

**Цель:** Спроектировать, создать и заполнить базу данных для онлайн-библиотеки, а затем выполнить серию запросов для анализа данных.

**Часть 1: Проектирование и создание БД**

**Задача:** Создать базу данных online\_library.db со следующими таблицами:

**1. authors (Авторы)**

- author\_id (INTEGER, PRIMARY KEY)
- name (TEXT, NOT NULL)
- country (TEXT)

**2. genres (Жанры)**

- genre\_id (INTEGER, PRIMARY KEY)
- name (TEXT, NOT NULL, UNIQUE)

**3. books (Книги)**

- book\_id (INTEGER, PRIMARY KEY)
- title (TEXT, NOT NULL)
- author\_id (INTEGER, NOT NULL, FOREIGN KEY  
REFERENCES authors(author\_id))

- genre\_id (INTEGER, NOT NULL, FOREIGN KEY  
REFERENCES genres(genre\_id))

- published\_year (INTEGER)
- price (REAL, NOT NULL)
- quantity (INTEGER, NOT NULL DEFAULT 0)

**4. clients (Клиенты)**

- client\_id (INTEGER, PRIMARY KEY)
- first\_name (TEXT, NOT NULL)
- last\_name (TEXT, NOT NULL)
- email (TEXT, UNIQUE)

## 5. orders (Заказы)

- order\_id (INTEGER, PRIMARY KEY)
- client\_id (INTEGER, NOT NULL, FOREIGN KEY REFERENCES clients(client\_id))
- book\_id (INTEGER, NOT NULL, FOREIGN KEY REFERENCES books(book\_id))
- order\_date (TEXT, NOT NULL DEFAULT (date('now'))) -- Дата в формате 'YYYY-MM-DD'
- quantity (INTEGER, NOT NULL CHECK (quantity > 0))
- status (TEXT, NOT NULL DEFAULT 'В обработке' CHECK(status IN ('В обработке', 'Выполнен', 'Отменен')))

### Задание:

Напишите SQL-скрипт для создания всех таблиц с указанными полями и ограничениями (PRIMARY KEY, FOREIGN KEY, NOT NULL, UNIQUE, CHECK, DEFAULT).

### Часть 2: Заполнение БД тестовыми данными

**Задание:** Напишите SQL-скрипт для вставки тестовых данных в

каждую таблицу (минимум по 5 записей в каждую, чтобы запросы из Части 3 могли работать). Примеры данных:

- **Авторы:** Ф.М. Достоевский (Россия), Дж. Роулинг

(Великобритания), Дж. Оруэлл (Великобритания), Рэй Брэдбери (США), Эрих Мария Ремарк (Германия).

- **Жанры:** Роман, Фэнтези, Антиутопия, Научная фантастика, Классика.

- **Книги:** "1984" (Оруэлл, Антиутопия, 1949),

"Гарри Поттер и философский камень" (Роулинг, Фэнтези, 1997), "Идиот"

(Достоевский, Роман, 1869) и т.д. Укажите разные цены и количество на складе.

- **Клиенты:** Иван Иванов, Мария Петрова и др.

- **Заказы:** Создайте несколько заказов от разных клиентов на разные книги с разными датами и статусами.

Убедитесь, что при создании заказа quantity в orders не превышает quantity в books.

### **Часть 3: Написание запросов**

**Задание:** Напишите SQL-запросы, которые выводят следующую

информацию:

1. 2. 3. **Простой SELECT**: Показать название и год издания всех книг, отсортированных по году издания (от новых к старым).

**SELECT с JOIN (один)**: Показать название книги и полное имя автора для всех книг.

**SELECT с JOIN (несколько) и фильтрацией**: Показать имена клиентов и названия книг, которые они заказывали в январе 2024 года. Отсортировать по имени клиента.

**GROUP BY и агрегирующая функция**: Посчитать общее количество книг на складе (сумма quantity из таблицы books).

5. **GROUP BY + HAVING**: Показать жанры, у которых средняя цена книги выше 500 рублей.

6. **Вложенный запрос (подзапрос)**: Показать названия книг, которые никогда не заказывались. (Используйте NOT IN или NOT EXISTS).

7. **Работа с датами**: Показать все заказы за последние 30 дней (используйте функцию date('now')).

**8. UPDATE + условие:** Увеличить цену всех книг в жанре "Фэнтези" на 10%.

**9. DELETE + условие:** Удалить всех клиентов, которые не сделали ни одного заказа.

**10. Сложный запрос с агрегацией по нескольким таблицам:** Для каждого автора посчитать общее количество проданных книг (сумма quantity из orders для всех его книг). Вывести имя автора и общее количество.