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, status\_name

order by user\_id),

all\_dates as (select distinct date\_trunc('day', class\_start\_datetime) as dates\_of\_classes, user\_id

from SKYENG\_DB.classes

where class\_start\_datetime between '2016-01-01' and '2016-12-31'

order by dates\_of\_classes),

payments\_by\_dates as (select user\_id, date\_trunc('day',transaction\_datetime) as class\_date, sum(classes) as transaction\_balance\_change

from SKYENG\_DB.payments

where status\_name = 'success'

group by user\_id, class\_date

order by user\_id),

all\_dates\_by\_user as (select f.user\_id, d.dates\_of\_classes

from first\_payments f

join all\_dates d on f.user\_id = d.user\_id

where d.dates\_of\_classes>=f.first\_payment\_date),

classes\_by\_dates as (select user\_id,

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

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

from skyeng\_db.classes

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

and class\_type != 'trial'

group by user\_id, class\_date),

classes\_by\_dates\_dates\_cumsum as (select u.user\_id, k.classes,

sum(coalesce(k.classes,0)) over (partition by u.user\_id order by k.class\_date rows between unbounded preceding and current row) classes\_cs,

k.class\_date,

case when k.classes is not null then k.classes else '0' end

from all\_dates\_by\_user u

join classes\_by\_dates k on (u.user\_id=k.user\_id and u.dates\_of\_classes=k.class\_date) ),

payments\_by\_dates\_cumsum as (select u.user\_id,

u.dates\_of\_classes,

s.transaction\_balance\_change

,sum(s.transaction\_balance\_change) over (partition by u.user\_id order by u.dates\_of\_classes) as transaction\_balance\_change\_cs

from all\_dates\_by\_user u

left join payments\_by\_dates s on (u.user\_id=s.user\_id and u.dates\_of\_classes=s.class\_date)

order by s.class\_date),

balances as (select a.user\_id,

v.class\_date,

a.transaction\_balance\_change,

a.transaction\_balance\_change\_cs,

v.classes,

v.classes\_cs,

v.classes\_cs + a.transaction\_balance\_change\_cs as balance

from payments\_by\_dates\_cumsum a

join classes\_by\_dates\_dates\_cumsum v on (a.user\_id=v.user\_id and a.dates\_of\_classes=v.class\_date))

-- select \*

-- from balances

-- order by user\_id, class\_date

-- limit 1000

select class\_date,

sum(transaction\_balance\_change) sum\_transaction\_balance\_change,

sum(transaction\_balance\_change\_cs) sum\_transaction\_balance\_change\_cs,

sum(classes) sum\_classes,

sum(classes\_cs) sum\_classes\_cs,

sum(balance) sum\_balance

from balances

group by class\_date

order by class\_date