

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Отчет по лабораторной работе №2.20**

**Основы работы с SQLite3.**

**по дисциплине «Технологии программирования и алгоритмизации»**

Выполнил студент группы ИВТ-б-о-21-1

Лысенко И.А. « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 20\_\_ г.

Проверил Воронкин Р.А. \_\_\_\_\_

(подпись)

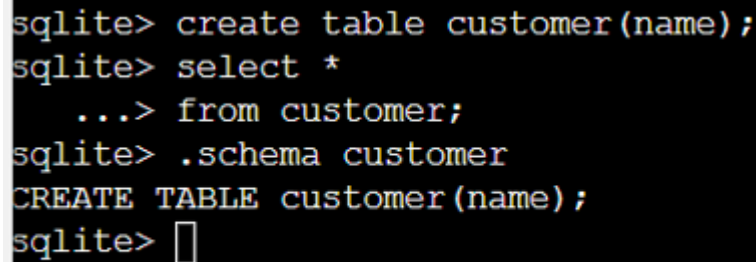
Ставрополь 2023

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

**Ход работы:**

1. Создал репозиторий на GitHub:  
[https://github.com/IsSveshuD/lab\\_2\\_20.git](https://github.com/IsSveshuD/lab_2_20.git)

2. Решил задачу:

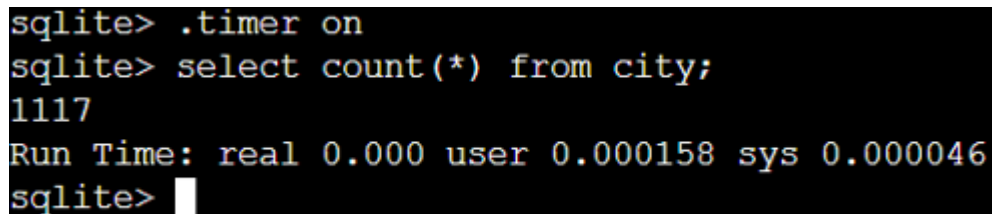


```
sqlite> create table customer(name);
sqlite> select *
...> from customer;
sqlite> .schema customer
CREATE TABLE customer(name);
sqlite> 
```

Рисунок 1 – Задача 1

.schema вернула CREATE TABLE customer(name);

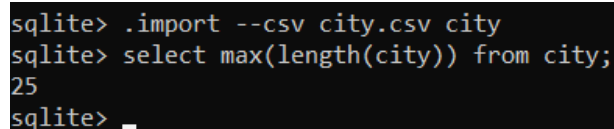
3. За вывод времени выполнения запроса отвечает команда .timer .



```
sqlite> .timer on
sqlite> select count(*) from city;
1117
Run Time: real 0.000 user 0.000158 sys 0.000046
sqlite> 
```

Рисунок 2 – Задача 2. Вывод времени запроса

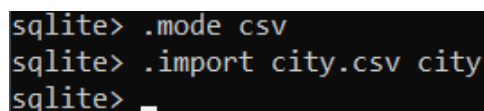
4. Команда: select max(length(city)) from city; вернула значение 25.



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

Рисунок 3 – Задача 3.

5. Чтобы импортировать БД без использования опции --csv, необходимо сначала ввести команды .mode csv, и только потом импортировать БД.



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

Рисунок 4 – Задача 4

6. Решите задачу: 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. Ответ: 174

```
sqlite> select
...> timezone, count(city) as city_count
...> from city where federal_district = 'Сибирский' or federal_district = 'Приволжский'
...> group by timezone
...> order by timezone ASC;
UTC+3,303
UTC+4,123
UTC+5,174
UTC+6,18
UTC+7,258
UTC+8,66
sqlite>
```

Рисунок 5 – Задача 5. Количество городов по часовым поясам.

7. Решите задачу: напишите в песочнице запрос, который найдет три ближайших к Самаре города, не считая саму Самару.

```
sqlite> with geo_x as (select geo_lat as geo_x from city where city = 'Самара'),
...> geo_y as (select geo_lon as geo_y from city where city = 'Самара'),
...> geo_xx as (select geo_lat as geo_xx, city from city),
...> geo_yy as (select geo_lon as geo_yy from city)
...> select sqrt((power((geo_x - geo_xx),2) + power((geo_y - geo_yy),2))) as distance,
...> city from (geo_x, geo_y, geo_xx, geo_yy)
...> where city != 'Самара'
...> order by distance asc limit 3;
```

distance	city
0.00105299999999886	Заречный
0.0094843000000004	Каменка
0.01199310000000051	Елизovo

Рисунок 6 – Задача 6. Три ближайших к Самаре города

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

```
sqlite> select timezone, count(*) city_count
...> from city
...> group by 1
...> order by 2 desc;
```

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

Рисунок 7 – Количество городов по часовым поясам с заголовками

9. Тоже самое, но с заголовками и в формате csv.

```
sqlite> .mode csv
sqlite> select timezone, count(*) city_count
...> from city
...> group by 1
...> order by 2 desc;
timezone,city_count
UTC+3,660
UTC+5,173
UTC+7,86
UTC+4,66
UTC+9,31
UTC+8,28
UTC+2,22
UTC+10,22
UTC+11,17
UTC+6,6
UTC+12,6
```

Рисунок 8 – Количество городов по часовым поясам в формате csv

10. Тоже самое, но с разделителем |.

```

sqlite> .separator |
sqlite> select timezone, count(*) city_count
...> from city
...> group by 1
...> order by 2 desc;
timezone|city_count
UTC+3|660
UTC+5|173
UTC+7|86
UTC+4|66
UTC+9|31
UTC+8|28
UTC+2|22
UTC+10|22
UTC+11|17
UTC+6|6
UTC+12|6

```

Рисунок 9 – Количество городов по часовым поясам с разделителем |

11. Для выполнения индивидуального задания выбрал базу данных по составу блюд ресторанов быстрого питания.

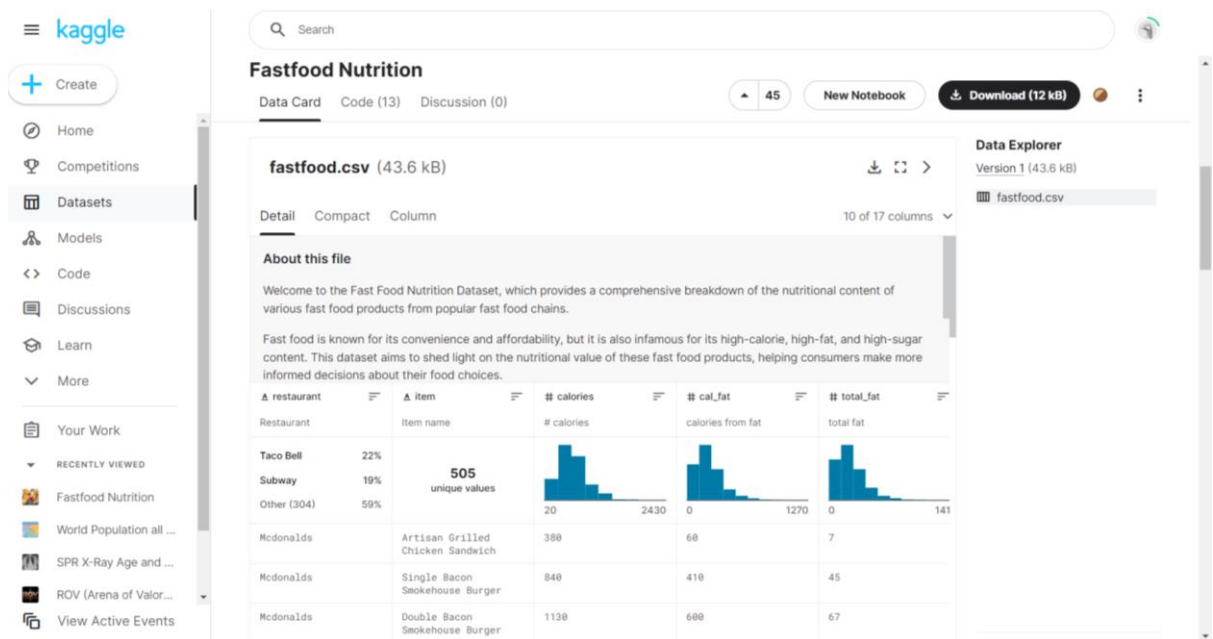


Рисунок 10 – Выбранная база данных

12. Определил какие блюда в McDonalds содержат 380 калорий.

```

sqlite> select item, calories from fastfood where restaurant = 'Mcdonalds' and calories =380;

```

item	calories
Artisan Grilled Chicken Sandwich	380
Filet-O-FishR	380
McDouble	380

Рисунок 11 – Запрос 1

13. Определил сколько всего блюд есть в каждом ресторане.

```
sqlite> select restaurant, count(*) as item_count
...> from fastfood group by 1 order by 2 desc;
```

restaurant	item_count
Taco Bell	115
Subway	96
Burger King	70
Mcdonalds	57
Arbys	55
Sonic	53
Dairy Queen	42
Chick Fil-A	27

Рисунок 12 – Запрос 2

14. Определил какие блюда и в каких ресторанах не содержат сахара.

```
sqlite> select restaurant, item, sugar from fastfood where sugar = 0 limit 10;
```

restaurant	item	sugar
Mcdonalds	3 piece Buttermilk Crispy Chicken Tenders	0
Mcdonalds	4 Piece Chicken McNuggets	0
Mcdonalds	6 Piece Chicken McNuggets	0
Mcdonalds	10 Piece Chicken McNuggets	0
Mcdonalds	20 Piece Chicken McNuggets	0
Chick Fil-A	4 piece Chicken Nuggets	0
Chick Fil-A	6 piece Chicken Nuggets	0
Chick Fil-A	4 Piece Grilled Chicken Nuggets	0
Chick Fil-A	6 Piece Grilled Chicken Nuggets	0
Chick Fil-A	8 piece Grilled Chicken Nuggets	0

Рисунок 13 – Запрос 3

15. Определил блюда, содержащие менее 50 холестерина, отсортировал по возрастанию, вывел первые 15 блюд.

```
sqlite> select restaurant, item, cholesterol from fastfood where restaurant = 'Taco Bell' and cholesterol < 50 order by 3 limit 15;
```

restaurant	item	cholesterol
Taco Bell	Cheesy Bean and Rice Burrito	0
Taco Bell	Fresco Bean Burrito	0
Taco Bell	Triple Layer Nachos	0
Taco Bell	Black Bean Burrito	10
Taco Bell	Cantina Power Burrito - Veggie	10
Taco Bell	Spicy Potato Soft Taco	10
Taco Bell	Cantina Power Bowl - Veggie	10
Taco Bell	Spicy Tostada	10
Taco Bell	Fresco Burrito SupremeR - Steak	15
Taco Bell	Fresco Grilled Steak Soft Taco	15
Taco Bell	Double Tostada	15
Taco Bell	7-Layer Burrito	20
Taco Bell	Beefy FritosR Burrito	20
Taco Bell	Fresco Crunchy Taco	20
Taco Bell	Fresco Soft Taco	20

Рисунок 14 – Запрос 4

## 16. Определил блюда, которые содержат 10 карбона

```
sqlite> select restaurant, item, total_carb from fastfood where item like '%Chicken%' and total_carb == 10;
```

restaurant	item	total_carb
Burger King	4 Piece Chicken Nuggets	10
Dairy Queen	Grilled Chicken Garden Greens Salad	10
Subway	Double Chicken Salad	10
Subway	Oven Roasted Chicken Salad	10

Рисунок 15 – Запрос 5

### Ответы на вопросы:

#### 1. Каково назначение реляционных баз данных и СУБД?

Главная функция СУБД – это управление данными (которые могут быть как во внешней, так и в оперативной памяти). СУБД обязательно поддерживает языки баз данных, а также отвечает за копирование и восстановление информации после каких-либо сбоев.

#### 2. Каково назначение языка SQL?

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.

#### 3. Из чего состоит язык SQL?

Язык SQL состоит из операторов, инструкций и вычисляемых функций.

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

С помощью SQLite создаются базы данных, представляющие собой один кроссплатформенный текстовый файл. Файл базы данных, в отличие от SQLite, не встраивается в приложение, не становится его частью, он существует отдельно. Так можно создать базу данных, пользуясь консольным `sqlite3`, после чего использовать ее в программе с помощью библиотеки SQLite языка программирования. При этом файл базы данных также хранится на локальной машине.

## **5. Как установить SQLite в Windows и Linux?**

В Ubuntu установить `sqlite3` можно командой `sudo apt install sqlite3`. Для операционной системы Windows скачивают свой архив (`sqlite-tools-win32-*.zip`) и распаковывают.

## **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`.

## **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?**

`PRIMARY KEY` – ограничитель, который заставляет СУБД проверять уникальность значения данного поля у каждой добавляемой записи.

## **10. Как сделать первичный ключ таблицы автоинкрементным?**



Добавить AUTOINCREMENT в столбце при создании таблицы.

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

Ограничитель NOT NULL используют, чтобы запретить оставление поля пустым.

DEFAULT задает значение 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. Чтобы включить поддержку внешних ключей в sqlite3, надо выполнить команду PRAGMA foreign\_keys = ON. После этого добавить в таблицу запись, в которой внешний ключ не совпадает ни с одним первичным из другой таблицы, не получится.

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

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

**14. Как выбрать данные из таблицы SQLite?**

С помощью оператора SELECT осуществляется выборочный просмотр данных из таблицы.

**15. Как ограничить выборку данных с помощью условия WHERE?**

Условие WHERE используется не только с оператором SELECT, также с UPDATE и DELETE. С помощью WHERE определяются строки, которые будут выбраны, обновлены или удалены. По сути, это фильтр.

**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.

**17. Как выполнить обновление записей в таблице SQLite?**

UPDATE ... SET – обновление полей записи

## **18. Как удалить записи из таблицы SQLite?**

DELETE FROM – удаление записей таблицы

## **19. Как сгруппировать данные из выборки 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?**

```
select federal_district as district, count(*) as city_count  
from city  
group by 1  
order by 2 desc;
```

## **27. Каково назначение "табличных выражений" в SQLite?**

Выражение `with history as (...)` создает именованный запрос. Название — `history`, а содержание — селект в скобках (век основания для каждого города).

К `history` можно обращаться по имени в остальном запросе, что мы и делаем.

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

```
.mode csv
```

## **29. Какие еще форматы для экспорта данных Вам известны?**

```
.mode list
```

```
.mode json.
```

**Вывод:** были исследованы возможности управления базами данных SQLite3.