

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт математики и информационных технологий имени
профессора Н.И.Червякова

Кафедра инфокоммуникаций.

Дисциплина: Кроссплатформенное программирования

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

Основы SQLite

Выполнила: студентка 3 курса
09.03.01 «Информатика и вычислительная
техника» группы ИВТ-б-о-19-1
Бондаренко В.В.

Проверил:
Воронкин Роман Александрович

Работа защищена с оценкой:

Ставрополь, 2021

Лабораторная работа №1

Основы SQLite

Цель работы исследовать базовые возможности системы управления базами данных SQLite.

Ход работы:

1. С помощью команды select вывела информацию о том, сколько городов в каждом из федеральных округов:

```
sqlite> select federal_district, city, population from city limit 10;
Южный|Адыгейск|12689
Южный|Майкоп|144055
Сибирский|Горно-Алтайск|62861
Сибирский|Алейск|28528
Сибирский|Барнаул|635585
Сибирский|Белокуриха|15072
Сибирский|Бийск|203826
Сибирский|Горняк|13040
Сибирский|Заринск|47035
Сибирский|Змеиногорск|10569
```

2. С помощью команды select вывела следующую информацию:

- 1) У каких городов в названии есть слово «Красный»?

```
sqlite> select address from city where city like '%Красный%';
Ростовская обл, г Красный Сулин
Саратовская обл, г Красный Кут
Тверская обл, г Красный Холм
```

- 2) Какие города появились за последние 30 лет?

```
sqlite> select region, city, foundation_year
...> from city
...> where foundation_year between 1990 and 2020;
Ингушетия|Магас|1995
Татарстан|Иннополис|2012
sqlite> select count(*)
...> from city
```

- 3) Сколько городов в Приволжском и Уральском округах?

```
sqlite> select count(*)
...> from city
...> where federal_district in ('Приволжский', 'Уральский');
315
```

3. Сформировала подзапрос: “Сколько городов было основано в каждом веке?»:

```

sqlite> with history as (
...> select city, (foundation_year/100)+1 as century
...> from city
...> )
...> select
...> century || '-й век' as dates,
...> count(*) as city_count
...> from history
...> group by century
...> order by century desc;
21-й век|1
20-й век|263
19-й век|189
18-й век|191
17-й век|137
16-й век|79
15-й век|39
14-й век|38
13-й век|27
12-й век|44
11-й век|8
10-й век|6
9-й век|4
5-й век|2
3-й век|1
1-й век|88

```

4. Выгрузила данные в CSV файл и проверила его содержимое;

```

sqlite> .mode csv
sqlite> .once samara.csv
sqlite> select kladr_id, city from city where region = 'Самарская';
sqlite> .exit

```

	A	B	C	D	E
1	6300000200000,"Жигулевск"				
2	6300001000000,"Кинель"				
3	6301700100000,"Нефтегорск"				
4	6300000300000,"Новокуйбышевск"				
5	6300000400000,"Октябрьск"				
6	6300000500000,"Отрадный"				
7	6300000900000,"Похвистнево"				
8	6300000100000,"Самара"				
9	6300000800000,"Сызрань"				
10	6300000700000,"Тольятти"				
11	6300000600000,"Чапаевск"				
12					
13					

5. Познакомилась с настройками экспорта в CSV файл (для наглядности выводил на экран):

```

sqlite> select kladr_id, city from city where region = 'Самарская' limit 3;
6300000200000|Жигулевск
6300001000000|Кинель
6301700100000|Нефтегорск
sqlite> .mode csv
sqlite> .headers on
sqlite> select kladr_id, city from city where region = 'Самарская' limit 3;
kladr_id,city
6300000200000,"Жигулевск"
6300001000000,"Кинель"
6301700100000,"Нефтегорск"
sqlite> .mode csv
sqlite> .headers on
sqlite> .separator ;
sqlite> select kladr_id, city from city where region = 'Самарская' limit 3;
kladr_id;city
6300000200000;Жигулевск
6300001000000;Кинель
6301700100000;Нефтегорск
sqlite> .mode list
sqlite> .headers on
sqlite> .separator ;
sqlite> .separator ,
sqlite> select kladr_id, city from city where region = 'Самарская' limit 3;
kladr_id,city
6300000200000,Жигулевск
6300001000000,Кинель
6301700100000,Нефтегорск

```

6. Вставила данные из таблицы city в другую БД:

```

sqlite> .mode insert cities
sqlite> select kladr_id, city from city where region = 'Самарская' limit 3;
INSERT INTO cities VALUES('6300000200000','Жигулевск');
INSERT INTO cities VALUES('6300001000000','Кинель');
INSERT INTO cities VALUES('6301700100000','Нефтегорск');

```

7. Экспортировала данные таблицы в формат html:

```

sqlite> .mode html
sqlite> select kladr_id, city from city where region = 'Самарская' limit 3;
<TR><TD>6300000200000</TD>
<TD>Жигулевск</TD>
</TR>
<TR><TD>6300001000000</TD>
<TD>Кинель</TD>
</TR>
<TR><TD>6301700100000</TD>
<TD>Нефтегорск</TD>
</TR>

```

8. Импортировала данные из файла samara.csv в новую таблицу samara:

```

sqlite> .mode csv
sqlite> .headers on
sqlite> .separator |
sqlite> .import --csv samara.csv samara
sqlite> select name from samara limit 5;
name
"Жигулевск"
"Кинель"
"Нефтегорск"
"Новокуйбышевск"
"Октябрьск"
sqlite> select * from samara limit 5;
kladr_id|name
6300000200000|"Жигулевск"
6300001000000|"Кинель"
6301700100000|"Нефтегорск"
6300000300000|"Новокуйбышевск"
6300000400000|"Октябрьск"

```

Общее задание:

1. Команда .schema показала, шаблон команды CREATE, с помощью которой создавалась таблица.

```

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

```

2. За вывод времени выполнения запроса отвечает команда .timer on/off

```
sqlite> .timer on
sqlite> select count(*) from city;
1117
Run Time: real 0.000 user 0.000000 sys 0.000000
sqlite>
```

3. Эта команда вернула число 25, что является максимальным количеством символов в столбце city.

```
sqlite> select max(length(city)) from city;
25
sqlite>
```

4. Если не используется –csv, то режим ввода является производным от режима вывода .mode, поэтому перед импортом нужно написать команду .mode csv

```
sqlite> .mode csv
sqlite> .import city.csv city_1
sqlite>
```

5. Написала запрос, который посчитает количество городов для каждого часового пояса в Сибирском и Приволжском федеральных округах. Вывела столбцы timezone и city_count, отсортировав по значению часового пояса:

```
sqlite> with kol_zones as (
...> select timezone as zones from city where federal_district in ('Приволжский', 'Уральский'))
...> select zones, count(*) as city_count from kol_zones group by zones order by zones desc;
zones|city_count
UTC+5|173
UTC+4|41
UTC+3|101
sqlite>
```

Ответ: 173

6. Написала запрос, который найдет три ближайших к Самаре города, не считая саму Самару.

```
sqlite> with target as (
...> select geo_lat, geo_lon from city where city = 'Самара'
...> )
...> select city from (
...> select city, geo_lat, geo_lon, (select geo_lat from target) as target_geo_lat,
...> (select geo_lon from target) as target_geo_lon from city where city != 'Самара'
...> )
...> order by sqrt(power(geo_lat - target_geo_lat, 2) + power(geo_lon - target_geo_lon, 2))
...> limit 3;
Новокуйбышевск
Чапаевск
Кинель
sqlite>
```

Ответ: Новокуйбышевск, Чапаевск, Кинель

Контрольные вопросы:

1. Какие существуют средства для импорта данных в SQLite?

```
.import --csv city.csv city
```

```
.mode csv
```

```
.import city.csv city
```

2. В чем недостатки локальных и централизованных СКВ?

Локальные СКВ

Легко запутаться в файлах и в каком конкретно ты находишься, можно изменить не тот файл или сохранить в не нужной директории

Централизованные

Централизованный сервер является уязвимым местом всей системы. Если сервер выключается на час, то в течение часа разработчики не могут взаимодействовать, и никто не может сохранить новые версии. Если же

повреждается диск с центральной базой данных и нет резервной копии, вы теряете абсолютно всё - всю историю проекта, разве что за исключением нескольких рабочих версий, сохранившихся на рабочих машинах пользователей. Локальные системы управления версиями подвержены той же проблеме: если вся история проекта хранится в одном месте, вы рискуете потерять всё.

3. Каково назначение команды .schema ?

показывает список и структуру всех таблиц в базе

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

Группировка производится, когда в оператора group используется функция by

5. Каково назначение "табличных выражений" в SQLite?

Наглядная демонстрация содержимого в таблице

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

```
sqlite> .mode csv
```

```
sqlite> .once samara.csv
```

```
sqlite> select kladr_id, city from city where region = 'Самарская';
```

```
sqlite> .exit
```

```
.mode json
```

```
select kladr_id, city  
from city where region = 'Самарская' limit 3;
```

7. Какие еще форматы для экспорта данных Вам известны?

```
.mode insert cities
```

```
.mode markdown
```

```
.mode html
```