**Задание.**

Клиенты оставляют заявки на звонок, после чего менеджеры звонят и обсуждают все необходимые вопросы.

Есть 2 таблицы с данными. В таблице tasks лежат заявки от клиентов, в таблице calls лежат звонки от менеджеров клиентам.

📁 Таблица tasks

* client\_id (идентификатор клиента)
* created\_datetime (время клиентской заявки)
* title (тема обращения)

📁 Таблица calls

* manager\_id (идентификатор менеджера)
* client\_id (идентификатор клиента, которому звонят)
* call\_datetime (время менеджерского звонка)

**Задание 2.1. Сколько заявок приходило каждый день в июне 2022 года?**

select date\_trunc(‘day’, created\_datetime) as “Месяц”, count(client\_id) as “Количество заявок”

from tasks

where date\_trunc(‘month’, created\_datetime)=’2022-06-01’

group by date\_trunc(‘day’, created\_datetime)

order by Date\_trunc(‘day’, created\_datetime)

**Задание 2.2. Список тем, для которых обращений было больше 10 в апреле 2022 года.**

select title as “Тема обращения”

from tasks

where date\_trunc(‘month’, created\_datetime)=’2022-04-01’

group by title

having count(\*)>10

**Задание 2.3. Список клиентов, которые оставляли заявку, но ни одного звонка от менеджера по ним не было.**

select a.client\_id

from tasks a

left join calls b

on a.client\_id=b.client\_id

where call\_datetime is null

group by a.client\_id

order by a.client\_id

**Задание 2.4. Для каждого клиента выведите три его последних обращения и постройте распределение количества этих обращений по теме.**

with tasks\_1 as

(select a.title

, b.client\_id

, a.created\_datetime

, row\_number() over (partition by b.client\_id order by a.created\_datetime desc) as cnt\_rank

from tasks a

left join calls b

on a.client\_id=b.client\_id

)

select title

, count(\*) as “Количество обращений”

from tasks\_1

where cnt\_rank <=3

group by title

order by title

**Задание 2.5. Для каждой темы обращения найдите среднее время, которое проходит между клиентскими обращениями.**

with time\_int as

( select title

, (created\_datetime-(lag(created\_datetime) over (partition by title order by created\_datetime)) as interval

from tasks)

select title

, avg(interval) as avg\_time\_interval

from time\_int

where interval is not null

group by title

order by title