МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский политехнический университет Петра Великого»

Институт компьютерных наук и технологий

Высшая школа искусственного интеллекта

Направление: 02.03.03 Математическое обеспечение и администрирование информационных систем

**Отчет о выполнении курсовой работы Телефонная книга**

по дисциплине «Основы программирования и алгоритмизации»

Выполнил: Алексеев Н. С.

Cтудент группы: 5130203/30002

Руководитель: Ненад Йовановски

Санкт-Петербург

2025 г.

**Оглавление**

[Введение: 3](#_Toc191135718)

[Постановка задачи: 4](#_Toc191135719)

[Структура приложения: 5](#_Toc191135720)

[Реализация 9](#_Toc191135721)

[Заключение: 11](#_Toc191135722)

[Список использованных источников 12](#_Toc191135723)

# Введение:

В современном мире, где мобильные устройства и средства связи играют ключевую роль в повседневной жизни, эффективное управление контактной информацией становится важной задачей. Телефонная книга, как инструмент для хранения и организации контактов, является неотъемлемой частью любого устройства или приложения, связанного с коммуникацией. В данном отчете рассматриваются основные аспекты разработки, функциональности и использования телефонной книги, а также анализируются её возможности и преимущества. Проект представляет собой приложение Телефонный справочник, разработанное с использованием библиотеки Qt. Приложение позволяет управлять списком контактов, включая добавление,

удаление, редактирование, поиск и сортировку контактов. Данные могут

храниться в текстовом файле или в базе данных SQLite.

# Постановка задачи:

Написать приложение телефонный справочник с использованием библиотеки Qt

Подзадача 0: Название и назначение классов;

Подзадача 1: Написать программу без QT, и данные контрактов хранить в файл;

Подзадача 2: Добавить графический интерфейс помощью QT, также для хранения и чтения файла использовать QFile;

Подзадача 3: Не удаляя хранение в файл, добавить новую возможность для хранения данных в базе данных (любая база, предлагаю sqlite / postgresql);

Подзадача \*4\*: Переопределить operator new & copy и move operator (для класса контактов) и обосновать количество копирований и созданный элементов и даже постараться уменьшить количество если можно – пожеланию

Обязательные поля для хранения: Имя, Фамилия, Отчество, адрес, дата рождения, email, телефонные номера (рабочий, домашний, служебный) в любом количестве.

Организовать проверку всех вводимых данных на корректность при помощи регулярных выражений

Фамилия, Имя или Отчество --- должны содержать только буквы и цифры различных алфавитов, а также дефис и пробел, но при этом должны начинаться только на буквы, и не могли бы оканчиваться или начинаться на дефис. Все незначимые пробелы перед и после данных должны удаляться.

Варианты записи телефона:

+78121234567

88121234567

+7(812)1234567

8(812)1234567

+7(812)123-45-67

8(812)123-45-67

Дата рождения должна быть меньше текущей даты, число месяцев в дате должно быть от 1 до 12, число дней от 1 до 31, причем должно учитываться различное число дней в месяце и високосные года. (В третьей подзадаче для обработки даты как вариант можно использовать классы QDateEdit или QCalendarWidget)

E-mail должен содержать в себе имя пользователя состоящее из латинских букв и цифр, символ разделения пользователя и имени домена(@), а также сам домен состоящий из латинских букв и цифр. Все незначащие пробелы (включая пробелы перед и после символа @) должны быть удалены.

Приложение должно позволять добавлять/удалять записи, а также редактировать все поля. Осуществлять сортировку отображаемых данных по указанному полю и поиск по нескольким полям.

Для удобного представления данных пользователю, можно использовать таблицы (см. классы QTableWidget, QTableView)

# ****Структура приложения:****

**Файлы проекта:**

**contact.h / contact.cpp: Класс Contact для хранения данных о контакте.**

**phonebook.h / phonebook.cpp: Класс PhoneBook для управления списком контактов.**

**filestorage.h / filestorage.cpp: Класс FileStorage для работы с текстовыми файлами.**

**databasestorage.h / databasestorage.cpp: Класс DatabaseStorage для работы с базой данных SQLite.**

**mainwindow.h / mainwindow.cpp: Класс MainWindow для реализации графического интерфейса.**

**main.cpp: Точка входа в приложение.**

**CMakeLists.txt: Файл для сборки проекта с помощью CMake.**

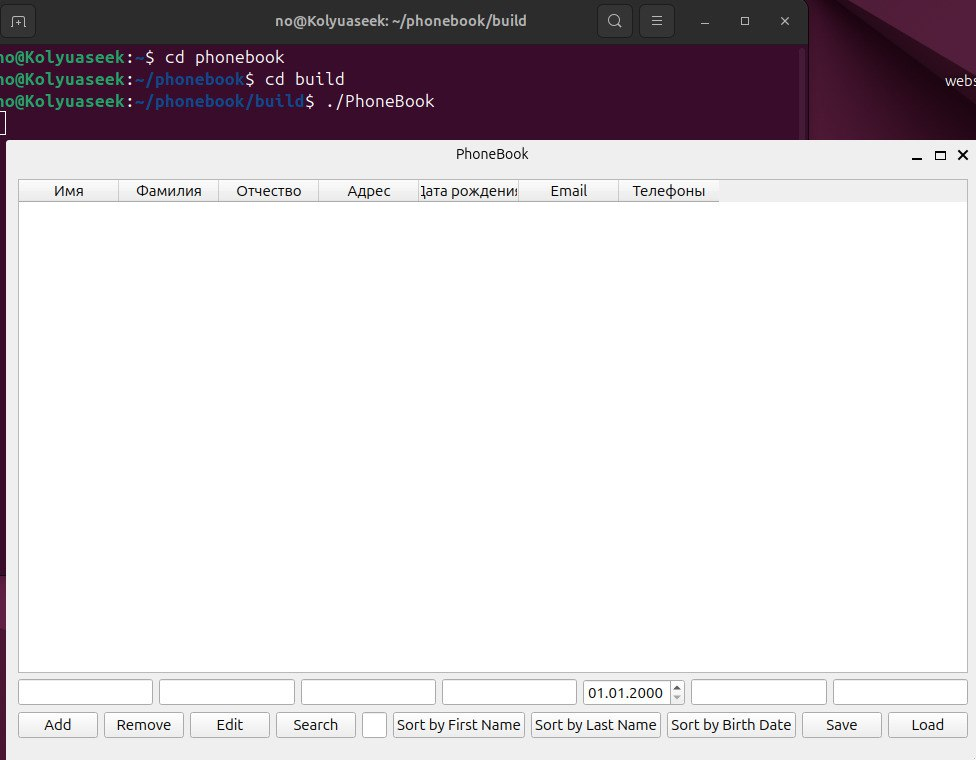


Рис 1 – Запуск нашего приложения

Функционал приложения, который изображен на рисунке (1)

Добавление контакта:

Пользователь может добавить новый контакт, заполнив поля: Имя, Фамилия, Отчество, Адрес, Дата рождения, Email, Телефонные номера.

Данные проверяются на корректность с использованием регулярных выражений

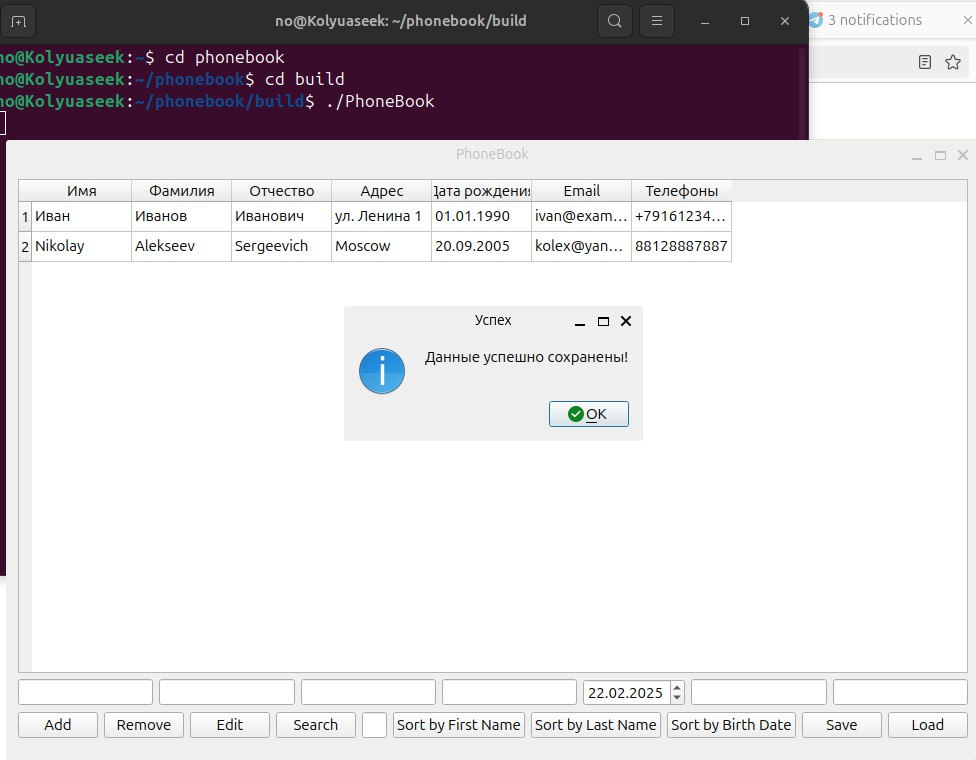


Рис 2 – Демонстрация добавление контакта

Удаление контакта:

Пользователь может удалить выбранный контакт из списка.

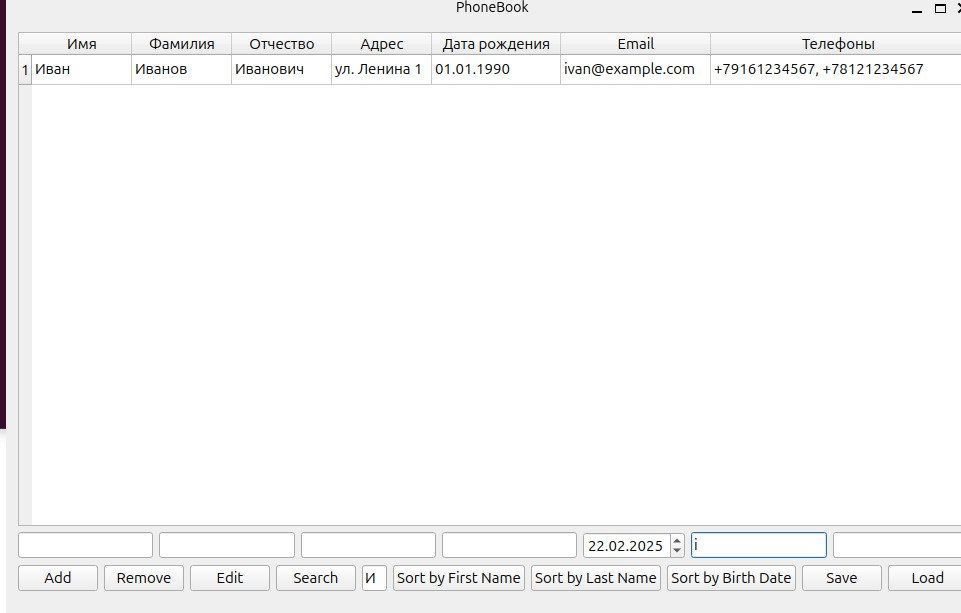


Рис 3 – Состояние системы после удаление контакта

На рисунке (3) мы видим, что пользователь удалил контакта “Nikolay” и все его данные. Для этого он нажал на контакт и нажал “Remove”.

Редактирование контакта:

Пользователь может изменить данные выбранного контакта.

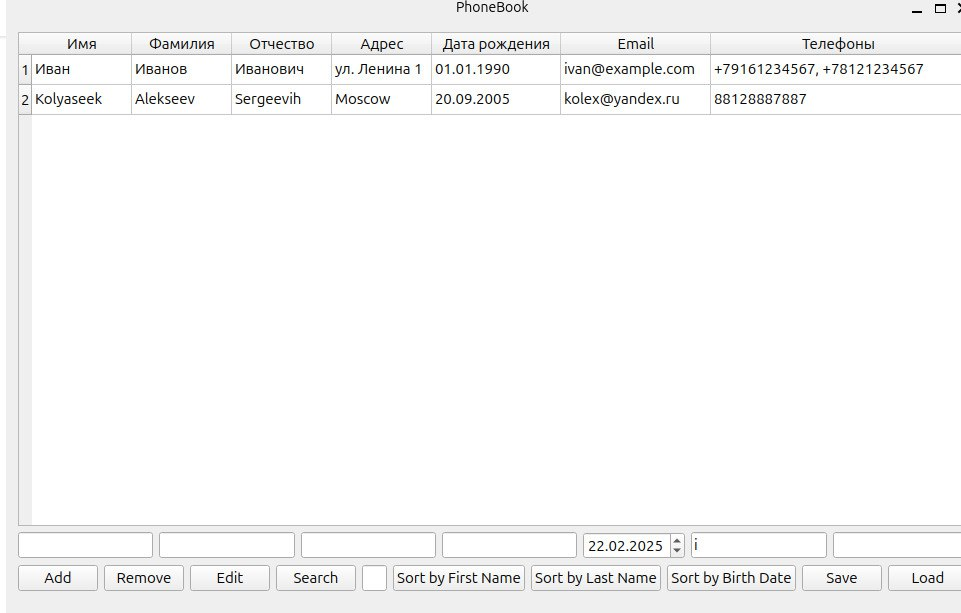


Рис 4 – Демонстрация изменения контакта

В данном случае пользователь изменил имя контакта “Nikolay” и сохранил изменения, нажав на кнопку “Edit”.

Поиск контактов:

Пользователь может искать контакты по нескольким полям (Имя, Фамилия, Email, Телефон).

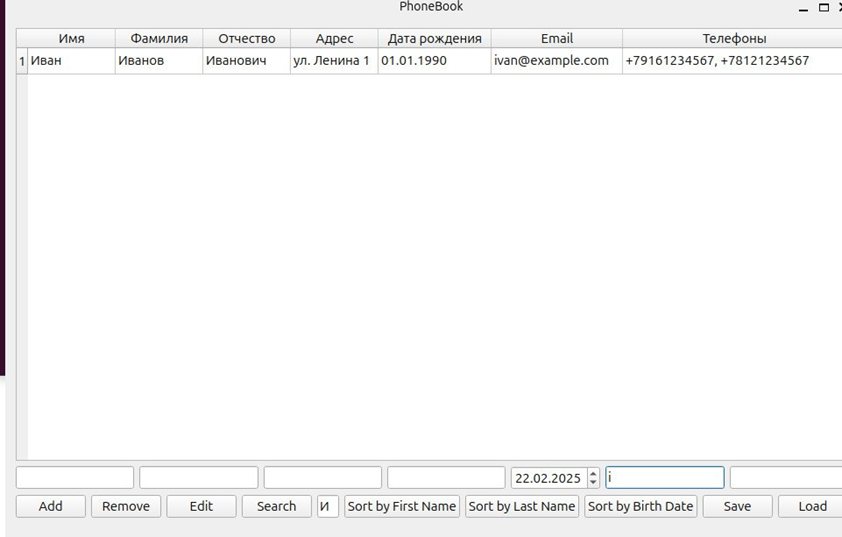


Рис 5 – Демонстрация поиска контакта

На рисунке (5) можно наблюдать поиск контакта, в данном случае поиск по имени “И”. Нам сразу же вывелись все имена, начинающие на букву “И”, по такому же принципу можно найти и другие контакты.

Сортировка контактов:

Пользователь может сортировать контакты по Имени, Фамилии или Дате рождения.

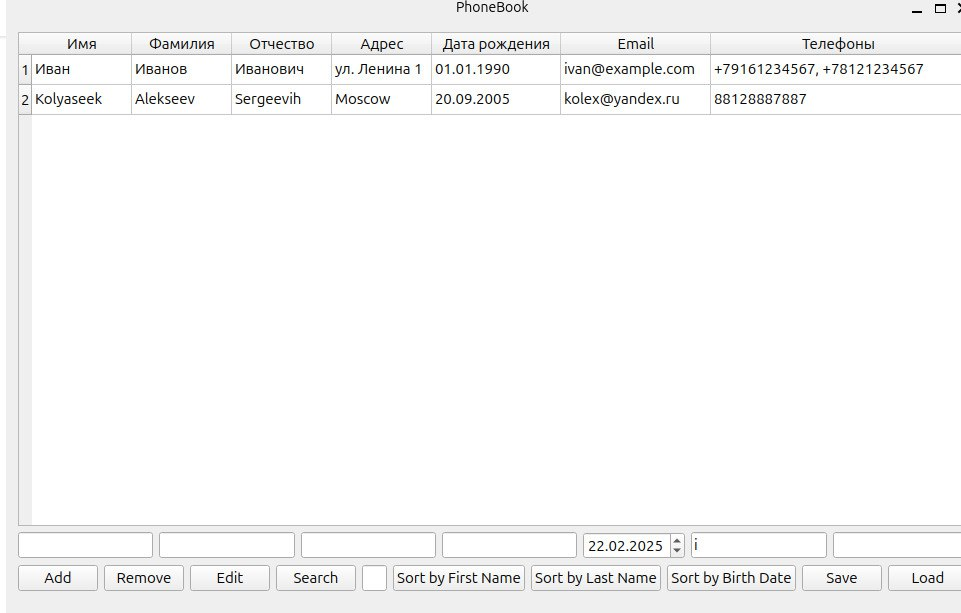


Рис 6 - Sort by Birth Date

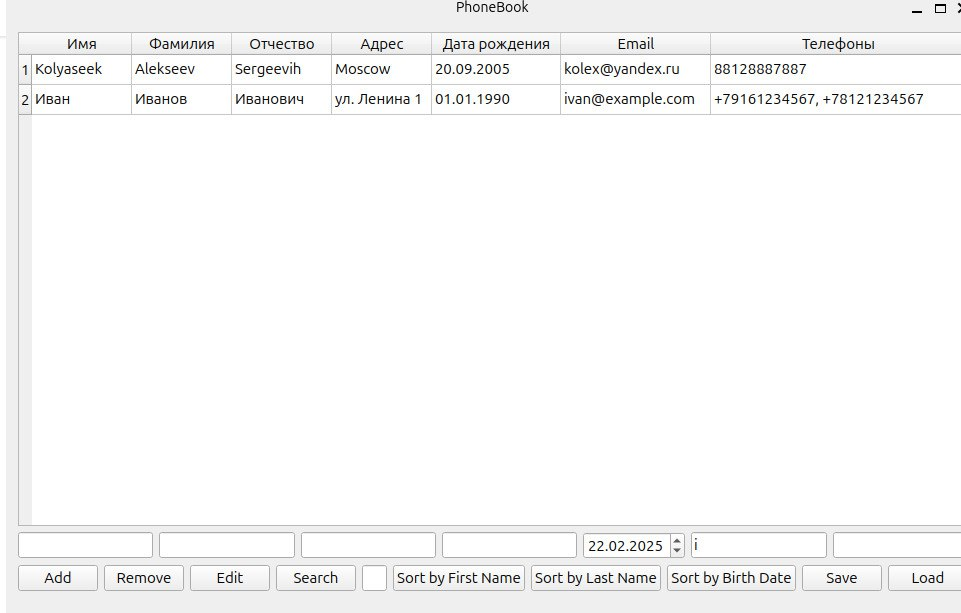


Рис 7 - Sort by First Name

Сохранение и загрузка данных:

Данные могут быть сохранены в текстовый файл или в базу данных SQLite.

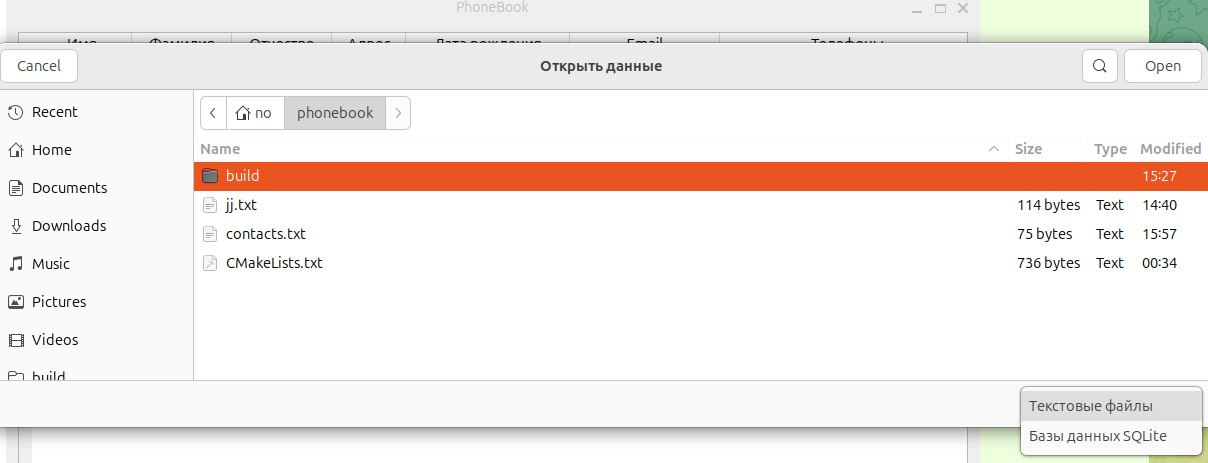


Рис 8 – Сохранение/Загрузка данных из текстового файла/ в файл

Данные могут быть загружены из текстового файла или базы данных.

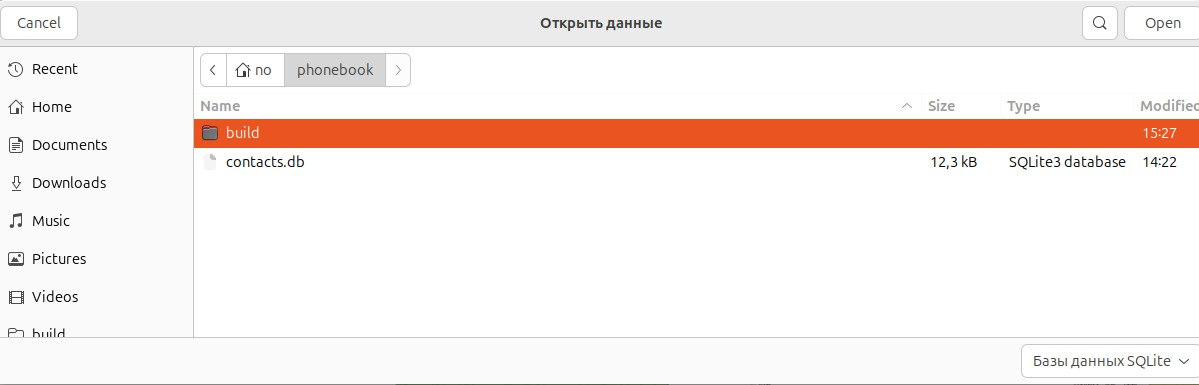


Рис 9 – Сохранение/Загрузка данных из базы данных/ в базу данных

# Реализация

Класс Contact:

Хранит данные о контакте: Имя, Фамилия, Отчество, Адрес, Дата рождения, Email, Телефонные номера.

Реализует проверку корректности данных.

Класс PhoneBook:

Управляет списком контактов.

Реализует методы для добавления, удаления, редактирования, поиска и сортировки контактов.

Класс FileStorage:

Сохраняет и загружает данные в текстовый файл.

Класс DatabaseStorage:

Сохраняет и загружает данные в базу данных SQLite.

Класс MainWindow:

Реализует графический интерфейс с использованием Qt.

Обеспечивает взаимодействие пользователя с приложением.

# Заключение:

Проект Телефонный справочник на Qt представляет собой законченное приложение для управления контактами, разработанное с использованием современных технологий и подходов. В процессе разработки были успешно решены все поставленные задачи, и приложение демонстрирует высокий уровень функциональности и удобства использования.

# Список использованных источников

Официальная документация Qt:

* + - Qt Documentation. [Электронный ресурс]. URL: https://doc.qt.io/
    - Описание классов, методов и примеров использования библиотеки Qt.

Книги по Qt и C++:

* + - Blanchette, J., Summerfield, M. "C++ GUI Programming with Qt 4". Prentice Hall, 2006.
    - Шлее, М. "Qt 5.10. Профессиональное программирование на C++". БХВ-Петербург, 2018.
    - Summers, M. "Advanced Qt Programming: Creating Great Software with C++ and Qt 4". Prentice Hall, 2010.

Руководства по SQLite:

* + - SQLite Documentation. [Электронный ресурс]. URL: https://www.sqlite.org/docs.html
    - Kreibich, J. A. "Using SQLite". O'Reilly Media, 2010.

Регулярные выражения:

* + - Friedl, J. E. F. "Mastering Regular Expressions". O'Reilly Media, 2006.
    - Документация по регулярным выражениям в Qt: https://doc.qt.io/qt-6/qregularexpression.html
    - Страуструп, Б. "Язык программирования C++". Бином, 2011.
    - Josuttis, N. M. "The C++ Standard Library: A Tutorial and Reference". Addison-Wesley, 2012.

Ресурсы по Git:

* + - Pro Git Book. [Электронный ресурс]. URL: https://git-scm.com/book/ru/v2