**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра «Математическая Кибернетика и Информационные технологии»

Лабораторная работа №8

Создание визуального интерфейса для базы данных

Выполнил: Студент группы

БВТ2201

Ибрагимов Т.Э.

Москва

2023

**Цель работы:** создать визуальный интерфейс для базы данных.

**Задание:**

1. Использование библиотеки PyQt5.
2. Использование адаптера psycopg2.
3. Приложение должно иметь при себе функционал позволяющий: просматривать базу данных в удобном для пользователя формате, удалять, добавлять и изменять записи в этой же базе данных.
4. Визуальная часть должна иметь при себе:
   1. Минимум 3 вкладки, в каждой из которых содержится информация из отдельной таблицы в базе данных.
   2. Внутри каждой вкладки информация должна отображаться в виде таблиц.
   3. Внутри каждой вкладки должна отображаться кнопка с обновлением информации.
   4. Внутри каждой таблицы должны отображаться все поля из таблицы в базе данных в виде колонок
   5. Внутри каждой таблицы после каждой строки записи должны быть отображены кнопки изменения и удаления записи
   6. В конце каждой таблицы должна находиться пустая строка с кнопкой для добавления новой записи.
   7. На вкладке с расписанием дни недели должны быть указаны в отдельных таблицах.

**Ссылка на GitHub со всеми лабами: https://github.com/dayerzz/Laby\_po\_VvIT\_2Sem**

**Ход работы**

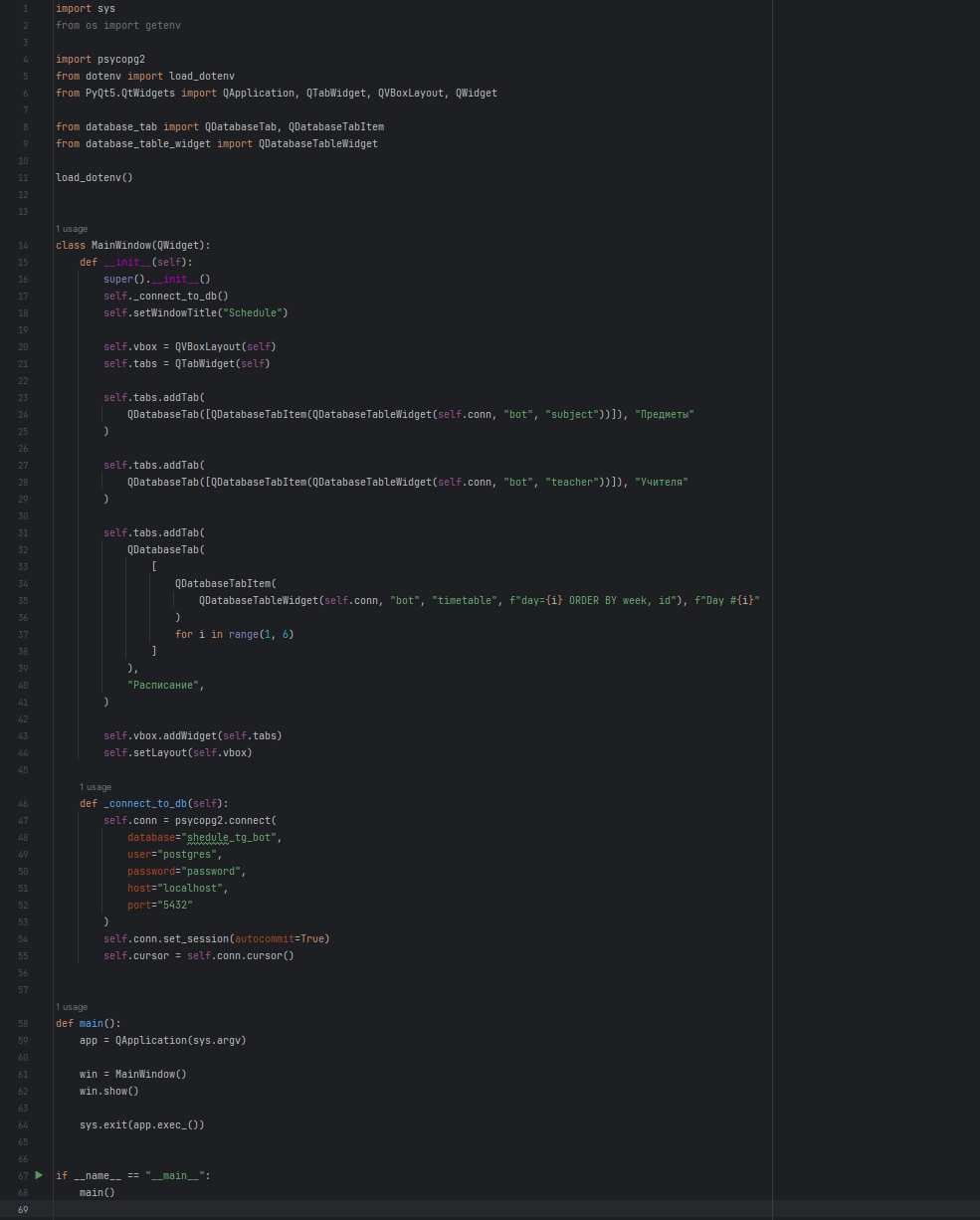


Рисунок 1 – код файла main.py

В файле main.py происходит подключение к БД, импортирование всех необходимых библиотек, а также создание 3 вкладок с таблицами (Предметы, Учителя и Расписание).

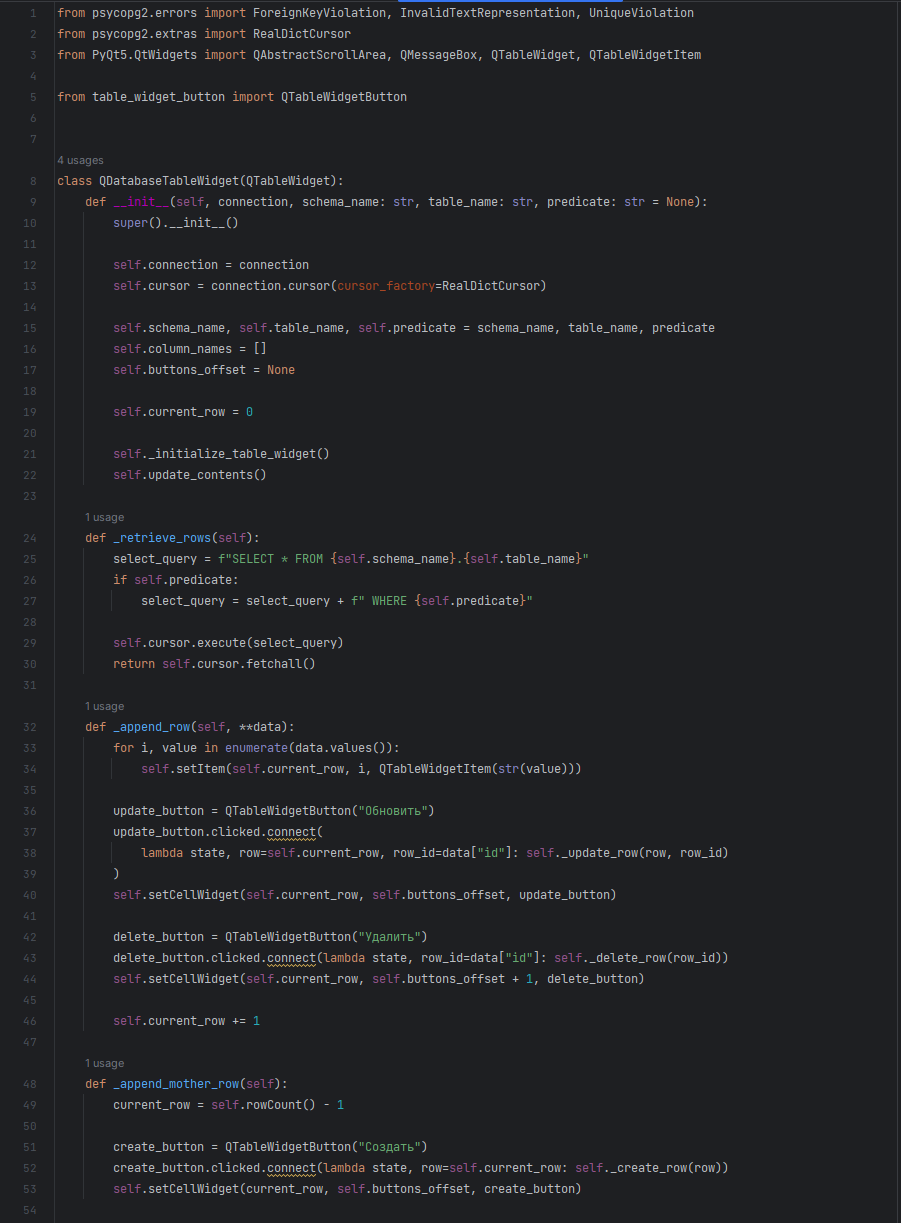


Рисунок 2 – код файла database\_table\_widget.py (часть 1)

В файле database\_table\_widget.py находится код для создание кнопок («Обновить», «Создать» и «Удалить»), каждая из которых служит для определённого вноса измениний в БД. Функция def \_retrieve\_rows(self) служит для получения списка столбцов из БД. Функция def \_initialize\_table\_widget(self) служит для создания виджета таблицы. Функция def update\_content(self) служит для обновления содержимого в БД. А также в этом файле есть исключение ошибок при обновлении или удалении данных в БД.

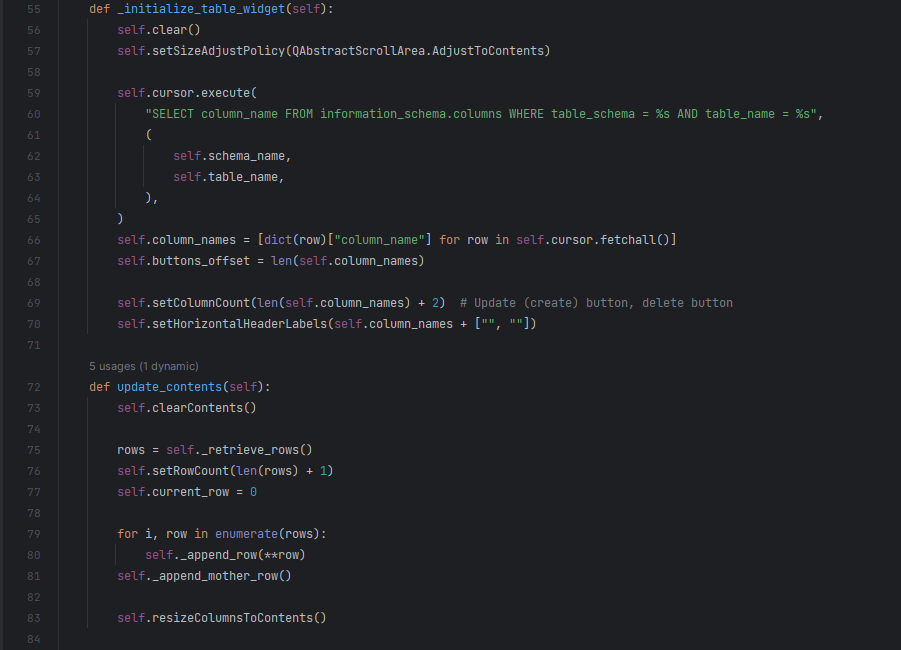


Рисунок 3 – код файла database\_table\_widget.py (часть 2)

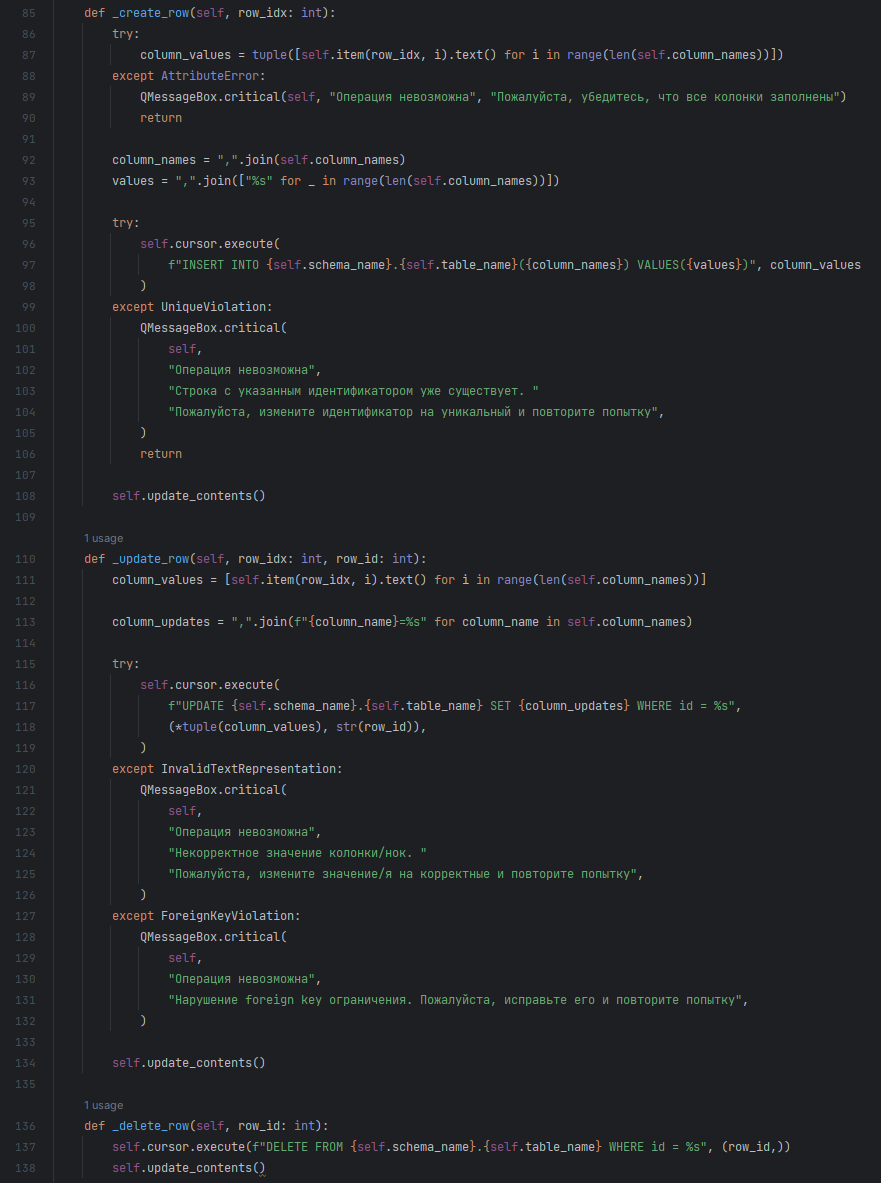


Рисунок 4 – код на вывод ошибок при удалении или обновлении записей в БД из файла database\_table\_widget.py

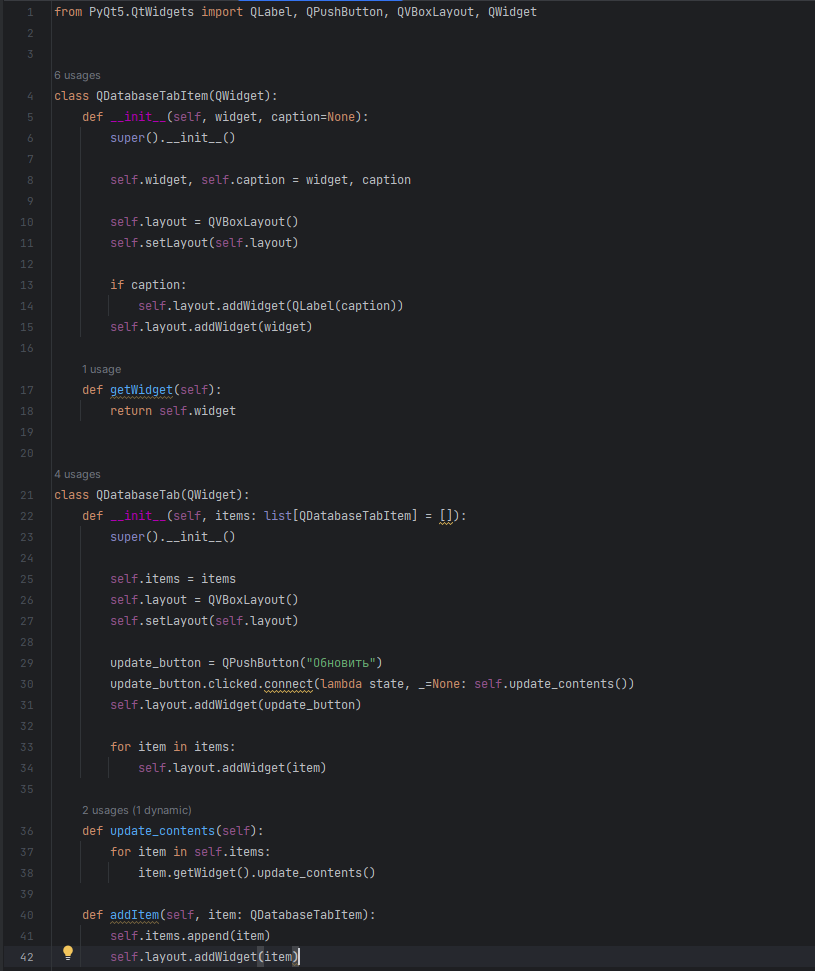


Рисунок 5 – код файла database\_tab.py

В файле database\_tab.py содержится код для полного обновления данных внесённых в таблицу.

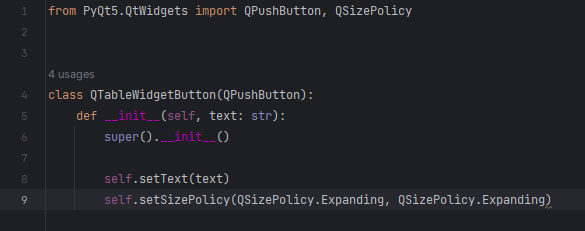


Рисунок 6 – код файла table\_widget\_button.py

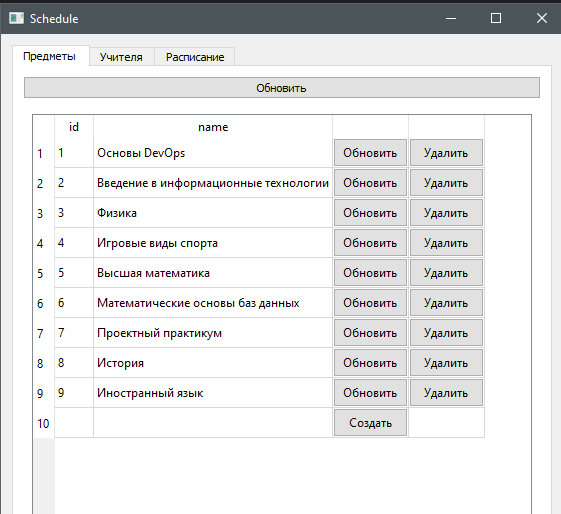


Рисунок 7 – вид вкладки «Предметы»

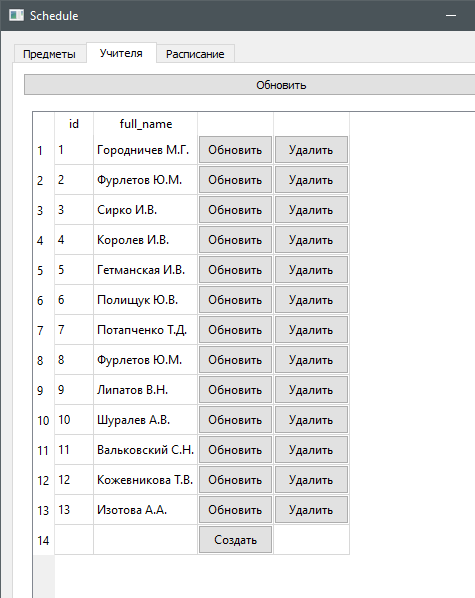


Рисунок 8 – вид вкладки «Учителя»

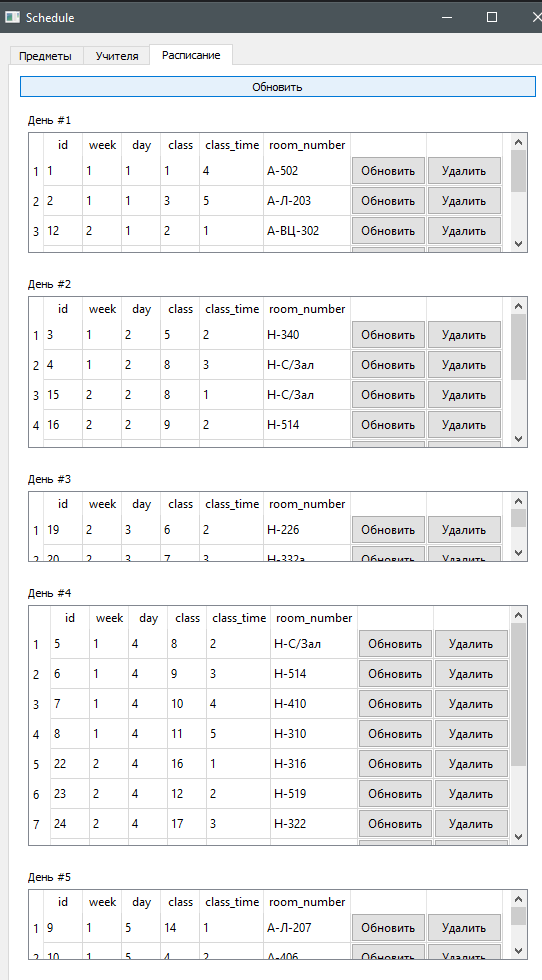


Рисунок 9 – вид вкладки «Расписание»

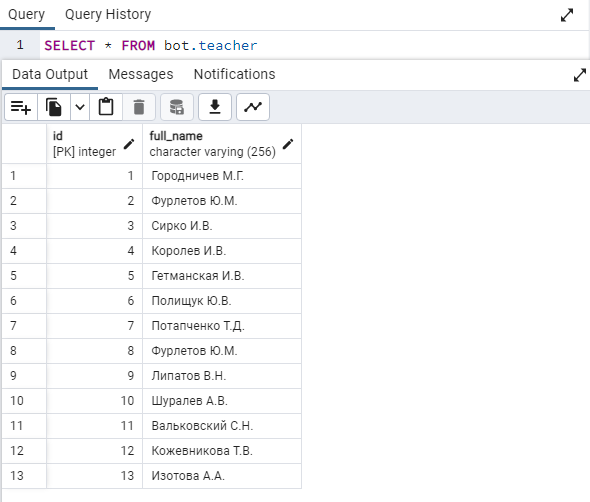


Рисунок 10 – вид таблицы teacher в БД до вноса изменений

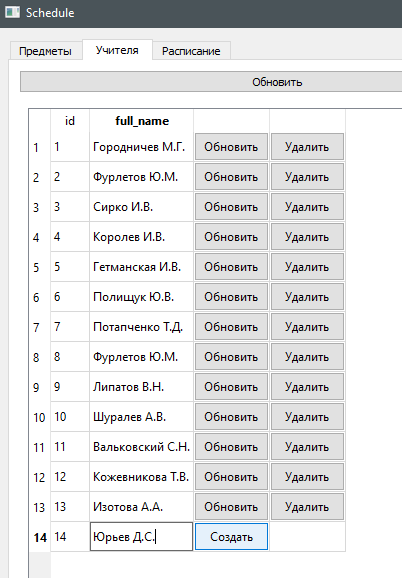


Рисунок 11 – добавление новой записи в таблицу teacher

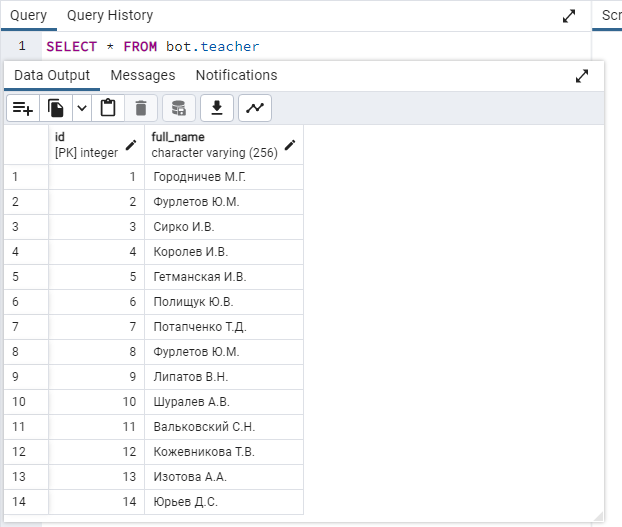


Рисунок 12 – вид таблицы teacher в БД после вноса изменений

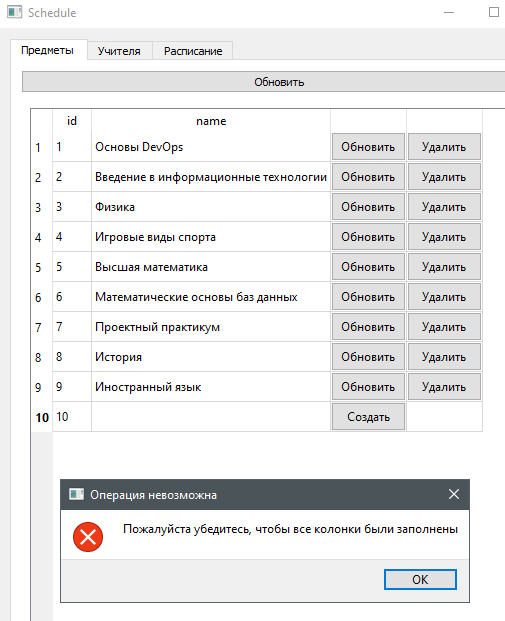


Рисунок 13 – вывод ошибки при неполном добавлении новых данных в БД

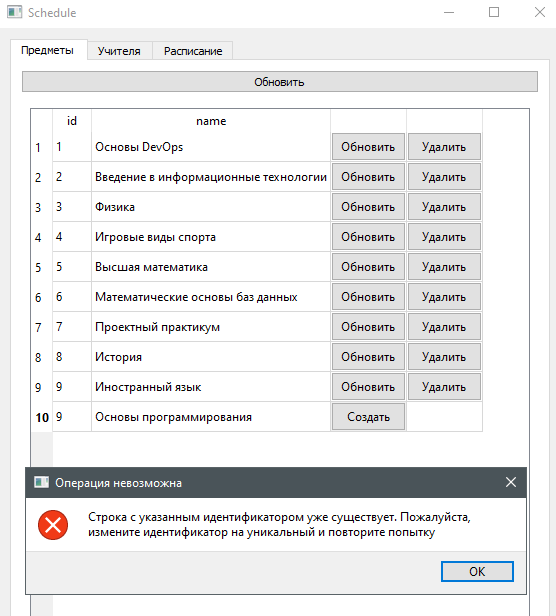


Рисунок 14 – вывод ошибки при добавлении данных в БД с уже существующим id

**Вывод:** в данной лабораторной работе я сделал собственный визуальный интерфейс для БД с расписанием использую различные библиотеки Python.