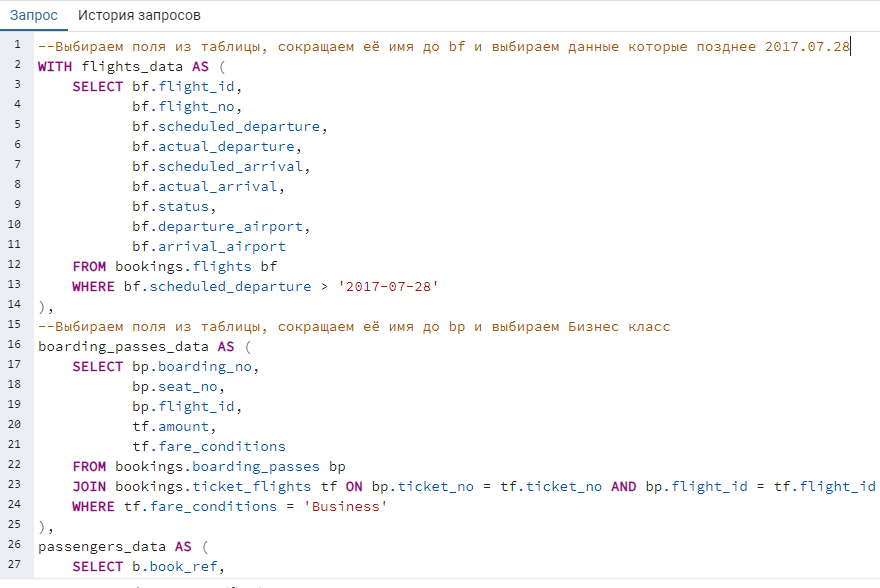
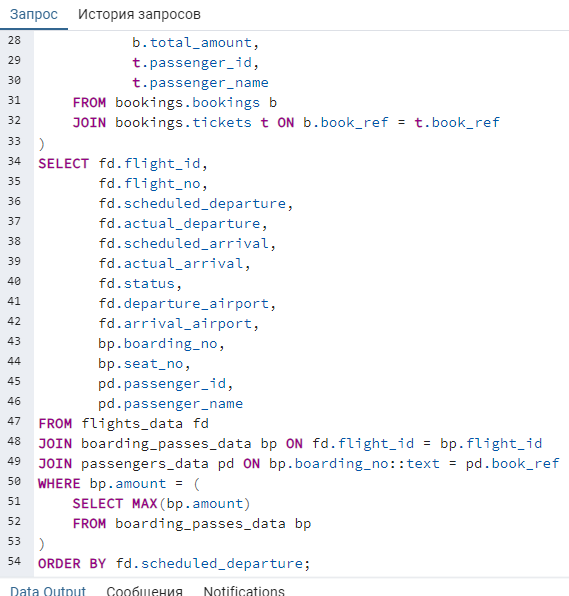
Ерхан Михаил

Группа Пи22-1в

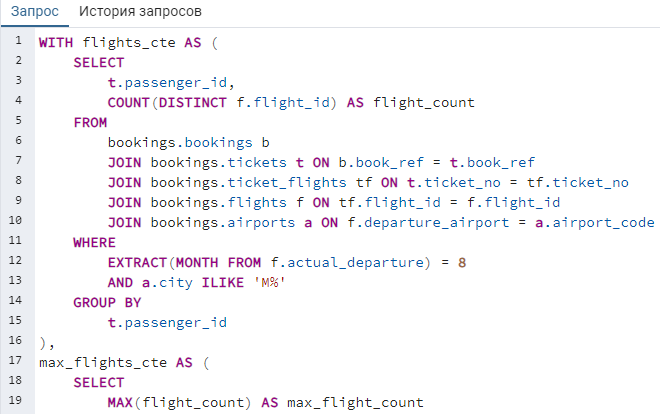
Практическая работа № 2

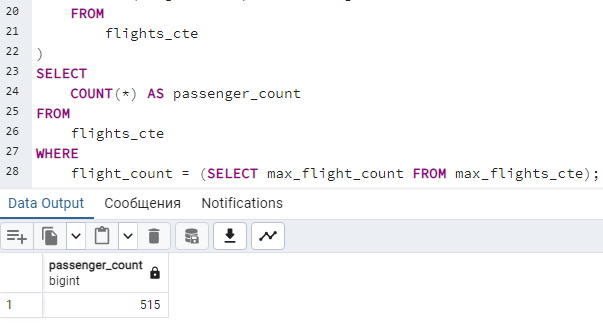
1. Напишите запрос на основе CTE, который ищет в базе перелеты, совершенные после 28.07.2017 года, и выводит по ним информацию о пассажирах, которые летели бизнес-классом по максимальной цене.





1. Напишите запрос на основе CTE, который ищет максимальное количество пассажиров, совершивших минимальное количество перелетов в августе, в рейсах, совершенных из городов, название которые начинается с буквы «М».

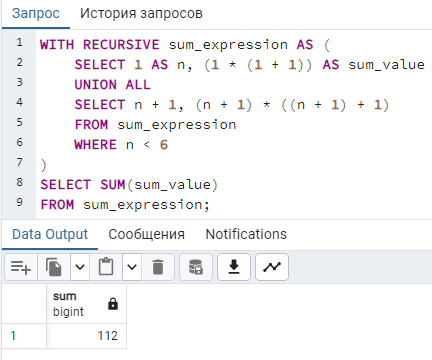




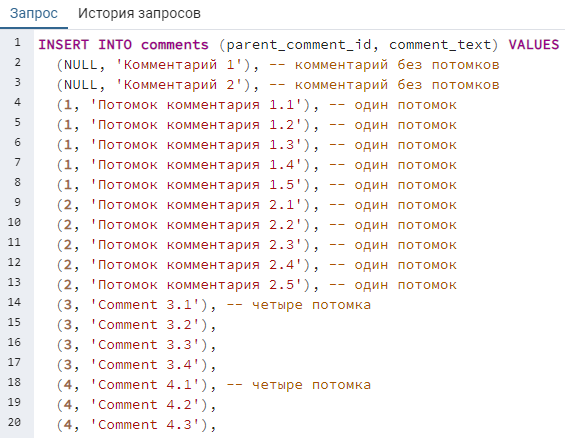
1. Создайте копию таблицы flights – flights\_copy. Напишите запрос, используя изменение данных в WITH, который перемещает из flights\_copy в таблицу flights\_log все рейсы из аэропортов городов Воронеж, Москва и Самара, совершенные c 29.07.2017 по 12.08.2017. При этом таблица flights\_log должна содержать только следующие столбцы: flight\_id, airports.airport\_code, airports.city, scheduled\_departure, scheduled\_arrival.

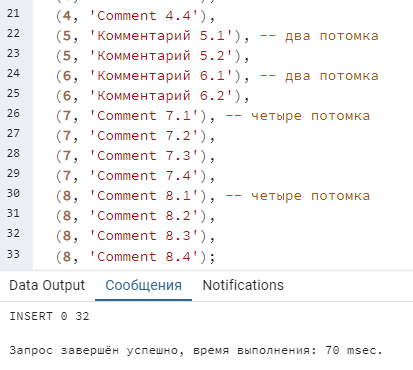


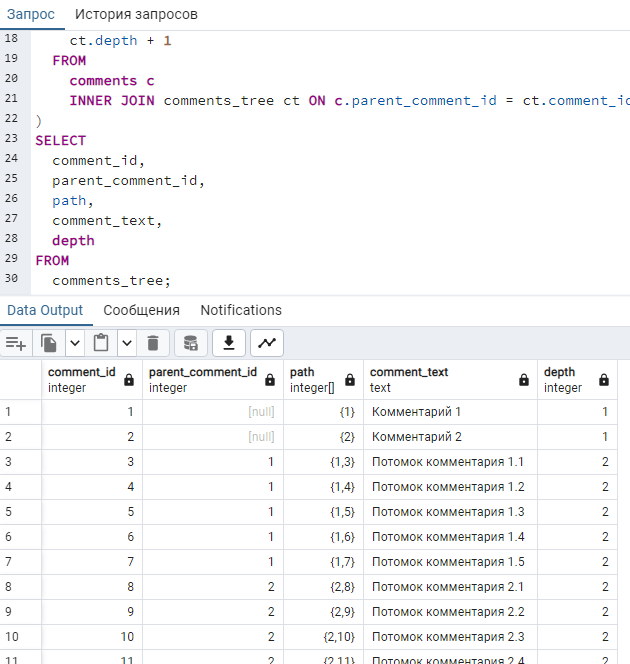
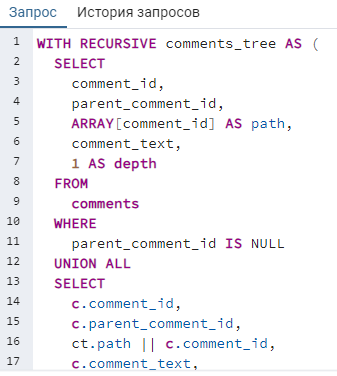
1. Реализуйте рекурсивный запрос для вычисления значения выражения: Сумма шести слагаемых n\*(n + 1) при начальном n0 = 1.



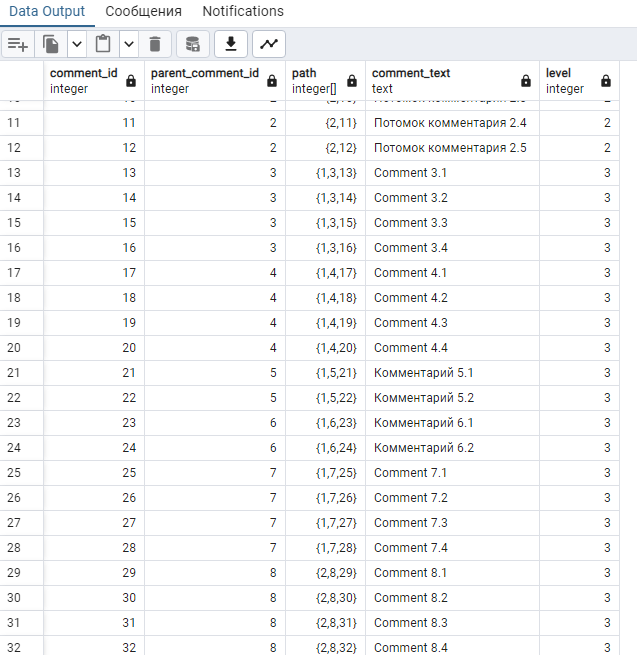
1. Создайте таблицу, в которой будут храниться комментарии чата, имеющие иерархическую структуру: каждый комментарий, может иметь до пяти потомков. Заполните таблицу выдуманными данными, в таблице должно быть не менее двух комментариев с пятью потомками, не менее 3 комментариев с четырьмя потомками, не менее 5 комментариев с двумя потомками, комментарии с одним потомком, комментарии без потомков (следует хранить не только id комментариев, но и сами комментарии). Выполните с помощью рекурсивных запросов обходы дерева комментариев в глубину и ширину, сделайте скриншоты запросов и результатов.



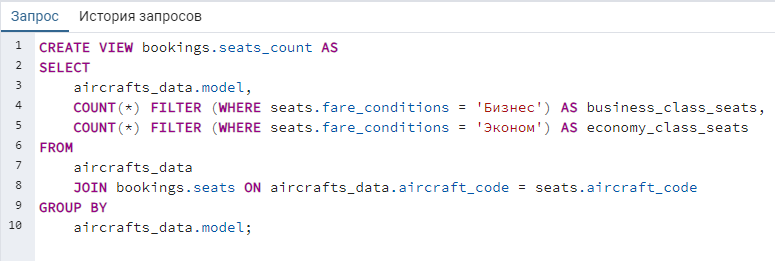


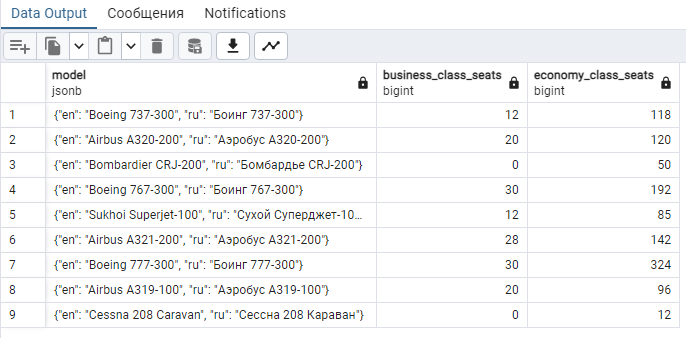




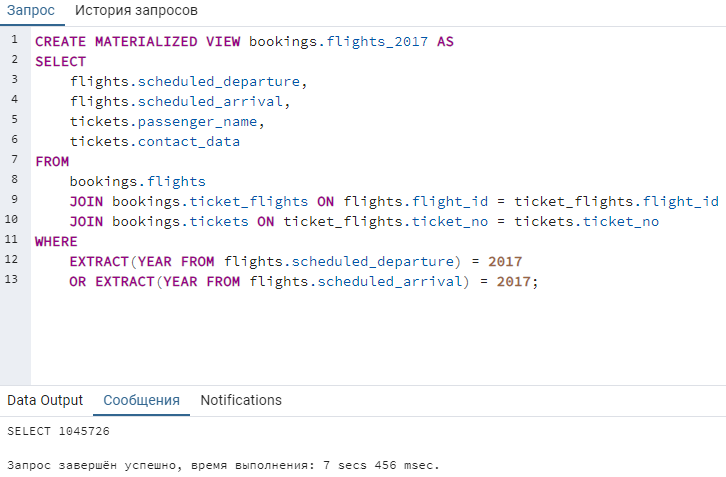


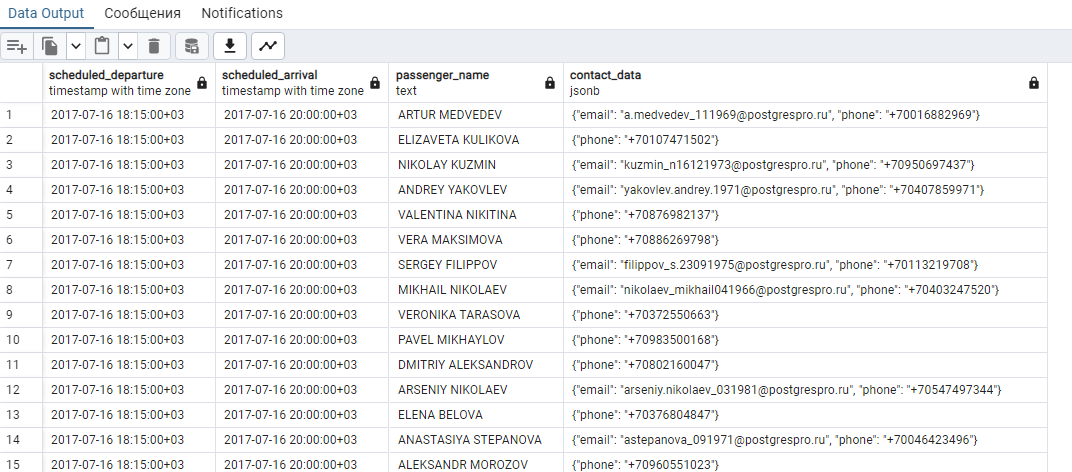
1. Создайте представление на основе запроса: подсчитать количество мест в салонах для всех моделей самолетов с учетом класса обслуживания (бизнес-класс и эконом-класс).





1. Создайте материализованное представление, включающее информацию о перелетах и пассажирах за 2017 год.





ERROR: представление airports зависит от объекта таблица airports\_data представление flights\_v зависит от объекта представление airports представление routes зависит от объекта представление airports ограничение flights\_arrival\_airport\_fkey в отношении таблица flights зависит от объекта таблица airports\_data ограничение flights\_departure\_airport\_fkey в отношении таблица flights зависит от объекта таблица airports\_data удалить объект таблица airports\_data нельзя, так как от него зависят другие объекты

1. Представления (views) могут быть обновляемыми. Это значит, что можно с помощью команд INSERT, UPDATE и DELETE, применяемых к представлению, внести изменения в таблицу, лежащую в основе этого представления. Создайте простое представление над одной из таблиц базы данных «Авиаперевозки», выполните несколько команд с целью внесения изменений в эту таблицу.

CREATE OR REPLACE VIEW bookings.updatable\_view AS

SELECT \*

FROM bookings.bookings

WHERE total\_amount > 200000;

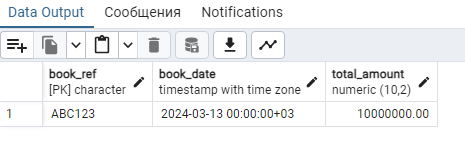
-----

INSERT INTO bookings.updatable\_view (book\_ref, book\_date, total\_amount)

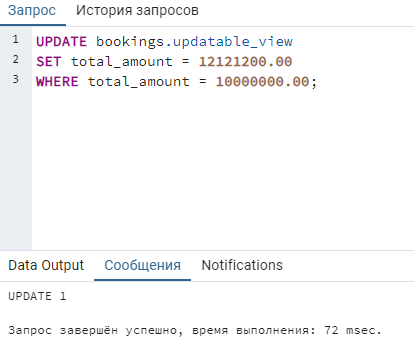
VALUES ('ABC123', '2024-03-13', 10000000.00);

-----

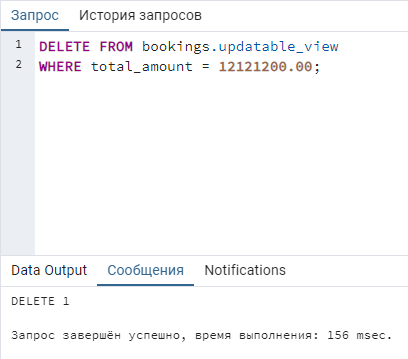
Проверка, что такая запись есть в таблице:



-----



-----



1. Определение таблицы можно изменить с помощью команды ALTER TABLE. Аналогичные команды существуют и для изменения представлений и материализованных представлений: ALTER VIEW и ALTER MATERIALIZED VIEW. Самостоятельно ознакомьтесь с их возможностями с помощью документации.

Допустим, у нас есть представление "flights\_info", которое содержит информацию о перелетах. Мы хотим изменить это представление и добавить новое поле "flight\_status".

ALTER VIEW flights\_info

AS

SELECT

flights.\*,

status.flight\_status

FROM

flights

INNER JOIN status ON flights.status\_id = status.status\_id;

Предположим, у нас есть материализованное представление "sales\_report", которое содержит отчет о продажах. Мы хотим изменить запрос в этом представлении, чтобы учесть дополнительные фильтры и агрегатные функции.

ALTER MATERIALIZED VIEW sales\_report

REFRESH FAST

AS

SELECT

product\_id,

SUM(quantity) AS total\_quantity,

SUM(amount) AS total\_amount

FROM

sales

WHERE

sale\_date >= '2022-01-01' AND sale\_date <= '2022-12-31'

GROUP BY

product\_id;