Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №6**

**«Восстановление системы»**

**«МДК 06.02** **Инженерно-техническая поддержка сопровождения информационных систем»**

Выполнил: студент учебной группы

ИСПк-302-52-00

Холин П.Э.

Преподаватель:

Самоделкин П.А.

Киров

2024

**Цель работы -** изучение приемов и методов организации непрерывной работы информационных систем, подлежащих защите.

**Задание**:

1. Самостоятельно найти (github) и локально развернуть 3 программы с использованием СУБД SQLite. Проанализировать работу программ, изучить SQL-запросы, рассмотреть структуру БД. Для работы с СУБД SQLite потребуется программа DB Browser for SQLite.
2. Отчет сдать преподавателю.

**Результаты выполнения задани****я**

В ходе выполнения работы были развернуты три информационные системы, использующтие СУБД SQLite.

1. Первая система – sqliteBasics (доступна по ссылке https://github.com/MariyaSha/sqliteBasics)

В проекте присутствует два файла: main.py и select.py.

Файл main.py содержит основные действия с БД. В начале происходит подключение к БД. Если файл не существует, то он будет создан.

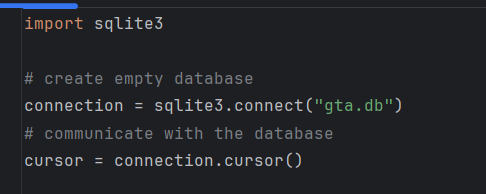


Рис. 1 – Подключение к БД

Далее идут данные для заполнения. release\_list содержит данные о годах выпуска игр, их названиях и вымышленных городах, в которых разворачивается действие. city\_list сопоставляет вымышленные города из игр реальным городам.



Рис. 2 – Данные для заполнения.

Создается таблица gta с тремя столбцами:

* release\_year (год выпуска игры),
* release\_name (название игры),
* city (вымышленный город).

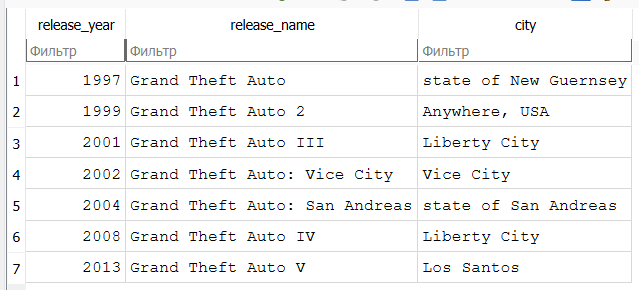


Рис. 3 – Структура таблицы gta

Таблица cities сопоставляет вымышленные города (gta\_city) и реальные города (real\_city).

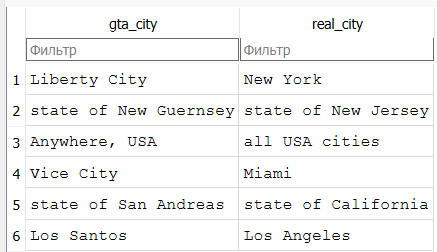


Рис. 4 – Структура таблицы cities

В файле select.py код демонстрирует загрузку существующей базы данных SQLite (gta.db), чтение данных из таблицы gta и вывод их в консоль.

Сначала происходит подключение к БД gta.db. Затем выполняется SQL-запрос.

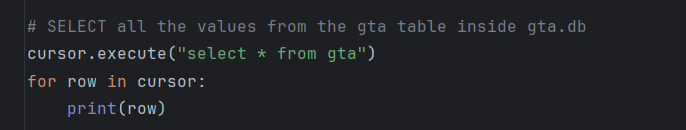


Рис. 5 – SQL-запрос

Запрос извлекает все строки и столбцы из таблицы gta. Далее итерация по результатам запроса. Каждый результат (row) — это кортеж, содержащий одну строку таблицы. В итоге результат выводится в консоль.

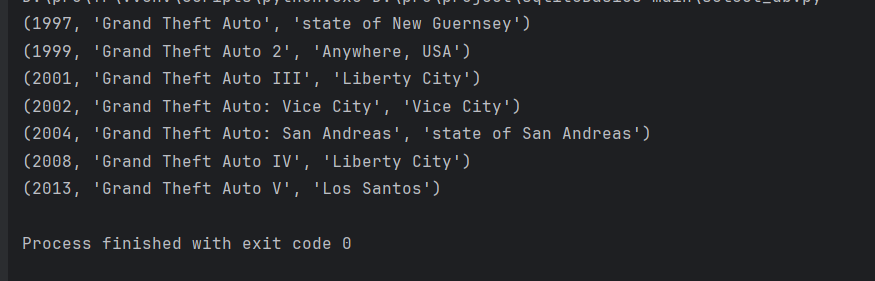


Рис. 6 – Результат запроса

1. Следующая система – pythontodolist (https://github.com/armandvh/pythontodolist)

Система представляет из себя приложение "ToDoList" на Python с графическим интерфейсом, используя библиотеки tkinter и tkcalendar. Приложение позволяет добавлять, удалять и загружать задачи, а данные сохраняются в базе данных SQLite.

В начале происходит создание главного окна. Затем подключение к базе данных. Если файл базы данных отсутствует, он создается автоматически. Также создается таблица tasks (если еще не существует) с колонками:

– Task (Описание задачи.

– time (Время выполнения).

– priority (Приоритет задачи).

– category (Категория задачи).

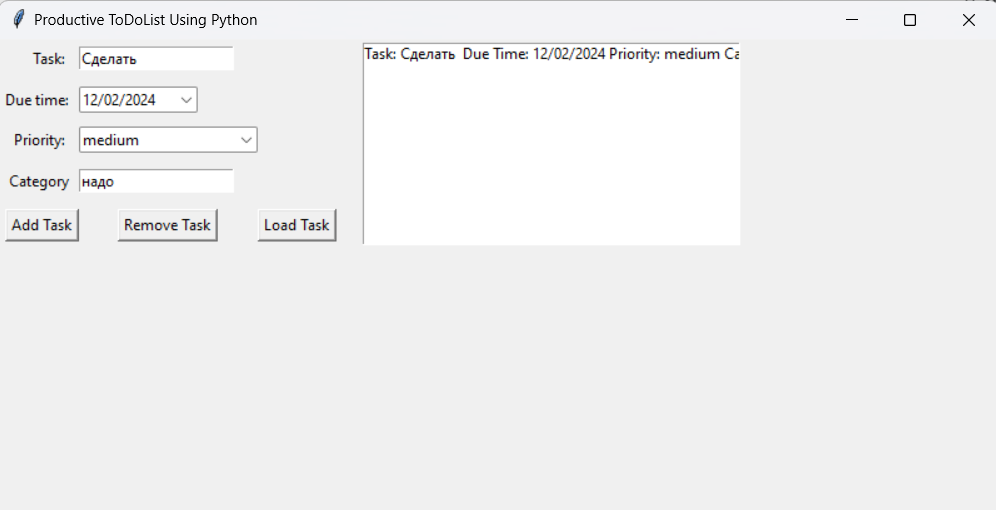
****

Рис. 7 – Главное окно приложения

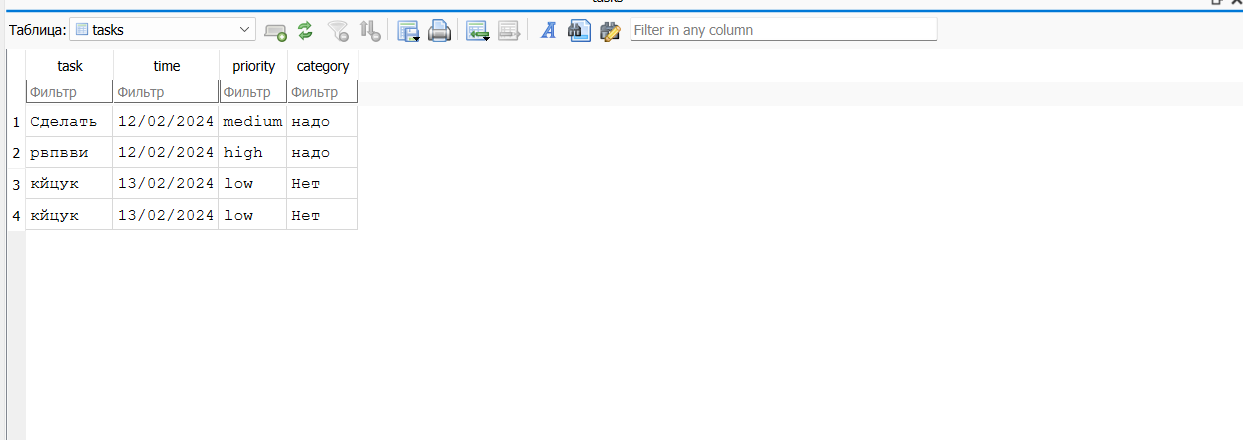


Рис. 8 – Структура таблицы tasks

1. Последняя система – StudentManagementAppSQL (https://github.com/Qde13/StudentManagementAppSQL).

Программа представляет собой систему управления студентами (добавление, редактирование, удаление, поиск) с использованием PyQt6 для создания графического интерфейса и SQLite для хранения данных.

В начале происходит подключение к БД database.db. Затем создается главное окно приложения. В системе реализованы следующие функции:

– Загрузка данных: при запуске данные загружаются из таблицы students базы данных SQLite и отображаются в таблице.

– Добавление: через диалог добавления пользователь может ввести данные нового студента, которые сохраняются в базе данных.

– Редактирование: пользователь выбирает строку таблицы, нажимает "Edit Record", изменяет данные через диалог редактирования, и изменения сохраняются.

– Удаление: пользователь выбирает строку таблицы, нажимает "Delete Record" и подтверждает удаление.

– Поиск: осуществляется поиск по имени.

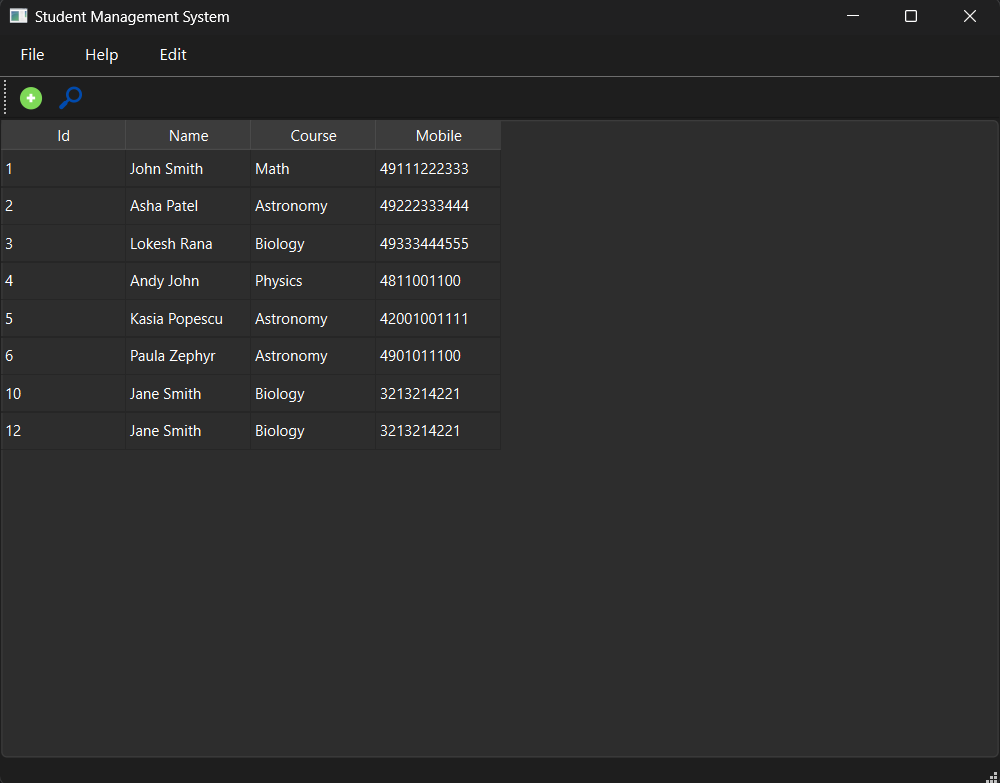


Рис. 9 – Главное окно приложения

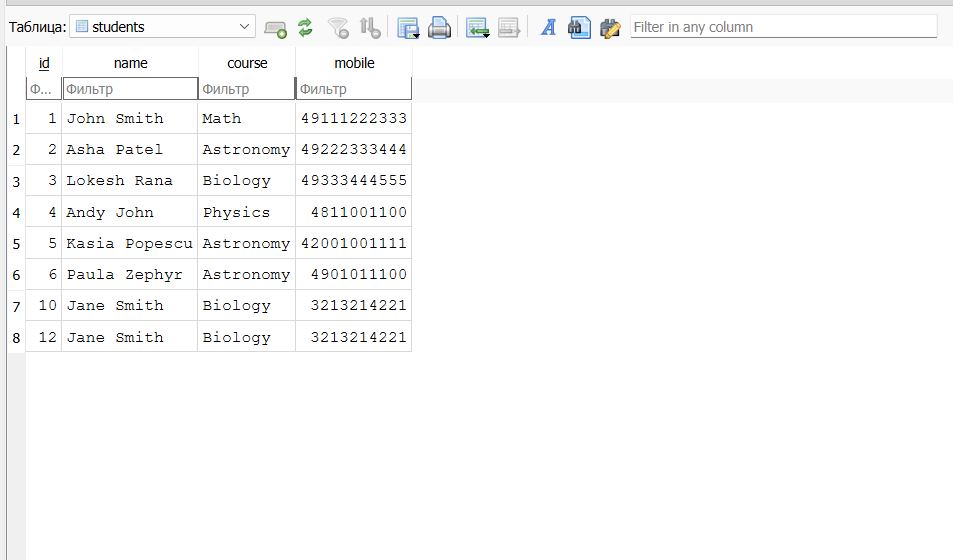


Рис. 10 – Структура таблицы students

**Выводы по работе**

В ходе работы были рассмотрены несколько информационных систем, использующих СУБД SQLite. Данные системы также были развернуты на локальной машине для изучения их структуры, функциональности.