

LTV В SQL

18 задач про клиентов и прибыль

```
SELECT client_id,  
       SUM(revenue - cost) AS ltv  
FROM orders  
GROUP BY client_id  
ORDER BY ltv DESC;
```



клиенты · repeat rate · CAC

выручка · прибыль · сегменты

Глеб Зайцев
LTV в SQL. 18 задач про
клиентов и прибыль

<https://litres.ru/74419923>

SelfPub; 2026

Аннотация

Практическая книга по расчёту LTV в SQL для аналитиков и предпринимателей. Внутри 18 задач на маленьких таблицах: выручка клиента, первая и вторая покупка, repeat rate, LTV по прибыли, каналы, сегменты, САС и компактный клиентский отчёт. Все примеры запускаются в SQLite.

Содержание

Перед первой задачей	4
Почему LTV нельзя считать одной строкой на заказ	4
Как работать с примерами	5
Маршрут по задачам	6
Практические задачи	7
Задача 1. Посчитать выручку клиента за период	7
Задача 2. Найти первую покупку клиента	10
Задача 3. Посчитать число заказов клиента	13
Задача 4. Посчитать средний чек клиента	16
Конец ознакомительного фрагмента.	17

Глеб Зайцев

LTV в SQL. 18 задач про

клиентов и прибыль

Перед первой задачей

Почему LTV нельзя считать

одной строкой на заказ

LTV живёт на уровне клиента, а не отдельной покупки.

Повторные заказы, клиенты без покупок, канал привлечения и прибыль после себестоимости могут радикально изменить вывод.

Эта книга показывает практические запросы для начальной LTV-аналитики без обещаний точного прогноза будущего.

Как работать с примерами

Сначала сворачивайте данные на уровень клиента, затем сравнивайте клиентов по каналам, сегментам или когортам.

Следите за знаменателем: клиенты без заказов часто должны оставаться в расчёте.

Все данные встроены в задачи и проверяются в SQLite.

Маршрут по задачам

1. Посчитать выручку клиента за период
2. Найти первую покупку клиента
3. Посчитать число заказов клиента
4. Посчитать средний чек клиента
5. Найти повторных покупателей
6. Сложить валовую прибыль клиента
7. Посчитать LTV по каналу привлечения
8. Учесть клиентов без заказов
9. Найти возраст клиента в днях до покупки
10. Разложить LTV по месяцу первой покупки
11. Выделить топ клиентов по LTV
12. Посчитать долю повторных клиентов
13. Найти вторую покупку клиента
14. Посчитать дни между первой и второй покупкой
15. Сравнить LTV и САС
16. Собрать LTV по сегментам
17. Найти клиентов без повторной покупки
18. Собрать компактный LTV-отчёт

Практические задачи

Задача 1. Посчитать выручку клиента за период

Рабочий вопрос

Нужно сложить все покупки клиента, чтобы увидеть базовую LTV-метрику без лишней магии.

Данные

```
CREATE TABLE orders(client_id INTEGER,  
revenue INTEGER);  
INSERT INTO orders VALUES (10,300),  
(10,500), (20,900), (30,200);
```

Запрос

```
SELECT client_id, SUM(revenue) AS  
customer_revenue
```

```
FROM orders
GROUP BY client_id
ORDER BY customer_revenue DESC,
client_id;
```

Ожидаемый результат

```
[[20, 900], [10, 800], [30, 200]]
```

Почему работает

SUM(revenue) по client_id складывает повторные покупки одного клиента.

Запрос сделан на маленькой синтетической таблице, поэтому ответ можно проверить глазами и повторить в SQLite.

Явный ORDER BY или отдельная агрегация фиксируют смысл результата, а не оставляют его на случайный порядок строк.

Где легко ошибиться

Средний чек и LTV отвечают на разные вопросы: один заказ против всех заказов клиента.

Самостоятельная проверка

Кто лидер по выручке? Объясните ответ по исходным строкам и условию запроса.

Ответ: Клиент 20 с выручкой 900. Это следует из исходных строк и условия запроса.

Задача 2. Найти первую покупку клиента

Рабочий вопрос

Нужно определить дату первой покупки, чтобы строить когорты и возраст клиента.

Данные

```
CREATE TABLE orders(client_id INTEGER,  
paid_at TEXT);  
INSERT INTO orders VALUES  
(10, '2026-09-03'), (10, '2026-09-01'),  
(20, '2026-09-02');
```

Запрос

```
SELECT client_id, MIN(paid_at) AS  
first_paid_at  
FROM orders  
GROUP BY client_id  
ORDER BY client_id;
```

Ожидаемый результат

```
[[10, "2026-09-01"], [20, "2026-09-02"]]
```

Почему работает

MIN(paid_at) на формате YYYY-MM-DD даёт самую раннюю дату клиента.

Запрос сделан на маленькой синтетической таблице, поэтому ответ можно проверить глазами и повторить в SQLite.

Явный ORDER BY или отдельная агрегация фиксируют смысл результата, а не оставляют его на случайный порядок строк.

Где легко ошибиться

Если дата хранится в другом текстовом формате, простая сортировка строк может сломаться.

Самостоятельная проверка

Какая первая дата у клиента 10? Объясните ответ по исходным строкам и условию запроса.

Ответ: 2026-09-01. Это следует из исходных строк и усло-

вия запроса.

Задача 3. Посчитать число заказов клиента

Рабочий вопрос

Нужно отделить клиента с одной покупкой от клиента с повторными заказами.

Данные

```
CREATE TABLE orders(client_id INTEGER,  
order_id INTEGER);  
INSERT INTO orders VALUES (10,1), (10,2),  
(20,3), (30,4), (30,5);
```

Запрос

```
SELECT      client_id,      COUNT(*)      AS  
orders_count  
FROM orders  
GROUP BY client_id  
ORDER BY client_id;
```

Ожидаемый результат

```
[[10, 2], [20, 1], [30, 2]]
```

Почему работает

COUNT(*) после группировки показывает количество заказов каждого клиента.

Запрос сделан на маленькой синтетической таблице, поэтому ответ можно проверить глазами и повторить в SQLite.

Явный ORDER BY или отдельная агрегация фиксируют смысл результата, а не оставляют его на случайный порядок строк.

Где легко ошибиться

DISTINCT client_id дал бы только список клиентов и потерял бы частоту покупок.

Самостоятельная проверка

Кто сделал два заказа? Объясните ответ по исходным строкам и условию запроса.

Ответ: Клиенты 10 и 30. Это следует из исходных строк

и условия запроса.

Задача 4. Посчитать средний чек клиента

Рабочий вопрос

Нужно увидеть среднюю сумму заказа по каждому клиенту, а не только общую выручку.

Данные

```
CREATE TABLE orders(client_id INTEGER,  
revenue INTEGER);
```

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.