

**\*Важно! Ваше тестовое задания не будет использовано для коммерческих целей**

*Вступительная информация для Задания 1:*

Приложение - мобильная утилита для сканирования документов. Модель монетизации подписочная, есть пробный период 7 дней с дальнейшим переходом в оплату 4.99 USD в неделю. По ссылке ниже выгрузка с базы данных по оформлениям подписок и оплат. Каждая строка представляет собой отдельное событие (либо оформление пробной подписки, либо оплата после завершения пробного периода). Задание построено таким образом, чтобы проверить понимание принципов unit-экономики предприятия. Задание 1 предпочтительней делать с помощью Python.

**Задание 1**

1. Необходимо рассчитать текущий LTV юзера, используя когортный анализ (cohorting event - оформление пробного периода, когорта представляет собой кол-во возможных операций).
2. Спрогнозировать, каким будет LTV на полгода.
3. Построить график, который будет отображать кривую фактического LTV на фоне кривой прогнозируемого LTV.
4. Рассчитать ROMI на 4 недели и на полгода, если стоимость привлечения пользователя, который оформил подписку - 6 USD (ROMI нужно брать операционный, а не бухгалтерский, цель: узнать как окупятся наши инвестиции).

Ссылка на файл:

[https://drive.google.com/file/d/1L8843A80r50\\_e-hU79XV\\_ssAfzgQVUDP/view?usp=sharing](https://drive.google.com/file/d/1L8843A80r50_e-hU79XV_ssAfzgQVUDP/view?usp=sharing)

**Задание 2**

Ниже 2 задачи на проверку знаний SQL.

Для решения заданий следует использовать синтаксис - PostgreSQL. Ответы нужно представить в виде файла с запросами.

### Схема данных = структура Базы данных

- Это название таблицы
- Это название столбца/колонки

1. Написать 2 SQL запроса для поиска значений среднего и медианы по сумме продажи. Сумму транзакции округляем до целой части. Нельзя использовать стандартные функции среднего и медианы в SQL. Можно использовать только агрегатные функции SUM и COUNT.

Схема данных:

- orders
- id
- sale\_amount - в центах
- user\_id
- datetime

2. Написать SQL-запрос для поиска задублированных в результате ошибки транзакций.

Схема данных:

- purchases
- transaction\_id
- datetime
- amount
- user\_id

3. Написать SQL-запрос для построения воронки перехода из установки в оформление пробного периода и в покупку платной версии приложения в разрезе стран. На одного юзера возможна только одно оформление пробного периода и одна покупка платной версии. Покупка возможна только после истечения срока пробного периода. На выходе должна получиться таблица с колонками “country”, “installs”, “trials”, “purchases”, “conversion\_rate\_to\_trial”, “conversion\_rate\_to\_purchase”

Схема данных:

- events
- transaction\_id
- datetime
- event\_type (значение может быть либо “instal”, либо “trial”, либо “purchase”)
- user\_id
- country

### **Задание 3**

Продуктовый менеджер сформировал гипотезу, что новая версия экрана продаж будет лучше перформировать по сравнению со старой, в связи с тем, что на новом экране продаж более понятно описаны преимущества платной версии приложения.

Продуктовый менеджер ожидает увеличение уровня конверсии из установки в покупку платной версии приложения. Приложение тем временем уже очень активно продвигается на рынке, каждый день привлекается в среднем по 30 000 новых пользователей, текущий уровень конверсии из установки в покупку 5%.

Как бы вы предложили проверить данную гипотезу, какие инструменты и тесты бы использовали в процессе проверки? Написать алгоритм исчерпывающий и понятный для аналитика, и для менеджера (не обязательно 2 в 1).