Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет инфокоммуникационных технологий

**ОТЧЕТ**

**О ЛАБОРАТОРНОЙ РАБОТЕ № 4**

По теме: «Запросы на выборку данных к БД PostgreSQL»

По дисциплине: Проектирование и реализация баз данных

Специальность:

09.03.03 Мобильные и сетевые технологии

|  |  |  |
| --- | --- | --- |
| Проверил:  Говорова М.М.  Дата: «26» 05 2021г.  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | Выполнил(и):  студент(ы) группы К3240  Дао Куанг Ань |
|  |  |  |

Санкт-Петербург 2021

1. Цель работы: овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQLи использования подзапросов при модификации данных.

Вариант 10. БД «Автовокзал»

2. Практическое задание:

1. Создать запросы и представления на выборку данных к базе данных PostgreSQL

(согласно индивидуальному заданию, часть 2 и 3).

2. Составить 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с

использованием подзапросов.

3. Изучить графическое представление запросов.

4. Просмотреть историю запросов.

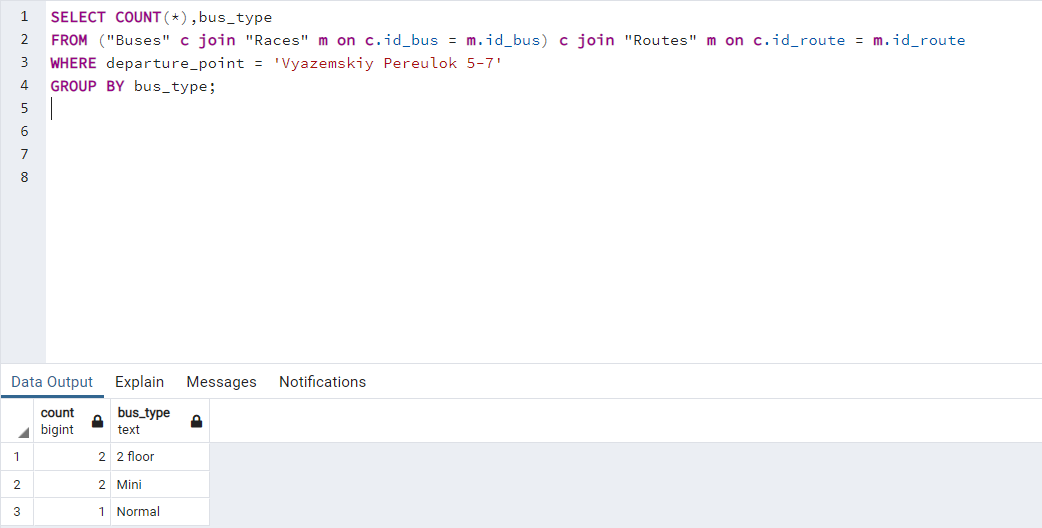
3. Выполнение задания

1. Вывести количество автобусов каждого типа, отправляющихся с автовокзала.

SELECT COUNT(DISTINCT m.id\_bus),bus\_type FROM ("Buses" c join "Races" m on c.id\_bus = m.id\_bus) c join "Routes" m on c.id\_route = m.id\_route

WHERE departure\_point = 'Vyazemskiy Pereulok 5-7'

GROUP BY bus\_type;



1. Вывести фамилии водителей и номера автобусов, отправившиеся в рейсы до 12 часов текущего дня.

SELECT DISTINCT id\_bus, driver\_name

FROM ("Drivers" c join "Races" m on c.id\_driver = m.id\_driver)

WHERE time\_departure < CURRENT\_DATE + INTERVAL '12 hours' AND time\_departure > CURRENT\_DATE

GROUP BY driver\_name,id\_bus;

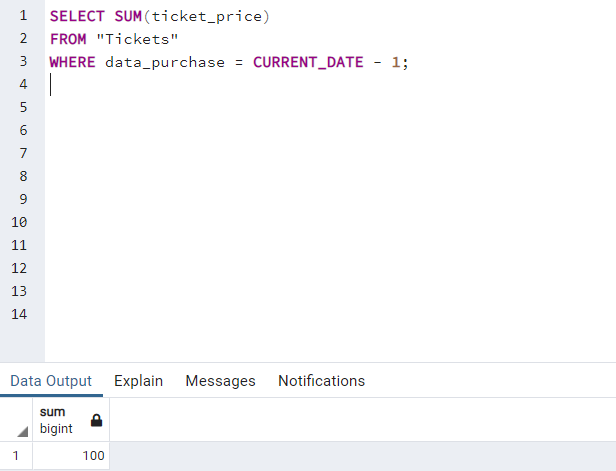


1. Рассчитать выручку от продажи билетов за прошедший день.

SELECT SUM(ticket\_price)

FROM "Tickets"

WHERE data\_purchase = CURRENT\_DATE - 1;

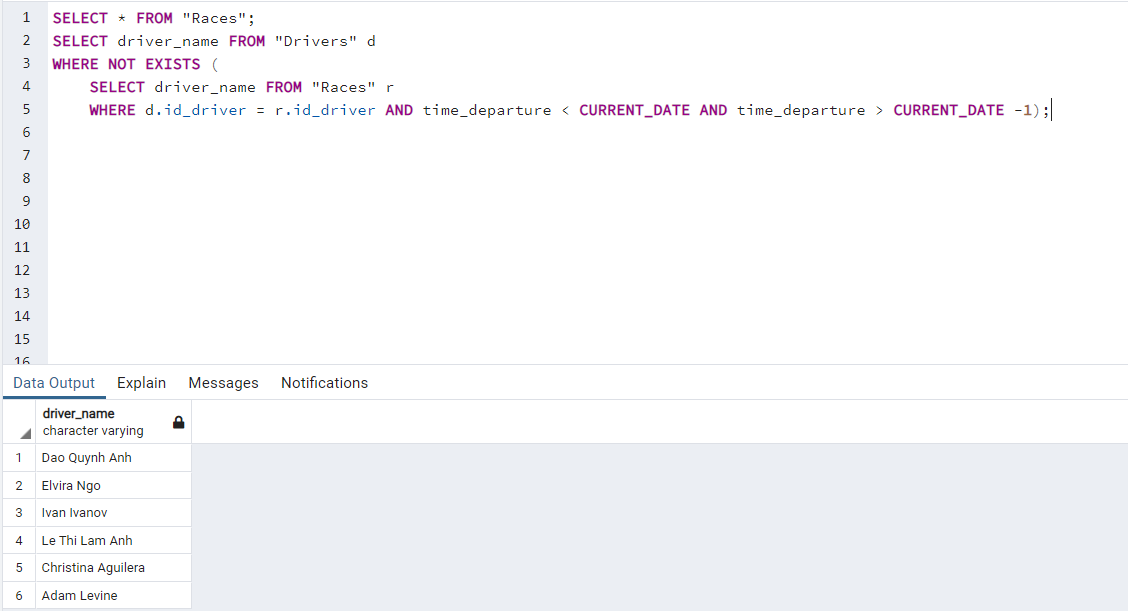


1. Вывести список водителей, которые не выполнили ни одного рейса за прошедший день.

SELECT DISTINCT driver\_name FROM "Drivers" d

WHERE NOT EXISTS (

SELECT driver\_name FROM "Races" r

WHERE d.id\_driver = r.id\_driver AND time\_departure < CURRENT\_DATE AND time\_departure > CURRENT\_DATE -1);

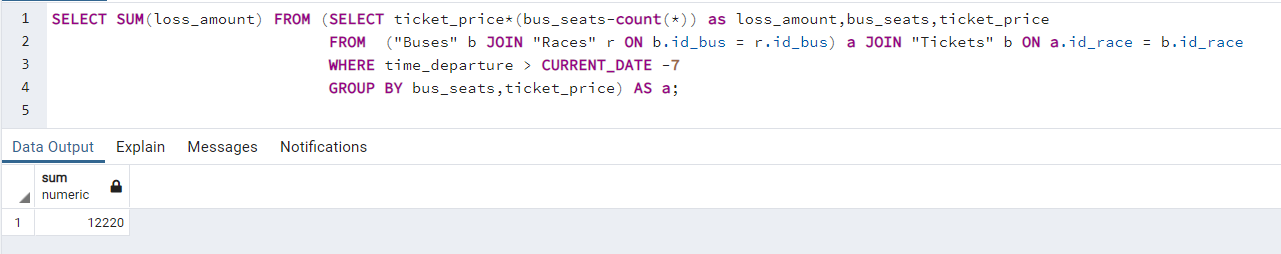
1. Вывести сумму убытков из-за непроданных мест в автобусе за прошедшую неделю.

SELECT SUM(loss\_amount) FROM (SELECT ticket\_price\*(bus\_seats-count(\*)) as loss\_amount,bus\_seats,ticket\_price

FROM ("Buses" b JOIN "Races" r ON b.id\_bus = r.id\_bus) a JOIN "Tickets" b ON a.id\_race = b.id\_race

WHERE time\_departure > CURRENT\_DATE -7

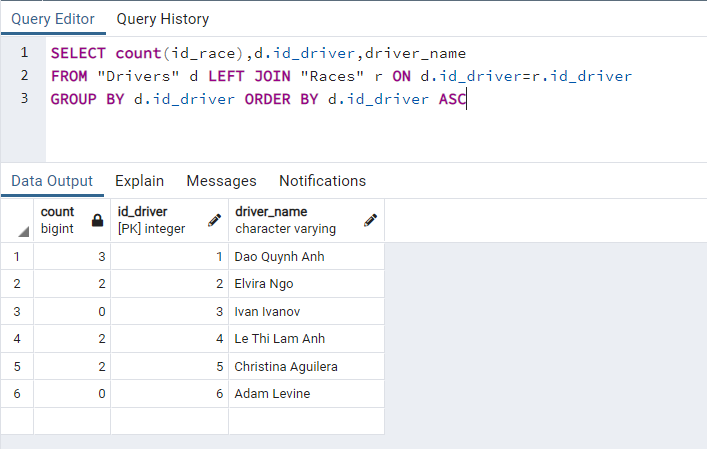
GROUP BY bus\_seats,ticket\_price) AS a;



1. Сколько рейсов выполнил каждый водитель за последний месяц.

SELECT count(id\_race),d.id\_driver,driver\_name

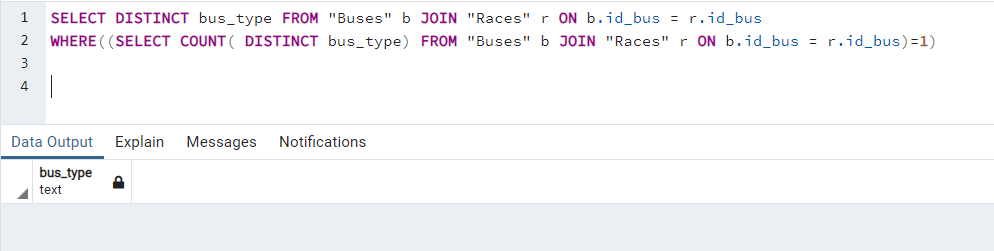
FROM "Drivers" d LEFT JOIN "Races" r ON d.id\_driver=r.id\_driver

GROUP BY d.id\_driver ORDER BY d.id\_driver ASC 

1. Вывести тип автобуса, который используется на всех рейсах.

SELECT DISTINCT bus\_type FROM "Buses" b JOIN "Races" r ON b.id\_bus = r.id\_bus

WHERE((SELECT COUNT( DISTINCT bus\_type) FROM "Buses" b JOIN "Races" r ON b.id\_bus = r.id\_bus)=1)



1. Вывести данные водителя, который провел максимальное время в пути за прошедшую неделю.

SELECT DISTINCT B.id\_driver,B.driver\_state,B.driver\_salary,B.driver\_name,B.driver\_passport\_serie,B.driver\_telephone,B.driver\_passport\_date,B.driver\_passport\_issued\_by,

SUM(R.time\_arrival-R.time\_departure)

FROM "Drivers" B JOIN "Races" R ON B.id\_driver = R.id\_driver

GROUP BY B.id\_driver

HAVING SUM(R.time\_arrival-R.time\_departure) >= ALL(SELECT SUM(R.time\_arrival-R.time\_departure)

FROM "Drivers" B JOIN "Races" R ON B.id\_driver = R.id\_driver);



Представления:

1. Количество свободных мест на все рейсы на завтра;

create or replace view public.available\_tickets

as

select b.bus\_seats-count(\*) as available\_tickets,r.id\_race,b.bus\_seats,ticket\_price,time\_departure,time\_arrival

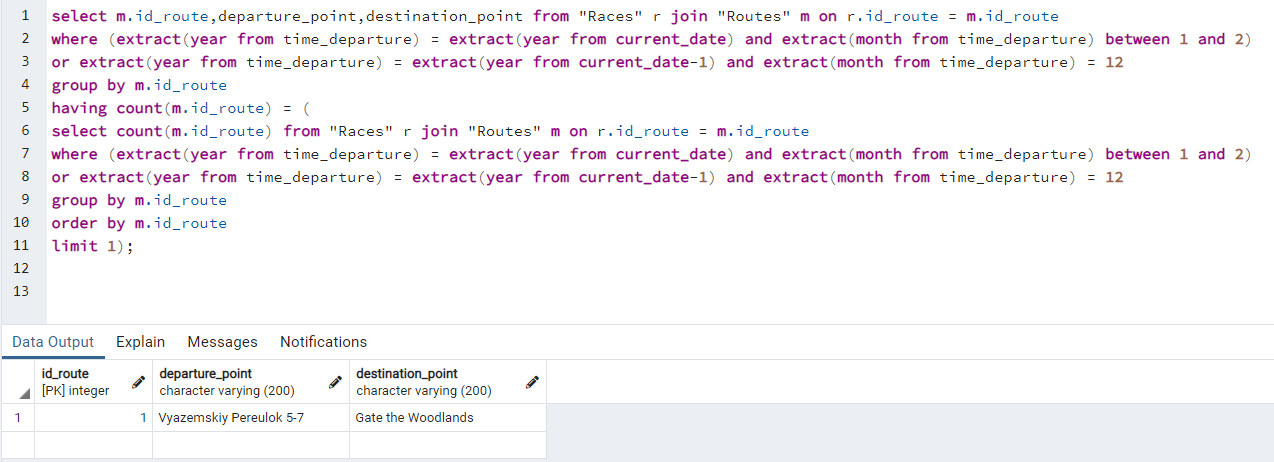
from "Buses" b join "Races"r on b.id\_bus=r.id\_bus join "Tickets" t on r.id\_race = t.id\_race

where time\_departure >= current\_date+1

group by r.id\_race,b.bus\_seats,ticket\_price,time\_departure,time\_arrival;

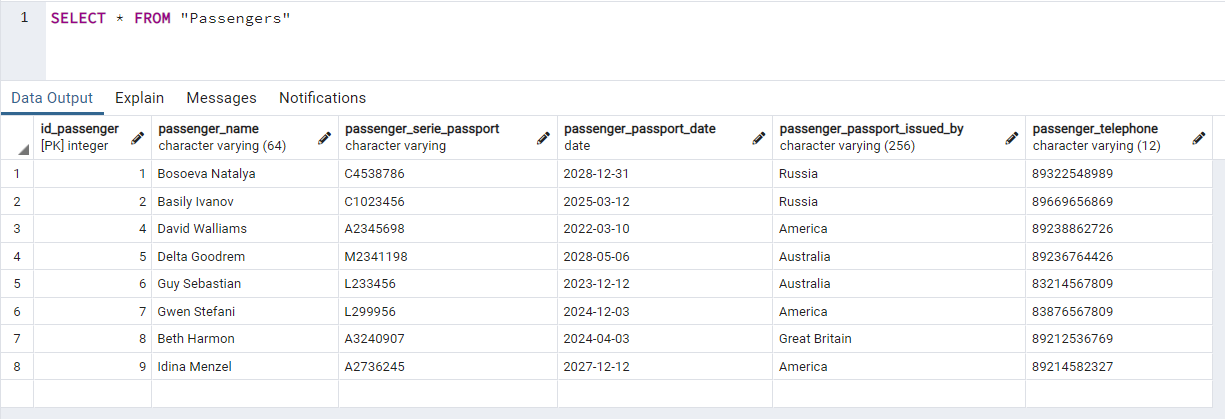
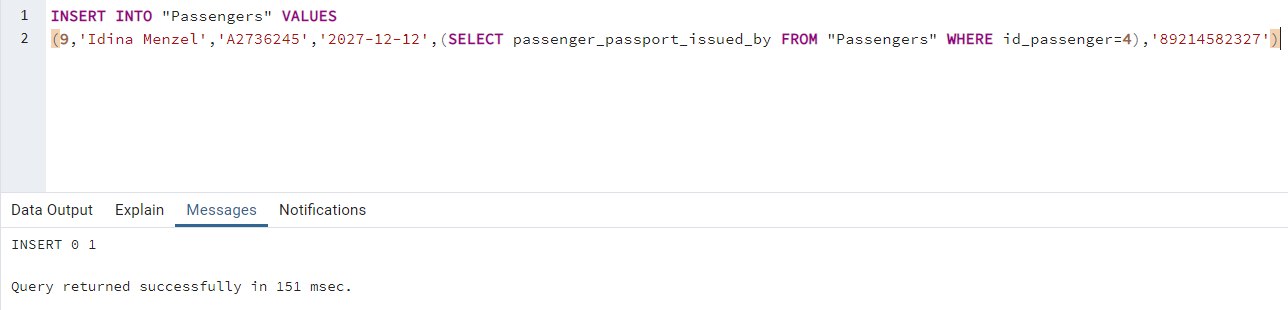


1. Самый популярный маршрут зимой.

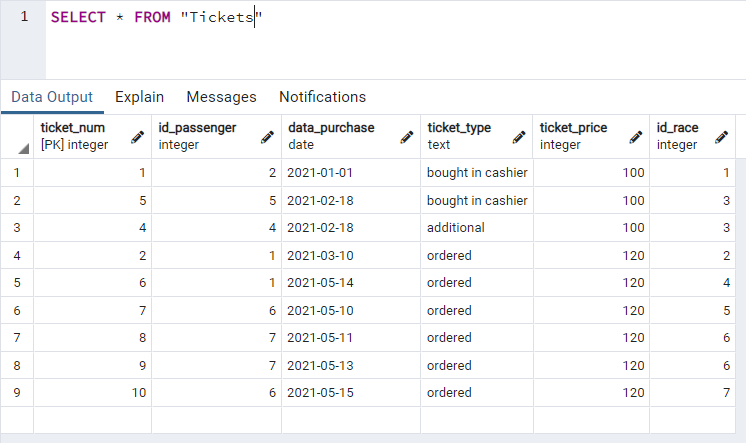
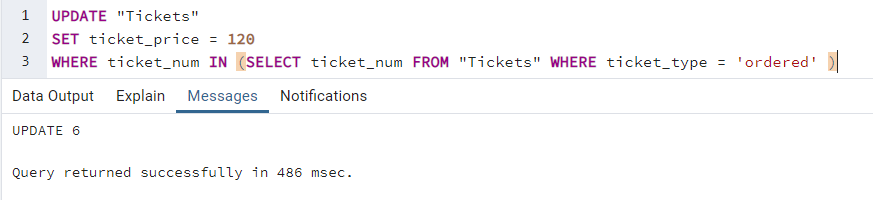


Запрос на модификацию данных:

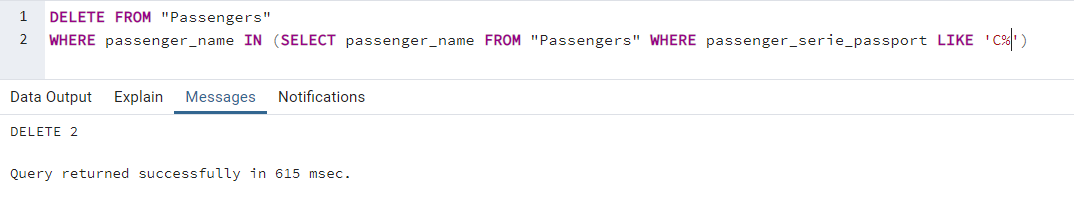
INSERT

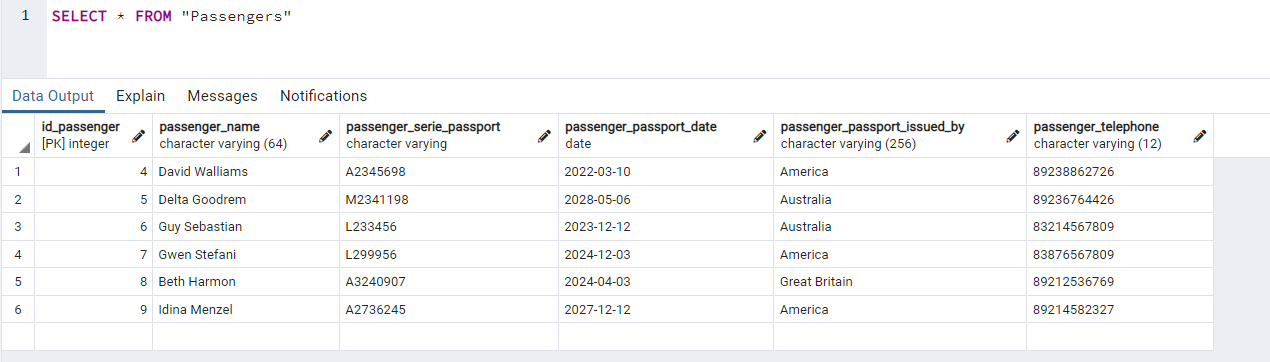


UPDATE



DELETE





4.Вывод

- Создал запросы и представления на выборку данных.

- Составил 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов.

- Изучил графическое представление запросов. - Просмотрел историю запросов.