|  |
| --- |
| **МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ** |
| **федеральное государственное бюджетное образовательное учреждение высшего образования «Российский экономический университет имени Г.В. Плеханова»**  **Московский приборостроительный техникум** |

**ОТЧЕТ**

по учебной практике

УП 01.01 «Прикладное программирование»

*индекс по УП и наименование практики*

Профессионального модуля ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем»

*индекс по УП и наименование профессионального модуля*

Специальность 09.02.07 «Информационные системы и программирование»

*код и наименование специальности*

Студент Савкова Кира Дмитриевна

Группа П50-3-22

Руководитель по практической подготовке от техникума

Арсений Владимирович Архангельский

«\_\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 года

# Практическая работа №2

## Подключение базы данных. Чтение, добавление, изменение, удаление данных из БД

Цель: Необходимо создать базу данных с несколькими связанными таблицами, нормализованными до третьей нормальной формы. После этого требуется создать приложение WPF и подключить его к этой базе данных. Реализация приложения должна включать два варианта: один с использованием Entity Framework, а другой с использованием DataSet. В приложении WPF должны быть различные окна или страницы для работы с разными таблицами. Данные из базы данных должны отображаться в элементе управления DataGrid в приложении WPF. Пользователь должен иметь возможность добавлять, изменять и удалять данные из таблицы через интерфейс приложения.

1. Приложение с DataSet
   1. Для начала необходимо открыть прошлый проект и в нем файл PRACTICA1DataSet1.xsd. Далее в 4 таблицах реализовать следующие функции: возможность добавлять, изменять и удалять данные из таблицы. Для этого нужно Щелкнуть ПКМ по первой таблице на строчку Fill,GetDatа() и выбрать «Добавить запрос».

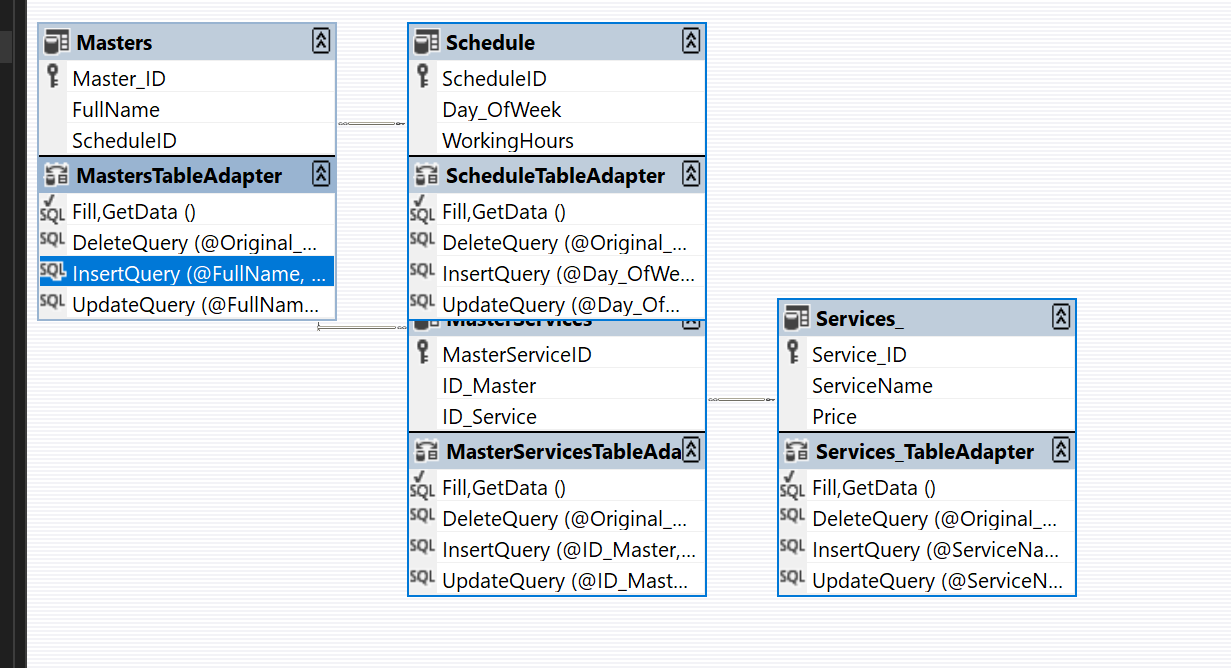


Рисунок - Вид таблиц

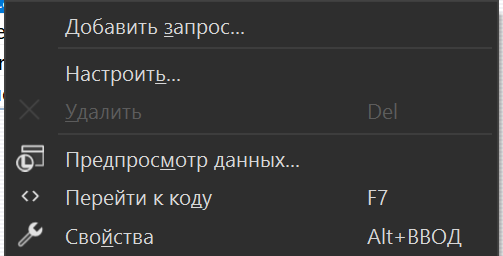


Рисунок - Добавление запроса

Так как мы хотим сделать обычный запрос, в самом первом пункте выберем «Использовать инструкции SQL». Для добавления выберем тип запроса (Inser, Delete, Update) и нажмем «Далее»

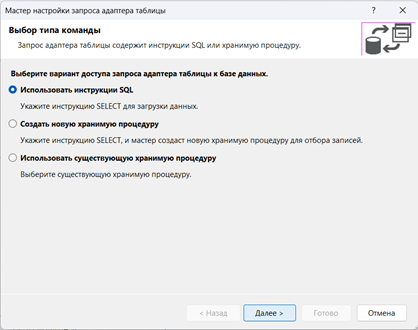


Рисунок - Выбор типа команды

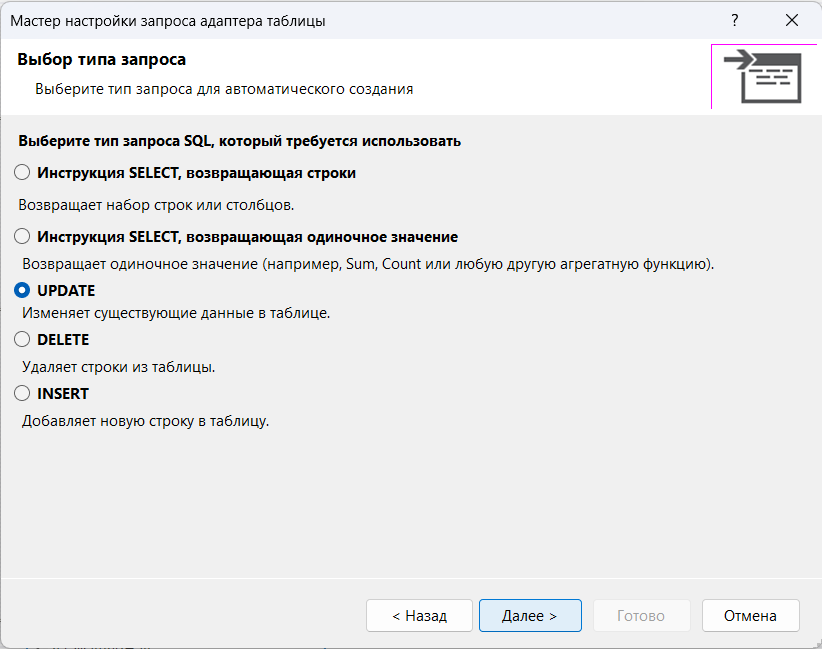


Рисунок - Выбор типа запроса

После необходимо воспользоваться конструктором запросов и заполнить наш запрос.

Основа нашего приложения остается та же, нам нужно в логике дописать необходимые фукции, а в интерфейсе добавить кнопки и текстобксы для первых 2х окон и комбобоксы для 2 и 3 окон. Расположение кнопок пропишем с помощью Grid.

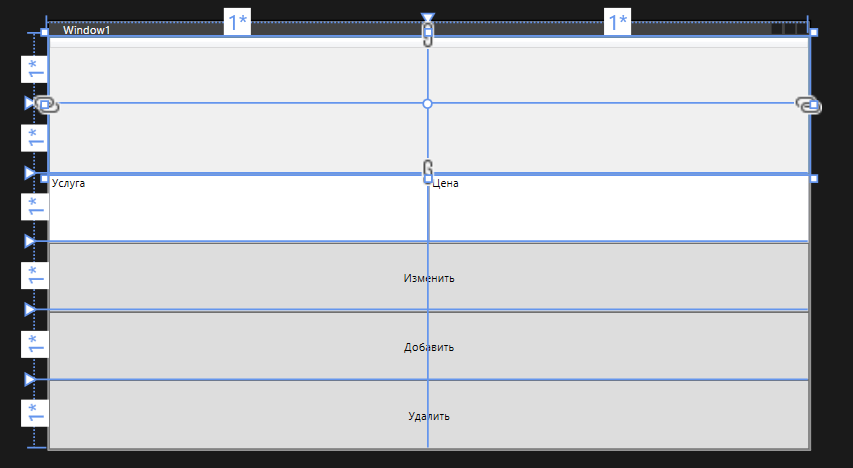


Рисунок - Вид 1го окна

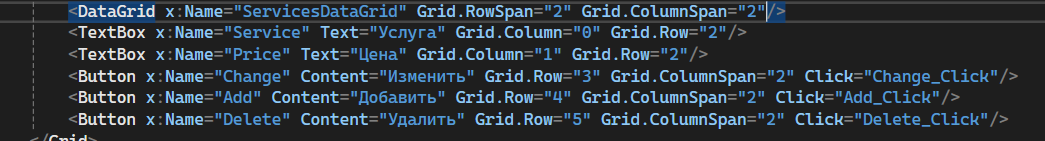


Рисунок - Создание кнопок

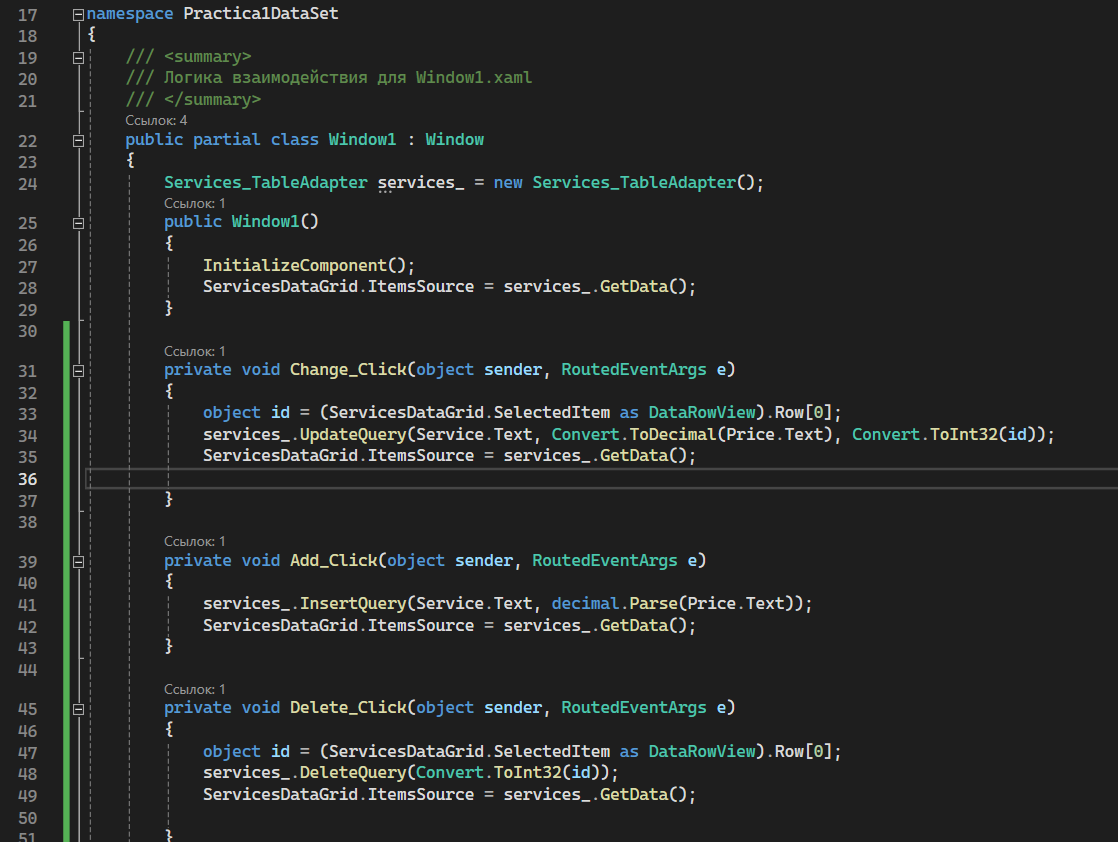


Рисунок - Логика первого окна

Методы Change\_Click, Add\_Click и Delete\_Click обрабатывают нажатия на соответствующие кнопки в пользовательском интерфейсе:

Change\_Click: Обновляет запись в таблице Services\_ с помощью метода UpdateQuery класса Services\_TableAdapter.

Add\_Click: Добавляет новую запись в таблицу Services\_ с помощью метода InsertQuery класса Services\_TableAdapter.

Delete\_Click: Удаляет выбранную запись из таблицы Services\_ с помощью метода DeleteQuery класса Services\_TableAdapter.

В дальнейшем в каждом из окон будут прописываться такие же методы по шаблону, но с разными датагридами и данными таблиц.

Рассмотрим подробнее данные методы в 1м окне:

Метод вызывается при нажатии кнопки "Change". Он обновляет запись в таблице Services\_ с помощью метода UpdateQuery класса Services\_TableAdapter.

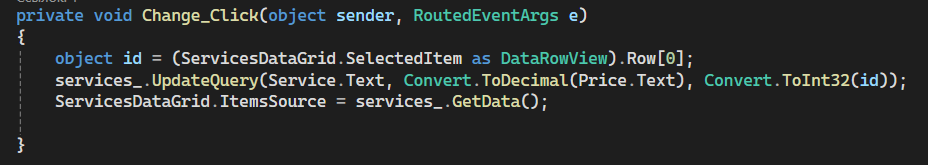


Рисунок - Метод изменения

Метод вызывается при нажатии кнопки "Add". Он добавляет новую запись в таблицу Services\_ с помощью метода InsertQuery класса Services\_TableAdapte

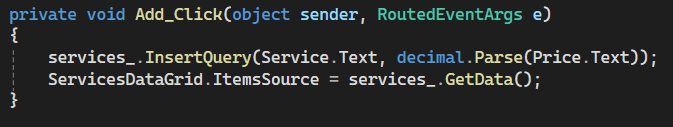


Рисунок - Метод добавления

Метод вызывается при нажатии кнопки "Delete". Он удаляет выбранную запись из таблицы Services\_ с помощью метода DeleteQuery класса Services\_TableAdapter.

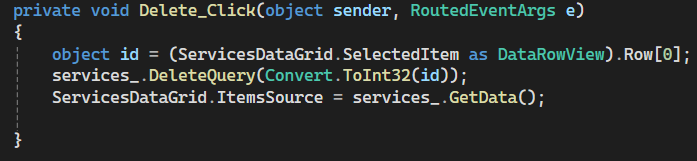


Рисунок - Метод удаления

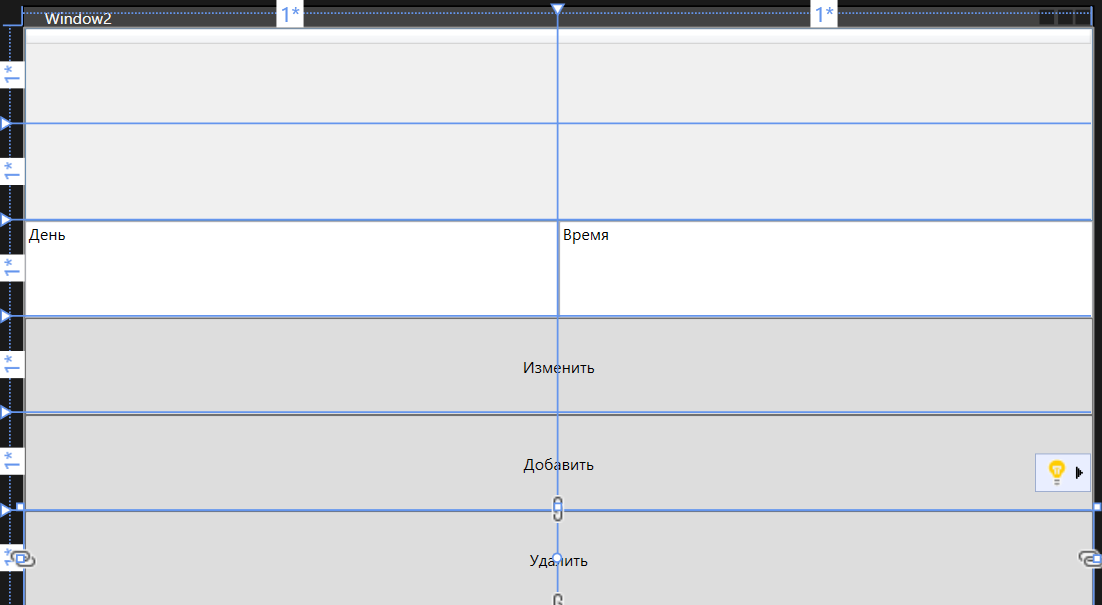


Рисунок - Вид 2го окна

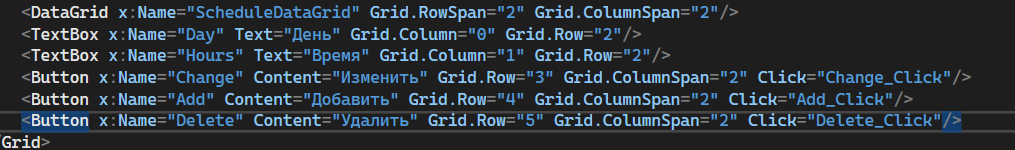


Рисунок - Создание кнопок

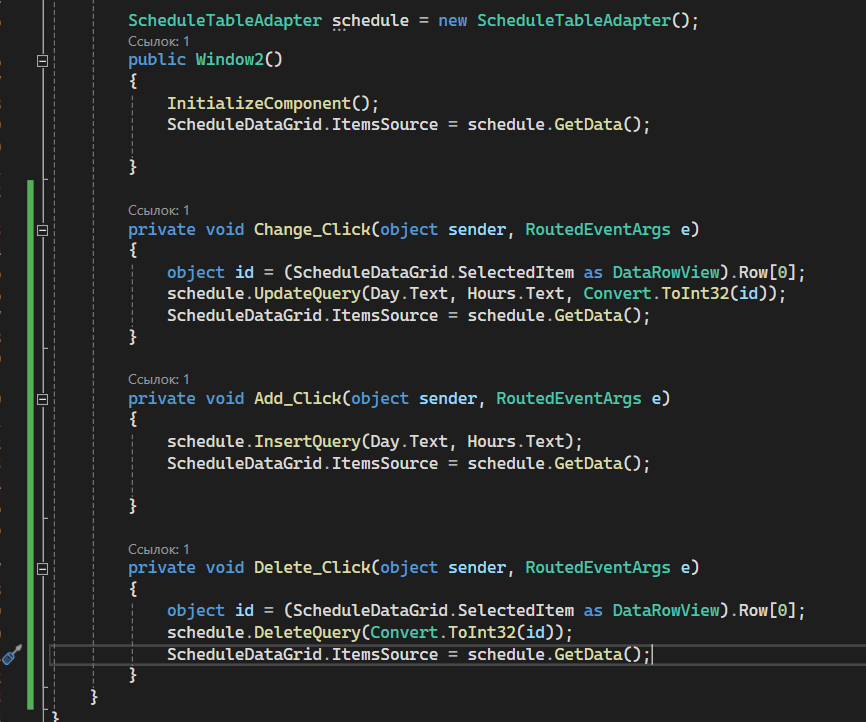


Рисунок - Методы 2го окна

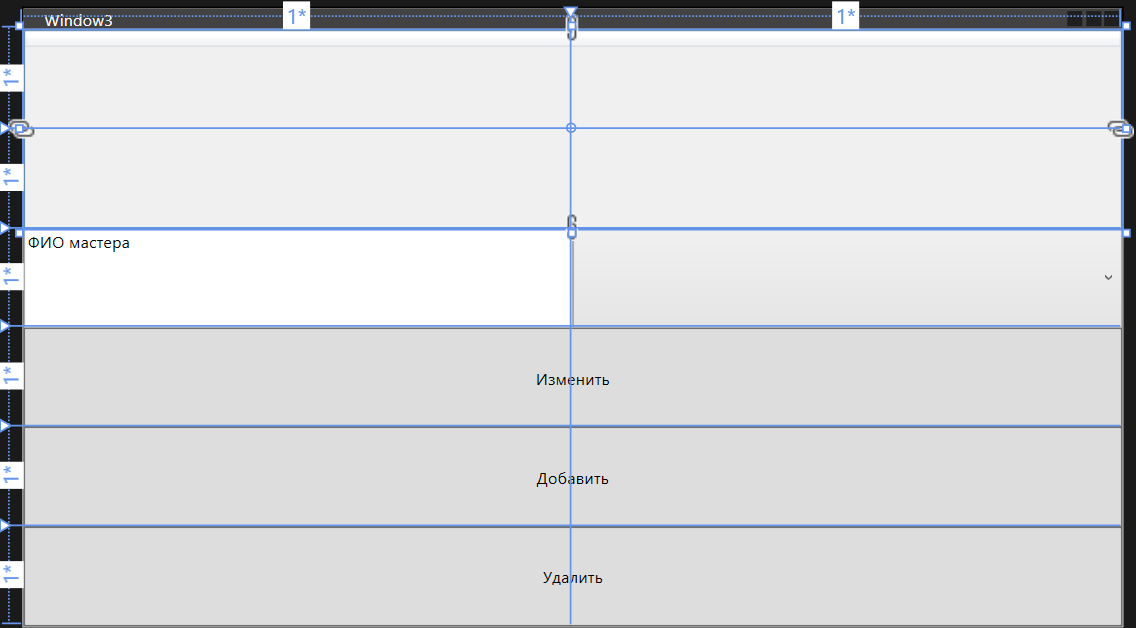


Рисунок - Вид 3го окна

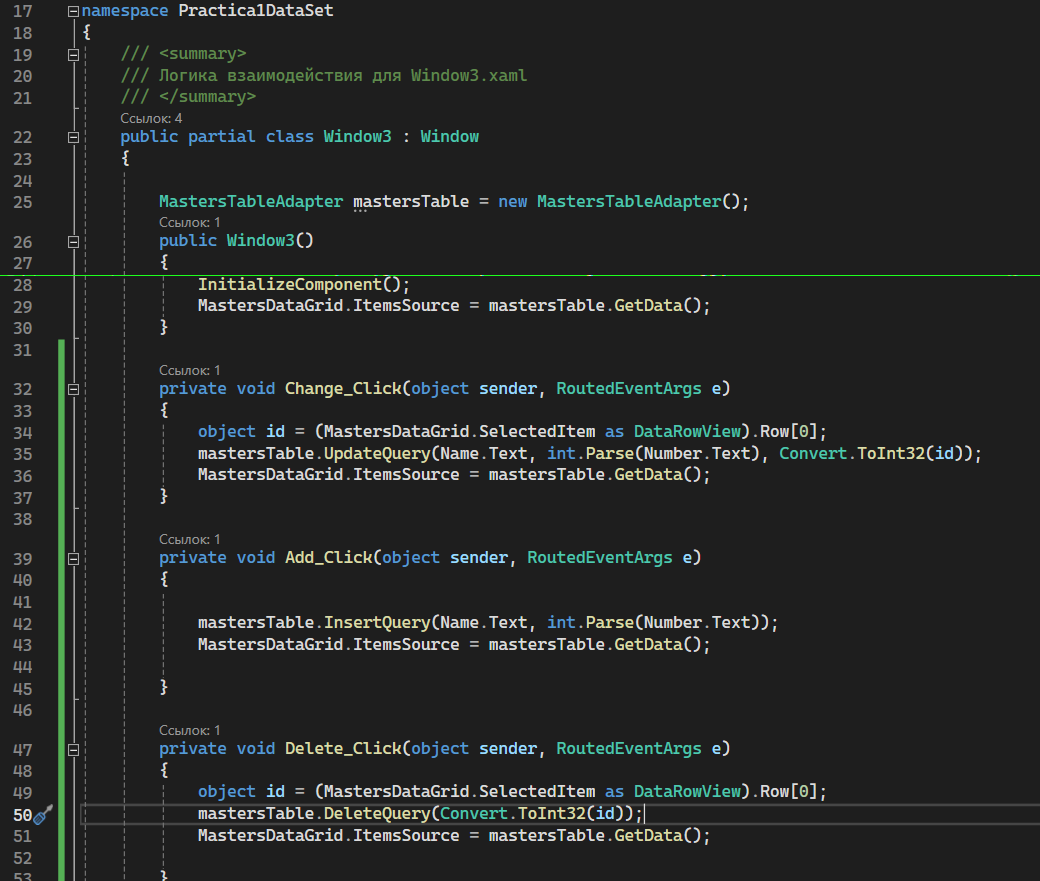
\

Рисунок - Методы 3го окна

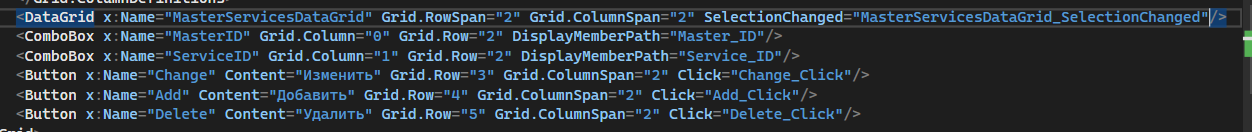


Рисунок - Создание кнопок

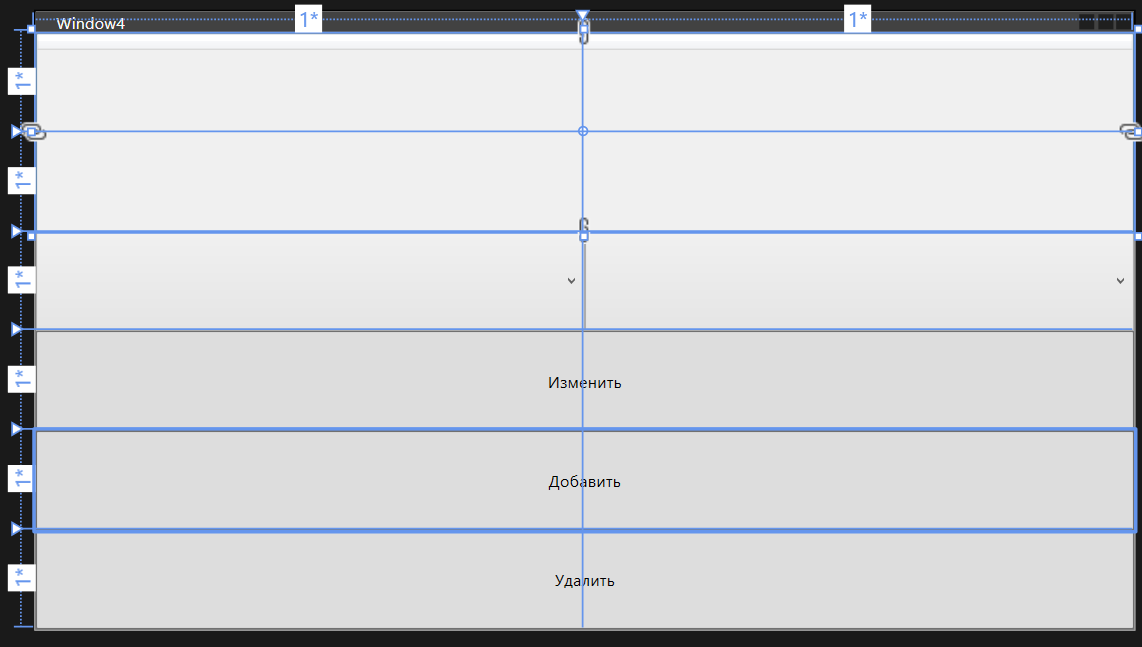


Рисунок - Вид 4го окна

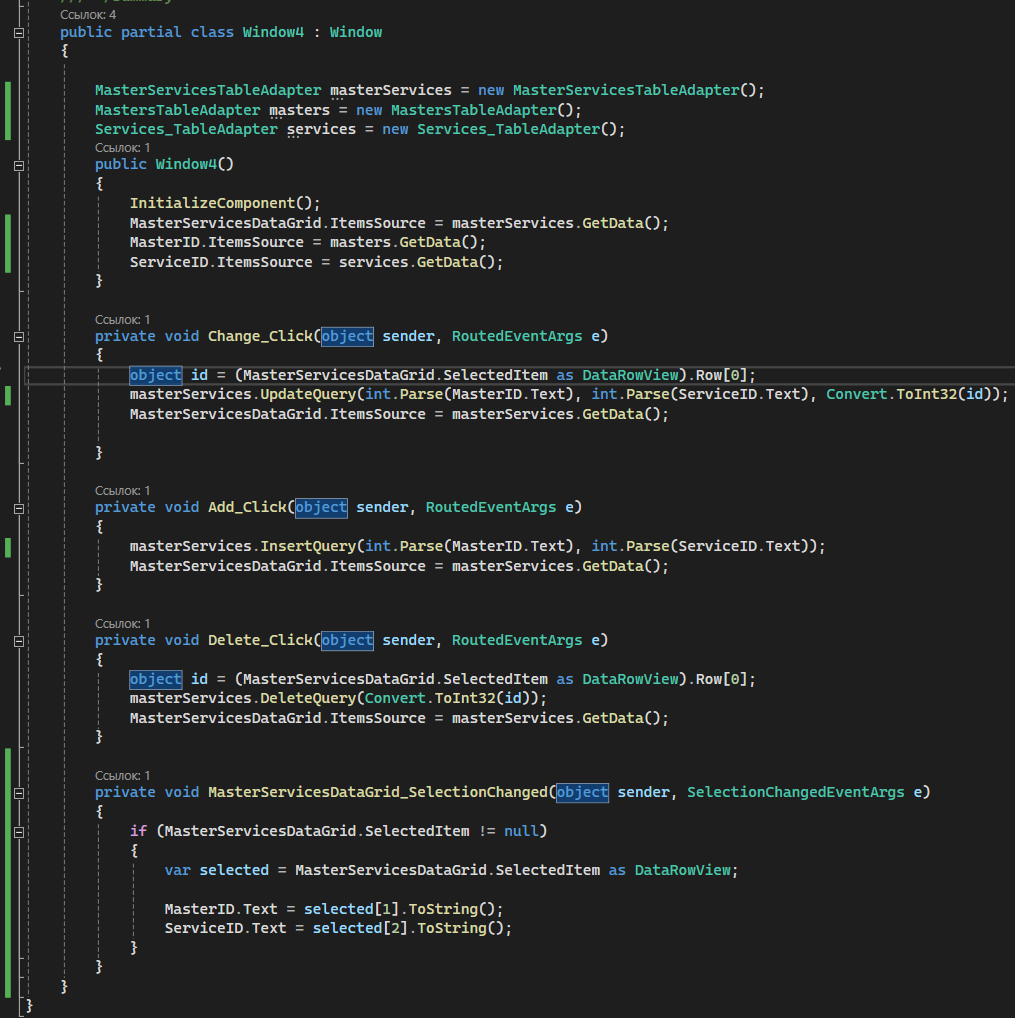


Рисунок - Методы 4го окна

Результат работы программы:

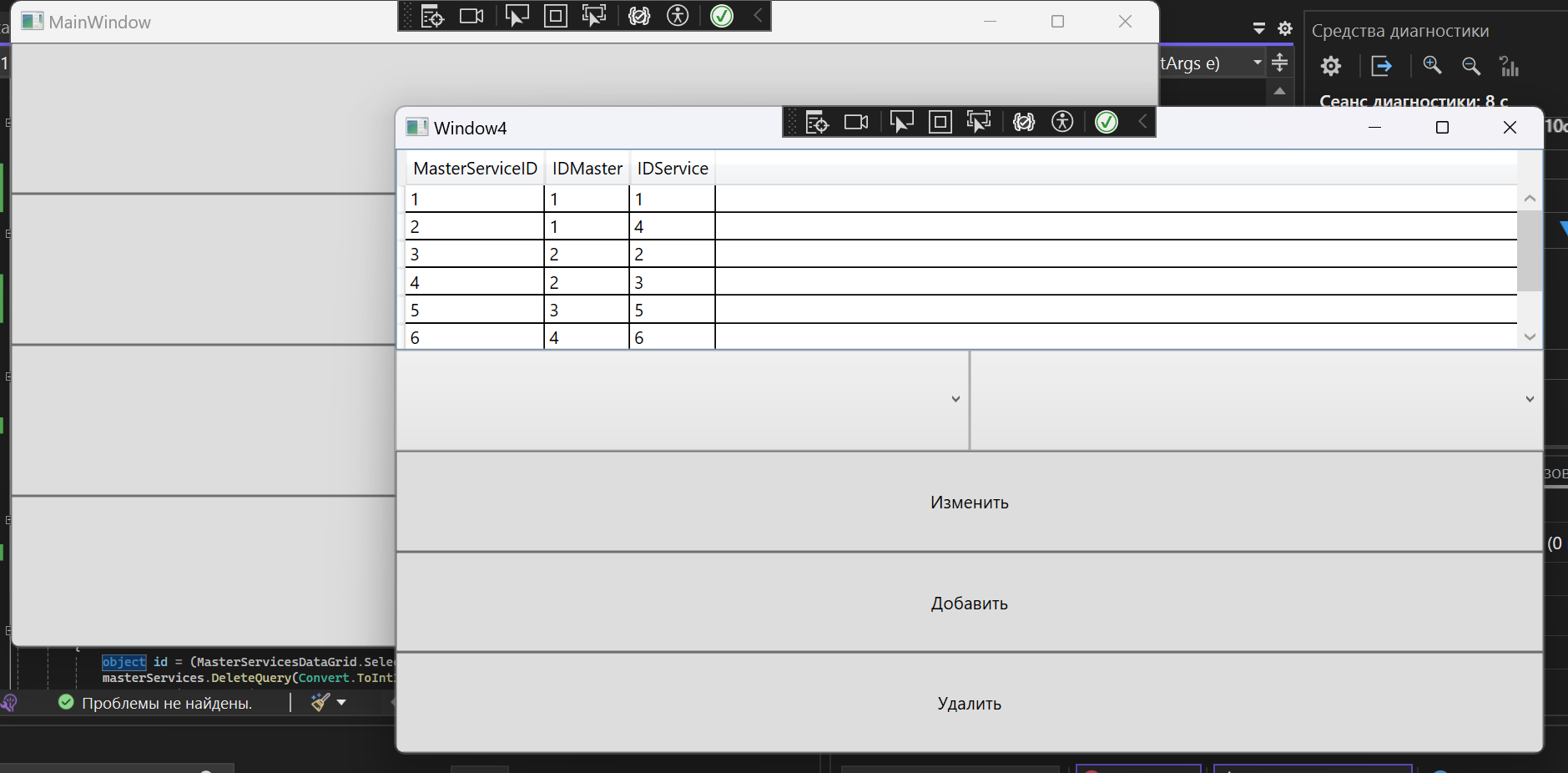


Рисунок – Результат

1. Приложение с Entity Framework

Необходимо также открыть приложение с прошлой работы. В каждом окне прописываем 2 текстбокса и 3 кнопки (копируем это с приложения DataSet, так как интерфейс остается неизменным).

Рассмотрим логику методов:

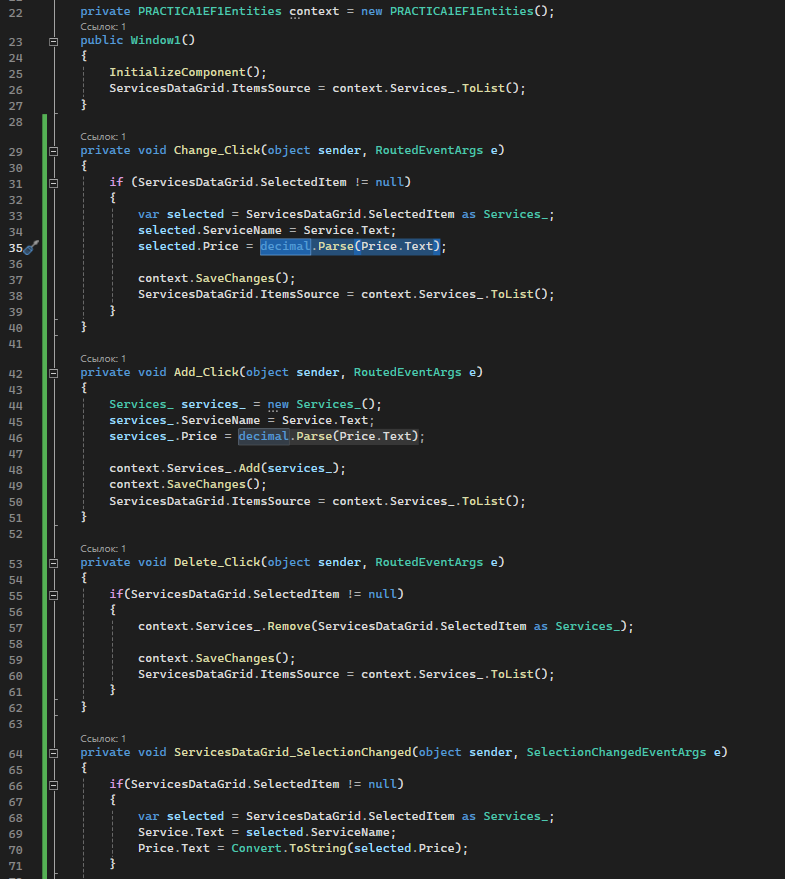


Рисунок - Методы 1го окна

Этот метод вызывается при нажатии кнопки "Change". Он изменяет выбранную запись в таблице Services\_ в базе данных и сохраняет изменения с помощью метода SaveChanges().

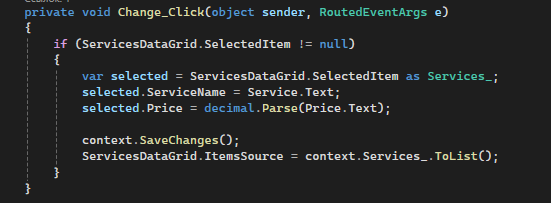


Рисунок - Метод изменения

Этот метод вызывается при нажатии кнопки "Add". Он создает новый объект Services\_, заполняет его данными из текстовых полей и добавляет его в таблицу Services\_ в базе данных.

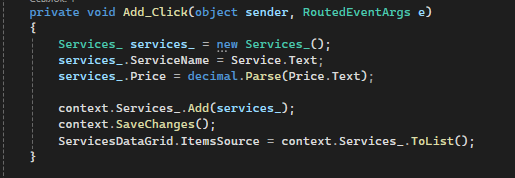


Рисунок - Метод добавления

Этот метод вызывается при нажатии кнопки "Delete". Он удаляет выбранную запись из таблицы Services\_ в базе данных и сохраняет изменения.

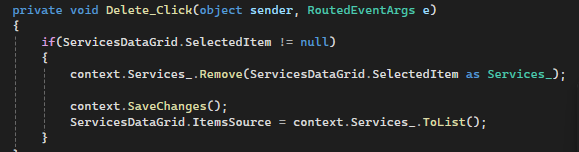


Рисунок - Метод удаления

Этот метод вызывается при изменении выбора в ServicesDataGrid. Он устанавливает значения текстовых полей в соответствии с выбранной записью.

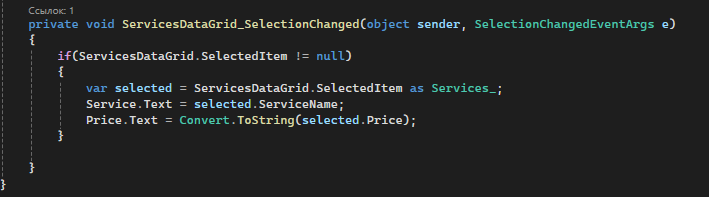


Рисунок - Метод который вызывается при изменении выбора ServicesDataGrid

По шаблону прописываем эти методы для остальных таблиц и окон:

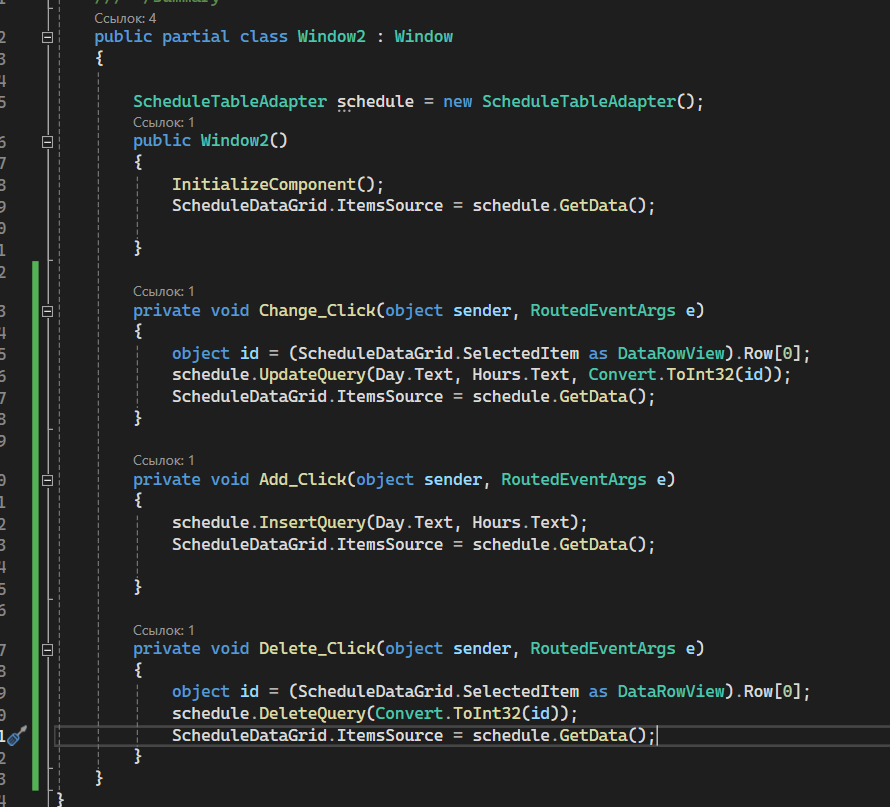


Рисунок - Реализация методов во втором окне

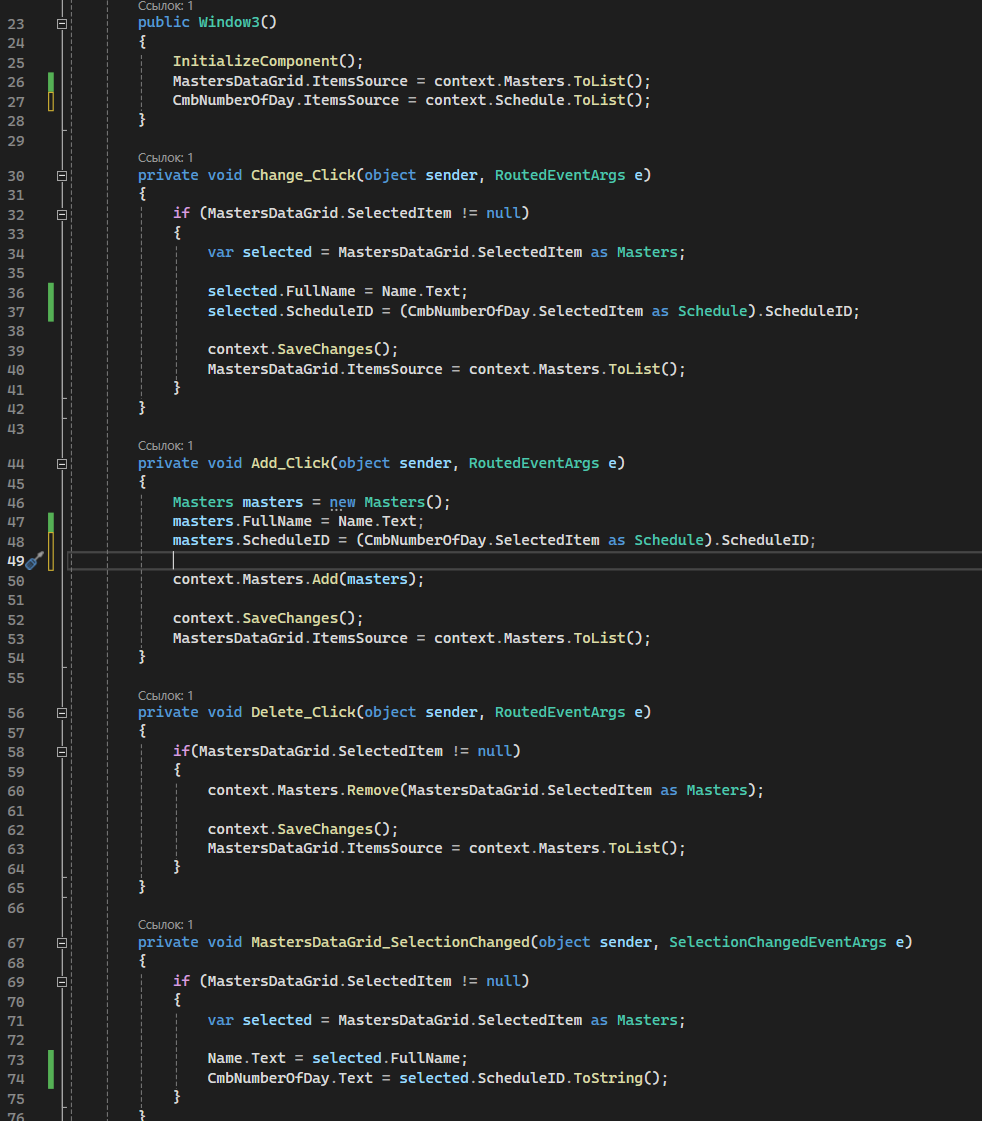


Рисунок - Реализация методов в 3 окне

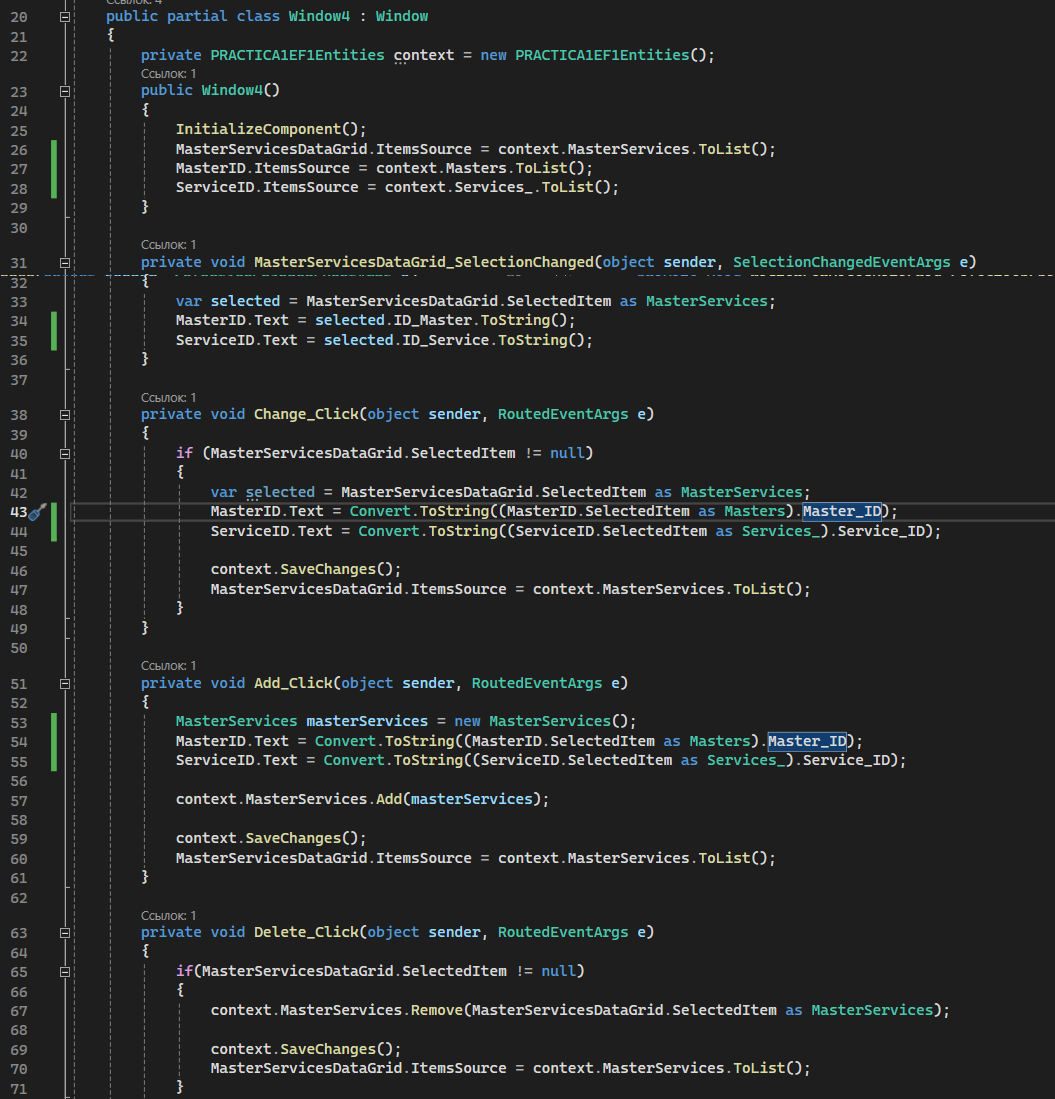


Рисунок - Реализация методов 4 окне

Вывод: Было реализовано приложение WPF, которое включает в себя базу данных с несколькими связанными таблицами, нормализованными до третьей нормальной формы. Приложение разработано в двух вариантах: с использованием Entity Framework и с использованием DataSet. Внутри приложения представлены различные окна или страницы, позволяющие пользователю взаимодействовать с разными таблицами. Данные из базы данных отображаются в элементе управления DataGrid, а также пользователь имеет возможность добавлять, изменять и удалять данные через интерфейс приложения.