|  |  |
| --- | --- |
| Департамент образования Ярославской области  Государственное профессиональное образовательное автономное учреждение  Ярославской области  «ЯРОСЛАВСКИЙ ПРОМЫШЛЕННО-ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ  им. Н.П. Пастухова» | |
| **Отчет по УЧЕБНОЙ практике**  **ПМ.02 Осуществление интеграции программных модулей**  УП.02.09.02.07.18ИП1.2 - 12 | |
|  | Студент  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_В.Н.Копанов  «\_\_\_\_» апреля 2021 г. |
| Руководитель практики  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.О. Куксов  «\_\_\_\_» апреля 2021 г. |
| 2021 | |

**Содержание**

[Практическая работа №1 3](#_Toc70507510)

[Практическая работа №2 9](#_Toc70507511)

[Практическая работа №3 12](#_Toc70507512)

[Практическая работа №4 14](#_Toc70507513)

# **Практическая работа №1**

**Тема:** подключение базы данных. Авторизация пользователя.

**Цель работы:** по заданной схеме данных разработать базу данных, создать программное приложение с возможностью авторизации в системе.

**Ход выполнения:**

Для выполнения практической работы была взята схема под номером 4\_4, представленная на рисунке 1.

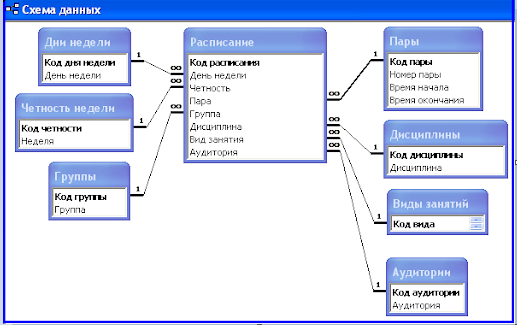


Рисунок 1 Схема данных

По данной схеме впоследствии, при помощи инструментов программного продукта MS SQL была создана диаграмма базы данных, которая представлена на рисунке 2.

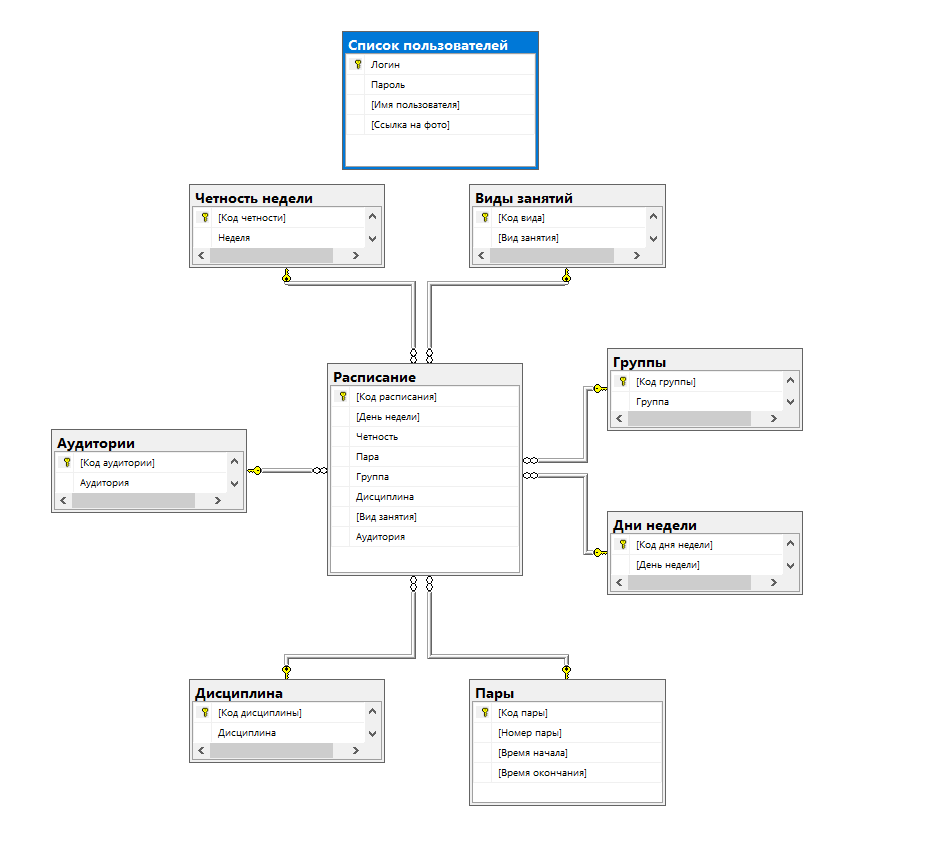


Рисунок 2 Диаграмма базы данных

После создания БД, она была заполнена, подключена к приложению, отображение модели БД, представлено на рисунке 3.

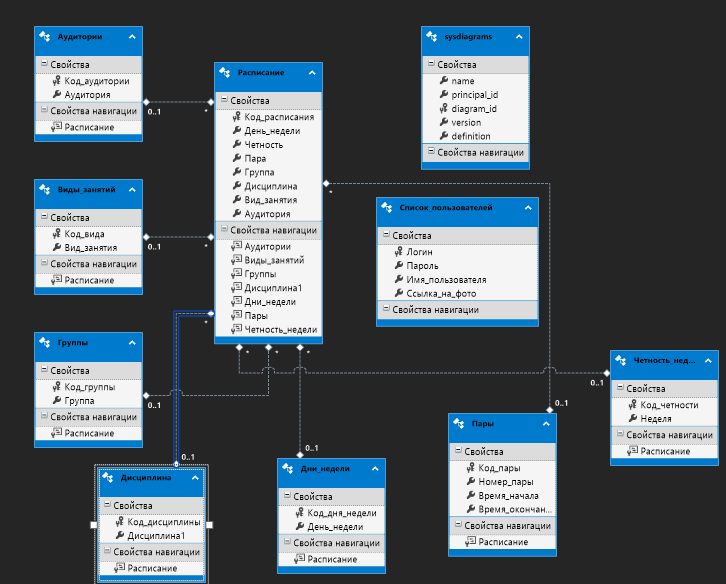


Рисунок 3 Модель БД в приложении

После завершения работы с БД и ее подключения к приложению, было создано окно «Авторизации», при помощи инструментов WPF C#, на форме авторизации размещены элементы «button»(Отвечающий за вход в систему), «textbox»(Отвечающий за введение логина), «passwordbox»(отвечающий за ввод пароля), «label»(Название формы), была реализована логика авторизации пользователя при помощи запроса к БД, а именно к таблице «Список\_Пользователей», которая хранит данные о логине, пароле и ссылки на фото пользователя. В качестве параметра с помощью лямбда выражения было задано условие отбора записей в таблице (в данном случае, если логин и пароль равны значениям в textbox и passwordbox). Авторизация реализована при помощи обращения к интерфейсу IQueryable<>,который принимает таблицу из бд и хранит в себе результат запроса к БД, при этом возвращает сколько раз данные значения повторялись в бд и повторялись ли вообше, реализация представлена на рисунке 4.

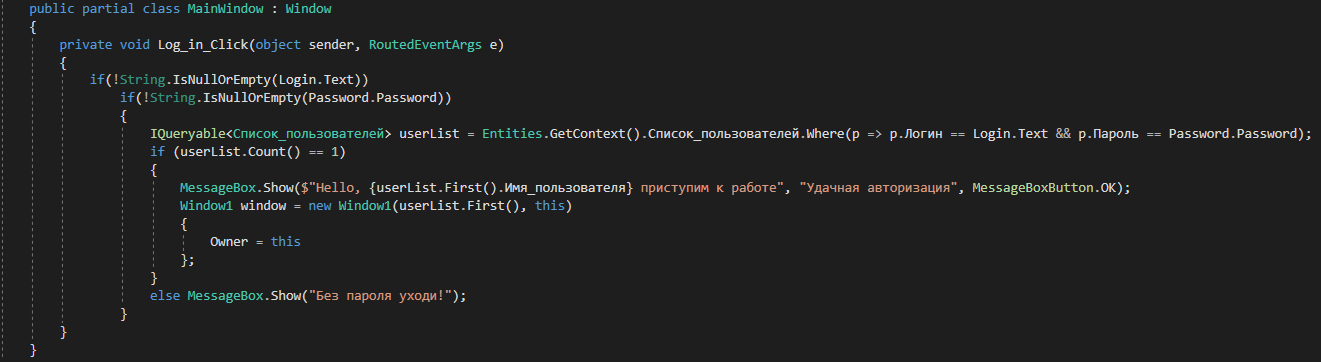


Рисунок 4 Реализация авторизации пользователя

Форма авторизации и процесс авторизации представлены на рисунках 4 и 5.

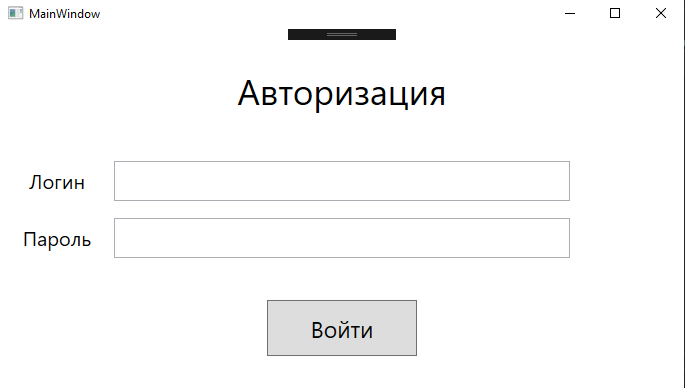


Рисунок 5 Форма авторизации

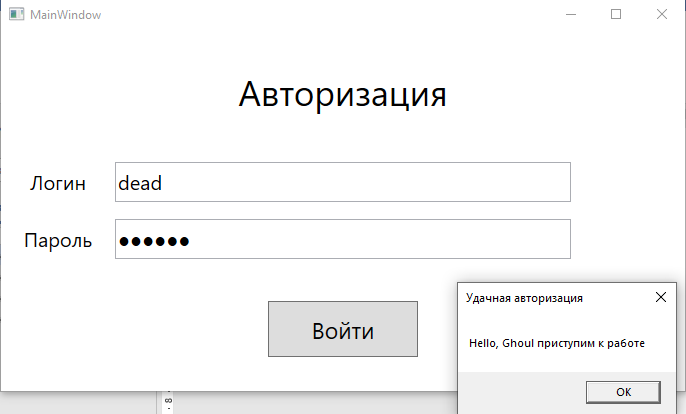


Рисунок 6 Процесс авторизации

**Вывод:** Исходя из схемы данных была реализована база данных, которая впоследствии была подключена к приложению, была реализована форма авторизации

**Контрольные вопросы**

1. Что такое «база данных»?

База данных - набор сведений, хранящихся некоторым упорядоченным способом. Можно сравнить базу данных со шкафом, в котором хранятся документы.

1. Из каких этапов состоит проектирование базы данных?

* Концептуальное (инфологическое) проектирование
* Логическое (даталогическое) проектирование
* Физическое проектирование

1. Что подразумевается под таблицей и полем в SQL?

Таблица SQL – структурированный набор данных в виде таблицы, столбцы данной таблицы являются полями SQL.

1. Что такое первичный ключ (Primary key)?

Первичный ключ - столбец, значения которого во всех строках различны. Первичные ключи могут быть логическими (естественными) и суррогатными (искусственными).

1. Что такое внешний ключ (Foreign key)?

Внешний ключ или FOREIGN KEY также является атрибутом ограничения и обеспечивает связь двух таблиц. По сути, это поле или несколько полей, которые ссылаются на PRIMARY KEY в родительской таблице.

1. Какие связи существуют в реляционных базах данных?
2. Что подразумевается под целостностью данных?
3. Что такое нормализация и каковы ее преимущества?
4. Как создать подключение к базе данных?

# **Практическая работа №2**

**Тема:** отображение данных на форме.

**Цель работы:** реализовать отображение данных базы данных на форме.

**Ход работы:**

Была создана форма, содержащая в себе основной функционал программы, данная форма представлена на рисунке 7

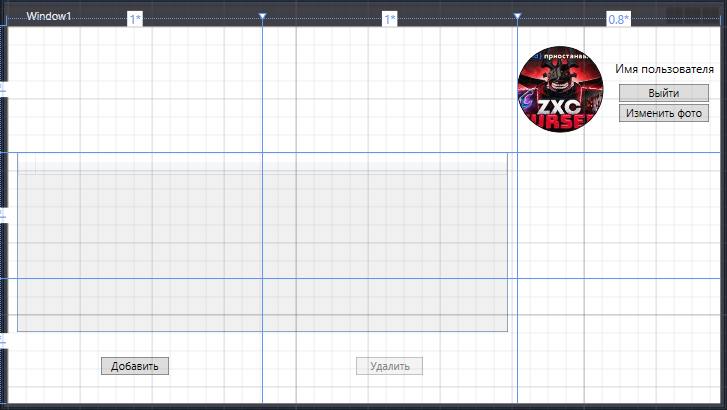


Рисунок 7 Форма профиля

На данной форме присутствуют такие элементы как «datagrid», необходимый для отображения информации о расписании, «ellipse», необходимы для отображения фото и элементы «button» и «label» для выхода из профиля и для отображения информации о пользователе. Также на форме находятся элементы «button» для практической работы № 3 и № 4, они необходимы для изменения фотографии профиля, а также для создания и удаления новых заказов.

После создания формы, была реализована логика добавления фотографии профиля на элемент «ellipse» в случае, если в базе данных присутствует ссылка на фотографию, а также логика внесения расписания в элемент «datagrid», данная логика была реализована при помощи запросов к базе данных с последующим ее изменением и подкачки новых данных их БД, и автогенерации столбцов в элементе «datagrid», результат реализации представлен на рисунке 8, а реализация добавления фото представлена на рисунке 9

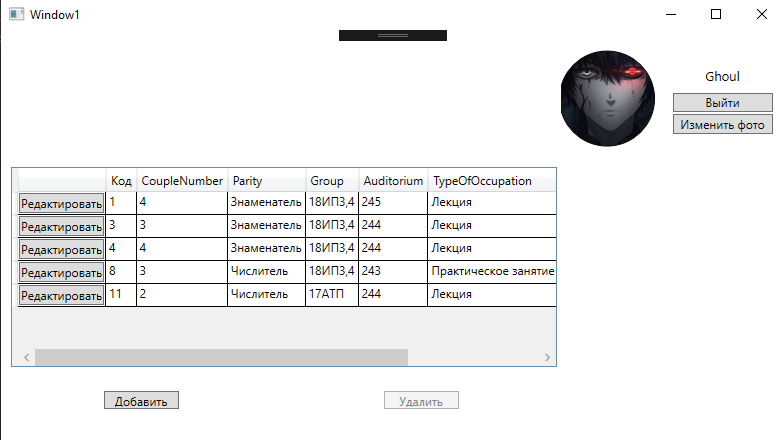


Рисунок 8 Форма профиля, с фотографией и расписанием

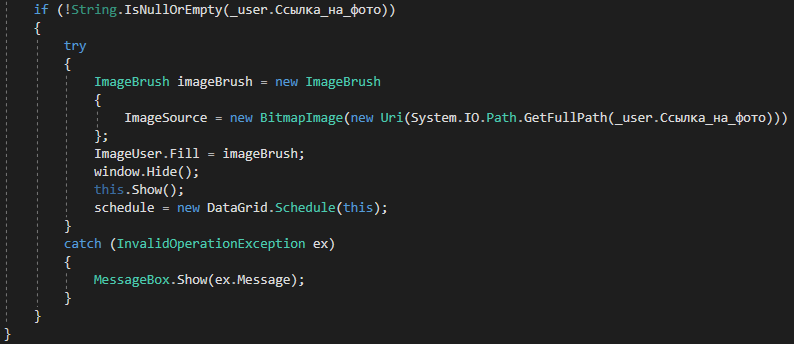


Рисунок 9 Реализация добавления фото

После реализации формы профиля был реализован вывод информации в MessageBox для просмотра информации о паре, указанной в расписании. в которых указывается информация о времени пары. После создания формы была реализована логика добавления информации о продукте в соответствующие элементы «label» и добавления картинки в «rectangle», при помощи запросов к базе данных, также была реализована логика вызова MessageBox, при помощи события, которое находится в столбце “CoupleNumber” в «datagrid» Результат реализации и соответственно сам MessageBox представлена на рисунке 10.

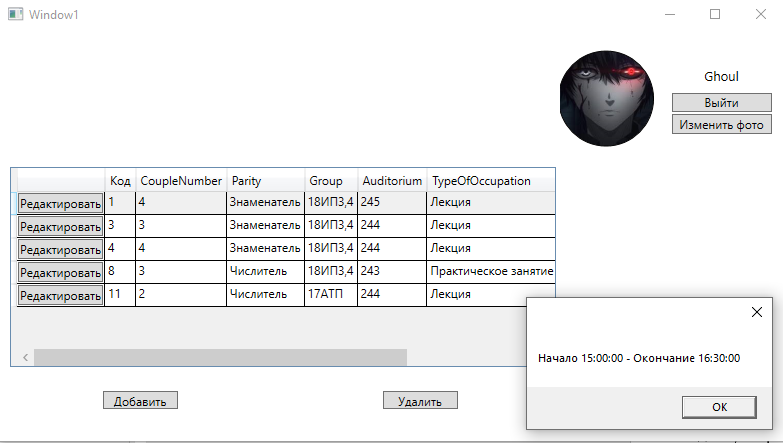


Рисунок 10 Форма с информацией о времени пары.

**Вывод:** была создана форма профиля и вызов MessageBox’a с информацией о времени пары, реализовано отображение информации о расписании в элементе «datagrid», а также отображение изображения, привязанного к профилю.

**Контрольные вопросы**

1. Что такое Windows Presentation Foundation? В чем преимущество перед WinForms?
2. Что такое XAML? Для чего он используется в WPF?
3. Что такое LINQ? Как работает технология LINQ?

# **Практическая работа №3**

**Тема:** редактирование данных в БД с помощью приложения.

**Цель работы:** реализовать редактирование данных из базы данных с помощью программного приложения.

**Ход работы:**

Для редактирования заказов была создана форма, содержащая элементы «Combobox», представляющие из себя список с определёнными элементами, в данном случае со всеми полями в таблице «Расписание» и всеми связанными с ней таблицами.

Был создан метод UpdateDataGrid(), в котором происходит запрос к БД для заполнения DataGrid информацией из БД, также происходит подписание DataGrid на событие Mouse\_DoubleClick, данный метод необходим для обновления базы данных. Реализация метода представлена на рисунке 11.

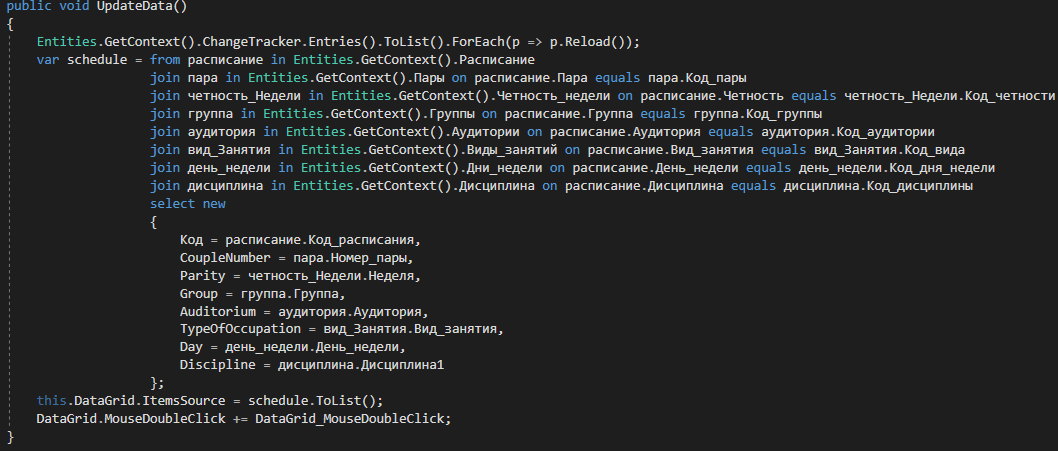


Рисунок 11 Метод UpdateData

Для редактирования расписания в элементе «datagrid» была создана кнопка перехода к форме для редактирования. Форма редактирования представлена на рисунке 12

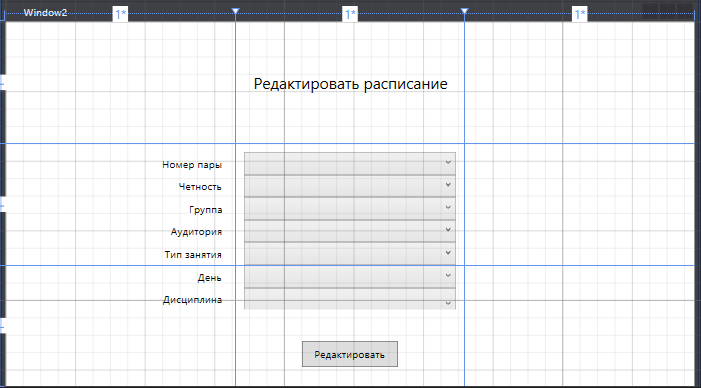


Рисунок 12 Форма редактирования

По нажатию кнопки «Редактировать» происходит обновление изменённых параметров базы данных, после чего обновлённые данные заносятся в расписание.

Также была реализована логика изменения фото профиля, по нажатию кнопки «Изменить фото», всплывает окно, с выбором объекта, после замены фото происходит обновление БД, и новое фото заменяется старым.

Новое фото, представлено на рисунке 13

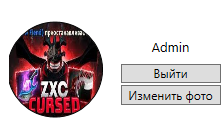


Рисунок 13 Новое фото профиля

**Контрольные вопросы**

1. Что такое «реляционная алгебра»? Для чего она используется?
2. Перечислите основные операции реляционной алгебры. Расскажите подробно об одной из них.
3. Какую операцию реляционной алгебры вы использовали в данной работе и зачем?

# **Практическая работа №4**

**Тема**: добавление и удаление данных из БД с помощью приложения.

**Цель работы**: реализовать добавление и удаление данных из базы данных с помощью программного приложения.

**Ход работы:**

Для добавления новых заказов была создана форма, содержащая 7 элементов «combobox», содержащая все поля таблицы «Расписание».

Форма добавления предмета реализована при помощи запросов к базе данных, а также её обновления программно. Форма нового заказа представлена на рисунке 14

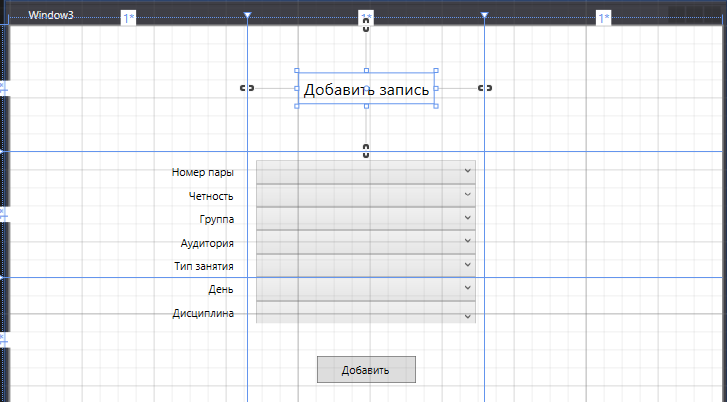


Рисунок 14 Форма создания нового заказа

Создание нового заказа представлено на рисунке 15

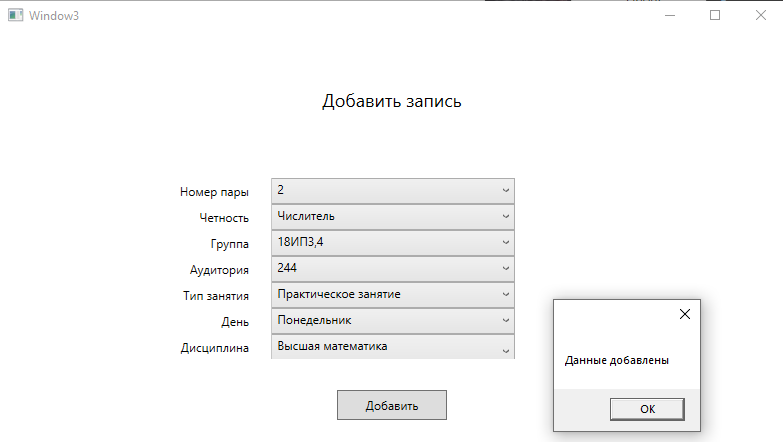


Рисунок 15 Создание заказа

Созданный заказ представлен на рисунке 16

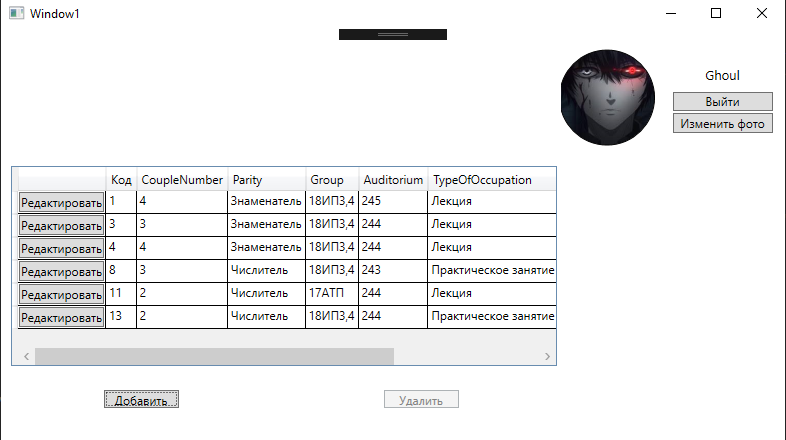


Рисунок 16 Созданный в «datagrid» заказ

А также на форме профиля добавлены 2 кнопки для добавления данных (переход на новую форму) и удаления данных. Кнопка удаления не активна, пока не будет выбрана строка в DataGrid.

При помощи свойства Remove(), была реализована возможность удаления заказа по нажатию кнопки “Удалить”. Данный функционал представлен на рисунках 17, 18.

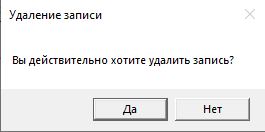


Рисунок 17 Удаление данных

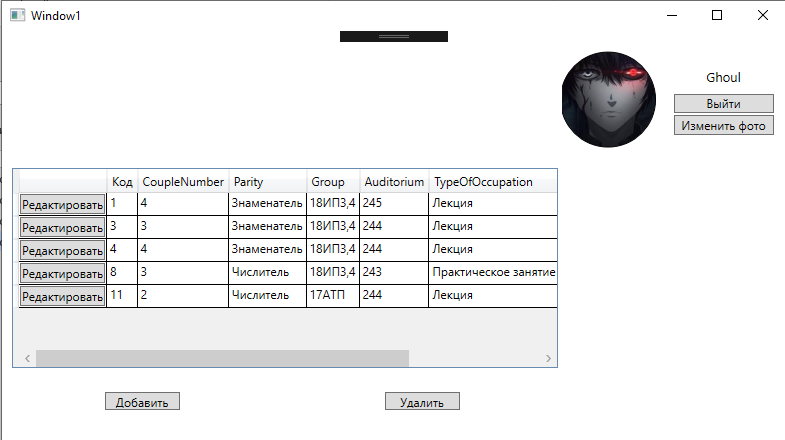


Рисунок 18 Удаление данных

**Вывод:** была создана новая форма, предназначенная для добавления новой пары, а также добавлены две кнопки, одна для перехода на форму создания нового расписания, другая для удаления расписания.

**Контрольные вопросы**

Опишите структуру команды «INSERT INTO»

Опишите структуру команды «DELETE»