

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования «Национальный исследовательский университет ИТМО»  
Факультет инфокоммуникационных технологий

Лабораторная работа №2  
«Запросы на выборку и модификацию данных, представления и  
индексы в PostgreSQL»

По дисциплине  
«Проектирование и реализация баз данных»

Выполнил:  
Шевченко.М.Д.  
Группа:  
К3240  
Преподаватель:  
Говорова М.М.



Санкт-Петербург  
2022 г

## ЦЕЛЬ РАБОТЫ

Овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

Практическое задание (10 вариант. Автовокзал)

**Задание 1.** Создать запросы:

- Вывести загрузку преподавателей в понедельник (в часах).
- Найти недельную нагрузку студентов каждой группы (в часах).
- Вывести список свободных лекционных аудиторий в заданное время.
- Вывести количество аудиторий каждого типа.
- Вывести еженедельное количество часов занятий для каждой группы.
- Найти номера аудиторий каждого типа, имеющих максимальное количество мест.
- Вывести фамилии преподавателей, которые всегда проводят практические занятия в одной и той же аудитории.

**Задание 2.** Создать представление:

- содержащее данные о расписании заданной группы на каждый день;
- средняя недельная аудиторная нагрузка по группам

**Задание 3.**

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

**Задание 4.**

Изучить графическое представление запросов и просмотреть историю запросов

**Задание 5.**

Создать простой и составной индексы для двух произвольных запросов и сравнить время выполнения запросов без индексов и с индексами. Для получения плана запроса использовать команду EXPLAIN.

## Запросы к базе данных

1. Вывести загрузку преподавателей в понедельник (в часах).

```
select (extract(epoch from (select sum(Количество_часов) from Занятие where День_недели = 'Понедельник'))/3600) as Количество_часов
```

Результат		План выполнения	
	Количество_часов double precision		
1	4		

2. Найти недельную нагрузку студентов каждой группы (в часах).

```
select Код_группы, sum(Количество_часов) from Занятие  
group by Код_группы
```

Результат		План выполнения		Сообщ	
	Код_группы character varying (8)		sum interval		
1	1.2-Б2/а		08:00:00		
2	1.3-Б3/а		04:30:00		
3	1.1-Б1/а		04:00:00		

3. Вывести список свободных лекционных аудиторий в заданное время.

```
select * from Аудитория where (Код_площадки in (select Код_площадки from Занятие  
where '10:00' < Время_проведения_занятия or '10:00' > Время_проведения_занятия)) and  
Тип = 'Лекционная'
```

Результат		План выполнения		Сообщения		Notifications	
	Номер_аудитории character varying (10)		Кол_во_мест integer		Тип character varying (40)		Код_площадки [PK] character (5)
1	352		20		Лекционная		20001

4. Вывести количество аудиторий каждого типа.

```
select Тип, count(*) as count from Аудитория  
group by Тип
```

Результат		План выполнения		Сообщ	
	Тип character varying (40)		count bigint		
1	Лаборатория		1		
2	Лекционная		1		
3	Поточная		1		

5. Вывести еженедельное количество часов занятий для каждой группы.

select Код\_группы, Код\_дисциплины, sum(Количество\_часов) from Занятие  
group by Код\_группы, Код\_дисциплины

Результат	План выполнения	Сообщения	Notifications
Код_группы character varying (8)	Код_дисциплины integer	sum interval	
1 1.2-Б2/а	2	04:00:00	
2 1.1-Б1/а	1	04:00:00	
3 1.3-Б3/а	2	04:30:00	
4 1.2-Б2/а	3	04:00:00	

6. Найти номера аудиторий каждого типа, имеющих максимальное количество мест.

select Номер\_аудитории, Тип, Max(Кол\_во\_мест) from Аудитория  
group by Номер\_аудитории, Тип

Результат	План выполнения	Сообщения	Notifications
Номер_аудитории character varying (10)	Тип character varying (40)	max integer	
1 312	Поточная	50	
2 352	Лекционная	20	
3 550	Лаборатория	30	

7. Вывести фамилии преподавателей, которые всегда проводят практические занятия в одной и той же аудитории.

select ФИО\_преподавателя from Преподаватель where Таб\_номер\_преподавателя in  
(select Таб\_номер\_преподавателя from (select Таб\_номер\_преподавателя, Вид,  
count(distinct Код\_площадки) from Занятие group by Таб\_номер\_преподавателя, Вид) as  
count where (count = 1 and Вид = 'Практика'))

Результат	План выполнения	С
ФИО_преподавателя character varying (75)		
1 Иванов Иван Иванович		
2 Сергиенко Петр Валерьевич		

insert into Аудитория values ('314','15','Лаборатория','Ломоносова, д. 9','20004')

select Distinct on(Тип) Max(Кол\_во\_мест) from Аудитория  
Group by Тип

select Номер\_аудитории, Тип from Аудитория where Кол\_во\_мест = (select Distinct on(Тип)  
Max(Кол\_во\_мест) from Аудитория Group by Тип)

```
select distinct Тип, Кол_во_мест from Аудитория order by Кол_во_мест desc, Тип
```

```
select *
```

```
from
```

```
(
```

```
select distinct Тип, max(Кол_во_мест)
```

```
from Аудитория
```

```
group by Тип
```

```
)
```

```
as Списочек
```

```
inner join (select distinct Тип, Номер_аудитории from Аудитория) as Табличечка
```

```
on Списочек.Тип = Табличечка.Тип
```

## Представления

1)

Create view Расписание\_группы\_на\_каждый\_день as

(

```
select Код_группы, Название, День_недели, Время_проведения_занятия from Занятие
inner join Дисциплина on Дисциплина.Код_дисциплины = Занятие.Код_дисциплины
where Код_группы = '1.2-Б2/а' order by День_недели
```

)

Результат					План выполнения	Сообщения	Notifications
	Код_группы character varying (8)	Название character varying (40)	День_недели character (11)	Время_проведения_занятия time without time zone			
1	1.2-Б2/а	ООП	Пятница	11:30:00			
2	1.2-Б2/а	Экономика	Суббота	13:30:00			

2)

```
select Код_группы, Код_площадки, sum(Количество_часов) from Занятие
group by Код_группы, Код_площадки
```

	Код_группы character varying (8)	Код_площадки character (5)	sum interval
1	1.3-Б3/а	20001	02:30:00
2	1.3-Б3/а	20003	02:00:00
3	1.1-Б1/а	20001	04:00:00
4	1.2-Б2/а	20002	08:00:00

INSERT, DELETE, UPDATE

```
insert into Преподаватель(Должность, ФИО_преподавателя)
values
(
    (select Должность from Преподаватель where (Таб_номер_преподавателя =
'11111')),
    (select ФИО_преподавателя from Преподаватель where (Должность = 'Ассистент'))
)
```

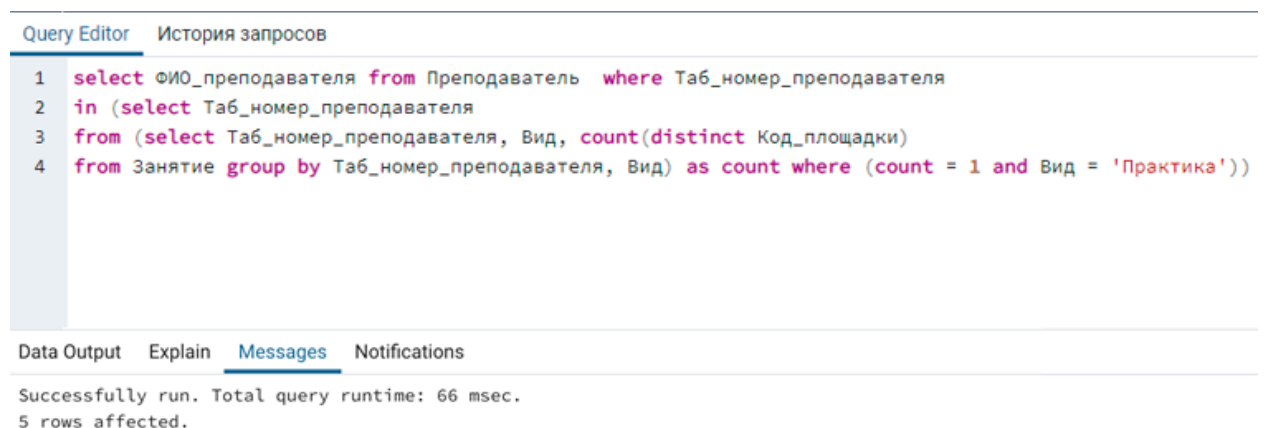
```
delete from Преподаватель
where
(
    Таб_номер_преподавателя = (select Таб_номер_преподавателя from
Преподаватель where (Должность = 'Старший преподаватель'))
)
```

```
update Аудитория
set Кол_во_мест =
(
    select Кол_во_мест - 5
    from Аудитория
    where Тип = 'Лекционная'
)
where Кол_во_мест < '30'
```

Индексы:

1)

До:



The screenshot shows a SQL Query Editor window with two tabs: "Query Editor" and "История запросов". The "Query Editor" tab is active, displaying a SQL query. Below the query, there is a status bar indicating the query was successfully run, the total runtime was 66 msec, and 5 rows were affected. The status bar also includes tabs for "Data Output", "Explain", "Messages", and "Notifications", with "Messages" currently selected.

```
1 select ФИО_преподавателя from Преподаватель where Таб_номер_преподавателя
2 in (select Таб_номер_преподавателя
3 from (select Таб_номер_преподавателя, Вид, count(distinct Код_площадки)
4 from Занятие group by Таб_номер_преподавателя, Вид) as count where (count = 1 and Вид = 'Практика'))
```

Query Editor История запросов

Data Output Explain Messages Notifications

Successfully run. Total query runtime: 66 msec.  
5 rows affected.

Индекс: create index Преподаватель\_ид on Преподаватель(ФИО\_преподавателя)

После:

Query Editor История запросов

```

1 select ФИО_преподавателя from Преподаватель where Таб_номер_преподавателя
2 in (select Таб_номер_преподавателя
3 from (select Таб_номер_преподавателя, Вид, count(distinct Код_площадки)
4 from Занятие group by Таб_номер_преподавателя, Вид) as count where (count = 1 and Вид = 'Практика'))

```

Data Output Explain Messages Notifications

Successfully run. Total query runtime: 39 msec.  
5 rows affected.

🔍 ⚙️ 📄 📥



2)

До:

```

1 select * from Аудитория where (Код_площадки in (select Код_площадки from Занятие
2 where '10:00' < Время_проведения_занятия or '10:00' > Время_проведения_занятия)) and

```

Data Output Explain Messages Notifications

Successfully run. Total query runtime: 75 msec.  
5 rows affected.

Составной индекс: create index Площадка\_ид on Аудитория(Код\_площадки)

После:

```

1 select * from Аудитория where (Код_площадки in (select Код_площадки from Занятие
2 where '10:00' < Время_проведения_занятия or '10:00' > Время_проведения_занятия)) and

```

Data Output Explain Messages Notifications

Successfully run. Total query runtime: 37 msec.  
5 rows affected.



