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»
Вариант №22

Выполнил:
Михеева Елена Александровна
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., канд. техн. наук,
доцент, доцент кафедры
инфокоммуникаций

_____
(подпись)

Отчет защищен с оценкой _____ Дата защиты_____

Ставрополь, 2024 г.

Тема: Основы работы с SQLite3

Цель: исследовать базовые возможности системы управления базами данных SQLite3.

Порядок выполнения работы:

1. Отработали программу примера лабораторной работы.

```
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/example.py add
 -h
usage: workers add [-h] [--db DB] -n NAME [-p POST] -y YEAR

options:
  -h, --help            show this help message and exit
  --db DB               The database file name
  -n NAME, --name NAME  The worker's name
  -p POST, --post POST  The worker's post
  -y YEAR, --year YEAR  The year of hiring
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/example.py add --d
b example.db -n "lena" -p "designer" -y 2012
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/example.py dis
play
Список работников пуст.
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/example.py display
 -h
usage: workers display [-h] [--db DB]

options:
  -h, --help  show this help message and exit
  --db DB     The database file name
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/example.py display
 --db example.db
+-------+-------------------------------+-----------------------+------------+
|  No   |             Ф.И.О.            |       Должность       |    Год     |
+-------+-------------------------------+-----------------------+------------+
|     1 | lena                          | designer              |       2012 |
+-------+-------------------------------+-----------------------+------------+
```

Рисунок 1. Результаты работы программы примера

2. Приступили к выполнению индивидуального задания №1.

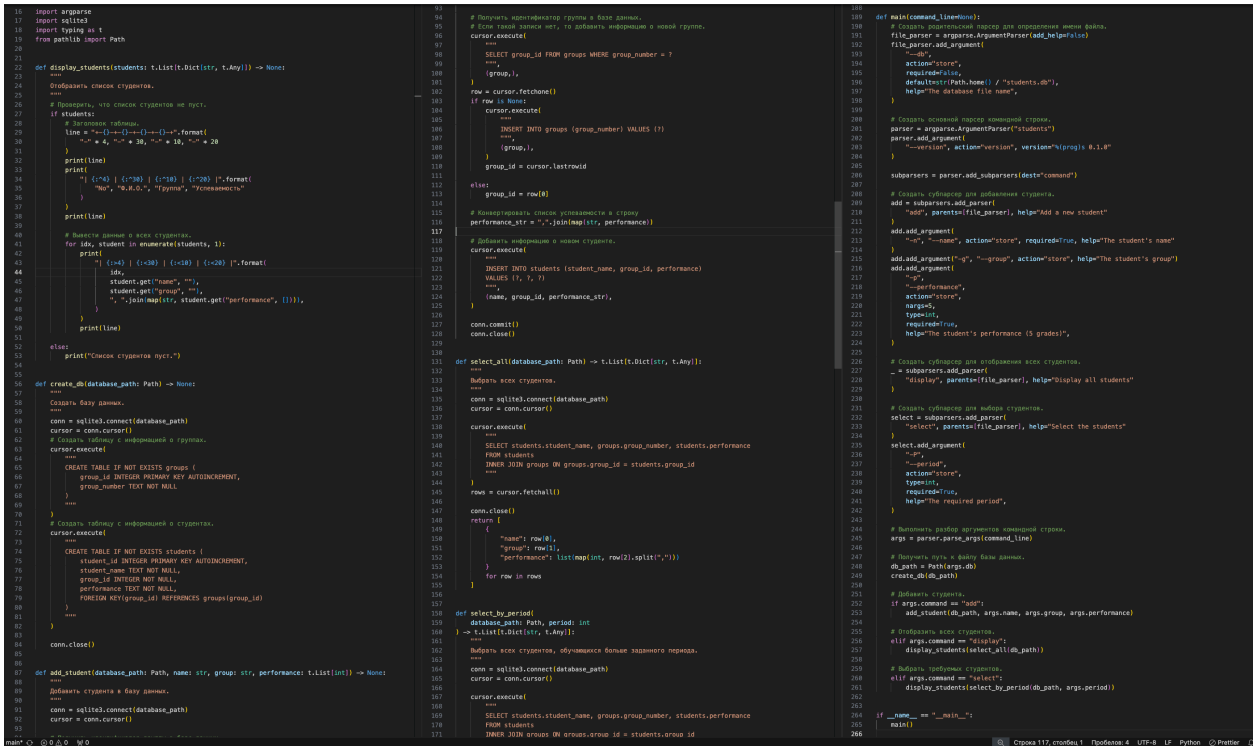

Рисунок 2. Код индивидуального задания №1

```
usage: students [-h] [--version] {add,display,select} ...

positional arguments:
  {add,display,select}
    add                 Add a new student
    display             Display all students
    select              Select the students

options:
  -h, --help            show this help message and exit
  --version             show program's version number and exit
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/task1.py add -h
usage: students add [-h] [--db DB] -n NAME [-g GROUP] -p PERFORMANCE PERFORMANCE PERFORMANCE PERFORMANCE PERFORMANCE

options:
  -h, --help            show this help message and exit
  --db DB               The database file name
  -n NAME, --name NAME  The student's name
  -g GROUP, --group GROUP
                        The student's group
  -p PERFORMANCE PERFORMANCE PERFORMANCE PERFORMANCE PERFORMANCE, --performance PERFORMANCE PERFORMANCE PERFORMANCE PERFORMANCE PERFORMANCE
                        The student's performance (5 grades)
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/task1.py add --db gradebook.db -n "jfy" -g "78ds" -p 1 2 3 4 5
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/task1.py display -db gradebook.db
usage: students [-h] [--version] {add,display,select} ...
students: error: unrecognized arguments: -db gradebook.db
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/task1.py display --db gradebook.db
+-------+-------------------------------+-------------+-----------------------+
|  No   |             Ф.И.О.            |   Группа    |     Успеваемость      |
+-------+-------------------------------+-------------+-----------------------+
|     1 | jfy                           | 78ds        | 1, 2, 3, 4, 5         |
+-------+-------------------------------+-------------+-----------------------+
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/task1.py display
Список студентов пуст.
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/task1.py add -n "jfy" -g "78ds" -p 1 2 3 4 5
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 % python3 programms/task1.py display
+-------+-------------------------------+-------------+-----------------------+
|  No   |             Ф.И.О.            |   Группа    |     Успеваемость      |
+-------+-------------------------------+-------------+-----------------------+
|     1 | jfy                           | 78ds        | 1, 2, 3, 4, 5         |
+-------+-------------------------------+-------------+-----------------------+
(venv) (base) elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.21 %
```

Рисунок 3. Результат работы программы задания №1

Выполнено задание №2: для своего варианта лабораторной работы 2.17 необходимо реализовать хранение данных в базе данных pgsql. Информация в базе данных должна храниться не менее чем в двух таблицах.

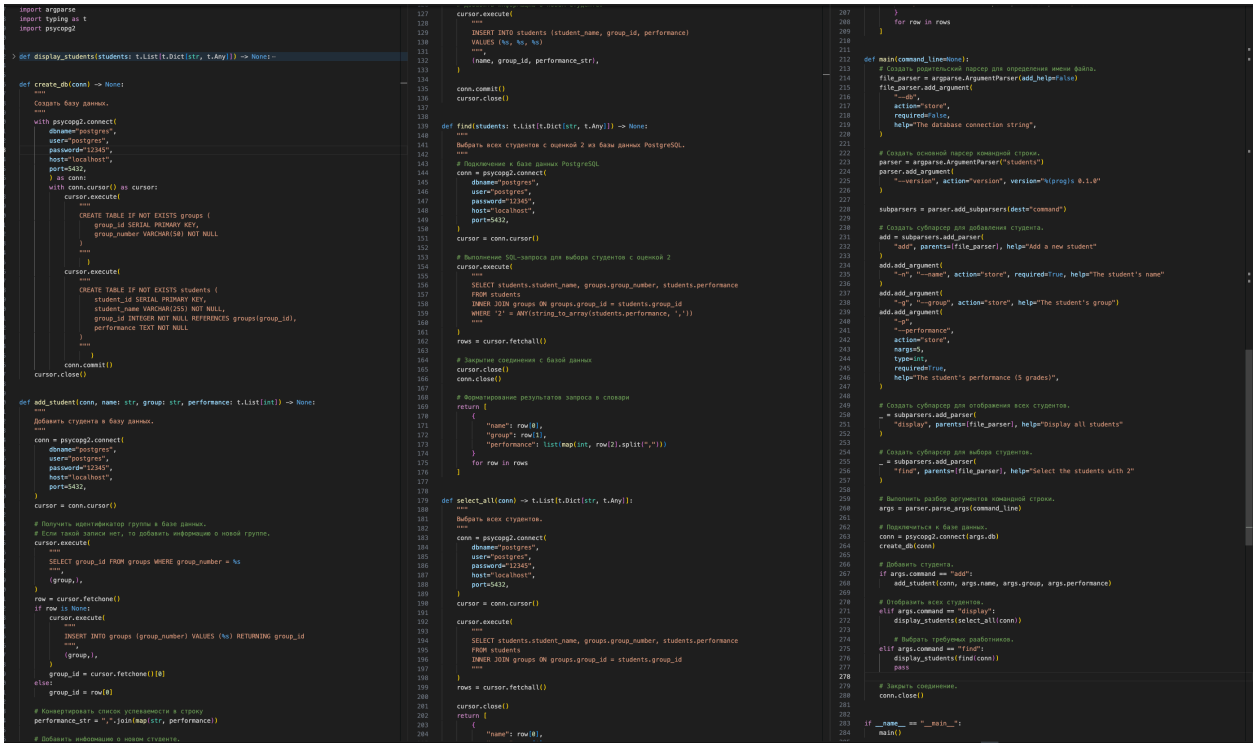

Рисунок 4. Программа задания №2

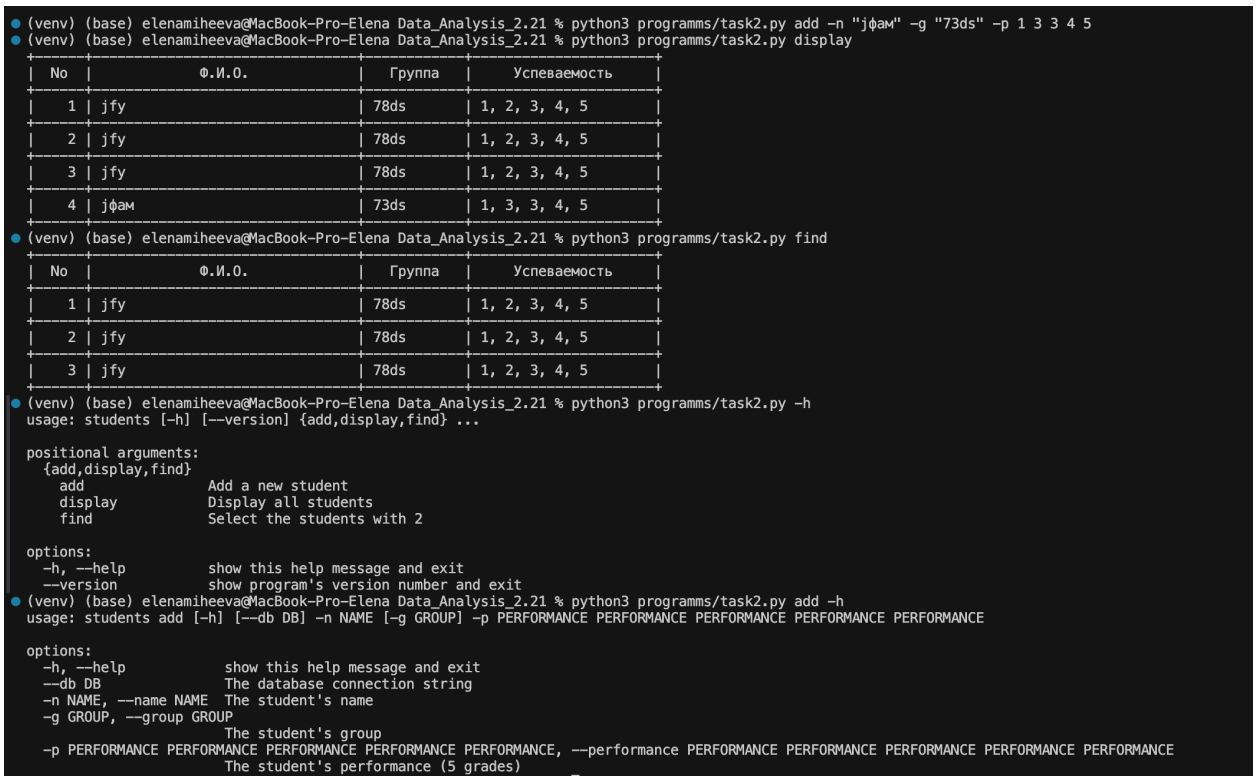

Рисунок 5. Результат работы программы задания №2

Ответы на контрольные вопросы:

1. Каково назначение реляционных баз данных и СУБД?

Ответ: реляционные базы данных (РБД) предназначены для хранения, управления и организации данных в структурированном виде, используя

таблицы, состоящие из строк и столбцов. Система управления базами данных (СУБД) является программным обеспечением, которое обеспечивает взаимодействие пользователей с базой данных, управление данными, выполнение запросов, обеспечение безопасности и целостности данных, а также поддержку транзакций и обеспечения согласованности данных.

2. Каково назначение языка SQL?

Ответ: SQL (Structured Query Language) предназначен для управления и манипуляции данными в реляционных базах данных. С помощью SQL можно создавать и изменять структуры баз данных (такие как таблицы и индексы), выполнять запросы для извлечения данных, вставлять, обновлять и удалять данные, а также управлять доступом к базе данных и выполнять административные задачи.

3. Из чего состоит язык SQL?

Ответ: 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?

Ответ: SQLite отличается от клиент-серверных СУБД следующим: установка и настройка (SQLite не требует установки и настройки серверного ПО, тогда как клиент-серверные СУБД требуют настройки серверного окружения); производительность и масштабируемость (SQLite подходит для небольших приложений и низкой нагрузки, а клиент-серверные СУБД лучше справляются с высокими нагрузками и масштабируемостью); функциональные возможности (Клиент-серверные СУБД обычно предоставляют более широкие возможности, такие как репликация, шардирование и сложные аналитические функции).

5. Как установить SQLite в Windows и Linux?

Ответ: для Windows скачивают свой архив и распаковывают. Далее настраивают путь к каталогу, добавляя адрес каталога к переменной PATH. В Linux аналогичная установка.

6. Как создать базу данных SQLite?

Ответ: с помощью sqlite3 создать или открыть существующую базу данных можно двумя способами. 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). $ sqlite3 “Название базы данных”. Также, работая в самой программе, можно выполнить команду open your.db.

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

Ответ: в SQLite текущей базой данных является та, с которой установлено соединение. SQLite не поддерживает команду, которая явно показывает текущую базу данных, так как обычно в рамках одного соединения используется только одна база данных. Однако можно использовать команду

.databases в утилите командной строки SQLite, чтобы увидеть список подключенных баз данных и их имена.

8. Как создать и удалить таблицу в SQLite?

Ответ: “CREATE TABLE” – создание таблицы. “DROP TABLE” – удаление таблицы.

9. Что является первичным ключом в таблице?

Ответ: первичный ключ в таблице – это одно или несколько полей, значения которых однозначно идентифицируют каждую запись в таблице. Первичный ключ должен быть уникальным и не содержать NULL значений. В большинстве случаев в качестве первичного ключа используется одно поле.

10. Как сделать первичный ключ таблицы автоинкрементным?

Ответ: в SQLite можно сделать первичный ключ автоинкрементным, используя тип данных INTEGER PRIMARY KEY AUTOINCREMENT. Это обеспечивает автоматическое увеличение значения ключа при вставке новой записи.

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

Ответ: инструкция NOT NULL используется для указания того, что поле не может содержать значение NULL (Для каждой строки таблицы данное поле обязательно должно иметь значение). Инструкция DEFAULT задаёт значение по умолчанию для поля, которое будет использоваться, если при вставке данных не указано значение для этого поля.

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

Ответ: внешние ключи используются для установления связи между двумя таблицами. Внешний ключ в одной таблице указывает на первичный ключ другой таблицы, тем самым обеспечивая целостность данных и предотвращая создание несогласованных записей. Инструкция FOREIGN KEY в момент создания таблицы позволяет создать внешний ключ. Внешний

ключ указывает, что значения в одном поле должны совпадать со значениями первичного ключа в другой таблице.

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

Ответ: для вставки строки в таблицу SQLite используется инструкция INSERT INTO.

14. Как выбрать данные из таблицы SQLite?

Ответ: для выбора данных из таблицы в SQLite используется инструкция SELECT, позволяющая выбрать одну или несколько строк из таблицы, а также определённые поля.

15. Как ограничить выборку данных с помощью условия WHERE?

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

16. Как упорядочить выбранные данные?

Ответ: данные можно 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?

Ответ: UPDATE ... SET – обновление полей записи, UPDATE “Имя таблицы”, SET “Столбец” = значение, WHERE условие.

18. Как удалить записи из таблицы SQLite?

Ответ: DELETE FROM – удаление записей таблицы, DELETE FROM “Имя таблицы” WHERE условие.

19. Как сгруппировать данные из выборки/из таблицы SQLite?

Ответ: оператор GROUP BY 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?

Ответ: в SQL есть функции агрегирования данных: count(), sum(), avr(), min(), max(), способные выполнить задачу.

21. Как выполнить объединение нескольких таблиц в операторе SELECT?

Ответ: после FROM указываются обе сводимые таблицы через JOIN.

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

Ответ: подзапросы помогают уменьшить работу путём создания дополнительного запроса внутри основного, а шаблоны реализуют поиск по таблице, если неизвестно полное название данных в строке.

23. Каково назначение представлений VIEW в SQLite?

Ответ: для сохранения результата выборки для дальнейшего использования в языке SQL используется оператор CREATE VIEW, который создает представление – виртуальную таблицу. В неё сохраняется результат запроса.

24. Какие существуют средства для импорта данных в SQLite?

Ответ: .import позволяет импортировать данные. Если указать перед этим .mode, то достаточно указание файла и название таблицы. Если нет – нужно указать –тип файла.

25. Каково назначение команды .schema ?

Ответ: schema показывает, какие столбцы есть в таблице, тип их данных и прочее.

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

Ответ: для группировки GROUP BY, а для сортировки ORDER BY.

27. Каково назначение "табличных выражений" в SQLite?

Ответ: табличные выражения в SQLite используются для структурирования запросов и улучшения их читаемости и гибкости.

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

Ответ: для экспорта сначала указывается режим (.mode csv или json), после чего идёт экспорт .output “название и расширение файла” . После этого вывод будет перенаправлен в данный файл.

29. Какие еще форматы для экспорта данных Вам известны?

Ответ: данные в SQLite можно экспортировать в форматах XML, HTML, а также в виде SQL скриптов.

Вывод: в ходе выполнения практической работы было исследовано взаимодействие с базами данных SQLite3 с помощью языка программирования Python.