**Основное задание курсовой**

1. Наша задача — смоделировать изменение балансов студентов. Баланс — это количество уроков, которое есть у каждого студента.
2. Также мы хотим создать таблицу, где будут балансы **каждого студента** за каждый день.
3. Чтобы проверить, всё ли в порядке с нашими данными, составить список гипотез и вопросов, нам важно понимать:
   * сколько всего уроков было на балансе **всех учеников** за каждый календарный день;
   * как это количество менялось под влиянием транзакций (оплат, начислений, корректирующих списаний) и уроков (списаний с баланса по мере прохождения уроков).

**Дополнительное задание**

**Задание 1:**

* Посмотрите на изменения балансов студентов (на примере топ-1000 строк), собранных из CTE.
* Какие данные вас смущают? Какие вопросы стоит задавать дата-инженерам и владельцам таблиц?

**Задание 2:**

* Создайте визуализацию (линейную диаграмму) итогового результата.
* Какие выводы можно сделать из получившейся визуализации?

**Образ результата**

* В результате должен получиться запрос, который собирает данные о балансах студентов за каждый прожитый ими день.
* Вопросы дата-инженерам и владельцам таблиц.
* Визуализация (линейная диаграмма) итогового результата.

**План решения задачи**

1. Посмотреть какие данные у нас есть и что они значат
2. Выбрать те столбцы, которые понадобиться
3. Задать интересующие вопросы.
4. Продумать образ результата, как должна выглядеть итоговая таблица.
5. Написать план, как добиться такой таблицы.
6. И написать какие инструменты необходимы для решения задачи
7. Проанализировать ответ на интересующие вопросы.
8. Выполнить всё в sql.
9. Дополнительные задания.

**Решение задачи**

**Шаг 1,2.** Посмотреть какие данные у нас есть и что они значат. Выбрать те столбцы, которые понадобиться

**Введенные наименования таблиц и столбцов**

Таблицы в CTE:

* whitewash\_balance - таблица с начислениями на баланс
* write\_off\_balance – таблица со списаниями с баланса
* balance\_all – таблица, состоящая из двух предыдущих

Строки:

* whitewash\_classes – количество уроков начисленных на баланс
* write\_off\_classes – количество уроков списанных с баланса
* balance – баланс ученика в конкретный день.

# SKYENG\_DB

### Classes ***Витрина с уроками***

* **user\_id** — уникальный идентификатор юзера.
* **id\_class** — уникальный идентификатор урока.
* **class\_start\_datetime** — время начала урока.
* **class\_end\_datetime** — время конца урока.
* **class\_status** — статус урока (успешно проведен / отменен и т. д.).
* **class\_status\_datetime** — время проставления статуса по уроку.

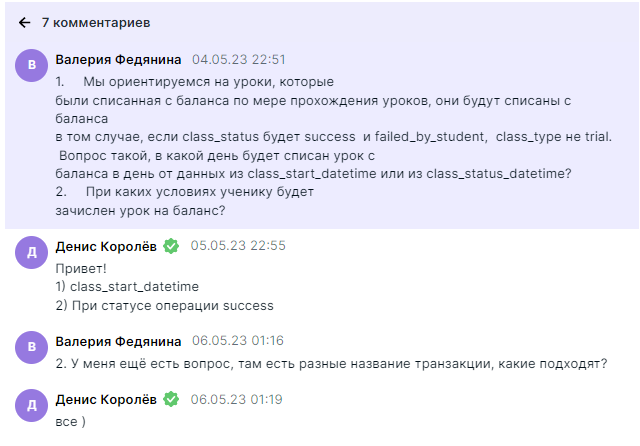
### Payments ***Витрина с платежами по урокам***

* **user\_id** — уникальный идентификатор юзера.
* **operation\_name** — название проведенной операции.
* **status\_name** — статус проведенной операции (исполнена / не исполнена и т. д.).
* **classes** — количество оплаченных уроков.
* **transaction\_datetime** — время проведения операции.

# **Шаг 3,7.** Задать интересующие вопросы.

Вопросы, которые необходимо задать для большего понимания задачи:

1. Мы ориентируемся на уроки, которые были списанная с баланса по мере прохождения уроков, они будут списаны с баланса в том случае, если class\_status будет success и failed\_by\_student, class\_type не trial. Вопрос такой, в какой день будет списан урок с баланса в день от данных из class\_start\_datetime или из class\_status\_datetime?
2. При каких условиях ученику будет зачислен урок на баланс?

Были получены ответы:

# **Шаг 4.** Продумать образ результата, как должна выглядеть итоговая таблица.

|  |  |
| --- | --- |
| **date** | **−** |
| **user\_id** | **balance** |

# **Шаг 5,6,8.** Написать план, как добиться такой таблицы.

Чтобы прийти к результирующей таблице, необходимо понять сколько в конкретную дату, у конкретного юзера на балансе уроков. Предположим, что у нас 1 ученик и 1 день (1.01.2020). Как понять сколько у нас на балансе уроков в этот день?

Баланс у нас меняется в зависимости от того сколько списалось уроков и сколько упало на баланс.

1. Посчитать сколько уроков упало на баланс.
   * Необходимо отфильтровать данные, которые удовлетворяют условиям, удовлетворяющим попадания урока на баланс.
   * Посчитать сколько юзеру упало уроков в конкретный день.
2. Посчитать сколько уроков списалось с баланса.
   * Необходимо отфильтровать данные, которые удовлетворяют условиям, удовлетворяющим списания урока с баланса.
   * Посчитать сколько у юзера списалось уроков в конкретный день.

Теперь у нас есть 2 таблицы, которые необходимо соединить. Я выбрала full join равенством переменных user\_id, date. Получилась таблица:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| whitewash\_balance | | | write\_off\_balance | | |
| date | user\_id | whitewash\_classes | date | user\_id | write\_off\_classes |

Теперь я приведу таблицу в более удобный вид:

|  |  |  |  |
| --- | --- | --- | --- |
| date | user\_id | whitewash\_classes | write\_off\_classes |

С помощью case when для дат, чтобы дата теперь была на каждое изменение в whitewash\_classes и write\_off\_classes

Теперь необходимо считать баланс уроков. Для этого возьмем whitewash\_classes и из него будем вычитать write\_off\_classes, единственное в whitewash\_classes у нас есть пустые ячейки. Сдублировать их с прошлых дней будет не верным, так как каждый следующий день значение whitewash\_classes должно становиться whitewash\_classes- write\_off\_classes за прошлый день.

Подумав ещё над решением этой задачи, я пришла к выводу, что использовать здесь join не нужно, а нужно использовать union – вертикальное соединение запроса, при этом в таблице write\_off\_balance вместо столбца write\_off\_classes я использую − write\_off\_classes, чтобы получить отрицательные числа.

Соединив таблица, я использую оконную функцию sum() с накоплением. Чтобы не было повторяющихся данных в случае, когда у нас есть пополнение и списание баланса в один и тот же день, я использую distinct date.

Получена результирующая таблица, где есть балансы **каждого студента** за каждый день.

<https://metabase.sky.pro/question/67415>

with

whitewash\_balance as

(

select date\_trunc('day',transaction\_datetime) as date

,user\_id

,sum(classes) as whitewash\_classes

from skyeng\_db.payments

where status\_name = 'success'

group by 1,2

),

write\_off\_balance as

(

select date\_trunc('day',class\_start\_datetime) as date

,user\_id

,count(\*) as write\_off\_classes

from skyeng\_db.classes

where class\_status in ('success','failed\_by\_student')

and class\_type !='trial'

group by 1,2

),

balance\_all as

(

select date

, user\_id

, whitewash\_classes as balance

from whitewash\_balance

union all

select date

, user\_id

, -write\_off\_classes as balance

from write\_off\_balance

order by user\_id,date

)

select distinct date

, user\_id

, sum(balance) over (partition by user\_id order by date) as balance

from balance\_all

order by 2,1

limit 1000

Получим ещё одну таблицу, в которой будет указано сколько всего уроков было на балансе **всех учеников** за каждый календарный день, для этого уберем user\_id из select.

<https://metabase.sky.pro/question/67416>

with

whitewash\_balance as

(

select date\_trunc('day',transaction\_datetime) as date

,user\_id

,sum(classes) as whitewash\_classes

from skyeng\_db.payments

where status\_name = 'success'

group by 1,2

),

write\_off\_balance as

(

select date\_trunc('day',class\_start\_datetime) as date

,user\_id

,count(\*) as write\_off\_classes

from skyeng\_db.classes

where class\_status in ('success','failed\_by\_student')

and class\_type !='trial'

group by 1,2

),

balance\_day as

(

select date

, user\_id

, whitewash\_classes as balance

from whitewash\_balance

union all

select date

, user\_id

, -write\_off\_classes as balance

from write\_off\_balance

order by user\_id,date

)

select distinct date

, sum(balance) over (partition by user\_id order by date) as balance

from balance\_day

order by 1

Таблица, с топ 30 студентов, у которых суммарный баланс уроков самый низкий.

<https://metabase.sky.pro/question/67435>

with

whitewash\_balance as

( select date\_trunc('day',transaction\_datetime) as date

,user\_id

,sum(classes) as whitewash\_classes

from skyeng\_db.payments

where status\_name = 'success'

group by 1,2

),

write\_off\_balance as

(

select date\_trunc('day',class\_start\_datetime) as date

,user\_id

,count(\*) as write\_off\_classes

from skyeng\_db.classes

where class\_status in ('success','failed\_by\_student')

and class\_type !='trial'

group by 1,2

),

balance\_all as

(

select date

, user\_id

, whitewash\_classes as balance

from whitewash\_balance

union all

select date

, user\_id

, -write\_off\_classes as balance

from write\_off\_balance

order by user\_id,date

)

select distinct user\_id

,sum(balance) as balance

from

(

select distinct date

, user\_id

, sum(balance) over (partition by user\_id order by date) as balance

from balance\_all

order by 2,1

) t

where balance <0

group by 1

having sum(balance) < 0

order by 2

limit 30

**Шаг 9.** Дополнительные задания**.**

**Задание 1:**

* Посмотрите на изменения балансов студентов (на примере топ-1000 строк), собранных из CTE.
* Какие данные вас смущают? Какие вопросы стоит задавать дата-инженерам и владельцам таблиц?

Посмотрев на балансы студентов, можно заметить, студентов могут уходить в минус, то есть ученик, посмотрел урок, при это ещё его не оплатил. Такие ученики, могут вообще не оплатить и тогда принести убыток для компании.

Вопросы для дата-инженерам и владельцам таблиц:

* Может ли быть такое, что не все транзакции по оплате выведены на ветрину payments и из-за этого балансы неверно отображаются?
* Какие существуют ограничение, которые не дают возможность проходить уроки, не имея уроков на балансе?

Для того, чтобы понять, насколько это распространено. Построив топ с самым большим количеством студентов, у которых суммарный баланс уроков самый низкий.

Стало понятно, что очень большое количество учеников, имеют отрицательный суммарный баланс. Так же видно, что это очень большие суммы, что говорит об серьезности проблемы.

**Задание 2:**

* Создайте визуализацию (линейную диаграмму) итогового результата.
* Какие выводы можно сделать из получившейся визуализации?

По диаграммам я вижу, что студенты могут уходить в минус, и это влияет на то, какое количество уроков на балансе всех учеников. Основываясь на эти данные, можно сделать вывод, что из-за того, что у студентов есть возможность уходить в минусовой баланс по урокам и это влияет на общий баланс, это может привести к недополученным деньгам, а в последствии к банкротству.