# Задача №2

## Контекст

Имеется база данных для недавно созданного загородного клуба. В ней имеется информация о членах этого клуба, объектов для

отдыха, таких как теннисные корты, и истории бронирования. Помимо прочего, клуб хочет понять, как они могут использовать

свою информацию для анализа использования/спроса на объекты. \_\_Обратите внимание\_\_: этот набор данных предназначен

исключительно для интересного набора упражнений, а схема базы данных несовершенна в нескольких аспектах — пожалуйста, не

воспринимайте ее как пример хорошего дизайна.

В БД в схеме `cd` имееются 3 таблицы.

Таблица `cd.members`

postgresql

CREATE TABLE cd.members(

memid INTEGER NOT NULL,

surname CHARACTER VARYING(200) NOT NULL,

firstname CHARACTER VARYING(200) NOT NULL,

address CHARACTER VARYING(300) NOT NULL,

zipcode INTEGER NOT NULL,

telephone CHARACTER VARYING(20) NOT NULL,

recommendedby INTEGER,

joindate TIMESTAMP NOT NULL,

CONSTRAINT members\_pk PRIMARY KEY (memid),

CONSTRAINT fk\_members\_recommendedby FOREIGN KEY (recommendedby)

REFERENCES cd.members(memid) ON DELETE SET NULL

);

У каждого участника есть идентификатор (не обязательно последовательный), основная информация об адресе, ссылка на

участника, который рекомендовал их (если есть), и отметка времени, когда они присоединились.

Таблица `cd.facilities`

postgresql

CREATE TABLE cd.facilities(

facid INTEGER NOT NULL,

name CHARACTER VARYING(100) NOT NULL,

membercost NUMERIC NOT NULL,

guestcost NUMERIC NOT NULL,

initialoutlay NUMERIC NOT NULL,

monthlymaintenance NUMERIC NOT NULL,

CONSTRAINT facilities\_pk PRIMARY KEY (facid)

);

В таблице перечислены все доступные для бронирования объекты, которыми располагает загородный клуб. Клуб хранит

информацию об идентификаторе/имени, стоимости бронирования как членов, так и гостей, первоначальную стоимость строительства объекта и предполагаемые ежемесячные расходы на содержание.

Таблица `cd.bookings`

postgresql

CREATE TABLE cd.bookings(

bookid INTEGER NOT NULL,

facid INTEGER NOT NULL,

memid INTEGER NOT NULL,

starttime TIMESTAMP NOT NULL,

slots INTEGER NOT NULL,

CONSTRAINT bookings\_pk PRIMARY KEY (bookid),

CONSTRAINT fk\_bookings\_facid FOREIGN KEY (facid) REFERENCES cd.facilities(facid),

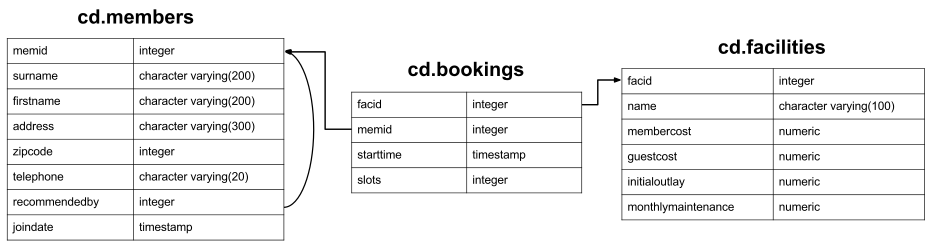
CONSTRAINT fk\_bookings\_memid FOREIGN KEY (memid) REFERENCES cd.members(memid)

);

И таблица, отслеживающая бронирование объектов. В нем хранится идентификатор объекта, член, который сделал бронирование,

начало бронирования и количество получасовых «слотов», на которые было сделано бронирование.

Схематически это выглядит примерно так:



## Постановка

Подсчитайте количество рекомендаций, сделанных каждым участником. Упорядочить по идентификатору участника.

Примечание: Надо выводить только тех участников, которые сделали хотя бы 1 рекомендацию

## Ожидаемый формат ответа

Ваш запрос должен возвращать таблицу формата:

| recommendedby | count |

|---------------|-------|

| 1 | 5 |

| 2 | 3 |

| 3 | 1 |

| ... | ... |