**Отчет по Лаб. работе 4. Бабкин Иван КС-33**

1. Сформулировать на русском языке (в виде текста) осмысленные запросы. Для сформулированного запроса написать код на языке SQL. Выполнить запросы в СУБД PostgreSQL, приложить результаты в виде картинки.

1. Придумать запрос на применение агрегирующих оконных функций. **Запрос:**

Рассчитать прибыль относительно даты

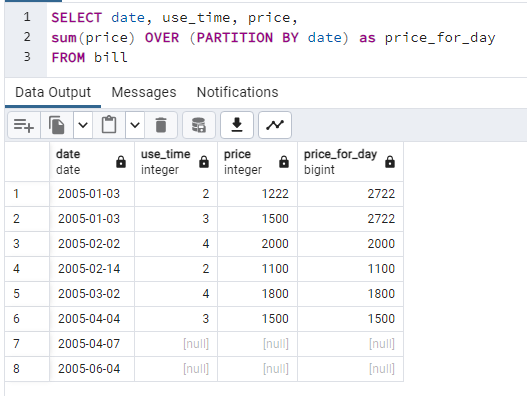
**Код:**

SELECT date, use\_time, price,

sum(price) OVER (PARTITION BY date) as price\_for\_day

FROM bill

**Скриншот выполнения работы программы:**

****

2. Придумать запрос на применение ранжирующих оконных функций.

**Запрос:**

Вывести данные, разбив их на блоки относительно использованного времени, отсортировать каждый блок относительно цены.

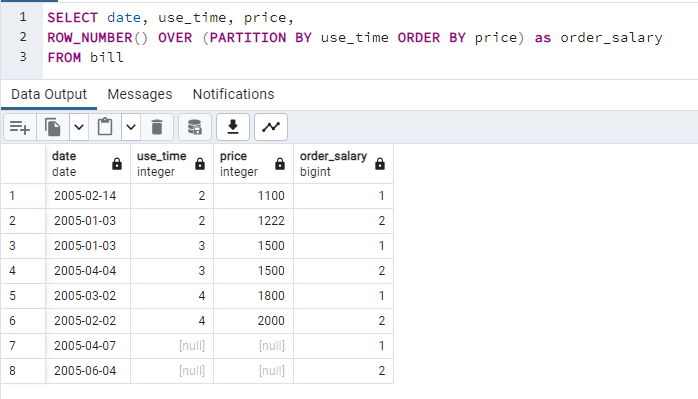
**Код:**

SELECT date, use\_time, price,

ROW\_NUMBER() OVER (PARTITION BY use\_time ORDER BY price) as order\_salary

FROM bill

**Скриншот выполнения работы программы:**

****

**Объединение таблиц самих с собой**

3. Вывести пары атрибутов одной сущности при определенном условии.

**Запрос:**

Вывести все уникальные цены за все время, где покупки стоили больше 1300

**Код:**

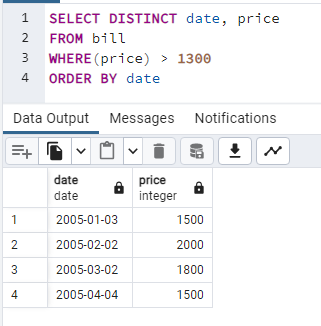
SELECT DISTINCT date, price

FROM bill

WHERE(price) > 1300

ORDER BY date

**Скриншот выполнения работы программы:**

****

4. Тоже, что и в пункте 3 с устранением избыточности без оператора DISTINCT

**Запрос:**

Вывести все уникальные покупки за каждый день, стоимость которых больше 1300руб.

**Код:**

SELECT date, max(price)

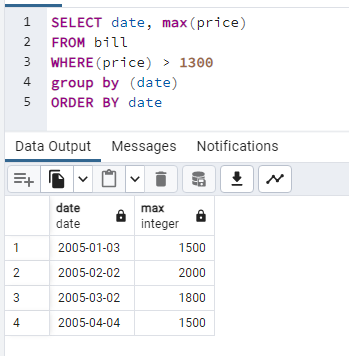
FROM bill

WHERE(price) > 1300

group by (date)

ORDER BY date

**Скриншот выполнения работы программы:**

****

**Подзапросы**

5. Придумать однотабличный запрос, использующий подзапрос в условии отбора строк поле фразы WHERE.

**Запрос:**

Вывести все покупки цена которых выше средней

**Код:**

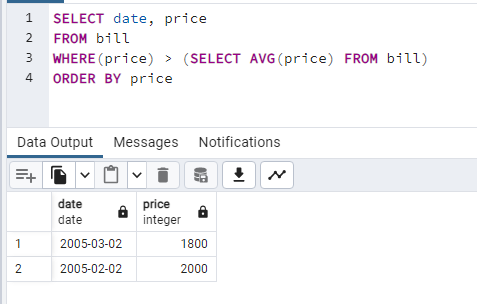
SELECT date, price

FROM bill

WHERE(price) > (SELECT AVG(price) FROM bill)

ORDER BY price

**Скриншот выполнения работы программы:**

****

6. Придумать многотабличный запрос, использующий подзапрос в условии отбора строк поле фразы WHERE.

**Запрос:**

Вывести все типы лодок, заказанных у вендора 1

**Код:**

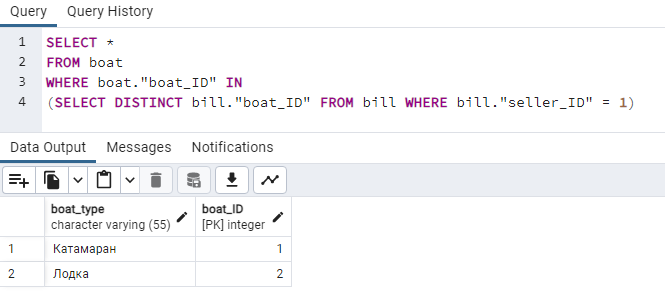
SELECT \*

FROM boat

WHERE boat."boat\_ID" IN

(SELECT DISTINCT bill."boat\_ID" FROM bill WHERE bill."seller\_ID" = 1)

**Скриншот выполнения работы программы:**

****

7. Придумать запрос, использующий подзапрос с агрегатной функцией в условии отбора строк поле фразы WHERE.

**Запрос:**

Вывести всех продавцов, продажи которых выше 2400руб.

**Код:**

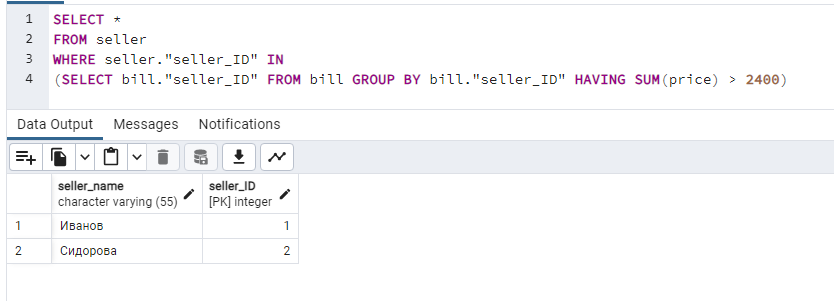
SELECT \*

FROM seller

WHERE seller."seller\_ID" IN

(SELECT bill."seller\_ID" FROM bill GROUP BY bill."seller\_ID" HAVING SUM(price) > 2400)

**Скриншот выполнения работы программы:**

****

8. Придумать запрос на использование подзапросов, которые выдают много строк с помощью оператора IN. **Запрос:**

Вывести все типы лодок, которые не были куплены до 2005.02.02

**Код:**

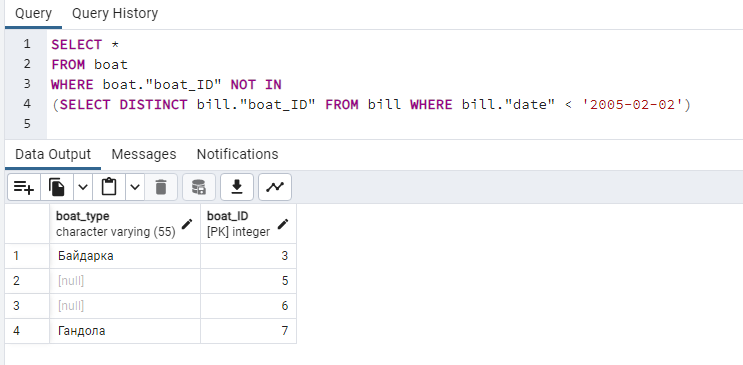
SELECT \*

FROM boat

WHERE boat."boat\_ID" NOT IN

(SELECT DISTINCT bill."boat\_ID" FROM bill WHERE bill."date" < '2005-02-02')

**Скриншот выполнения работы программы:**

****

9. Придумать запрос, использующий подзапрос в предложении HAVING.

**Запрос:**

Вывести всех продавцов, которые оформили аренду больше 1 раза

**Код:**

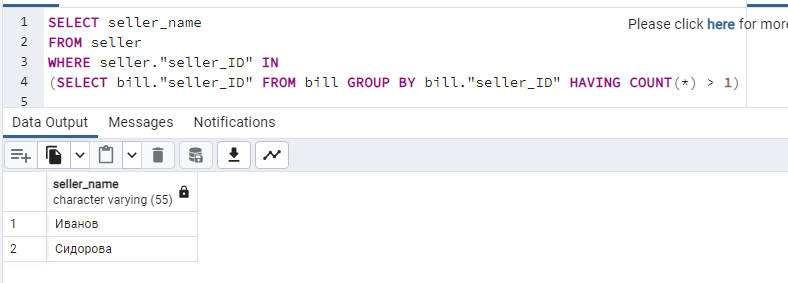
SELECT seller\_name

FROM seller

WHERE seller."seller\_ID" IN

(SELECT bill."seller\_ID" FROM bill GROUP BY bill."seller\_ID" HAVING COUNT(\*) > 1)

**Скриншот выполнения работы программы:**

****

10. Придумать запрос, использующий подзапрос в предложении FROM.

**Запрос:**

Вывести всех продавцов, а также количество их продаж

**Код:**

SELECT seller."seller\_name", subquery.bills\_count

FROM seller

JOIN (

SELECT bill."seller\_ID", COUNT(\*) AS bills\_count

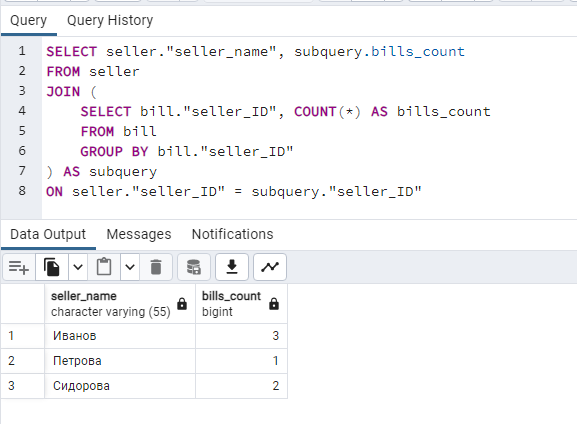
FROM bill

GROUP BY bill."seller\_ID"

) AS subquery

ON seller."seller\_ID" = subquery."seller\_ID"

**Скриншот выполнения работы программы:**

****

**Соотнесенные подзапросы**

11. Придумать запрос на использование соотнесенного подзапроса, который выдает много строк с помощью оператора IN.

**Запрос:**

Вывести всех продавцов, которые продавали Катамаран

**Код:**

SELECT \*

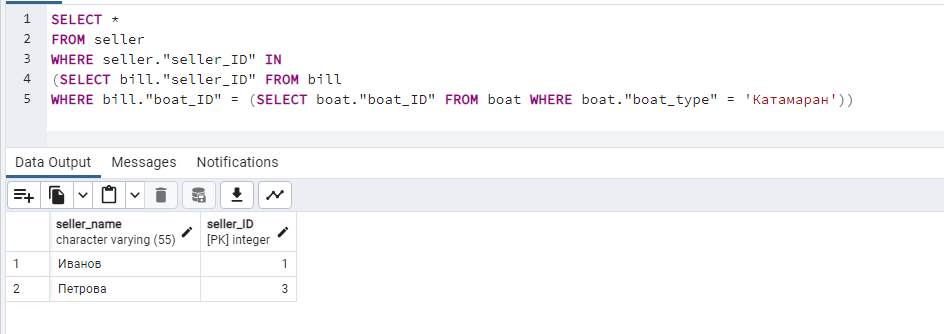
FROM seller

WHERE seller."seller\_ID" IN

(SELECT bill."seller\_ID" FROM bill

WHERE bill."boat\_ID" = (SELECT boat."boat\_ID" FROM boat WHERE boat."boat\_type" = 'Катамаран'))

**Скриншот выполнения работы программы:**

****

12. Придумать запрос на сравнение таблицы с собой.

**Запрос:**

Сравнить таблицу продавцов саму с собой и вывести все уникальные комбинации пар продавцов

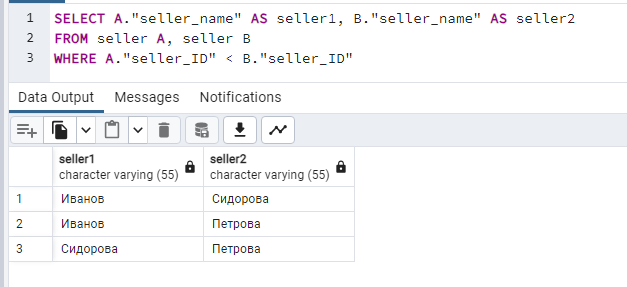
**Код:**

SELECT A."seller\_name" AS seller1, B."seller\_name" AS seller2

FROM seller A, seller B

WHERE A."seller\_ID" < B."seller\_ID"

**Скриншот выполнения работы программы:**

****