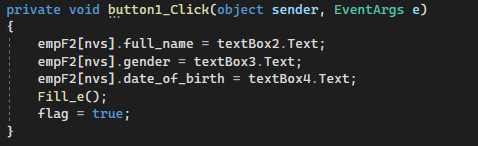


Рисунок 7. Программный код кнопки "Изменить".

На рисунке 8 представлена таблица после выбора строки и нажатия кнопки "Удалить" (строка с сотрудником Сидорова Алена Витальевна. была удалена). Нумерация сохраняется, выбранная строка удалена.

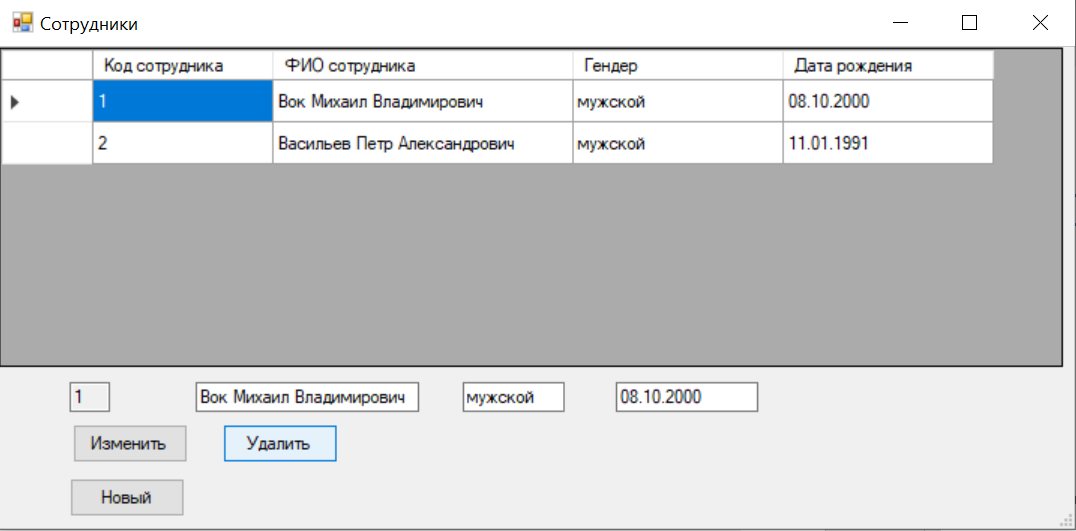


Рисунок 8. Кнопка "Удалить".

На рисунке 13 представлен программный код кнопки "Удалить", который обеспечивает удаление выбранной строки.

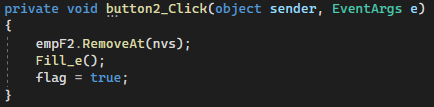


Рисунок 9. Программный код кнопки "Удалить".

На рисунке 10 показано, что произойдёт при нажатии на кнопку "Новый": текстовые поля с ФИО, гендером и датой рождения сотрудника очистятся, код - порядковый номер будет присвоен автоматически.

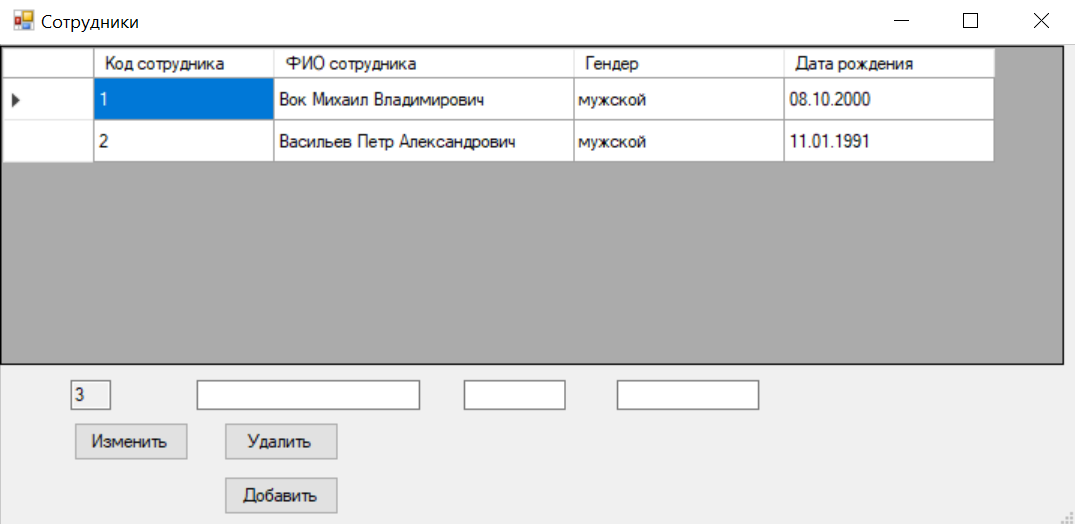


Рисунок 10. Кнопка "Новый".

На рисунке 11 представлен программный код, который очищает текстовые поля и присваивает новый номер полю "Код сотрудника" при нажатии на кнопку "Новый". При этом кнопка "Новый" становится невидимой, а кнопка "Добавить" – видимой.

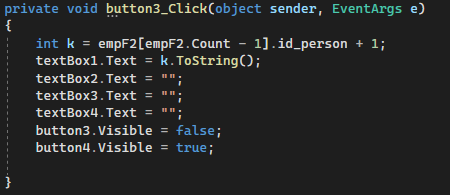


Рисунок 11. Программный код кнопки "Новый".

На рисунке 12 показан вид программы после нажатия кнопки "Добавить": в таблицу добавляется новая строка с введёнными ранее в текстовые поля данные.

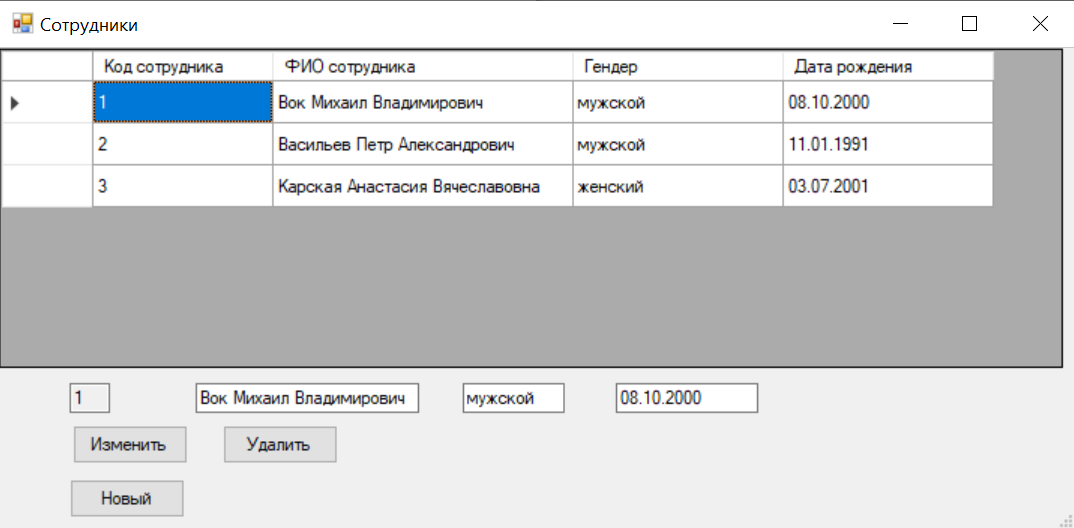


Рисунок 12. Кнопка "Добавить".

На рисунке 13 изображен программный код кнопки "Добавить", в котором происходит добавление информации и новой строки таблицы.

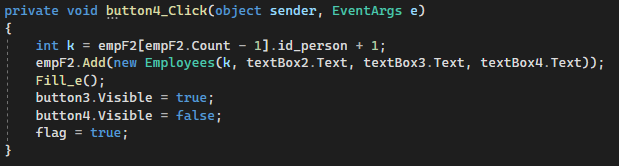


Рисунок 13. Программный код кнопки "Добавить".

На рисунке 14 представлен программный код сохранения всей информации в форме "Сотрудники" в файл "Employees.txt".

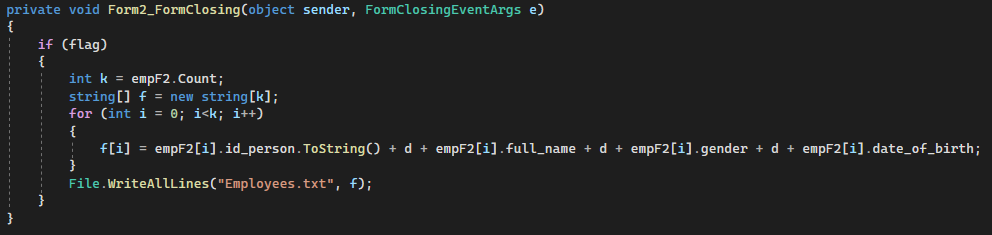


Рисунок 14. Программный код Сохранения информации в файл.

Рисунок 15 иллюстрирует программный код метода dataGridView1\_SelectionChanged\_1(object sender, EventArgs e), который при смене строки в таблице синхронизирует данные с текстовыми полями.

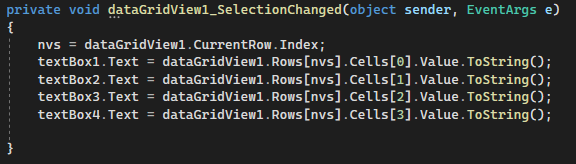


Рисунок 15. Программный код метода, определяющего выбранную строку.

## 2.3. Форма "Специальности"

Форма "Специальности" появляется при нажатии на кнопку "Специальности".

На рисунке 16 показан внешний вид раздела "Специальности". Все кнопки в данном разделе работают по методам, представленным из раздела "Сотрудники".

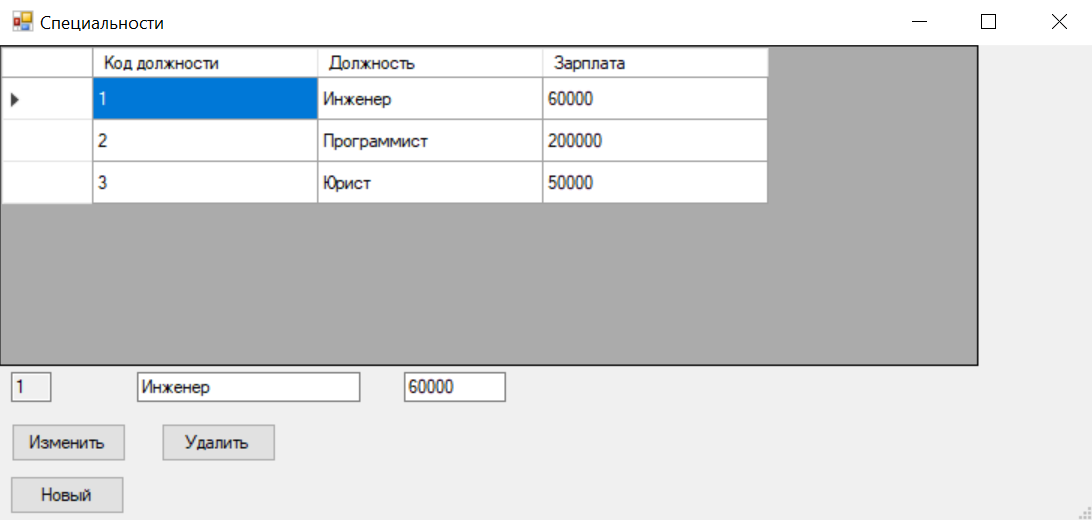


Рисунок 16. Интерфейс раздела "Специальности"

## 2.4. Форма "Назначения"

Форма "Назначения" появляется при нажатии на кнопку "Назначения".

На рисунке 17 показан внешний вид раздела "Назначения". Все кнопки в данном разделе работают по методам, представленным из раздела "Сотрудники".

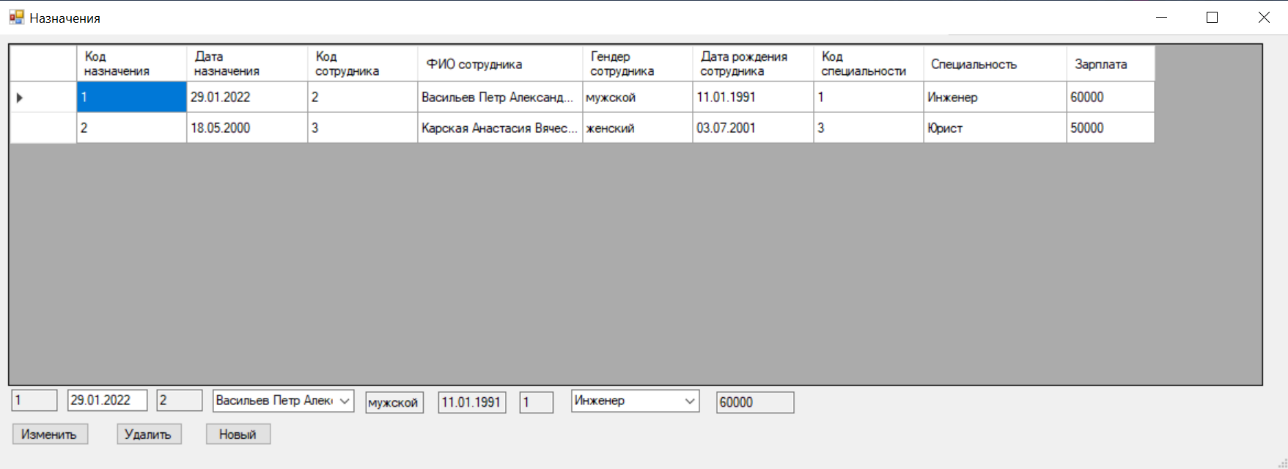


Рисунок 16. Интерфейс раздела "Специальности"

## 2.4. Форма "Итоги по специальности"

Форма "Итоги по специальности" появляется при нажатии на кнопку "Итоги по специальности".

На рисунке 17 изображен внешний вид формы "Итоги по специальности", на котором видны результаты обработки информации исходных файлов и подведён подсчёт суммы всех сотрудников, назначенных на выбранную специальность.

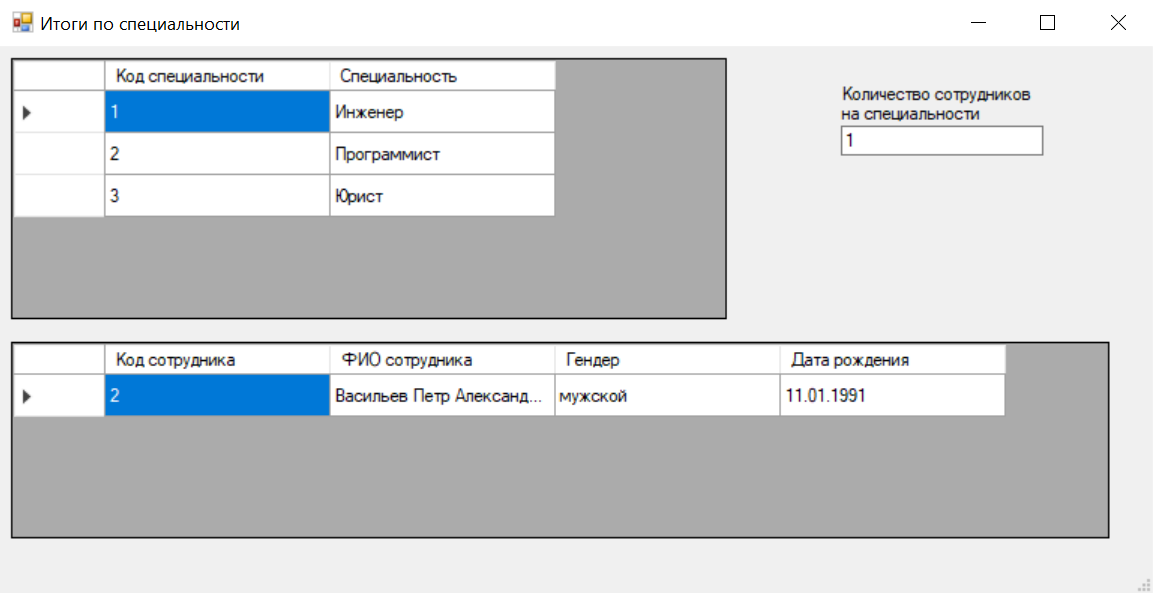


Рисунок 17. Интерфейс формы "Итоги по специальности"

На рисунке 18 представлено метод, определяющий, какая строка выбрана в таблице. После определения выбранной строки в верхней таблице, заполняется нижняя таблица на основе выбранного клиента.

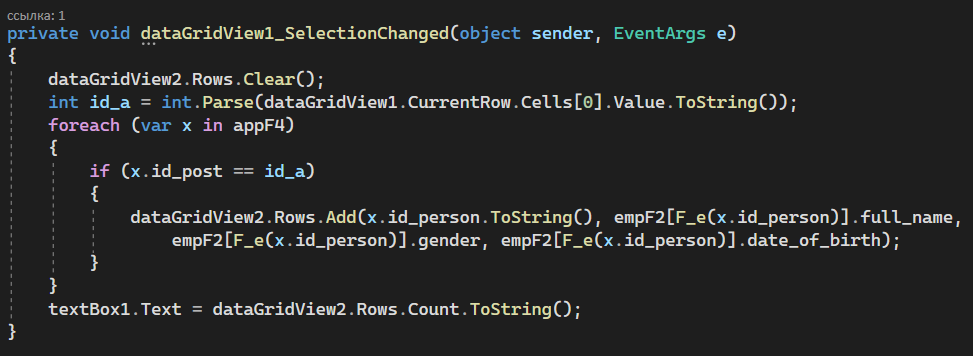


Рисунок 18. Фрагмент программного кода определения выбранных строк.

# 3. Заключение.

В ходе проекта было разработано приложение, работающее с текстовыми файлами (чтение, запись), позволяющее читать (смотреть сводную информацию), редактировать, добавлять, удалять и сохранять изменения. Для этого было создано 5 форм: "Начальная форма", "Сотрудники", "Специальности", "Назначения", "Итоги по специальности". Для каждой формы были разработаны методы для работы с необходимым файлом. DataGridView () - заполнение таблицы полученными данными. Для управления данными (редактирование и добавление) были созданы текстовые поля и раскрывающийся список (ComboBox). Данное приложение предназначено для реализации основных функций СУБД для автоматизации учета отдела кадров, имеющее удобный, интуитивно-понятный интерфейс, обеспечивающее доступ к информации в случае необходимости.