-- Шаг 1

with first\_payments as (

select user\_id

, date\_trunc('day', min(transaction\_datetime)) as first\_payment\_date

from skyeng\_db.payments

where status\_name = 'success'

group by user\_id

),

-- Шаг 2

all\_dates as (

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

from skyeng\_db.classes

where class\_start\_datetime between '2016-01-01' and '2017-01-01'

),

-- Шаг 3

all\_dates\_by\_user as (

select user\_id

, dt

from first\_payments

join all\_dates

on dt >= first\_payment\_date

),

-- Шаг 4

payments\_by\_dates as (

select user\_id

, date\_trunc('day', transaction\_datetime) as payment\_date

, sum(classes) as transaction\_balance\_change

from skyeng\_db.payments

where status\_name = 'success'

group by 1,2

),

-- Шаг 5

payments\_by\_dates\_cumsum as (

select all\_dates\_by\_user.user\_id

, dt

, coalesce(transaction\_balance\_change, 0) as transaction\_balance\_change

, sum(coalesce(transaction\_balance\_change, 0)) over (partition by all\_dates\_by\_user.user\_id order by dt) as transaction\_balance\_change\_cs

from all\_dates\_by\_user

left join payments\_by\_dates

on all\_dates\_by\_user.user\_id = payments\_by\_dates.user\_id

and payment\_date = dt

),

-- Шаг 6

classes\_by\_dates as (

select user\_id

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

, count(user\_id)\*-1 as classes

from skyeng\_db.classes

where class\_status = 'success'

or class\_status = 'failed\_by\_student'

group by 1,2

),

-- Шаг 7

classes\_by\_dates\_dates\_cumsum as (

select all\_dates\_by\_user.user\_id

, dt

, coalesce(classes, 0) as classes

, sum(coalesce(classes, 0)) over (partition by all\_dates\_by\_user.user\_id order by dt) as classes\_cs

from all\_dates\_by\_user

left join classes\_by\_dates

on all\_dates\_by\_user.user\_id = classes\_by\_dates.user\_id

and dt = class\_date

),

-- Шаг 8

balances as (

select payments\_by\_dates\_cumsum.user\_id

, payments\_by\_dates\_cumsum.dt

, transaction\_balance\_change

, transaction\_balance\_change\_cs

, classes

, classes\_cs

, sum(classes\_cs)+sum(transaction\_balance\_change\_cs) as balance

from payments\_by\_dates\_cumsum

join classes\_by\_dates\_dates\_cumsum

on payments\_by\_dates\_cumsum.user\_id = classes\_by\_dates\_dates\_cumsum.user\_id

and payments\_by\_dates\_cumsum.dt = classes\_by\_dates\_dates\_cumsum.dt

group by 1,2,3,4,5,6

)

-- --Задание 1

-- select \*

-- ,row\_number () over (order by user\_id, dt) as rn

-- from balances

-- -- limit 1000

-- --Какие вопросы стоит задавать дата-инженерам и владельцам таблицы payments?

-- --Результат запроса - выборка по 7ми ученикам. На примере этих учеников видим, что они не поддерживают стабильный баланс доступных занятий:

-- --то есть закупают какое-то количество классов, а следующую оплату совершают уже после отработки доступных занятий и выхода баланса в ноль.

-- --У некоторых учеников наблюдается отрицательный баланс (например, 3335256 с 9го ноября) - то есть они не заплатили за пройденные занятия.

-- Шаг 9

select dt

, sum(transaction\_balance\_change) as sum\_tbc

, sum(transaction\_balance\_change\_cs) as sum\_tbcc

, sum(classes) as sum\_c

, sum(classes\_cs) as sum\_cc

, sum(balance) as sum\_b

from balances

group by 1

order by 1

--Задание 2

--На графике 5 кривых, отражающих сумму значений метрики на каждую дату:

--sum\_tbc - оплачено занятий - в целом равномерная тенденция по году, максимальные значения достигаются в ноябре, то есть в этом месяце наиболее активные оплаты

--sum\_tbcc - оплачено занятий нарастающим итогом - активный рост наблюдаем с середины года

--sum\_c - пройдено уроков - в целом наблюдаем рост к концу года, но в сравнении с оплаченными занятиями не такой сильный

--sum\_cc - пройдено уроков нарастающим итогом - активный рост наблюдаем с середины года

--sum\_b - оплачено - пройдено уроков - равномерно растет на протяжении всего года, на конец года имеем положительный баланс, то есть оплачено уроков больше, чем пройдено.