--------------------- ДЗ 8 урок---------------------------------------------

----------------------1 часть. Скользящее среднее выплат по месяцам-----------------

-- 1. Постройте распределение выплаченных клиентами денег (\*\*\*amt\_payment\*\*\*) по месяцам поля \*\*\*date\_purchase\*\*\* (SQL-запрос)

-- 2. Дополните ваш запрос так, чтобы получилось три разных скользящих средних (по сумме денег по месяцам)

-- - \*\*MA(3)\*\* - скользящее среднее, принимающее текущее значение и два предыдущих.

-- - \*\*MA(7)\*\* - скользящее среднее, принимающее текущее значение и шесть предыдущих.

-- - \*\*MA\_2side(5)\*\* - двустороннее скользящее среднее, принимающее текущее значение, два предыдущих и два следующих.

-- - Перенесите результаты в Excel и постройте графики скользящего среднего

-- select month\_purchase

-- , avg(amt\_payment) over()

-- , avg(amt\_payment) over (order by month\_purchase rows between 2 preceding and current row) as ma\_3

-- , avg(amt\_payment) over (order by month\_purchase rows between 6 preceding and current row) as ma\_7

-- , avg(amt\_payment) over (order by month\_purchase rows between 2 preceding and 2 following) as MA\_5\_2side

-- from

-- (

-- select

-- date\_trunc('month',date\_purchase) month\_purchase

-- , sum(amt\_payment) amt\_payment

-- from skycinema.client\_sign\_up

-- group by month\_purchase

-- ) as t

-- group by month\_purchase, amt\_payment

-----------------------Задача 2. Витрина----------------------------------------------

-- Специалисты по машинному обучению попросили вас составить витрину для их модели.

-- Возьмите таблицу \*skybank.late\_collection\_clients и напишите скрипт\*, который сделает витрину со следующими полями:

-- - Внутренний идентификатор клиента [поле \*id\_client\*]

-- - Название города [поле \*name\_city\* из таблицы \*skybank.region\_dict\*]

-- - Числовой признак, который принимает значение 1 для мужчин и 0 для женщин [новое поле \*nflag\_gender\* на основании поля \*gender\*]

-- - Возраст [поле \*age\*]

-- - Числовая переменная, которая показывает, в первый ли раз клиент обратился к нам за кредитом [поле \*first\_time\*]

-- - Числовой признак, который принимает значение 1 при наличии мобильного телефона и 0 при его отсутствии [новое поле \*nflag\_cellphone\* на основании поля \*cellphone\*]

-- - Числовая переменная, которая показывает, активен ли клиент [поле \*is\_active\*]

-- - Номер клиентского сегмента [поле \*cl\_segm\*]

-- - Размер выданного кредита [поле \*amt\_loan\*]

-- - Количество кредитов, выданных в рамках данного типа кредита и данного города

-- - Дата выдачи кредита [поле \*date\_loan\*] (необходимо привести к формату даты)

-- - Тип выданного кредита [поле \*credit\_type\*]

-- - Суммарный объем кредитов выданных в данном городе

-- - Доля данного кредита среди всех кредитов, выданных в данном городе

-- - Суммарный объем кредитов выданных в рамках данного типа кредита

-- - Доля данного кредита среди всех кредитов, выданных в рамках данного типа кредита

-- - Суммарный объем кредитов выданных в рамках данного типа кредита и данного города

-- - Доля данного кредита среди всех кредитов, выданных в рамках данного типа кредита и данного города

-- - Количество кредитов, выданных в данном городе

-- - Количество кредитов, выданных в рамках данного типа кредита

-- select \*

-- from skybank.late\_collection\_clients

select id\_client

, name\_city

, case when gender = 'M' then 1 else 0 end as nflag\_gender

, age

, first\_time

, case when cellphone is not null then 1 else 0 end as cellphone\_flag

, is\_active

, cl\_segm

, amt\_loan

, count(\*) over(partition by name\_city) as cnt\_city

, count(\*) over(partition by credit\_type) as cnt\_crtype

, count(\*) over (partition by credit\_type, name\_city) as cnt\_crtype\_city

, date\_loan::date

, credit\_type

, sum(amt\_loan) over(partition by name\_city) as sum\_city\_loan

, sum(amt\_loan) over(partition by credit\_type) / sum(amt\_loan) over()::float as cr\_type\_share

, sum(amt\_loan) over(partition by credit\_type, name\_city) as sum\_crtype\_city

, sum(amt\_loan) over(partition by credit\_type, name\_city)::float / sum(amt\_loan) over() as cr\_type\_city\_share

from skybank.late\_collection\_clients a

left join skybank.region\_dict b

on a.id\_city = b.id\_city