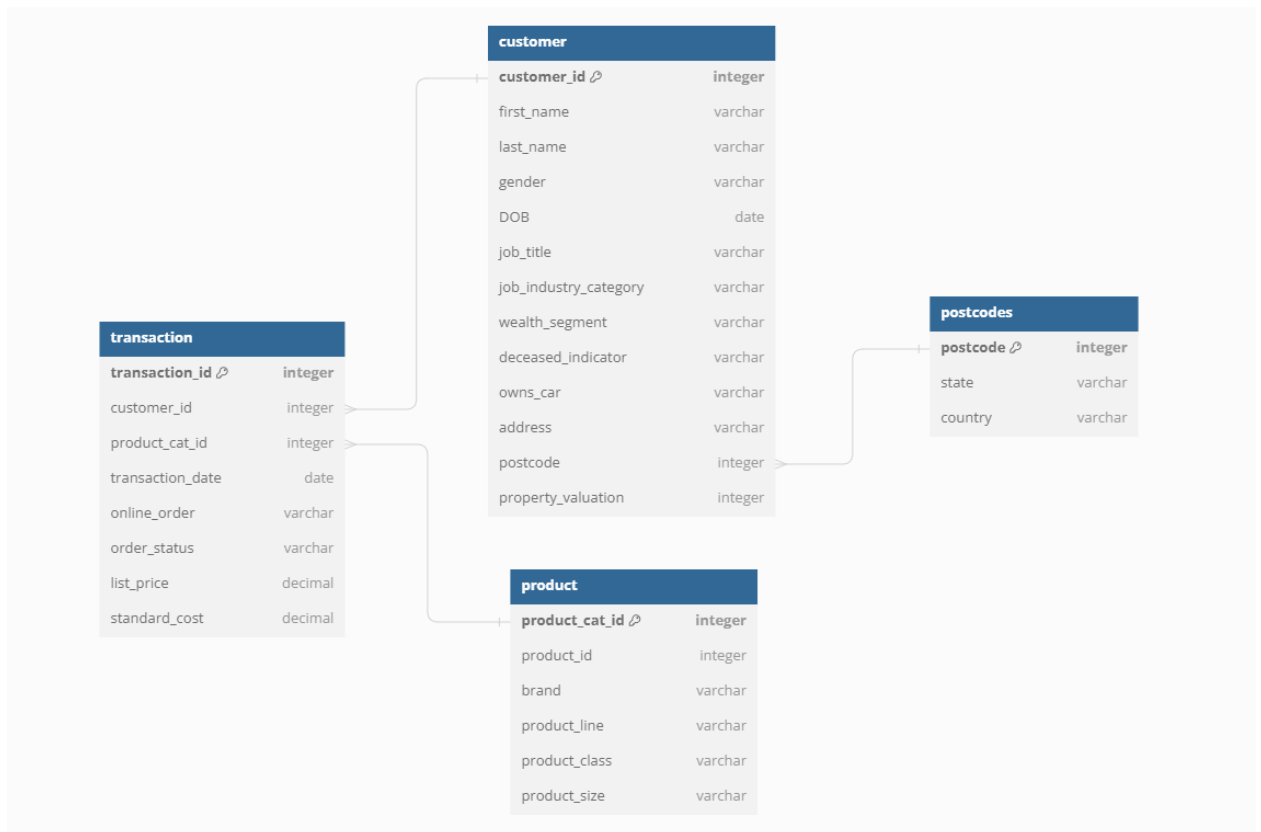


## 1. Скриншот из dbdiagram.io

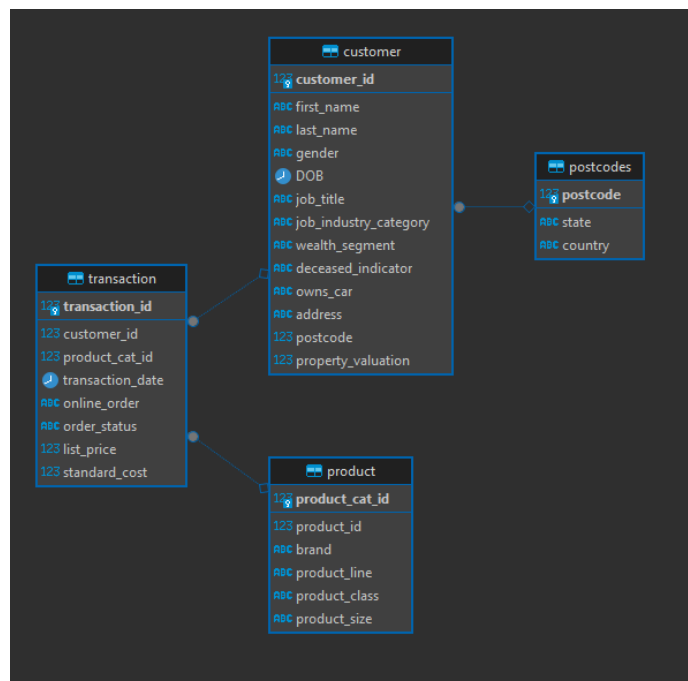


## 2. Нормализация

1. Для начала в таблице **transaction** определяем первичный ключ – столбец `transaction_id`.
2. Приводим таблицу **transaction** к 3NF, поскольку в ней присутствуют атрибуты продукта, которые не зависят напрямую от первичного ключа (`transaction_id`), в частности это поля: `brand`, `product_line`, `product_class`, `product_size`, `product_id`.
3. Поскольку `product_id` не уникален и работает в связке из 4 полей, выносим все эти поля в отдельную таблицу **product**, в которой создаем составной первичный ключ, идентифицирующий каждую уникальную комбинацию всех пяти полей. Данный ключ именуем `product_cat_id`.
4. В таблице **transaction** оставляем только внешний ключ `product_cat_id`, который ссылается на созданную в п.3 таблицу **product**.
5. Таким образом таблица **transaction** приведена к 3NF (существует первичный ключ, все неключевые столбцы зависят только от первичного ключа `transaction_id`, значения в столбцах атомарны), а также декомпозирована на саму себя и новую таблицу **product**.
6. Далее приводим таблицу **customer** к 3NF, в ней присутствуют поля, которая не зависят напрямую от `id` клиента, в частности `state` и `country`.
7. Сначала добавляем на поле `customer_id` первичный ключ (уникальный идентификатор клиента).

8. Добавляем еще один внешний ключ `customer_id` в первой таблице **transaction**, который ссылается на таблицу **customer**.
9. Поскольку каждый `postcode` в таблице **customer** имеет свой штат и страну, выносим эти поля в отдельную таблицу **postcodes**, в которой выбираем в качестве первичного ключа поле `postcode`.
10. В таблице **customer** создаем внешний ключ на поле `postcode`, который ссылается на таблицу **postcodes**.
11. Так, таблица **customer** была приведена к 3NF (существует первичный ключ, все неключевые столбцы зависят только от первичного ключа `customer_id`, значения в столбцах атомарны), а также декомпозирована на самую себя и новую таблицу **postcodes**.

### 3. Скрин из DBeaver с моделью данных



### 4. Загрузка данных в таблицы с помощью панели DBeaver

#### transaction:

transaction								
	123 transaction_id	123 customer_id	123 product_cat_id	transaction_date	ABC online_order	ABC order_status		
1	2 463	3 500	0	2017-04-25	False	Approved		
2	6 309	3 500	106	2017-01-09	True	Approved		
3	8 416	3 500	110	2017-02-16	False	Approved		
4	8 562	3 500	16	2017-08-08	False	Approved		
5	14 870	3 500	6	2017-01-11	False	Approved		
6	19 836	3 500	124	2017-03-14	True	Approved		
7	747	3 499	38	2017-06-21	False	Approved		
8	2 433	3 499	30	2017-02-25	True	Approved		
9	2 794	3 499	66	2017-01-12	False	Approved		
10	3 902	3 499	25	2017-06-22	True	Approved		
11	7 386	3 499	117	2017-03-01	True	Approved		
12	12 434	3 499	77	2017-04-25	False	Approved		

**product:**

postgres> Script public postgres> Script-1 postcodes customer product transaction

Свойства Данные Диаграмма postgres Базы данных

product Введите SQL выражение чтобы отфильтровать результаты

	123 product_cat_id	123 product_id	ABC brand	ABC product_line	ABC product_class	ABC product_size	
1	0	2	Solex	Standard	medium	medium	
2	1	3	Trek Bicycles	Standard	medium	large	
3	2	37	OHM Cycles	Standard	low	medium	
4	3	88	Norco Bicycles	Standard	medium	medium	
5	4	78	Giant Bicycles	Standard	medium	large	
6	5	25	Giant Bicycles	Road	medium	medium	
7	6	22	WeareA2B	Standard	medium	medium	
8	7	15	WeareA2B	Standard	medium	medium	
9	8	67	Solex	Standard	medium	large	
10	9	12	WeareA2B	Standard	medium	medium	
11	10	5	Trek Bicycles	Mountain	low	medium	
12	11	61	OHM Cycles	Standard	low	medium	

**customer:**

postgres> Script public postgres> Script-1 postcodes customer x product transaction

Свойства Данные Диаграмма postgres Базы данных

**customer** Введите SQL выражение чтобы отфильтровать результаты

	123 customer_id ▼	ABC first_name ▼	ABC last_name ▼	ABC gender ▼	DOB ▼	ABC job_title ▼	ABC job_indt
1	1	Laraine	Medendorp	F	1953-10-12	Executive Secretary	Health
2	2	Eli	Bockman	Male	1980-12-16	Administrative Officer	Financial Se
3	3	Arlin	Dearle	Male	1954-01-20	Recruiting Manager	Property
4	4	Talbot		Male	1961-10-03		IT
5	5	Sheila-kathryn	Calton	Female	1977-05-13	Senior Editor	n/a
6	6	Curr	Duckhouse	Male	1966-09-16		Retail
7	7	Fina	Merali	Female	1976-02-23		Financial Se
8	8	Rod	Inder	Male	1962-03-30	Media Manager I	n/a
9	9	Mala	Lind	Female	1973-03-10	Business Systems Development Analyst	Argiculture
10	10	Fiorenze	Birdall	Female	1988-10-11	Senior Quality Engineer	Financial Se
11	11	Uriah	Bisatt	Male	1954-04-30		Property
12	12	Sawyer	Flattman	Male	1994-07-21	Nuclear Power Engineer	Manufactur

**postcodes:**

The screenshot shows the PostgreSQL GUI with the 'postcodes' table selected. The table has 4 columns: 'id', 'postcode', 'state', and 'country'. The data is as follows:

id	postcode	state	country
1	2 000	NSW	Australia
2	2 007	NSW	Australia
3	2 008	NSW	Australia
4	2 009	NSW	Australia
5	2 010	NSW	Australia
6	2 011	NSW	Australia
7	2 015	NSW	Australia
8	2 016	NSW	Australia
9	2 017	NSW	Australia
10	2 018	NSW	Australia
11	2 019	NSW	Australia
12	2 020	NSW	Australia