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 РАБОТЕ №2.21**

# дисциплины «Основы кроссплатформенного программирования»

Выполнил:

Баканов Артем Вадимович

2 курс, группа ИТС-б-о-22-1,

11.03.02 «Инфокоммуникационные технологии и системы связи», направленность (профиль) «Инфокоммуникационные системы и

сети», очная форма обучения

(подпись)

Руководитель практики:

Воронкин Р.А., канд. тех. наук, доцент,

доцент кафедры инфокоммуникаций

(подпись)

Отчет защищен с оценкой Дата защиты

Ставрополь, 2023 г.

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

**Порядок выполнения работы:**

1. Создала новый репозиторий и клонировала его на свой компьютер.

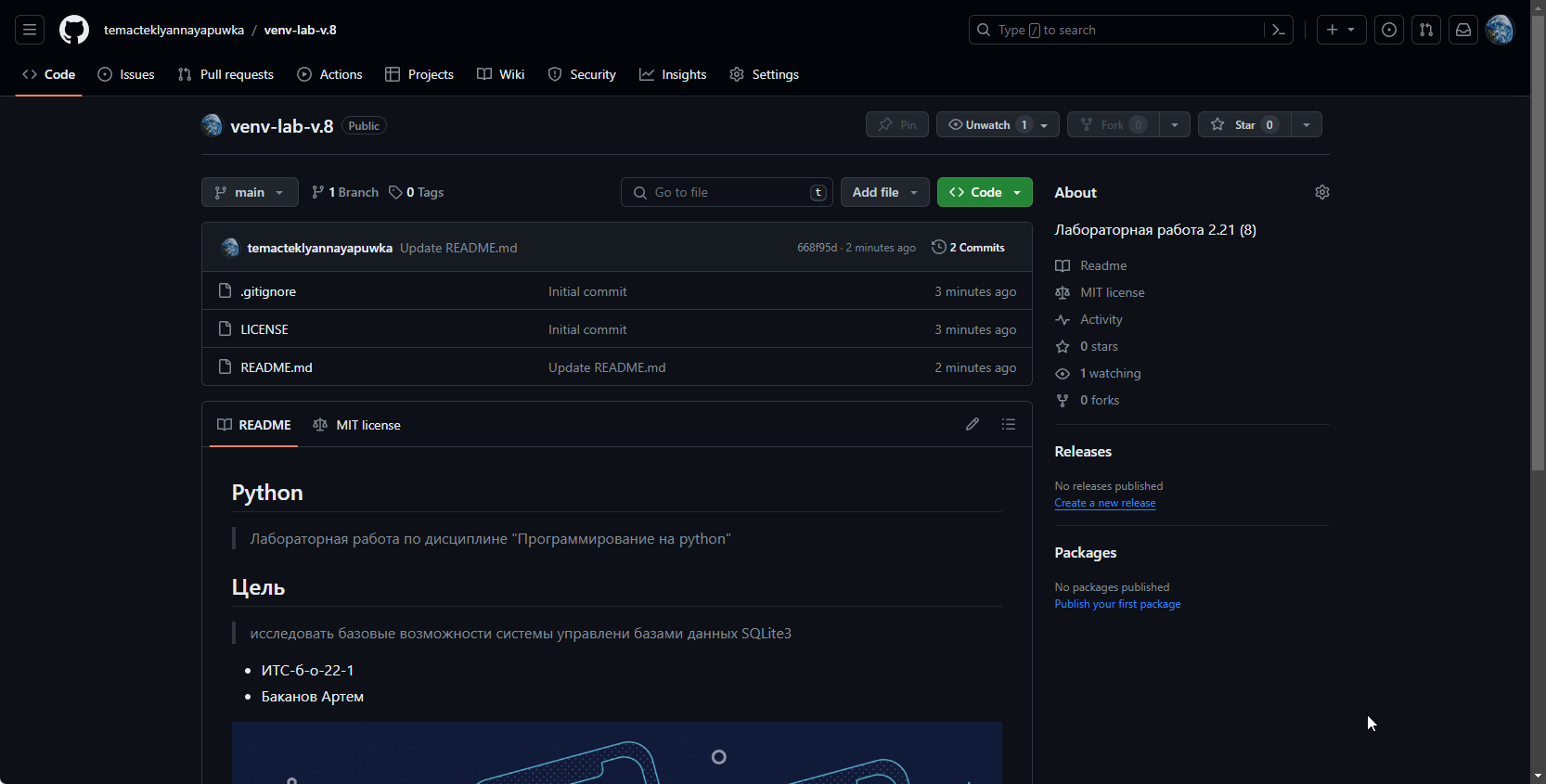


Рисунок 1 – Создан новый репозиторий

1. Клонировал репозиторий на свой компьютер. В ходе данной лабораторной работы работала с моделью ветвления git-flow.

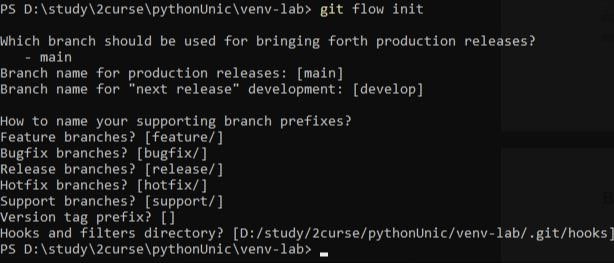


Рисунок 2 – Клонирование и модель ветвления git-flow

1. Создал виртуальное окружение Anaconda с именем репозитория.

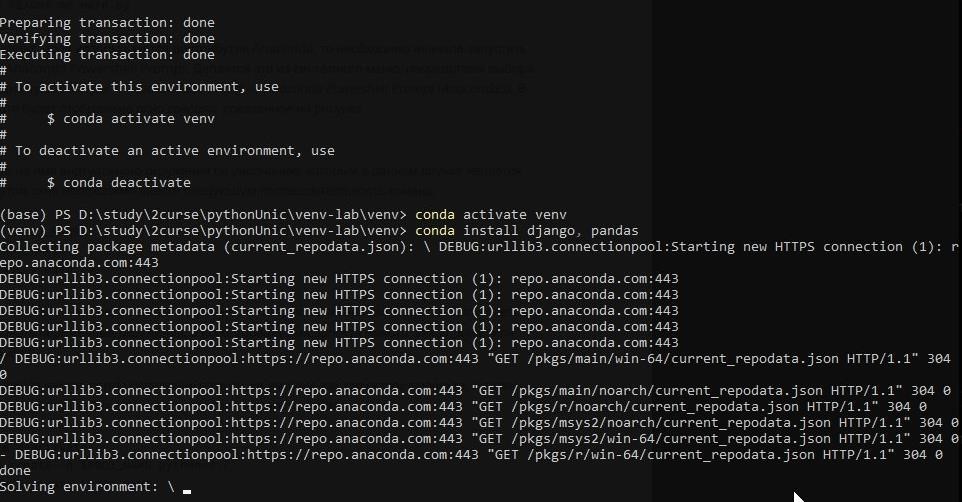


Рисунок 3 – Создание виртуального окружения

**Выполнение индивидуального задания**

С помощью команд sqlite3 создадим базу данных, а затем в ней создадим таблицы

Добавим магазины и товары в них с помощью команд консоли, а затем отобразим содержимое таблицы

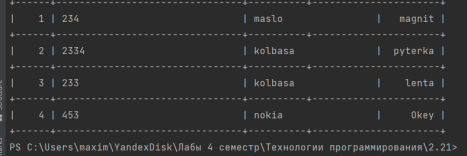


Рисунок 4 – Содержимое таблицы с магазинами

При помощи команды выбора магазина найдем нужный магазин

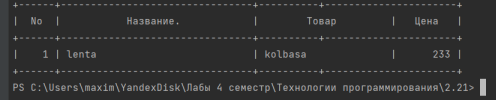


Рисунок 5 – Нужный магазин найден

**Ответы на контрольные вопросы**

1. Непосредственно модуль sqlite3 – это API к СУБД SQLite. Своего рода адаптер, который переводит команды, написанные на Питоне, в команды, которые понимает SQLite. Как и наоборот, доставляет ответы от SQLite в python-программу.

2. Для взаимодействия с базой данных SQLite3 в Python необходимо создать объект cursor. Вы можете создать его с помощью метода cursor() . Курсор SQLite3 – это метод объекта соединения. Для выполнения инструкций SQLite3 сначала устанавливается соединение, а затем создается объект курсора с использованием объекта соединения

3. При создании соединения с SQLite3 автоматически создается файл базы данных, если он еще не существует. Этот файл базы данных создается на диске, мы также можем создать базу данных в оперативной памяти с помощью функции :memory: with the connect. Такая база данных называется базой данных в памяти.

4. С помощью команды закрытия close().

5. Чтобы вставить данные в таблицу, используется оператор INSERT INTO.

6. Чтобы обновить данные в таблице, просто создайте соединение, затем создайте объект курсора с помощью соединения и, наконец, используйте оператор UPDATE.

7. Оператор SELECT используется для выбора данных из определенной таблицы. Если вы хотите выбрать все столбцы данных из таблицы, вы можете использовать звездочку (\*).

8. SQLite3 rowcount используется для возврата количества строк, которые были затронуты или выбраны последним выполненным SQLзапросом.

9. Чтобы перечислить все таблицы в базе данных SQLite3, вы должны запросить данные из таблицы sqlite\_master, а затем использовать fetchall() для получения результатов из инструкции SELECT

10. При создании таблицы мы должны убедиться, что она еще не существует. Аналогично, при удалении/удалении таблицы она должна существовать. Чтобы проверить, не существует ли таблица уже, мы используем IF NOT EXISTS с оператором CREATE TABLE следующим образом.

11. Метод executemany можно использовать для вставки нескольких строк одновременно.

12. В базе данных Python SQLite3 мы можем легко хранить дату или время, импортируя модуль datetime . Следующие форматы являются наиболее часто используемыми форматами для datetime:

**Вывод:** в результате выполнения работы были приобретены навыки по работе c базовыми возможностями системы управления базами данных SQLite3.