

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 №2.20
дисциплины «Анализ данных»

Выполнил:
Магдаев Даламбек Магомедович
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

Руководитель практики:
Воронкин Р.А., доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2024 г.

Тема: Основы работы с SQLite3

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

Порядок выполнения работы:

1. Создал новый репозиторий, клонировал его, в нем создал ветку developer и перешел на нее.
2. Решил задачу 1: выполните в песочнице команды.

```
Last login: Thu May 16 16:03:07 2024 from 127.0.0.1
SQLite version 3.38.0 2022-02-22 18:58:40
Enter ".help" for usage hints.
Connected to a transient in-memory database.
Use ".open FILENAME" to reopen on a persistent database.
sqlite> create table customer(name);
sqlite> select *
...> from customer;
sqlite> .schema customer
CREATE TABLE customer(name);
sqlite> █
```

Рисунок 1. Выполнение задачи

Что вернула команда .schema?

Ответ: .schema вернула CREATE TABLE customer(name)

3. Решил задачу 2: с помощью команды .help найдите в песочнице команду, которая отвечает за вывод времени выполнения запроса. Если ее включить, в результатах запроса добавится строка: Run Time: real XXX user XXX sys XXX.

```
sqlite> .timer on
sqlite> select count(*) from city;

┌ count (*) ─┐
│ 1117       │
└───────────┘

Run Time: real 0.001 user 0.000175 sys 0.000152
```

Рисунок 2. Выполнение задачи

Какая команда должна быть вместо SOMETHING?

Ответ: .timer on

4. Решил задачу 3: загрузите файл city.csv в песочнице, затем выполните такой запрос: select max(length(city)) from city

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

max(length(city))
25

```
Run Time: real 0.003 user 0.002265 sys 0.000000
sqlite>
```

Рисунок 3. Выполнение задачи

Какое число он вернул?

Ответ: запрос вернул число 25

5. Решил задачу 4: загрузите файл city.csv в песочнице с помощью команды .import, но без использования опции --csv. Эта опция появилась в недавней версии SQLite, так что полезно знать способ, для старых версий.

```
sqlite> .help import
.import FILE TABLE      Import data from FILE into TABLE
Options:
  --ascii                Use \037 and \036 as column and row separators
  --csv                  Use , and \n as column and row separators
  --skip N               Skip the first N rows of input
  --schema S             Target table to be S.TABLE
  -v                    "Verbose" - increase auxiliary output
Notes:
  * If TABLE does not exist, it is created. The first row of input
    determines the column names.
  * If neither --csv or --ascii are used, the input mode is derived
    from the ".mode" output mode
  * If FILE begins with "|" then it is a command that generates the
    input text.
sqlite> .mode csv
sqlite> .import city.csv city
sqlite> select max(length(city)) from city;
25
```

Рисунок 4. Выполнение задачи

Какая команда должна быть вместо do_something?

Ответ: необходимо перед загрузкой написать команду `.mode csv`

6. Решил задачу 5: 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. Укажите в ответе значение `city_count` для `timezone = UTC+5`.

```
sqlite> SELECT
...> DISTINCT timezone AS timee,
...> COUNT() AS count_city
...> FROM city
...>
...> GROUP BY timee
...> ORDER BY timee DESC;
```

timee	count_city
UTC+9	31
UTC+8	28
UTC+7	86
UTC+6	6
UTC+5	173
UTC+4	66
UTC+3	660
UTC+2	22
UTC+12	6
UTC+11	17
UTC+10	22

Рисунок 5. Выполнение задачи

Ответ: в часовом поясе UTC+5 58 городов Сибирского и Приволжского федеральных округов.

7. Решил задачу 6: напишите в песочнице запрос, который найдет три ближайших к Самаре города, не считая саму Самару. Укажите в ответе названия этих трех городов через запятую в порядке удаления от Самары.

```

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;

```

city	euclidean_distance
Новокуйбышевск	12.8298063265514
Чапаевск	24.7389598092284
Кинель	36.4840951529617

Рисунок 6. Выполнение задачи

Ответ: 3 ближайших города в порядке удаления от Самары: Новокуйбышевск, Чапаевск, Кинель.

8. Решил задачу 7: напишите в песочнице запрос, который посчитает количество городов в каждом часовом поясе. Отсортируйте по количеству городов по убыванию.

```

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
UTC+9	31
UTC+8	28
UTC+7	86
UTC+6	6
UTC+5	173
UTC+4	66
UTC+3	660
UTC+2	22
UTC+12	6
UTC+11	17
UTC+10	22

Рисунок 7. Выполнение задачи

9. Выполнил индивидуальное задание: загрузите в SQLite выбранный Вами датасет в формате CSV (датасет можно найти на сайте Kaggle). Сформируйте более пяти запросов к таблицам БД. Выгрузите результат выполнения запросов в форматы CSV и JSON.

```
sqlite> .mode box
sqlite> select title, authors, average_rating from books
...> where length(title) < 50 order by average_rating desc limit 10;
```

title	authors	average_rating
title	authors	average_rating
Willem de Kooning: Late Paintings	Julie Sylvester/David Sylvester	5.0
Middlesex Borough (Images of America: New Jersey)	Middlesex Borough Heritage Committee	5.0
Taxation of Mineral Rents	Ross Garnaut	5.0
The New Big Book of America	Todd Davis/Marc Frey	5.0
Fanning the Flame: Bible Cross and Mission	Chris Green/Chris Wright/Paul Douglas Gardner	5.0
Tyrannosaurus Wrecks (Stanley #1)	Laura Driscoll/Alisa Klayman-Grodsky/Eric Weiner	5.0
The Irish Anatomist: A Study of Flann O'Brien	Keith Donohue	5.0
Existential Meditation	Simon Cleveland	4.91
Little Big Book for God's Children	Alice Wong/Lena Tabori	4.88

Рисунок 8. Запрос 1 – Выбрать 10 книг с самым большим рейтингом

```
sqlite> select title, ratings_count from books
...> order by ratings_count desc limit 5;
```

title	ratings_count
title	ratings_count
The Hobbit or There and Back Again	2530894
The Catcher in the Rye	2457092
Angels & Demons (Robert Langdon #1)	2418736
Harry Potter and the Prisoner of Azkaban (Harry Potter #3)	2339585

Риснок 9. Запрос 2 – Выбрать 5 книг на английском языке с наименьшим числом оценок

```
sqlite> select language_code as language, count(title) as book_count from books
...> group by language;
```

language	book_count
9780674842113	1
9780851742717	1
ara	1
en-CA	3
en-GB	159
en-US	1040
eng	6815
enm	3
fre	113
ger	73
glg	1
grc	11

Рисунок 10. Запрос 3 – Посчитать количество книг на каждом языке

```
sqlite> select authors, (sum(average_rating) / count(title)) as rating from books where length(authors) < 50
...> group by authors order by rating desc limit 10;
```

authors	rating
William C. Dowling	5.0
Todd Davis/Marc Frey	5.0
Sheri Rose Shepherd	5.0
Ross Garnaut	5.0
Nicholas Evans/Rhonda Evans	5.0
Middlesex Borough Heritage Committee	5.0
Keith Donohue	5.0
Julie Sylvester/David Sylvester	5.0
James E. Campbell	5.0
Chris Green/Chris Wright/Paul Douglas Gardner	5.0

Рисунок 11. Запрос 4 – Выбрать 10 авторов с наивысшим общим рейтингом за все написанные ими книги

```
sqlite> select title, text_reviews_count from books
...> where length(title) < 50 and text_reviews_count between 1000 and 2000 limit 15;
```

title	text_reviews_count
Anna Karenina	1851
I am Charlotte Simmons	1688
The Portrait of a Lady	1951
Slouching Towards Bethlehem	1825
Play It As It Lays	1706
The New York Trilogy	1873
The Brooklyn Follies	1157
Killing Yourself to Live: 85% of a True Story	1109
We the Living	1086
Digging to America	1841
Quicksilver (The Baroque Cycle #1)	1735
The Design of Everyday Things	1379
Pompeii	1726
The Egypt Game	1130
Gates of Fire	1629

Рисунок 12. Запрос 5 – Выбрать любые 15 книг с числом письменных рецензий от 1000 до 2000

```
sqlite> select title, language_code as language, num_pages from books
...> where length(title) < 50 and language_code in ('fre', 'spa') and num_pages > 500
...> limit 15;
```

title	language	num_pages
Timbuktu / Leviathan / Moon Palace	fre	1075
Angeles y demonios (Robert Langdon #1)	spa	508
Da Vinci Code (Robert Langdon #2)	fre	744
El Coleccionista De Huesos (Lincoln Rhyme #1)	spa	640
Hyperion (Los Cantos de Hyperion #1)	spa	618
Endymion	fre	572
El codigo Da Vinci (Robert Langdon #2)	spa	557
Chaman (Familia Cole #2)	spa	768
V.	fre	640
Los Versos Sata?nicos	spa	679

Рисунок 13. Запрос 6 – Выбрать любые 5 книг на французском или испанском языке с рейтингом выше 4 и числом страниц 500

Ответы на контрольные вопросы:

1. Каково назначение реляционных баз данных и СУБД?

Слово "реляция" происходит от "relation", то есть "отношение". Это означает, что в РБД существуют механизмы установления связей между таблицами. Делается это с помощью так называемых первичных и внешних ключей.

2. Каково назначение языка SQL?

SQL – это язык программирования декларативного типа. В отличие от привычных нам процедурных языков, в которых есть условия, циклы и функции, в декларативных языках подобных алгоритмических конструкций почти нет. Декларативные выражения представляют собой скорее запросы, описание того, что хочет получить человек.

Язык SQL предназначен для создания и изменения реляционных баз данных, а также извлечения из них данных. Другими словами, SQL – это инструмент, с помощью которого человек управляет базой данных. При этом ключевыми операциями являются создание таблиц, добавление записей в таблицы, изменение и удаление записей, выборка записей из таблиц, изменение структуры таблиц.

3. Из чего состоит язык SQL?

Сам язык SQL состоит из операторов, инструкций и вычисляемых функций. Зарезервированные слова, которыми обычно выступают операторы, принято писать заглавными буквами. Однако написание их не прописными, а строчными буквами к ошибке не приводит.

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

SQLite – это система управления базами данных, отличительной особенностью которой является ее встраиваемость в приложения. Это значит, что большинство СУБД являются самостоятельными приложениями, взаимодействие с которыми организовано по принципу клиент-сервер.

Программа-клиент посылает запрос на языке SQL, СУБД, которая в том числе может находиться на удаленном компьютере, возвращает результат запроса.

В свою очередь SQLite является написанной на языке C библиотекой, которую динамически или статически подключают к программе. Для большинства языков программирования есть свои привязки (API) для библиотеки SQLite. Так в Python СУБД SQLite импортируют командой `import sqlite3`. Причем модуль `sqlite3` входит в стандартную библиотеку языка и не требует отдельной установки.

5. Как установить SQLite в Windows и Linux?

Для операционной системы Windows скачивают свой архив (`sqlite-tools-win32-*.zip`) и распаковывают. Далее настраивают путь к каталогу, добавляя адрес каталога к переменной `PATH` (подобное можно сделать и в Linux).

6. Как создать базу данных SQLite?

С помощью `sqlite3` создать или открыть существующую базу данных можно двумя способами.

Во-первых, при вызове утилиты `sqlite3` в качестве аргумента можно указать имя базы данных. Если БД существует, она будет открыта. Если ее нет, она будет создана и открыта.

7. Как выяснить в SQLite какая база данных является текущей?

Выяснить, какая база данных является текущей, можно с помощью команды `.databases` утилиты `sqlite3`. Если вы работаете с одной БД, а потом открываете другую, то текущей становится вторая БД.

8. Как создать и удалить таблицу в SQLite?

Таблицы базы данных создаются с помощью директивы `CREATE TABLE` языка SQL. После `CREATE TABLE` идет имя таблицы, после которого в скобках перечисляются имена столбцов и их тип. Для удаления целой таблицы из базы

данных используется директива DROP TABLE, после которой идет имя удаляемой таблицы.

9. Что является первичным ключом в таблице?

Первичный ключ в таблице базы данных – это уникальный идентификатор для каждой строки таблицы. Он гарантирует уникальность записи и используется для создания связей между таблицами (внешних ключей).

В архитектуре реляционной базы данных, первичный ключ может состоять из одного столбца (простой или одиночный ключ) или может комбинировать несколько столбцов (составной ключ), если уникальность может быть гарантирована только их комбинацией. Значения первичного ключа должны быть уникальными и не могут быть NULL.

10. Как сделать первичный ключ таблицы автоинкрементным?

Если нам не важно, какие конкретно идентификаторы будут записываться в поле `_id`, а важна только уникальность поля, следует назначить полю еще один ограничитель – автоинкремент – `AUTOINCREMENT`.

11. Каково назначение инструкций NOT NULL и DEFAULT при создании таблиц?

Ограничитель NOT NULL используют, чтобы запретить оставление поля пустым. По умолчанию, если поле не является первичным ключом, в него можно не помещать данные. В этом случае полю будет присвоено значение NULL. В случае NOT NULL вы не сможете добавить запись, не указав значения соответствующего поля.

Однако, добавив ограничитель DEFAULT, вы сможете не указывать значение. DEFAULT задает значение по умолчанию. В результате, когда данные в поле не передаются при добавлении записи, поле заполняется тем, что было указано по умолчанию.

12. Каково назначение внешних ключей в таблице? Как создать

внешний ключ в таблице?

С помощью внешнего ключа устанавливается связь между записями разных таблиц. Внешний ключ в одной таблице для другой является первичным. Внешние ключи не обязаны быть уникальными. В одной таблице, может быть, несколько внешних ключей, при этом каждый будет устанавливать связь со своей таблицей, где он является первичным.

13. Как выполнить вставку строки в таблицу базы данных SQLite?

С помощью оператора INSERT языка SQL выполняется вставка данных в таблицу.

```
INSERT INTO <table_name> (<column_name1>, <column_name2>, ...)
VALUES (<value1>, <value2>, ...);
```

14. Как выбрать данные из таблицы SQLite?

С помощью оператора SELECT.

15. Как ограничить выборку данных с помощью условия WHERE?

WHERE позволяет задавать условие, согласно которому отображаются только удовлетворяющие ему строки.

16. Как упорядочить выбранные данные?

При выводе данных их можно не только фильтровать с помощью WHERE, но и сортировать по возрастанию или убыванию с помощью оператора ORDER BY. ASC – сортировка от меньшего значения к большему. DESC – сортировка от большего значения к меньшему.

17. Как выполнить обновление записей в таблице SQLite?

UPDATE ... SET – обновление полей записи

UPDATE имя_таблицы

SET имя_столбца = новое_значение WHERE условие;

18. Как удалить записи из таблицы SQLite?

DELETE FROM – удаление записей таблицы DELETE FROM имя_таблицы WHERE условие;

19. Как сгруппировать данные из выборке из таблицы SQLite?

В SQL кроме функций агрегирования есть оператор GROUP BY, который выполняет группировку записей по вариациям заданного поля.

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

Для этих целей в языке SQL предусмотрены различные функции агрегирования данных. Наиболее используемые – count(), sum(), avr(), min(), max().

21. Как выполнить объединение нескольких таблиц в операторе SELECT?

После FROM указываются обе сводимые таблицы через JOIN. В данном случае неважно, какую указывать до JOIN, какую после. После ключевого слова ON записывается условие сведения. Условие сообщает, как соединять строки разных таблиц.

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

Шаблоны реализуют поиск по таблице, если неизвестно полное название данных в строке. Подзапросы помогают уменьшить работу путём создания дополнительного запроса внутри основного.

23. Каково назначение представлений VIEW в SQLite?

Бывает удобно сохранить результат выборки для дальнейшего использования. Для этих целей в языке SQL используется оператор CREATE VIEW, который создает представление – виртуальную таблицу. В эту

виртуальную таблицу как бы сохраняется результат запроса.

24. Какие существуют средства для импорта данных в SQLite?

`.import --csv city.csv city`

25. Каково назначение команды `.schema` ?

Показывает какие столбцы есть в таблице, тип их данных и прочие свойства.

26. Как выполняется группировка и сортировка данных в запросах SQLite?

Группировка – `GROUP BY`, сортировка – `ORDER BY`.

27. Каково назначение "табличных выражений" в SQLite?

Табличные выражения в SQLite используются для структурирования запросов и улучшения их читаемости и гибкости. Табличные выражения можно считать временными наборами записей, которые доступны в пределах одного запроса.

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

`.mode csv` или `.mode json`

`.once <имя файла>` или `.output` любой запрос

Вывод: в результате выполнения работы были исследованы базовые возможности системы управления базами данных SQLite3.