Домашнее задание №1

1. Исходная структура данных

Изначально в базе данных было две таблицы: **customer** и **transaction**, которые содержали ряд избыточных данных и транзитивных зависимостей.

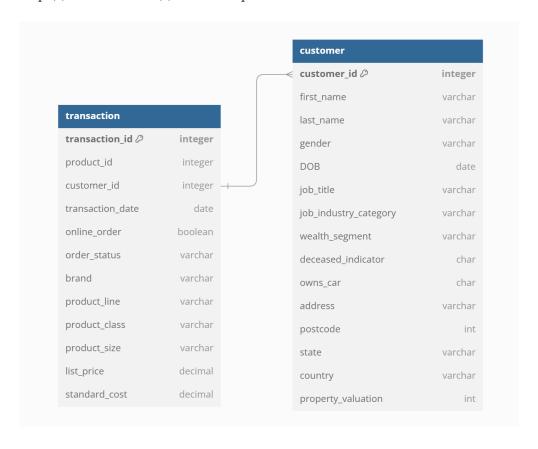


Рис. 1: Исходная структура данных

Отдельно важно отметить, что изначальные данные являлись **неконсистентными**. В рамках данного учебного задания не предпринималось мер по согласованию данных: несогласующиеся данные были просто удалены из базы. В общем случае такой подход, разумеется, некорректен, однако согласование данных на данном этапе потребовало бы приложение усилий, несоразмерных остальному заданию.

2. Почему исходная структура не находилась в нормальной форме?

Нарушение первой нормальной формы (1НФ)

• Повторяющиеся группы данных: Aтрибуты brand, product_line, product_class, product_size дублировались в каждой записи транзакции для одного и того же продукта. Эти данные логически принадлежат таблице продуктов (product).

Нарушение второй нормальной формы (2НФ)

• В таблице transaction атрибуты list_price и standard_cost зависят только от product_id, а не от всего составного ключа transaction_id. Это частичная зависимость.

Нарушение третьей нормальной формы (3НФ)

• В таблице customer атрибуты state и country зависят от postcode, а не от customer_id. Это транзитивная зависимость.

3. Этапы нормализации

Шаг 1: Приведение к первой нормальной форме (1НФ)

Преобразование:

Удалены повторяющиеся группы данных: создана отдельная таблица product для хранения информации о продуктах (brand, product_line, product_class, product_size).

Результат:

Все данные стали атомарными, без повторяющихся групп.

Шаг 2: Приведение ко второй нормальной форме (2НФ)

Преобразование:

Устранили частичные зависимости: поля list_price и standard_cost перенесены в таблицу product, так как они зависят только от product_id.

Результат:

Каждый неключевой атрибут теперь зависит от полного первичного ключа.

Шаг 3: Приведение к третьей нормальной форме (3НФ)

Преобразование:

Устранили транзитивные зависимости: создана отдельная таблица location, содержащая уникальные комбинации postcode, state и country. Из таблицы customer удалены state и country, связь с таблицей location теперь осуществляется через postcode.

Результат:

Каждый неключевой атрибут зависит только от первичного ключа своей таблицы, и транзитивные зависимости устранены.

4. Итоговая структура базы данных

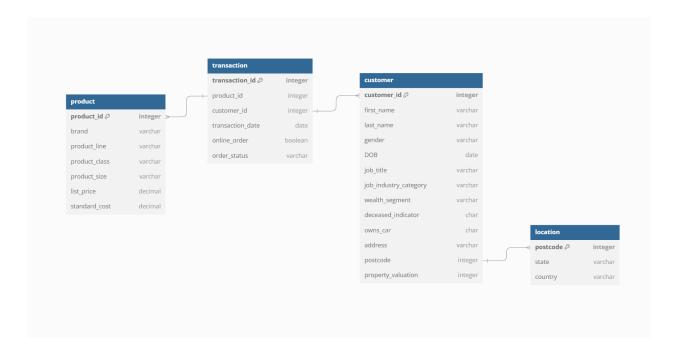


Рис. 2: Итоговая структура базы данных

После нормализации структура базы данных состоит из четырёх таблиц:

- 1. customer хранит информацию о клиентах и их почтовых индексах.
- 2. transaction содержит данные о транзакциях, ссылаясь на customer и product.

- 3. product содержит информацию о продуктах, включая их характеристики, цену и себестоимость.
- 4. location содержит уникальные данные о почтовых индексах, регионах и странах.

Почему текущая структура соответствует 3НФ?

- Все таблицы находятся в первой нормальной форме $(1 \text{H}\Phi)$ данные атомарны, без повторяющихся групп.
- Все таблицы находятся во второй нормальной форме $(2H\Phi)$ каждый неключевой атрибут зависит от полного первичного ключа.
- Все таблицы находятся в третьей нормальной форме (3HФ) транзитивные зависимости устранены, данные логически разделены между таблицами.

5. Скриншоты из DBeaver, подтверждающие загрузку данных

Название	ID объекта	Владелец	Таблично	е пространство	Примерное число строк
≡ customer	16 452	<u>postgres</u>			4 000
Ⅲ location	16 447	<u>postgres</u>			874
≡ product	16 464	<u>postgres</u>			101
⊞ transaction	16 469	<u>postgres</u>			19 997

Рис. 3: Список таблиц в базе данных

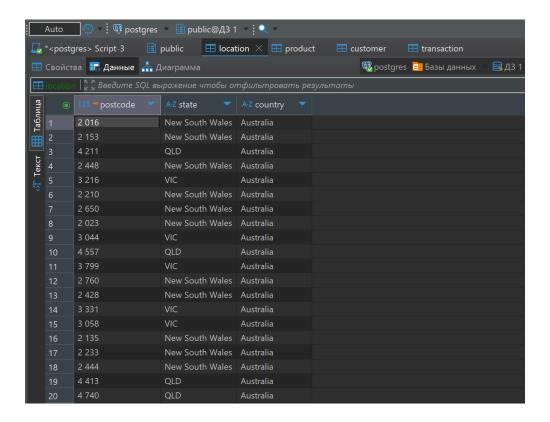


Рис. 4: Фрагмент таблицы location

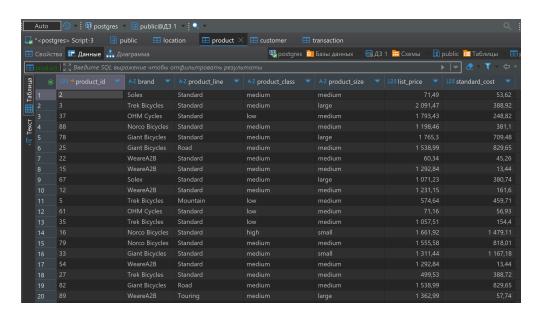


Рис. 5: Фрагмент таблицы product

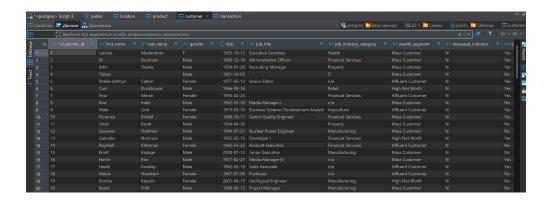


Рис. 6: Фрагмент таблицы customer

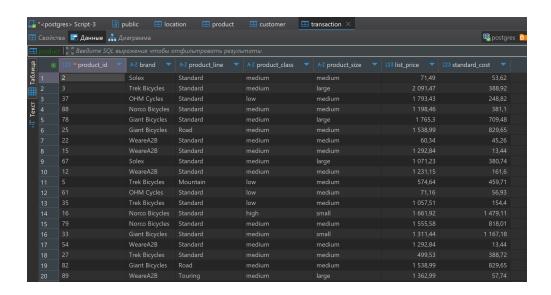


Рис. 7: Фрагмент таблицы transition

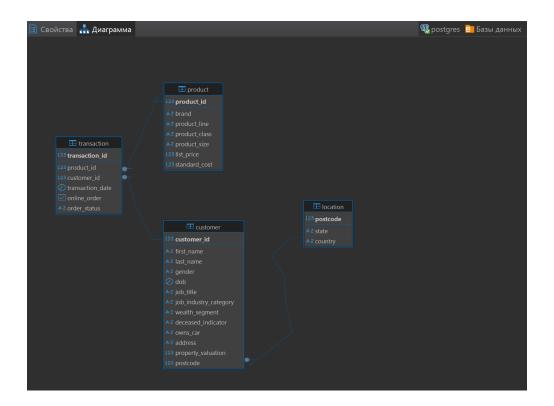


Рис. 8: Диаграмма базы данных