

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**  
**И ПРЕДСТАВЛЕНИЯ В POSTGRESQL»**

по дисциплине: Проектирование и реализация баз данных

Специальность:  
09.03.03 Прикладная информатика

Проверила: \_\_\_\_\_ Выполнил:  
Говорова М.М. \_\_\_\_\_ студент группы K3241  
Дата: «\_\_» \_\_\_\_\_ 20\_\_ г. Хорошкеева К. А.  
Оценка \_\_\_\_\_

# Санкт-Петербург 2020/2021

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

## База данных «Пассажир»

### Часть 2. Запросы к базе данных

1. Свободные места на все поезда, отправляющиеся с вокзала в течение следующих суток.

Query Editor

Query History

```
1 select train.train_number, train.departure_date,
2 "train_carriage"."Id_carriage", id_seat from train, train_carriage, seat
3 where train.train_number = train_carriage.number_train_carriage and
4 "train_carriage"."Id_carriage" = seat.id_train_carriage and
5 date(departure_date) = '2020-03-01' and
6 not exists(select * from ticket where ticket.id_seat = seat.id_seat)
```

Data Output

Explain

Messages

Notifications

	train_number integer	departure_date date	Id_carriage integer	id_seat integer	
1	2	2020-03-01	2	6	

2. Список пассажиров, отправившихся в Москву всеми рейсами за прошедшие сутки.

Query Editor

История запросов

```
1 SELECT departure_date, ticket.id_passenger, full_name
2 FROM public.ticket, public.passenger
3 where ticket.id_last_station = 2
4 and ticket.id_passenger = passenger.id_passenger;
```

Результат

План выполнения

Сообщения

Notifications

	departure_date date	id_passenger integer	full_name character varying (50)
1	2021-06-24	1	Иванов Иван Иванович
2	2021-06-24	2	Семенов Семен Семенович
3	2021-06-25	3	Новиков Иван Семенович

3. Номера поездов, на которые проданы все билеты на следующие сутки.

Query Editor

Query History

```
1 select train.train_number, train.train_name
2 from train
3 full join train_carriage on train.train_number = "train_carriage"."Id_carriage"
4 left join seat on "train_carriage"."Id_carriage" = seat.id_train_carriage
5 left join ticket on seat.id_seat = ticket.id_seat
6 group by train.train_number, ticket.id_ticket
7 having date(ticket.departure_date) = '24.06.2021'
8 and count(ticket.id_ticket) = sum(train_carriage.number_of_seats_in_the_train_carriage) / count(ticket.id_ticket)
```

Data Output

Explain

Messages

Notifications

train_number [PK] integer	train_name text
1	Владивосток

4. Свободные места в купейные (сидячие) вагоны всех рейсов до Москвы на текущие сутки.

5. Выручка от продажи билетов на все поезда за прошедшие сутки.

Query Editor

История запросов

1

SELECT departure\_date, sum(ticket\_price) FROM public.ticket GROUP BY departure\_date;

Результат

План выполнения

Сообщения

Notifications

	departure_date date	sum bigint
1	2021-06-25	1200
2	2021-06-24	1718

6. Общее количество билетов, проданных по всем направлениям в вагоны типа купе.

Query Editor

Query History

```
1 select count(*) from ticket, seat
2 where ticket.id_seat = seat.id_seat
3 and seat.seat_type = 'купе'
```

Data Output

Explain

Messages

Notifications

count	
bigint	
1	1

7. Номера и названия поездов, все вагоны которых были заполнены менее чем наполовину за прошедшие сутки.

### Часть 3. Представления

1. Представление для пассажиров о наличии свободных мест на заданный рейс.

Query Editor

Query History

```
1 select train.train_number, train.departure_date,
2 "train_carriage"."Id_carriage", id_seat from train, train_carriage, seat
3 where train.train_number = train_carriage.number_train_carriage and
4 "train_carriage"."Id_carriage" = seat.id_train_carriage and
5 date(departure_date) = '2020-03-01' and
6 not exists(select * from ticket where ticket.id_seat = seat.id_seat)
```

Data Output

Explain

Messages

Notifications

	train_number integer	departure_date date	Id_carriage integer	id_seat integer	
1	2	2020-03-01	2	6	

2. Представление с количеством непроданных билетов на все поезда, формирующиеся за прошедшие сутки (номер поезда, тип вагона, количество).

Query Editor

Query History

```
1 create view seats_left2 as
2 select train.train_number, train.train_name, seat.seat_type,
3 train_carriage.number_of_seats_in_the_train_carriage - count(*) as free_seats
4 from train
5 join train_carriage on train.train_number = train_carriage.number_train_carriage
6 join seat on "train_carriage"."Id_carriage" = seat.id_train_carriage
7 join ticket on seat.id_seat = ticket.id_seat
8 group by train.train_number, seat.seat_type, train_carriage.number_of_seats_in_the_train_carriage, ticket.id_ticket
9 having date(ticket.departure_date) = '2021-06-24';
10
11 select * from seats_left2
```

Data Output

Explain

Messages

Notifications

	train_number integer	train_name text	seat_type text	free_seats bigint	
1	1	Владивост...	купе	2	
2	1	Владивост...	Люкс	2	
3	1	Владивост...	Люкс	2	
4	2	Победа	купе	2	

### Часть 4. Запросы на изменение данных

1. Update

Query Editor		Query History	
<pre>1 UPDATE public.passenger 2     SET passport_number=101010101, full_name = 'Роботников Робот Роботович' 3     WHERE id_passenger = 2;</pre>			
Data Output		Explain	Messages
UPDATE 1			
Query returned successfully in 61 msec.			

До обновления

После обновления данных

Query EditorQuery History

1SELECT \* FROM public.passenger

2ORDER BY id\_passenger ASC

Data OutputExplainMessagesNotifications

	id_passenger [PK] integer	passport_number integer	full_name text
1	1	123456	Иванов Иван Иванович
2	2	333333	Семенов Семен Семенович
3	3	2121221	Новиков Иван Семенович
4	4	213123	Васильев Григорий Никитьевич

Query EditorQuery History

1SELECT \* FROM public.passenger

2ORDER BY id\_passenger ASC

Data OutputExplainMessagesNotifications

	id_passenger [PK] integer	passport_number integer	full_name text
1	1	123456	Иванов Иван Иванович
2	2	101010101	Роботников Робот Роботович
3	3	2121221	Новиков Иван Семенович
4	4	213123	Васильев Григорий Никитьевич

2. Delete

Query EditorQuery History

1DELETE FROM public.passenger

2WHERE id\_passenger = 5;

Data OutputExplainMessagesNotifications

DELETE 1

Query returned successfully in 51 msec.

До удаления

После удаления

Query EditorQuery History

1SELECT \* FROM public.passenger

2ORDER BY id\_passenger ASC

Data OutputExplainMessagesNotifications

	id_passenger [PK] integer	passport_number integer	full_name text
1	1	123456	Иванов Иван Иванович
2	2	333333	Семенов Семен Семенович
3	3	2121221	Новиков Иван Семенович
4	4	213123	Васильев Григорий Никитьевич
5	5	12333333	Васильев Николай Максимович

Query EditorQuery History

1SELECT \* FROM public.passenger

2ORDER BY id\_passenger ASC

Data OutputExplainMessagesNotifications

	id_passenger [PK] integer	passport_number integer	full_name text
1	1	123456	Иванов Иван Иванович
2	2	333333	Семенов Семен Семенович
3	3	2121221	Новиков Иван Семенович
4	4	213123	Васильев Григорий Никитьевич

После удаления

3. Insert

## До обновления

Query Editor	Query History		
<pre>1  INSERT INTO public.passenger( 2      id_passenger, passport_number, full_name) 3      VALUES (5, 12333333, 'Васильев Николай Максимович');</pre>			
Data Output	Explain	Messages	Notifications
INSERT 0 1			
Query returned successfully in 66 msec.			

## После обновления

Query Editor

Query History

```

1  SELECT * FROM public.passenger
2  ORDER BY id_passenger ASC

```

Data Output

Explain

Messages

Notifications

	id_passenger [PK] integer	passport_number integer	full_name text	
1	1	123456	Иванов Иван Иванович	
2	2	333333	Семенов Семен Семенович	
3	3	2121221	Новиков Иван Семенович	
4	4	213123	Васильев Григорий Никитичевич	
5	5	12333333	Васильев Николай Максимович	

## Новые запросы на изменение (с подзапросами)

### 1. Insert

Добавить Иванову Ивану Ивановичу билет Москва-Комарово класса люкс

### До вставки

Query Editor

История запросов

1

select \* from ticket

Результат

План выполнения

Сообщения

Notifications

	<div>id_ticket</div> <div>[PK] integer</div>	<div>departure_date</div> <div>date</div>	<div>ticket_price</div> <div>integer</div>	<div>arrival_date</div> <div>date</div>	<div>id_first_station</div> <div>integer</div>	<div>id_last_station</div> <div>integer</div>	<div>id_passenger</div> <div>integer</div>	<div>id_seat</div> <div>integer</div>
1	1	2021-06-24	359	2021-06-24	1	2	1	1
2	2	2021-06-24	359	2021-06-24	1	2	2	2
3	3	2021-06-24	1000	2021-06-25	1	3	3	3
4	4	2021-06-25	600	2021-06-25	2	3	2	3
5	5	2021-06-25	600	2021-06-25	3	2	3	3

### После вставки

```

1  INSERT INTO public.ticket(
      id_ticket, departure_date, ticket_price, arrival_date, id_first_station, id_last_station, id_passenger, id_seat
2
3  VALUES (
4      6, '2021-06-27', 500, '2021-06-27',
5      (select id_station from station where station_name like 'Москва' limit 1),
6      (select id_station from station where station_name like 'Комарово' limit 1),
7      (select id_passenger from passenger where full_name like 'Иванов Иван Иванович' limit 1),
8      (select id_seat from seat where seat_type like 'Люкс' limit 1));
9
10 select * from ticket;

```

Результат

План выполнения

Сообщения

Notifications

	id_ticket [PK] integer	departure_date date	ticket_price integer	arrival_date date	id_first_station integer	id_last_station integer	id_passenger integer	id_seat integer
1	1	2021-06-24	359	2021-06-24	1	2	1	1
2	2	2021-06-24	359	2021-06-24	1	2	2	2
3	3	2021-06-24	1000	2021-06-25	1	3	3	3
4	4	2021-06-25	600	2021-06-25	2	3	2	3
5	5	2021-06-25	600	2021-06-25	3	2	3	3
6	6	2021-06-27	500	2021-06-27	2	3	1	1

## 2. Update

Добавить в вагоны где количество мест меньше среднего еще по 5 мест

До обновления

```
1 select * from train_carriage|
```

Результат План выполнения Сообщения Notifications

	Id_train_carriage [PK] integer	number_train_carriage integer	number_of_seats_in_the_train_carriage integer
1	1	1	40
2	2	2	41
3	3	3	44
4	4	1	4
5	5	2	9
6	6	3	15

После обновления

```
1 UPDATE public.train_carriage
2     SET number_of_seats_in_the_train_carriage = number_of_seats_in_the_train_carriage + 5
3     WHERE number_of_seats_in_the_train_carriage < (
4         select avg(number_of_seats_in_the_train_carriage) from train_carriage);
5
6 select * from train_carriage|
```

Результат План выполнения Сообщения Notifications

	Id_train_carriage [PK] integer	number_train_carriage integer	number_of_seats_in_the_train_carriage integer
1	1	1	40
2	2	2	41
3	3	3	44
4	4	1	9
5	5	2	14
6	6	3	20





## 3. Delete

Удалить людей, у которых нет билетов

До удаления

```
1 select * from passenger
```





Результат    План выполнения    Сообщения    Notifications

	 id_passenger [PK] integer 	passport_number integer 	full_name character varying (50) 
1	1	123456	Иванов Иван Иванович
2	2	333333	Семенов Семен Семенович
3	3	2121221	Новиков Иван Семенович
4	4	100000000	Борисов Борис Борисович

После удаления

```
1 DELETE FROM public.passenger
2     WHERE id_passenger not in
3     (select id_passenger from ticket);
4
5 select * from passenger
```

Результат    План выполнения    Сообщения    Notifications

	 id_passenger [PK] integer 	passport_number integer 	full_name character varying (50) 
1	1	123456	Иванов Иван Иванович
2	2	333333	Семенов Семен Семенович
3	3	2121221	Новиков Иван Семенович

## Выводы

SQL-запросы SELECT дают возможно создание представлений для удобства работы с выборками. возможность модифицировать, добавлять и удалять данные. Еще есть возможность делать различные выборки из данных и подсчитывать их числовые характеристики