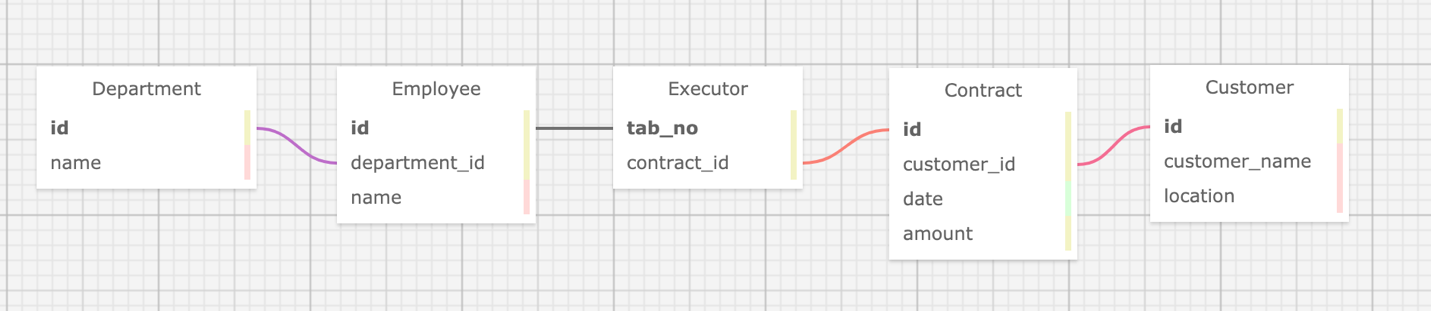
Дана следующая структура данных:



Пояснения:

* Имеется таблица сотрудников **Employees** с уникальным табельным номером, номером отдела и именем
* Таблица отделов **Department** хранит в себе номер и наименование
* Контракты с заказчиками хранятся в таблице Contract в связке с уникальным номером, идентификатором заказчика, датой заключения и стоимостью работ
* Информация о заказчиках хранится в таблице **Customer** с указанием айди заказчика, именем и местоположением
* Сводной является таблица **Executor** с информацией о табельном номере исполнителя и номере контракта

**-- вывод основной информации по таблице**

SELECT d.name AS department, e.name AS employees, ex.tab\_no, cu.customer\_name, cu.location, co.amount AS contract\_amount, co.contract\_date FROM employees e

JOIN department d ON d.id = e.department\_id

JOIN executor ex ON ex.tab\_no = e.id

JOIN contract co ON co.id = ex.contract\_id

JOIN customer cu ON cu.id = co.customer\_id

**-- 1. информация о всех контрактах, связанных с сотрудниками департамента «Logistic». /\*Скрипт с выводом названия отдела, чтоб проверить, что выборку сделали правильно\*/**

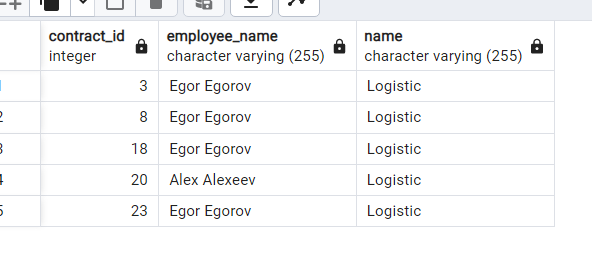
SELECT co.id AS contract\_id, e.name AS employee\_name, d.name FROM employees e

JOIN department d ON d.id = e.department\_id

JOIN executor ex ON ex.tab\_no = e.id

JOIN contract co ON co.id = ex.contract\_id

WHERE d.id = (SELECT id FROM department WHERE name = 'Logistic')



**-- 2. Найти среднюю стоимость контрактов, заключенных сотрудником Ivan Ivanov**

/\*Скрипт с выводом по ФИ и уникальному номеру сотрудника, чтоб проверить, что не суммировался тезка при наличии такого.\*/

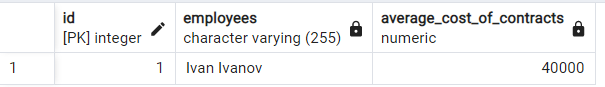
SELECT e.id, e.name AS employees, ROUND(AVG(co.amount),0) AS average\_cost\_of\_contracts FROM employees e

JOIN executor ex ON ex.tab\_no = e.id

JOIN contract co ON co.id = ex.contract\_id

GROUP BY e.id

HAVING e.name='Ivan Ivanov'



**-- 3. Найти самую часто встречающуюся локации среди всех заказчиков. Вывести: location, count**

SELECT location, count(location) FROM customer

GROUP BY location

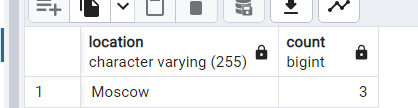
HAVING count(location) = (SELECT count(location) FROM customer

GROUP BY location

ORDER BY count(location)

DESC

LIMIT 1)



Необходимо использовать именно вышеуказанный скрипт, так как если использовать, например:

SELECT location, count(location) FROM customer

GROUP BY location

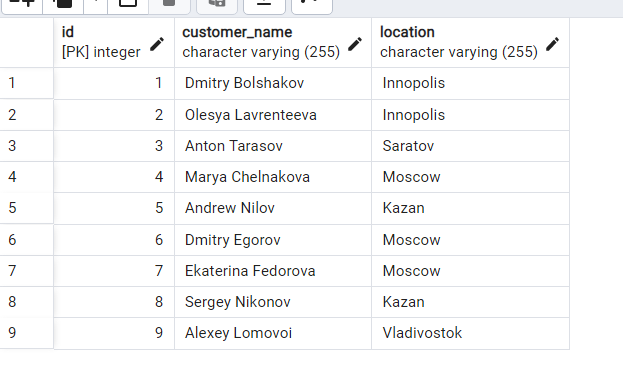
ORDER BY count(location)

DESC

LIMIT 1

То он имел бы право на существование только, если часто встречающаяся локация была бы гарантированно у одного заказчика, а если заказчиков с одинаковой частотой несколько, то он не отработает. В этом можно убедиться, если поменять в таблице данные и сравнять частоту локаций у двух заказчиков

Было:



У нас встречается три раза Москва, сделаем три Иннополиса (изменения проводила в онлайн редакторе, чтобы не нарушить целостность данных в оригинальной базе)

UPDATE customer

SET location = 'Innopolis'

WHERE location= 'Saratov'



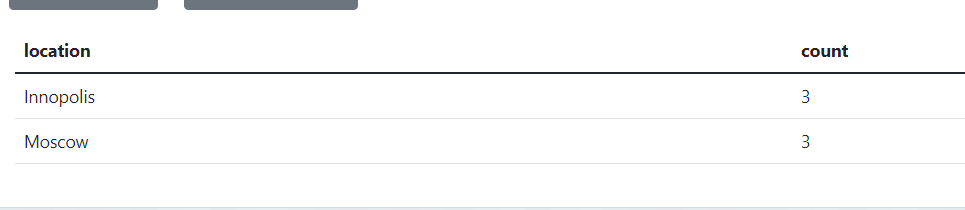


По неверному скрипту выводится





По верному:





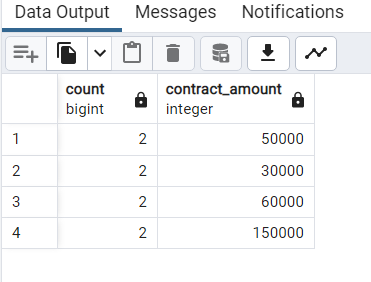
ЧТД 😊

**-- 4. Найти контракты одинаковой стоимости. Вывести count, amount**

SELECT count(amount), amount AS contract\_amount FROM contract

GROUP BY amount

having count(amount)>1



**-- 5. Найти заказчика с наименьшей средней стоимостью контрактов. Вывести customer\_name, среднее значение amount**

SELECT cu.customer\_name, ROUND(AVG(co.amount),0) AS average\_cost\_of\_contracts FROM employees e

JOIN executor ex ON ex.tab\_no = e.id

JOIN contract co ON co.id = ex.contract\_id

JOIN customer cu ON cu.id = co.customer\_id

GROUP BY cu.customer\_name

HAVING AVG(co.amount)=(SELECT AVG(co.amount) FROM employees e

JOIN executor ex ON ex.tab\_no = e.id

JOIN contract co ON co.id = ex.contract\_id

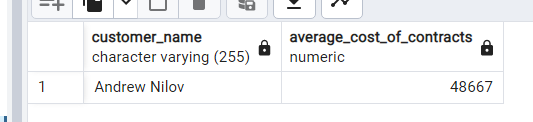
JOIN customer cu ON cu.id = co.customer\_id

GROUP BY cu.customer\_name

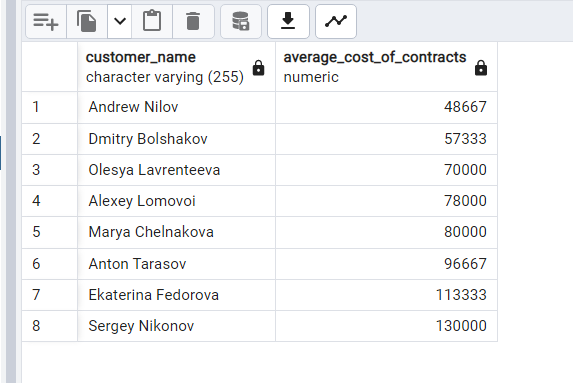
ORDER BY AVG(co.amount)

ASC

LIMIT 1)



Общая таблица заказчиков с указанием средней стоимости контрактов с сортировкой по возрастанию для проверки правильности выведенных данных:



**-- 6. Найти отдел, заключивший контрактов на наибольшую сумму. Вывести: department\_name, sum**

-- так как наибольшая сумма контрактов может быть равна у нескольких отделов, необходимо использовать скрипт с вложенным подзапросом:

SELECT d.name AS department\_name, sum(co.amount) AS sum\_contract\_amount FROM employees e

JOIN department d ON d.id = e.department\_id

JOIN executor ex ON ex.tab\_no = e.id

JOIN contract co ON co.id = ex.contract\_id

GROUP BY d.name

HAVING sum(co.amount)= (SELECT sum(co.amount) FROM employees e

JOIN department d ON d.id = e.department\_id

JOIN executor ex ON ex.tab\_no = e.id

JOIN contract co ON co.id = ex.contract\_id

GROUP BY d.name

ORDER BY sum(co.amount)

DESC

LIMIT 1)

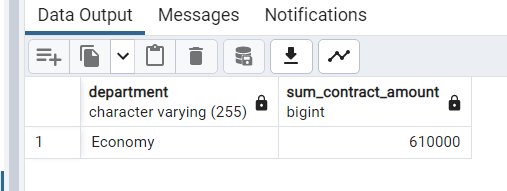
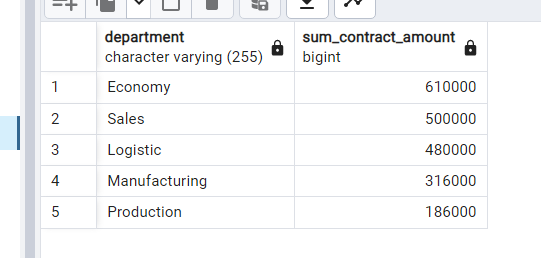


Таблица с суммарной стоимостью контрактов по отделам, для наглядности, что в предыдущем скрипте вывелся нужный:



Теперь рассмотрим пример, демонстрирующий почему именно скрипт с вложенным подзапросом, необходимо использовать.

Добавим два контракта, которые сравняют суммарные стоимости контрактов, отделов Economy и Sales:

INSERT INTO public.contract(

id, customer\_id, contract\_date, amount)

VALUES(26,7,'23.06.15',260000);

INSERT INTO executor(

tab\_no, contract\_id)

VALUES(4,26);

INSERT INTO public.contract(

id, customer\_id, contract\_date, amount)

VALUES(27,3,'23.08.19',150000);

INSERT INTO executor(

tab\_no, contract\_id)

VALUES(5,27);

Получаем в выведенной объединенной таблице две дополнительные строки:

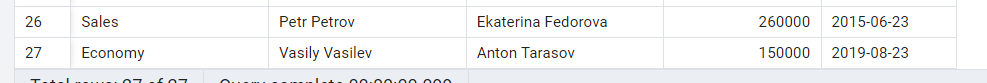
SELECT d.name AS department, e.name AS employees, cu.customer\_name, co.amount AS contract\_amount, co.contract\_date FROM employees e

JOIN department d ON d.id = e.department\_id

JOIN executor ex ON ex.tab\_no = e.id

RIGHT JOIN contract co ON co.id = ex.contract\_id

JOIN customer cu ON cu.id = co.customer\_id



Выводим суммарные стоимости контрактов всех отделов:

SELECT d.name AS department, sum(co.amount) AS sum\_contract\_amount FROM employees e

JOIN department d ON d.id = e.department\_id

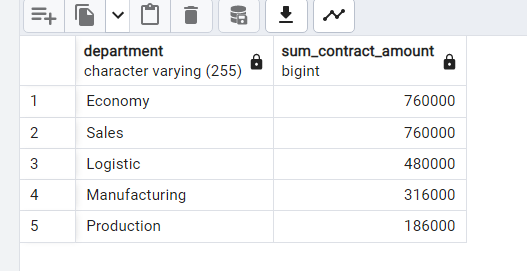
JOIN executor ex ON ex.tab\_no = e.id

JOIN contract co ON co.id = ex.contract\_id

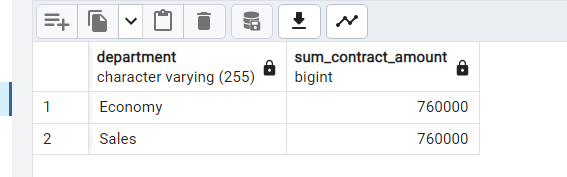
GROUP BY d.name

ORDER BY sum(co.amount)

DESC



Как видим, у двух два отделов равны наибольшие суммы контрактов. Теперь применим наш первоначальный скрипт, чтоб вывести отделы. Получаем:



Из таблицы видно, что вывелись все отделы с наибольшей суммой контрактов. Что и требовалось доказать 😊