# МИНИCTEPCTBO НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное образовательное учреждение высшего образования**

# «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

**Кафедра инфокоммуникаций**

# Лабораторная работа 2.21

Взаимодействие с базами данных SQLite3 с помощью языка программирования Python

|  |
| --- |
| Выполнил студент группы ИВТ-б-о-21-1 |
| Пентухов С. А. « » 20 г. |
| Подпись студента |
| Работа защищена « » 20 г. |
| Проверил Воронкин Р.А.  (подпись) |

**Цель работы:** исследовать базовые возможности системы управления базами данных SQLite3.

# Ход работы

1. Добавим магазины и товары в них с помощью команд консоли, а затем отобразим содержимое таблицы.

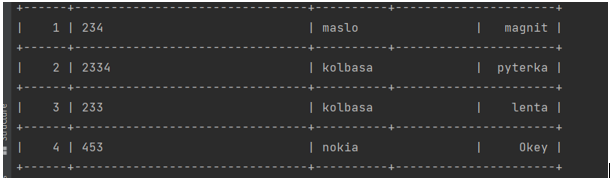


Рис. 1 Содержимое таблицы с магазинами

При помощи команды выбора магазина найдем нужный магазин.

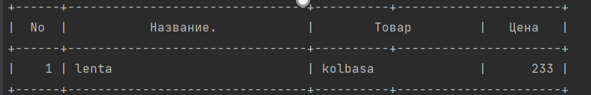


Рис. 2 Нужный магазин найден

Задание повышенной сложности

Самостоятельно изучите работу с пакетом python-psycopg2 для работы с базами данных PostgreSQL. Для своего варианта лабораторной работы 2.17 необходимо реализовать возможность хранения данных в базе данных СУБД PostgreSQL.

Установим PostgreSQL и запустим его на компьютере, его интерфейс.

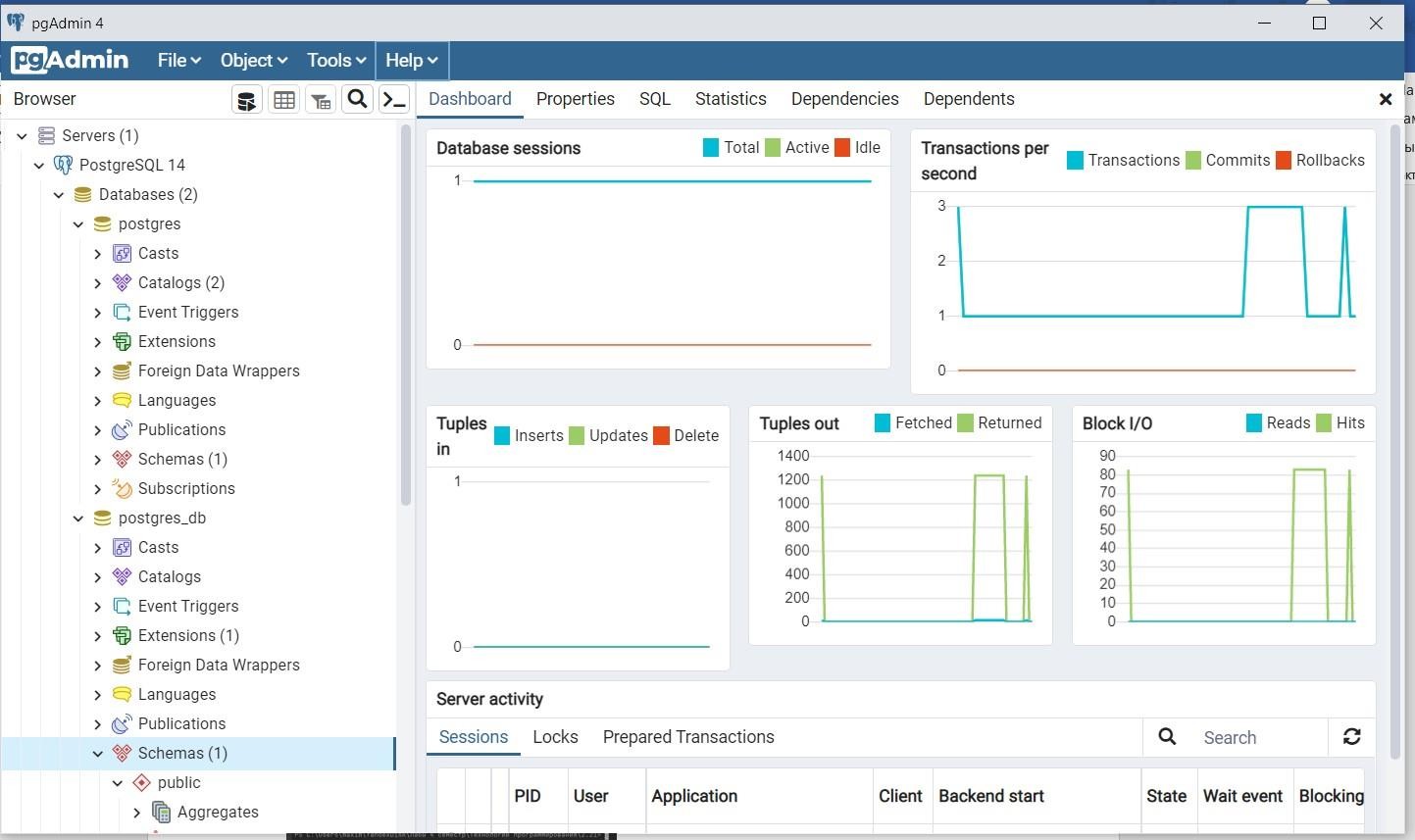


Рис. 3 Интерфейс СУБД PostegreSQL

Создадим таблицы и связи между ними, наполним их данными и отобразим.

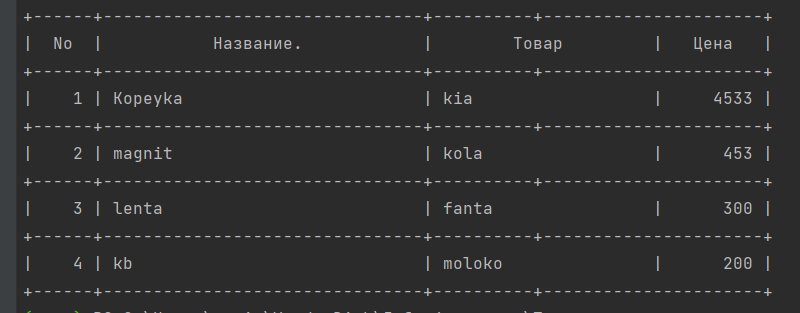


Рис. 4 Таблица с данными

Реализуем поиск и используем.

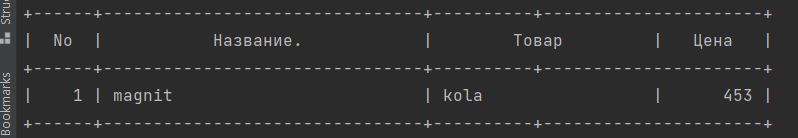


Рис. 5 Результат поиска

Вывод: в результате выполнения работы были приобретены навыки по работе c базовыми возможностями системы управления базами данных SQLite3.

Контрольные вопросы

1. Непосредственно **модуль sqlite3** – это API к СУБД SQLite. Своего рода адаптер, который переводит команды, написанные на Питоне, в команды, которые понимает SQLite. Как и наоборот, доставляет ответы от SQLite в python-программу.
2. Для взаимодействия с базой данных SQLite3 в Python необходимо создать объект cursor. Вы можете создать его с помощью метода cursor() . Курсор SQLite3 – это метод объекта соединения. Для выполнения инструкций SQLite3 сначала устанавливается соединение, а затем создается объект курсора с использованием объекта соединения
3. При создании соединения с SQLite3 автоматически создается файл базы данных, если он еще не существует. Этот файл базы данных создается на диске, мы также можем создать базу данных в оперативной памяти с помощью функции :memory: with the connect. Такая база данных называется базой данных в памяти.
4. С помощью команды закрытия close().
5. Чтобы вставить данные в таблицу, используется оператор INSERT INTO.
6. Чтобы обновить данные в таблице, просто создайте соединение, затем создайте объект курсора с помощью соединения и, наконец, используйте оператор UPDATE.
7. Оператор SELECT используется для выбора данных из определенной таблицы. Если вы хотите выбрать все столбцы данных из таблицы, вы можете использовать звездочку (\*).
8. SQLite3 rowcount используется для возврата количества строк, которые были затронуты или выбраны последним выполненным SQL- запросом.
9. Чтобы перечислить все таблицы в базе данных SQLite3, вы должны запросить данные из таблицы sqlite\_master, а затем использовать fetchall() для получения результатов из инструкции SELECT
10. При создании таблицы мы должны убедиться, что она еще не существует. Аналогично, при удалении/удалении таблицы она должна существовать. Чтобы проверить, не существует ли таблица уже, мы используем IF NOT EXISTS с оператором CREATE TABLE следующим образом.
11. Метод executemany можно использовать для вставки нескольких строк одновременно.
12. В базе данных Python SQLite3 мы можем легко хранить дату или время, импортируя модуль datetime . Следующие форматы являются наиболее часто используемыми форматами для datetime::