МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
 РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Чувашский государственный университет имени И.Н. Ульянова»

Факультет информатики и вычислительной техники

Кафедра математического и аппаратного обеспечения   
 информационных систем

Лабораторная работа №2 по дисциплине “Безопасность систем баз данных”

Тема:SQL-запросы: форматирование и стиль.

Проектирование БД: ER-модели и модели данных.

Выполнил: студент гр. ИВТ 21-22

Иванов Н.А. \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Проверил

Старший преподаватель

Марков А.В. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Цель работы:**

1. Выбрать code style.
2. Сформулировать основные лексические(как именовать), синтаксические(как располагать) правила.
3. Переделать скрипты из лабораторной работы №1 в соответствии с code style.
4. Изучить средства для построения диаграмм, выбрать одно из них(необязательно из тех, которые приведены в работе).
5. Создать в выбранном средстве диаграммы для базы данных из предыдущих лабораторных работ.

**Результат:**

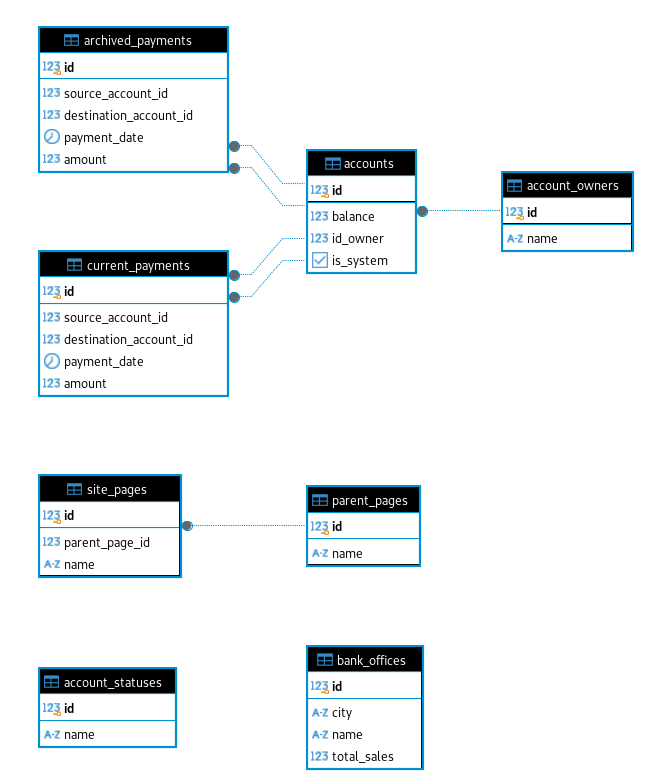
Выбранный code style: <https://www.sqlstyle.guide/>

Следуя code style Simon Holywell, который акцентирует внимание на таких принципах как:

1. Четкость и согласованность именования.
2. Использование ключевых слов в верхнем регистре.
3. Правильное форматирование отступов для улучшения читаемости.

Изначальный SQL-скрипт и новый sql-скрипт по code style:  
<https://github.com/bwl-andromeda/db>

ER-диаграмма построенная с помощью DBeaver:



**Вывод по работе:**

1. Оптимизация структуры базы данных:

- Все таблицы были пересмотрены, добавлены ключевые ограничения (NOT NULL, CHECK и каскадные удаления).

- Добавлены индексы для ускорения операций JOIN и фильтрации по ключевым колонкам (например, id\_owner, source\_account\_id и destination\_account\_id).

- В таблицах реализованы каскадные удаления (ON DELETE CASCADE), что облегчает управление зависимостями между данными, особенно при удалении связанных записей.

2. Улучшение SQL-запросов:

- Оптимизированы запросы для увеличения производительности за счёт использования индексов и замены LEFT JOIN на JOIN там, где это возможно.

- Применены агрегатные функции и индексация для ускорения подсчёта платежей и поиска записей.

- Запросы стали проще, читаемее и быстрее за счёт улучшения структуры базы данных.

3. Создание ER-диаграммы:

- Мы также научились создавать ER-диаграммы (диаграммы сущность-связь), которые позволяют визуализировать структуру базы данных и связи между таблицами.

- Для этого использовался инструмент:

- DBeaver — этот инструмент позволяет автоматически генерировать ER-диаграмму на основе реальной базы данных. Это полезно для анализа существующих баз данных или в процессе их проектирования. Благодаря этому мы получили готовую диаграмму, отражающую все таблицы, связи и типы данных.

4. Улучшение кода:

- Код приведён к стилю Simon Holywell, что улучшило читаемость за счёт использования верхнего регистра для ключевых слов SQL, правильных отступов и форматирования.

- Реализована структурированная вставка данных в таблицы с проверкой корректности данных и целостности связей.

Важные навыки и инструменты, которые мы освоили:

- Оптимизация базы данных для повышения производительности и поддержания целостности данных.

- Генерация ER-диаграммы с помощью инструментов draw.io и DBeaver.

- Следование code style и стандартизация SQL-запросов для повышения читаемости и удобства поддержки кода.