Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №6
дисциплины «Анализ данных»

Выполнила:
Кубанова Ксения Олеговна
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и вычислительная техника», очная форма обучения

_____
(подпись)

Руководитель практики:
Воронкин Р. А.

_____
(подпись)

Отчет защищен с оценкой _____ Дата защиты_____

Ставрополь, 2024 г.

Тема: основы работы с SQLite3

Цель: приобрести навыки работы с основами SQLite3.

Порядок выполнения работы

Задание 7.

Выполнить в песочнице следующие команды:

```
sqlite> create table customer(name);
sqlite> select *
   ...> from customer;
sqlite> .schema customer
CREATE TABLE customer(name);
```

Рисунок 1 – задание 7

Задание 8.

Подобрать к коду команду, отсчитывающую время выполнения запроса.

```
sqlite> .timer on
sqlite> select count(*) from city;
1117
Run Time: real 0.000 user 0.000182 sys 0.000052
```

Рисунок 2 – timer on

Задание 9.

Импортировать таблицу и выполнить запрос.

```
sqlite> .import --csv city.csv city
sqlite> select max(length(city)) from city;
25
```

Рисунок 3 – выполнение запроса

Задание 10.

Загрузить таблицу без помощи -csv

```
sqlite> .separator ,
sqlite> .import city.csv city
```

Рисунок 4 – загрузка

Задание 11.

Напишите в песочнице запрос, который посчитает количество городов для каждого часового пояса в Сибирском и Приволжском федеральных округах. Выведите столбцы timezone и city_count , отсортируйте по значению часового пояса:

```
sqlite> select
   ...> distinct timezone as timee,
   ...> count() as count_city
   ...> from city
   ...> group by timee
   ...> order by timee desc;
timezone,1
UTC+9,62
UTC+8,56
UTC+7,172
UTC+6,12
UTC+5,346
UTC+4,132
UTC+3,1320
UTC+2,44
UTC+12,12
UTC+11,34
UTC+10,44
```

Рисунок 5 - отсортировка

Задание 12.

Напишите в песочнице запрос, который найдет три ближайших к Самаре города, не считая саму Самару.

```
sqlite> with samara as (select geo_lat as lat, geo_lon as lon from city where city=
"Самара")
   ...> select
   ...> city,
   ...> sqrt(
   ...> pow (geo_lon - lon, 2) + power(geo_lat - lat, 2)
   ...> ) * 69.09 as euclidean_distance
   ...> from
   ...> city, samara
   ...> where euclidean_distance > 0
   ...> order by euclidean_distance
   ...> limit 3;
Новокуйбышевск,12.8298063265514
```

Рисунок 6 – нахождение ближайших городов

Задание 13.

Напишите в песочнице запрос, который посчитает количество городов в каждом часовом поясе. Отсортируйте по количеству городов по убыванию.

```
sqlite> .mode csv
sqlite> .headers on
sqlite> .separator |
sqlite> .mode box
sqlite> select
   ...> distinct timezone as timee,
   ...> count() as count_city
   ...> from city
   ...> group by timee
   ...> order by timee desc;
```

timee	count_city
timezone	1
UTC+9	62
UTC+8	56
UTC+7	172
UTC+6	12
UTC+5	346
UTC+4	132
UTC+3	1320
UTC+2	44
UTC+12	12
UTC+11	34
UTC+10	44

Рисунок 7 – подсчет городов

Все задания были сохранены в формате sql соответственно: prim7-prim13.

Индивидуальное задание.

Необходимо загрузить свой датасет и выполнить минимум пять запросов к нему, после чего итог каждого запроса соответственно сохранить в формате csv и json.

К заданию был выбран датасет на тему грибов и их характеристик. В числовых форматах описаны 7 столбцов.

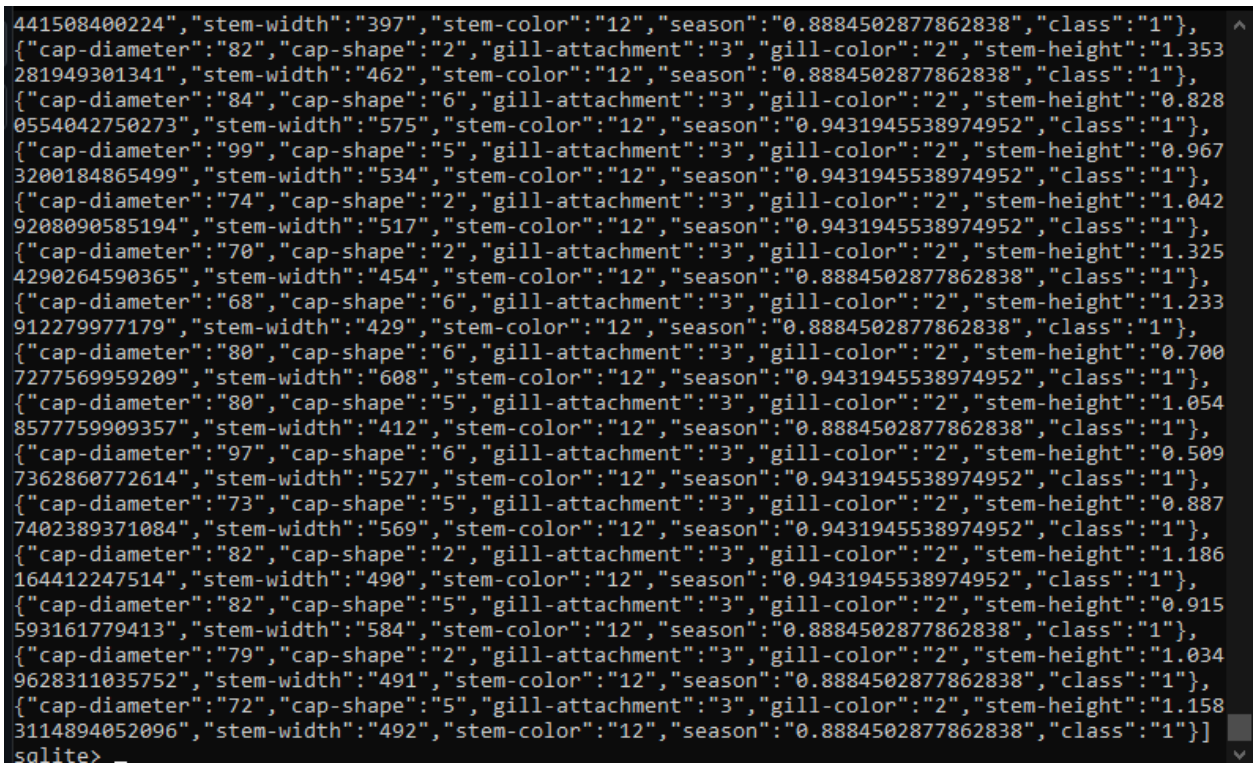

Рисунок 8 - SELECT * FROM mushroom_cleaned

Рисунок 9 – запросы с limit

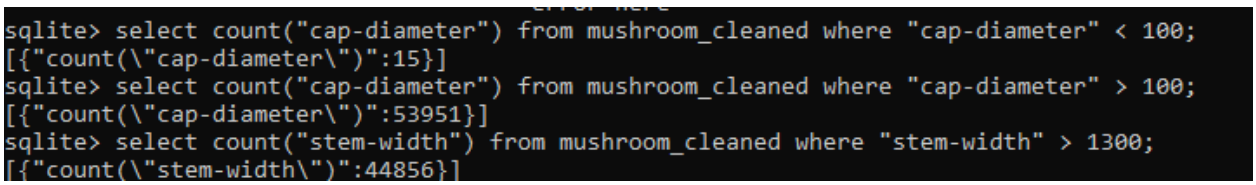

Рисунок 10 – логические запросы

Сами коды запросов были сохранены в ind.sql, а итоги этих запросов соответственно сохранены в csv и json.

Контрольные вопросы

1 Каково назначение реляционных баз данных и СУБД?

Назначение реляционных баз данных и систем управления базами данных (СУБД) состоит в организации и хранении структурированных

данных, обеспечении целостности данных, а также обеспечении возможности эффективного доступа к данным и их обработки.

2 Каково назначение языка SQL?

Назначение языка SQL (Structured Query Language) - это язык программирования, который используется для управления данными в реляционных базах данных. Он предоставляет возможность выполнения различных операций, таких как вставка, выборка, обновление и удаление данных.

3 Из чего состоит язык SQL?

Язык SQL состоит из различных типов операторов, таких как операторы выборки (SELECT), вставки (INSERT), обновления (UPDATE), удаления (DELETE), операторов определения структуры базы данных (CREATE, ALTER, DROP), операторов управления доступом (GRANT, REVOKE) и т. д.

4 В чем отличие СУБД SQLite от клиент-серверных СУБД?

Отличие СУБД SQLite от клиент-серверных СУБД заключается в том, что SQLite является встраиваемой базой данных, которая работает без необходимости отдельного серверного процесса, в то время как клиент-серверные СУБД используют клиент-серверную архитектуру, где клиентские приложения общаются с сервером баз данных для выполнения операций.

5 Как установить SQLite в Windows и Linux?

Установка SQLite в Windows и Linux может быть выполнена путем загрузки соответствующего бинарного дистрибутива с официального сайта SQLite и последующей установкой в соответствии с инструкциями для каждой операционной системы.

6 Как создать базу данных SQLite?

Для создания базы данных SQLite используется команда CREATE DATABASE вместе с указанием имени базы данных.

7 Как выяснить в SQLite какая база данных является текущей?

Чтобы узнать, какая база данных является текущей в SQLite, можно использовать команду PRAGMA database_list; в SQLite shell.

8 Как создать и удалить таблицу в SQLite?

Для создания таблицы в SQLite используется команда CREATE TABLE, а для удаления таблицы - команда DROP TABLE.

9 Что является первичным ключом в таблице?

Первичный ключ в таблице является уникальным идентификатором каждой записи в таблице, который гарантирует уникальность и целостность данных.

10 Как сделать первичный ключ таблицы автоинкрементным?

Для того чтобы сделать первичный ключ таблицы автоинкрементным, используется ключевое слово AUTOINCREMENT при определении столбца с первичным ключом.

11 Каково назначение инструкций NOT и DEFAULT при создании таблиц?

Инструкция NOT используется для указания, что столбец не может содержать NULL значения, а инструкция DEFAULT позволяет задать значение по умолчанию для столбца при вставке новой записи.

12 Каково назначение внешних ключей в таблице? Как создать внешний ключ в таблице?

Внешние ключи в таблице используются для создания связей между таблицами. Для их создания в SQLite используется инструкция FOREIGN KEY.

13 Как выполнить вставку строки в таблицу базы данных SQLite?

Для выполнения вставки строки в таблицу базы данных SQLite используется оператор INSERT INTO.

14 Как выбрать данные из таблицы SQLite?

Для выборки данных из таблицы SQLite используется оператор SELECT.

15 Как ограничить выборку данных с помощью условия WHERE?

Чтобы ограничить выборку данных с помощью условия WHERE, используется ключевое слово WHERE в операторе SELECT.

16 Как упорядочить выбранные данные?

Для упорядочивания выбранных данных в SQLite используется ключевое слово ORDER BY в операторе SELECT.

17 Как выполнить обновление записей в таблице SQLite?

Обновление записей в таблице SQLite выполняется с помощью оператора UPDATE.

18 Как удалить записи из таблицы SQLite?

Удаление записей из таблицы SQLite выполняется с помощью оператора DELETE FROM.

19 Как сгруппировать данные из выборке из таблицы SQLite?

Для группировки данных из выборки из таблицы SQLite используется ключевое слово GROUP BY в операторе SELECT.

20 Как получить значение агрегатной функции (например: минимум, максимум, количество записей и т. д.) в выборке из таблицы SQLite?

Для получения значения агрегатной функции в выборке из таблицы SQLite используется функция агрегирования, такая как MIN, MAX, COUNT и т. д.

21 Как выполнить объединение нескольких таблиц в операторе SELECT?

Объединение нескольких таблиц в операторе SELECT выполняется с помощью ключевого слова JOIN.

22 Каково назначение подзапросов и шаблонов при работе с таблицами SQLite?

Подзапросы и шаблоны используются для более сложных запросов к данным в SQLite, позволяя выполнять операции вложенных запросов и использовать шаблоны данных для обработки запросов.

23 Каково назначение представлений VIEW в SQLite?

Представления VIEW в SQLite представляют собой виртуальные таблицы, которые создаются на основе результатов запроса и могут быть использованы для упрощения выполнения сложных запросов или предоставления доступа к данным определенной группе пользователей.

24 Какие существуют средства для импорта данных в SQLite?

Существуют различные средства для импорта данных в SQLite, такие как SQLite командная строка, SQLite Studio, DB Browser for SQLite и другие.

25 Каково назначение команды .schema?

Команда .schema в SQLite используется для отображения схемы базы данных, включая структуру таблиц и ограничения.

26 Как выполняется группировка и сортировка данных в запросах SQLite?

Группировка и сортировка данных в запросах SQLite выполняются с использованием ключевых слов GROUP BY и ORDER BY соответственно.

27 Каково назначение "табличных выражений" в SQLite?

"Табличные выражения" в SQLite позволяют выполнять запросы, которые возвращают результаты в виде таблицы, которая может быть использована как обычная таблица в других частях запроса.

28 Как осуществляется экспорт данных из SQLite в форматы CSV и JSON?

Экспорт данных из SQLite в форматы CSV и JSON может быть осуществлен с помощью команды .mode в SQLite shell и использования соответствующих команд экспорта.

29 Какие еще форматы для экспорта данных Вам известны?

Кроме CSV и JSON, также можно экспортировать данные из SQLite в другие форматы, такие как XML, SQL, Excel и др., используя специализированные инструменты или скрипты.

Вывод: