WITH

--первая транзакция для каждого студента

first\_payments as

(

select user\_id

, transaction\_datetime as first\_payment\_date from

(

select user\_id

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

, dense\_rank() over (partition by user\_id order by transaction\_datetime) as d\_r

from skyeng\_db.payments

where status\_name = 'success'

order by user\_id, transaction\_datetime asc) t1

where d\_r=1

),

-----------------------------------------------------------------------

--таблица с датами за каждый календарный день 2016 года

all\_dates as

(

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

from skyeng\_db.classes

where date\_part('year', class\_start\_datetime) = 2016

),

-----------------------------------------------------------------------

all\_dates\_by\_user as

(

select user\_id

, dt

from first\_payments as fp

join all\_dates as ad

on fp.first\_payment\_date <= ad.dt

order by dt

),

-----------------------------------------------------------------------

payments\_by\_dates as

(

select user\_id

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

, sum(classes) as transaction\_balance\_change

from skyeng\_db.payments

where status\_name = 'success'

group by user\_id, payment\_date

),

-----------------------------------------------------------------------

payments\_by\_dates\_cumsum as

(

select alldat.\*

, case when transaction\_balance\_change is null then 0 else transaction\_balance\_change end as transaction\_balance\_change

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

from all\_dates\_by\_user as alldat

left join payments\_by\_dates as paymdat

on alldat.user\_id = paymdat.user\_id

and alldat.dt = paymdat.payment\_date

),

-------------------------------------------------------------------------

classes\_by\_dates as

(

select user\_id

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

, count(id\_class)\*(-1) as classes

from skyeng\_db.classes

where (class\_status = 'success' or class\_status = 'failed\_by\_student')

and (class\_type = 'regular' or class\_type = 'single')

group by user\_id, class\_date

),

----------------------------------------------------------------------------

classes\_by\_dates\_dates\_cumsum as

(

select aldatus.\*

, case when classes is null then 0 else classes end as classes

, sum(coalesce(classes, 0)) over (partition by aldatus.user\_id order by dt) as classes\_cs

from all\_dates\_by\_user as aldatus

left join classes\_by\_dates as class\_dat

on aldatus.dt = class\_dat.class\_date

and aldatus.user\_id = class\_dat.user\_id

),

---------------------------------------------------------------------------------

balances as

(

select \*

, classes\_cs + transaction\_balance\_change\_cs as balance

from

(

select pc.\*

, cc.classes

, cc. classes\_cs

from payments\_by\_dates\_cumsum as pc

join classes\_by\_dates\_dates\_cumsum as cc

on pc.dt = cc.dt

and pc.user\_id = cc.user\_id

order by pc.dt

) t4

)

-- Для этого просуммируем поля transaction\_balance\_change, transaction\_balance\_change\_cs, classes, classes\_cs, balance из CTE balances с группировкой и сортировкой по dt.

select dt

, sum (transaction\_balance\_change) as transaction\_balance\_change\_for\_all

, sum (transaction\_balance\_change\_cs) as transaction\_balance\_change\_cs\_for\_all

, sum (classes) as classes\_for\_all

, sum (classes\_cs) as classes\_cs\_for\_all

, sum (balance) as balance\_for\_all

from balances

group by dt

order by dt

