

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 №23**  
**дисциплины «Основы программной инженерии»**

Выполнил:  
Матвеев Александр Иванович  
2 курс, группа ПИЖ-б-о-22-1,  
09.03.04 «Программная инженерия»,  
направленность (профиль) «Разработка и  
сопровождение программного  
обеспечения», очная форма обучения

---

(подпись)

Проверил Воронкин Роман Александрович

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

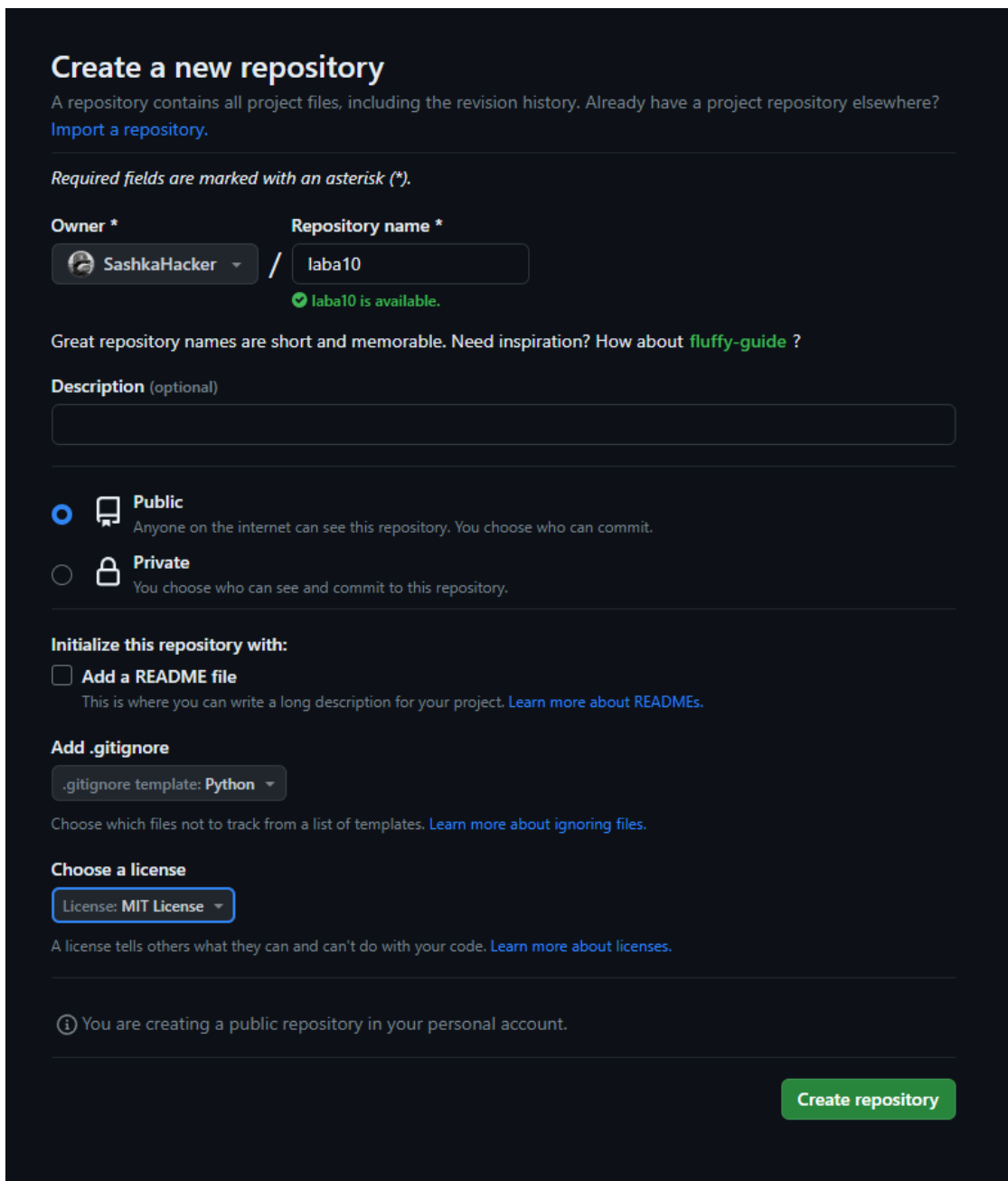
Ставрополь, 2024 г.

**Тема:** Взаимодействие с базами данных SQLite3 с помощью языка программирования Python.

**Цель работы:** Приобретение навыков по работе с базами данных SQLite3 при помощи языка программирования Python версии 3.x.

Ход работы.

### 1. Создание нового репозитория с лицензией MIT.



**Create a new repository**

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

*Required fields are marked with an asterisk (\*).*

**Owner \*** SashkaHacker / **Repository name \*** laba10

✔ laba10 is available.

Great repository names are short and memorable. Need inspiration? How about [fluffy-guide](#) ?

**Description** (optional)

☒ **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**  
You choose who can see and commit to this repository.

**Initialize this repository with:**

☐ **Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

**Add .gitignore**

.gitignore template: Python

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

**Choose a license**

License: MIT License

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

ⓘ You are creating a public repository in your personal account.

**Create repository**

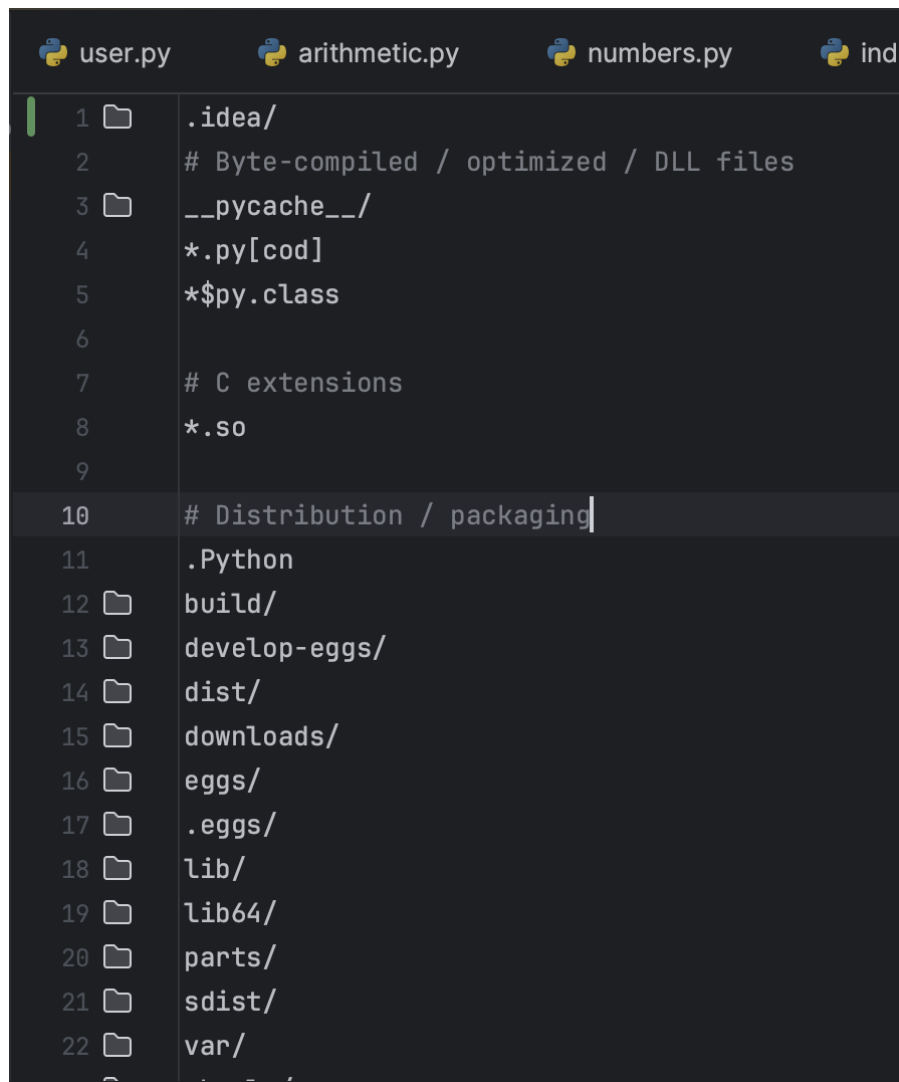
Рисунок 1 – Создание репозитория

2. Клонировал репозиторий на рабочий ПК.

```
[→ GitHub git clone https://github.com/SashkaHacker/laba16.git
Cloning into 'laba16'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.
```

Рисунок 2 – Клонирование репозитория

3. Дополнил файл .gitignore необходимыми инструкциями.



The image shows a code editor with a dark theme. At the top, there are four tabs: 'user.py', 'arithmetic.py', 'numbers.py', and 'ind'. The active tab is 'user.py'. The editor displays the content of a '.gitignore' file. The file is numbered from 1 to 22. The content is as follows:

```
1  .idea/
2  # Byte-compiled / optimized / DLL files
3  __pycache__/
4  *.py[cod]
5  *$py.class
6
7  # C extensions
8  *.so
9
10 # Distribution / packaging
11 .Python
12 build/
13 develop-eggs/
14 dist/
15 downloads/
16 eggs/
17 .eggs/
18 lib/
19 lib64/
20 parts/
21 sdist/
22 var/
```

Рисунок 3 – Файл .gitignore

#### 4. Проработка примера №1 из лабораторной работы.

```
(venv) → examples git:(develop) × python3 ex1.py add --db mydatabase.db -n "Иван Иванов" -р "Инженер" -у 2015
(venv) → examples git:(develop) × python3 ex1.py display --db mydatabase.db
```

No	Ф.И.О.	Должность	Год
1	Иван Иванов	Инженер	2015

Рисунок 4 – Демонстрация работы примера №1

```
(venv) → examples git:(develop) × python3 ex1.py select --db mydatabase.db --period 5
```

No	Ф.И.О.	Должность	Год
1	Иван Иванов	Инженер	2015

```
(venv) → examples git:(develop) ×
```

Рисунок 5 – Демонстрация работы примера №1

5. Выполнение индивидуального задания. Для своего варианта лабораторной работы 2.17 необходимо реализовать хранение данных в базе данных SQLite3. Информация в базе данных должна храниться не менее чем в двух таблицах.

```
PS C:\Users\Sashka\Documents\GitHub\laba23\individual> python .\individual1.py add --surname=Матвеев --name=Александр -р=Инженер --phone=893274983824 --date=28:05:2004
PS C:\Users\Sashka\Documents\GitHub\laba23\individual> python .\individual1.py display
```

№	Фамилия	Имя	Номер телефона	Дата рождения
1	Матвеев	Александр	893274983824	28:05:2004

Рисунок 6 – Демонстрация работы индивидуального задания

```
PS C:\Users\Sashka\Documents\GitHub\laba23\individual> python .\individual1.py select -p=893274983824
Фамилия: Матвеев
Имя: Александр
Номер телефона: 893274983824
Дата рождения: 28:05:2004
```

Рисунок 7 – Демонстрация работы индивидуального задания

6. Выполнение индивидуального задания повышенной сложности. Самостоятельно изучите работу с пакетом python-psycopg2 для работы с

базами данных PostgreSQL. Для своего варианта лабораторной работы 2.17 необходимо реализовать возможность хранения данных в базе данных СУБД PostgreSQL. Информация в базе данных должна храниться не менее чем в двух таблицах.

```
(env) PS C:\Users\Sashka\Documents\GitHub\laba23\individual> python .\individual_hard.py add --surname=Матвеев --name=Александр -p=Инженер --phone=893274983824 --date=28:05:2004
(env) PS C:\Users\Sashka\Documents\GitHub\laba23\individual> python .\individual_hard.py display
```

№	Фамилия	Имя	Номер телефона	Дата рождения
1	Матвеев	Александр	893274983824	28:05:2004

Рисунок 8 – Демонстрация работы индивидуального задания повышенной сложности

```
(env) PS C:\Users\Sashka\Documents\GitHub\laba23\individual> python .\individual_hard.py select -p=893274983824
Фамилия: Матвеев
Имя: Александр
Номер телефона: 893274983824
Дата рождения: 28:05:2004
```

Рисунок 9 – Демонстрация работы индивидуального задания повышенной сложности

### Контрольные вопросы:

1. Модуль `sqlite3` в Python предназначен для работы с легковесной дисковой базой данных SQLite. Он позволяет выполнять операции SQL, такие как создание таблиц, вставка, обновление и извлечение данных, без необходимости устанавливать отдельный сервер базы данных.

2. Соединение с базой данных SQLite3 выполняется с помощью функции `sqlite3.connect()`, которая принимает путь к файлу базы данных. Курсор базы данных - это объект, который используется для управления контекстом выполнения вашего запроса и извлечения результатов.

3. Чтобы подключиться к базе данных SQLite3, находящейся в оперативной памяти компьютера, используйте строку `":memory:"` в качестве аргумента функции `connect()`: `conn = sqlite3.connect(":memory:")`.

4. Чтобы корректно завершить работу с базой данных SQLite3, необходимо закрыть соединение с помощью метода `conn.close()`. Перед этим следует убедиться, что все транзакции были завершены с помощью `conn.commit()`.

5. Вставка данных в таблицу базы данных SQLite3 осуществляется с помощью метода `cursor.execute()`, в который передается SQL-запрос `INSERT INTO`.

6. Обновление данных таблицы базы данных SQLite3 осуществляется с помощью метода `cursor.execute()`, в который передается SQL-запрос `UPDATE`.

7. Выборка данных из базы данных SQLite3 осуществляется с помощью метода `cursor.execute()`, в который передается SQL-запрос `SELECT`. Результаты выборки можно извлечь с помощью методов `cursor.fetchone()`, `cursor.fetchmany()` или `cursor.fetchall()`.

8. Метод `rowcount` возвращает количество строк, которые были затронуты последним выполненным запросом (например, `INSERT`, `UPDATE`, `DELETE`).

9. Получить список всех таблиц базы данных SQLite3 можно с помощью запроса `SELECT name FROM sqlite_master WHERE type='table'`.

10. Проверку существования таблицы можно выполнить с помощью запроса `SELECT name FROM sqlite_master WHERE type='table' AND name='table_name'`. При добавлении используйте `CREATE TABLE IF NOT EXISTS`, а при удалении - проверьте наличие таблицы перед выполнением `DROP TABLE`.

11. Массовая вставка данных в базу данных SQLite3 может быть выполнена с помощью метода `cursor.executemany()`, который принимает SQL-запрос и последовательность параметров.

12. Работа с датой и временем в SQLite3 осуществляется с помощью встроенных функций SQLite, таких как `date()`, `time()`, `datetime()`, `strftime()`. SQLite хранит дату и время в текстовом формате, поэтому может потребоваться преобразование форматов при извлечении или вставке данных.