

СОДЕРЖАНИЕ

	Стр.
1 Введение	2
1.1 Почему эта тема важна.....	2
2 Теоритическая база	3
3 Цели и задачи.....	4
4 Технические требования к выполнению лабораторной работы	5
5 Ожидаемые результаты и выполнение разработанной лабо- раторной работы с демонстрацией результатов.....	6
6 Выводы	8
7 Список источников.....	9

1 Введение

Изучение баз данных и языка SQL имеет довольно большое значение в современном мире информационных технологий. Мы выбрали эту тему для проекта, так как понимаем важность умения работать с данными, организовывать их, а также извлекать нужную информацию. Эти навыки крайне полезны для будущей карьеры в области информационных технологий, бизнес-аналитики, разработки программного обеспечения и других смежных областей.

1.1 Почему эта тема важна

1. Повышение конкурентоспособности:

Знание баз данных и SQL сделает студентов более востребованными на рынке труда, поскольку большинство компаний нуждаются в специалистах, способных эффективно управлять данными.

2. Улучшение навыка принятия решений:

Понимание баз данных и SQL помогает в принятии обоснованных и логичных решений на основе данных, что является ключевым навыком во многих областях деятельности.

3. Развитие навыков анализа:

Изучение баз данных и SQL развивает у студентов умение анализировать данные, выявлять закономерности и тенденции, что является ценным навыком в различных профессиональных областях.

В итоге, изучение баз данных и SQL не только полезно в практическом плане, но также способствует развитию критического мышления и способностей к анализу данных, что делает эту тему важной для студентов. Также хочется добавить, что подобный предмет ждет нас в учебном плане в будущем, поэтому приобрести базовые знания по этой теме будет полезно каждому.

2 Теоритическая база

База данных — это упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе.

База данных обычно управляется системой управления базами данных (СУБД). Данные вместе с СУБД, а также приложения, которые с ними связаны, называются системой баз данных, или, для краткости, просто базой данных.

Базы данных используются для решения множества задач. Например, они могут использоваться для хранения информации о клиентах и заказах в компании, для анализа больших объемов данных, для поддержки электронной коммерции, а также для управления информацией о научных исследованиях.

Существует несколько типов БД, включая реляционные, иерархические, сетевые и объектно-ориентированные. Реляционные БД являются наиболее распространенным типом и основываются на принципе таблиц, состоящих из строк и столбцов. Каждая таблица представляет отдельную сущность, а столбцы представляют атрибуты этой сущности. Связи между таблицами устанавливаются с помощью ключей, которые позволяют объединять данные из разных таблиц.

Центральным понятием в БД является язык структурированных запросов (SQL), который позволяет выполнять различные операции с данными, такие как выборка, вставка, обновление и удаление. SQL является стандартным языком взаимодействия с реляционными БД.

3 Цели и задачи

1. Познакомиться с базой данных, состоящей из нескольких таблиц.
2. Разобраться в структуре sql-запроса и его возможных параметрах.
3. Найти необходимые данные с помощью sql-запросов.
4. Создать собственную БД.
5. Создать и связать таблицы для разработанной БД.
6. Реализовать функции для работы с БД.

4 Технические требования к выполнению лабораторной работы

1. Наличие pip.
2. Наличие Git.
3. Наличие sqlitestudio.
4. Наличие Python и подключенных библиотек.

5 Ожидаемые результаты и выполнение разработанной лабораторной работы с демонстрацией результатов

В результате выполненной лабораторной работы студент должен продемонстрировать базу данных, которая включает в себя таблицы с книгами, жанрами и пользователями, что связаны между собой через внешний ключ и поддерживают функции добавления, удаления и демонстрации имеющихся данных.

Вот изображение желаемых результатов в виде скриншотов из выполненной лабораторной работы:

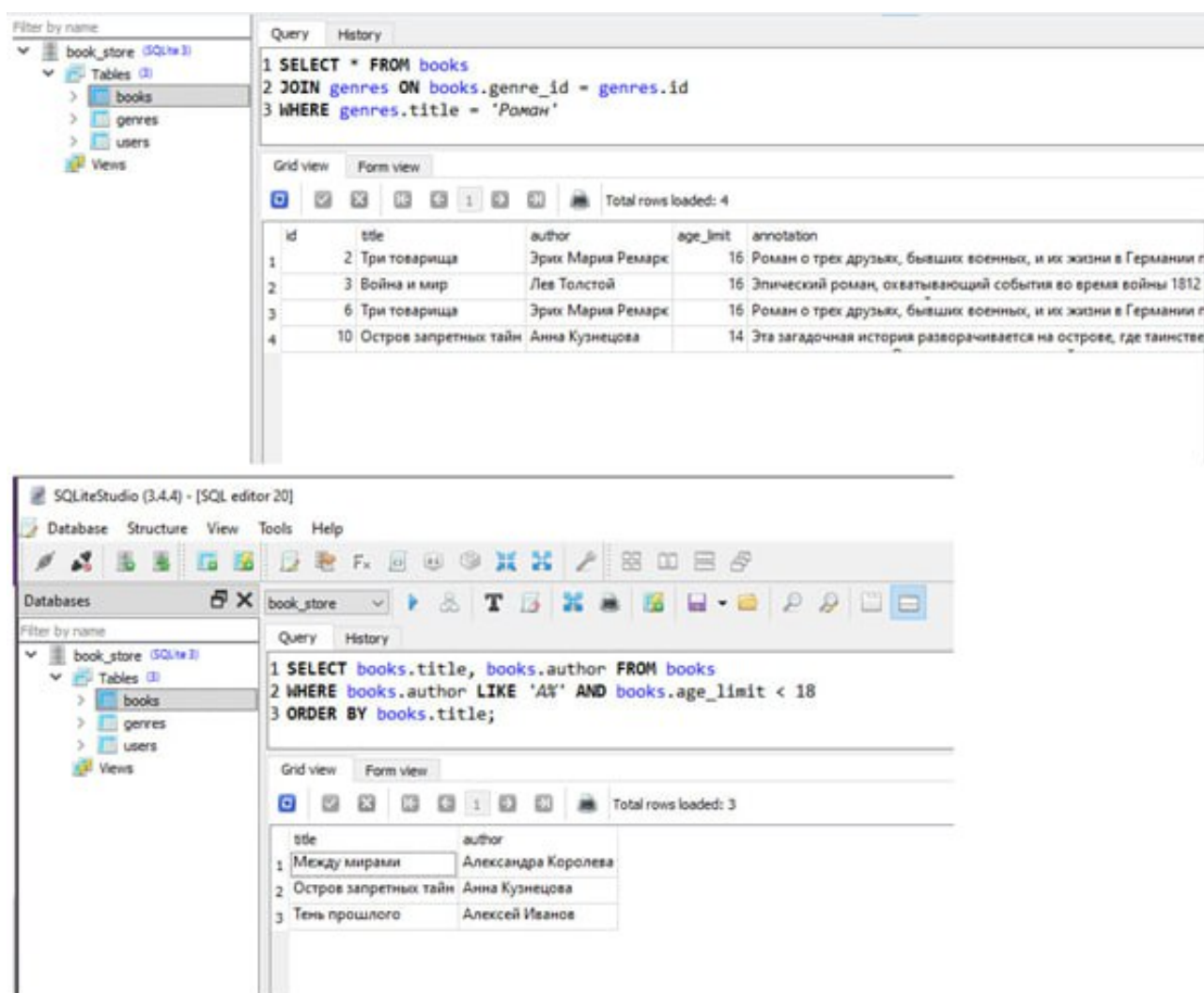


Рисунок 5.1 — SQL-запросы

```
Code Blame 13 lines (10 loc) · 498 Bytes Code 55% faster with GitHub Copilot

1 import sqlalchemy
2 from .db_session import SqlAlchemyBase
3 from sqlalchemy_serializer import SerializerMixin
4 from sqlalchemy import orm
5
6
7 class User(SqlAlchemyBase, SerializerMixin):
8     __tablename__ = 'users'
9
10    id = sqlalchemy.Column(sqlalchemy.Integer,
11                           primary_key=True, autoincrement=True)
12    nickname = sqlalchemy.Column(sqlalchemy.String, index=True, nullable=True, unique=False, autoincrement=True)
13    books = sqlalchemy.Column(sqlalchemy.String, nullable=True)
```

Рисунок 5.2 — Форма для таблицы с пользователями

```
Code Blame 13 lines (10 loc) · 498 Bytes Code 55% faster with GitHub Copilot

1 import sqlalchemy
2 from .db_session import SqlAlchemyBase
3 from sqlalchemy_serializer import SerializerMixin
4 from sqlalchemy import orm
5
6
7 class User(SqlAlchemyBase, SerializerMixin):
8     __tablename__ = 'users'
9
10    id = sqlalchemy.Column(sqlalchemy.Integer,
11                           primary_key=True, autoincrement=True)
12    nickname = sqlalchemy.Column(sqlalchemy.String, index=True, nullable=True, unique=False, autoincrement=True)
13    books = sqlalchemy.Column(sqlalchemy.String, nullable=True)
```

Рисунок 5.3 — Функции создания, получения и удаления пользователя

```
Code Blame 13 lines (10 loc) · 498 Bytes Code 55% faster with GitHub Copilot

1 import sqlalchemy
2 from .db_session import SqlAlchemyBase
3 from sqlalchemy_serializer import SerializerMixin
4 from sqlalchemy import orm
5
6
7 class User(SqlAlchemyBase, SerializerMixin):
8     __tablename__ = 'users'
9
10    id = sqlalchemy.Column(sqlalchemy.Integer,
11                           primary_key=True, autoincrement=True)
12    nickname = sqlalchemy.Column(sqlalchemy.String, index=True, nullable=True, unique=False, autoincrement=True)
13    books = sqlalchemy.Column(sqlalchemy.String, nullable=True)
```

Рисунок 5.4 — Функции получения книг пользователя, добавления жанра, получения жанров и удаления жанра

6 Выводы

Благодаря данной лабораторной работе студенты научились самостоятельно создавать простые базы данных, формировать sql-запросы и реализовывать функции для работы с базами данных.

7 Список источников

1 Документация по SQLAlchemy - URL: <https://docs.sqlalchemy.org/en/14/orm/index.html> (дата обращения по ссылке: 01.12.23)

2 Официальный сайт приложения Sqlitestudio - URL: <https://sqlitestudio.pl/> (дата обращения по ссылке: 01.12.23)

3 GitHub - крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки <https://github.com/>. - URL: <https://github.com/> (дата обращения по ссылке: 01.12.23)

4 METANIT.COM – сайт о программировании. - URL: <https://metanit.com/python/database/3.1.php> (дата обращения по ссылке: 01.12.23)