

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 дисциплине: Проектирование и реализация баз данных

Специальность: 09.03.03 Мобильные и сетевые технологии

Выполнил:  
студент 2 курса ИКТ группа К3241  
Сизей Омар

Проверила:  
Говорова Марина Михайловна

Санкт-Петербург 2021/2022

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

### **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.

### **Вариант 4. БД «Учет выполнения заданий»**

Описание предметной области: Сотрудники организации выполняют проекты. Проекты состоят из нескольких заданий. Каждый сотрудник может участвовать в одном или нескольких проектах, или временно не участвовать ни в каких проектах. Над каждым проектом может работать несколько сотрудников отделов, или временно проект может быть приостановлен, тогда над ним не работает ни один сотрудник. Над каждым заданием в проекте работает ровно один сотрудник. Каждый сотрудник числится в одном отделе.

БД должна содержать следующий минимальный набор сведений: Номер сотрудника. Фамилия сотрудника. Имя сотрудника. Отчество сотрудника. Должность сотрудника. Оклад сотрудника. Название организации. Номер организации. Адрес организации. Номер телефона отдела. Номер отдела. Название отдела. Код проекта. Название проекта. Сроки выполнения проекта. Руководитель проекта. Название организации. Номер организации. Адрес организации. Номер задания. Дата начала выполнения задания. Срок выполнения задания. Отметка о выполнении задания. Дата контроля выполнения задания. Причина невыполнения задания.

### **Схема:**



= project.id\_organisation and task.person\_number = employee.person\_number and  
employee.id\_division = division.id\_division

order by task.number\_task

ProjectDB/postgres@PostgreSQL 14

Query Editor   Query History

```
1 select task.number_task, project.code_project, organisation.id_organisation, employee.name, employee.last_name, division.name_division
2
3 from public.task, public.project, public.organisation, public.employee, public.division
4
5 where task.code_project = project.code_project and organisation.id_organisation = project.id_organisation and task.person_number = employee.person_number
6
7 order by task.number_task
```

Data Output   Explain   Messages   Notifications

	number_task integer	code_project integer	id_organisation integer	name character varying (20)	last_name character varying (20)	name_division character varying (20)
1	1122	1077	388	Katya	Ivanov	Cleaners
2	1133	1088	299	Alena	Petrov	Cleaners
3	1144	1099	100	Ivan	Smirnov	Finance

- Составить список проектов, работа над которыми была начата больше месяца назад.

select code\_project, name\_project, date\_start from public.project

where current\_date - date\_start > 30 order by code\_project

ProjectDB/postgres@PostgreSQL 14

Query Editor   Query History

```

1 select code_project, name_project, start_date from public.project
2 where current_date - start_date < 30 order by code_project

```

Data Output   Explain   Messages   Notifications

	code_project [PK] integer	name_project character varying (20)	start_date date
1	1077	Clean Geoizol	2022-05-30
2	1088	Clean Brandy	2022-05-25
3	1099	Clean Tiger	2022-05-20

- Вывести список сотрудников, оклад которых превышает средний оклад сотрудников своего отдела.

```
select employee.name, employee.last_name, post.salary from
public.employee, public.post
```

```
where post.salary > (select avg(post.salary) from public.post where
employee.id_division = employee.id_division)
```

```
1 select employee.name, employee.last_name, post.salary from public.employee, public.post
2
3 where post.salary > (select avg(post.salary) from public.post where employee.id_division = employee.id_division)
4
```

Data Output Explain Messages Notifications

	name character varying (20)	last_name character varying (20)	salary integer
1	Katya	Ivanov	80000
2	Alena	Petrov	80000
3	Ivan	Smirnov	80000
4	Bandu	camara	80000
5	Katya	Ivanov	90000
6	Alena	Petrov	90000
7	Ivan	Smirnov	90000
8	Bandu	camara	90000

- Найти отдел, работающий над максимальным количеством проектов.  
Тут в моих данных каждый отдел занят одним проектом, я сделал запрос который возвращает отдел, работающий над максимальным количеством заданий

```
select d.name_division, count(t.number_task)
```

```
from public.division as d, public.project as p, public.task as t, public.employee as e
  where t.code_project = p.code_project and e.person_number = t.person_number and
e.id_division = d.id_division
group by d.name_division
```

```
having count(t.number_task) = (select count(t.number_task)
```

```
from public.division as d, public.project as p, public.task as t, public.employee as e
  where t.code_project = p.code_project and e.person_number = t.person_number and
e.id_division = d.id_division
group by d.name_division order by count(t.number_task) desc limit 1)
```

ProjectDB/postgres@PostgreSQL 14 ▾

Query Editor   Query History

```
1 select d.name_division, count(t.number_task)
2 from public.division as d, public.project as p, public.task as t,
3 public.employee as e
4 where t.code_project = p.code_project and e.person_number = t.person_number
5 and e.id_division = d.id_division
6 group by d.name_division
7 having count(t.number_task) = (select count(t.number_task)
8 from public.division as d, public.project as p, public.task as t,
9 public.employee as e
10 where t.code_project = p.code_project and e.person_number = t.person_number
11 and e.id_division = d.id_division
12 group by d.name_division order by count(t.number_task) desc limit 1)
```

Data Output   Explain   Messages   Notifications

	name_division character varying (20)	count bigint
1	Cleaners	2

- Составить список сотрудников, проектов, заданий, в выполнении которых они участвуют и дат предполагаемого выполнения ими заданий. Учесть сотрудников, не участвующих в проектах.

```
SELECT employee.name as "Имя исполнителя", employee.last_name as "Фамилия
исполнителя", project.name_project as "Проект",
task.number_task as "Задание",
task.doing_date as "Дата выполнения задания" FROM public.task
INNER JOIN public.project ON project.code_project = task.code_project
RIGHT JOIN public.employee ON task.person_number = employee.person_number
INNER JOIN public.division ON employee.id_division = division.id_division ORDER
BY task.number_task
```

Query Editor

Query History

```

1  SELECT employee.name as "Имя исполнителя", employee.last_name as
2  "Фамилия исполнителя", project.name_project as "Проект", task.number_task
3  as "Задание",
4  task.doing_date as "Дата выполнения задания" FROM public.task
5  INNER JOIN public.project ON project.code_project = task.code_project
6  RIGHT JOIN public.employee ON task.person_number =
7  employee.person_number INNER JOIN public.division ON
8  employee.id_division = division.id_division ORDER BY task.number_task

```

Data Output

Explain

Messages

Notifications

	Имя исполнителя character varying (20)	Фамилия исполнителя character varying (20)	Проект character varying (20)	Задание integer	Дата выполнения задания date
1	Katya	Ivanov	Clean Geoizol	1122	2022-05-22
2	Alena	Petrov	Clean Brandy	1133	2022-05-28
3	Ivan	Smirnov	Clean Tiger	1144	2022-05-30
4	Omar	Ceesay	[null]	[null]	[null]
5	Sorbon	Kuznetsov	[null]	[null]	[null]

- Составить список сотрудников, не выполнивших задания в срок с указанием проектов и заданий, которые они должны были выполнить и количества дней просрочки выполнения заданий.

```
select employee.name, employee.last_name, project.name_project,
task.number_task, (current_date - task.doing_date) as days_overdue
from public.employee, public.project, public.task
where task.check_done not in ('yes') and task.doing_date < current_date and
employee.person_number = task.person_number and project.code_project =
```



task.code\_project

- Составить список проектов, в выполнении которого участвует более трех человек.

```
select project.name_project, count(employee.person_number) as cnt from
public.project, public.employee, public.task where
employee.person_number = task.person_number and task.code_project =
project.code_project
group by project.name_project
having count(employee.person_number) > 0 order by
count(employee.person_number) DESC
```

ProjectDB/postgres@PostgreSQL 14

Query Editor Query History

```
1 select project.name_project, count(employee.person_number) as cnt from
2 public.project, public.employee, public.task where
3 employee.person_number = task.person_number and task.code_project =
4 project.code_project
5 group by project.name_project
6 having count(employee.person_number) > 0 order by
7 count(employee.person_number) DESC
```

Data Output Explain Messages Notifications

	name_project character varying (20)	cnt bigint
1	Clean Tiger	1
2	Clean Geoizol	1
3	Clean Brandy	1

## Задание 2. Создать представления:

- для руководителей проектов, содержащее сведения об исполнителях, отделах, сроках выполнения заданий, включенных в проект.

```
create view public.management as
SELECT employee.name,
employee.last_name,
division.name_division,
project.name_project,
task.number_task,
task.doing_date
FROM employee,
division,
project,
task
WHERE employee.id_division = division.id_division AND employee.person_number =
task.person_number AND task.code_project = project.code_project;
```

```
1 select * from public.management
```

Data Output Explain Messages Notifications

	name character varying (20)	last_name character varying (20)	name_division character varying (20)	name_project character varying (20)	number_task integer	doing_date date
1	Katya	Ivanov	Cleaners	Clean Geoizol	1122	2022-05-22
2	Alena	Petrov	Cleaners	Clean Brandy	1133	2022-05-28
3	Ivan	Smirnov	Finance	Clean Tiger	1144	2022-05-30

- список проектов, срок выполнения которых истекает сегодня и которые включают больше трех невыполненных заданий.

```
create view public.project_due
```

```
as SELECT project.name_project, count(task.number_task) AS
```

```
count
```

```
FROM project, task
```

```
WHERE project.code_project = task.code_project AND
```

```
task.check_done::text = 'no'::text AND CURRENT_DATE =
```

```
task.doing_date
```

```
GROUP BY project.code_project
```

```
HAVING count(task.number_task) >= 3
```

```
ORDER BY (count(task.number_task)) DESC;
```

```
1 select * from public.project_due
```

Data Output Explain Messages Notifications

name_project character varying (20)	count bigint
----------------------------------------	-----------------

### Задание 3 Запросы DELETE UPDATE INSERT

Запрос DELETE:

Удалить сотрудника, который не имеет задания

До запроса:

The screenshot shows a database query editor interface. At the top, there are two tabs: "Query Editor" and "Query History". The "Query Editor" tab is active, displaying a SQL query across seven lines. The query is a DELETE statement that removes records from the 'public.employee' table where the 'person\_number' is not in a subquery. The subquery selects 'person\_number' from 'public.employee' and 'public.task' where they are not equal. Below the query editor, there are four tabs: "Data Output", "Explain", "Messages", and "Notifications". The "Messages" tab is active, showing the execution result: "DELETE 2" and "Query returned successfully in 282 msec."

```
1
2
3
4 delete from public.employee where person_number not in (
5 select task.person_number from public.employee, public.task where
6 employee.person_number <> task.person_number)
7
```

DELETED 2

Query returned successfully in 282 msec.

Запрос:

```
delete from public.employee where person_number not in (
select task.person_number from public.employee, public.task where
employee.person_number <> task.person_number)
```


После запроса:

The screenshot shows a PostgreSQL Query Editor interface. At the top, the connection is 'ProjectDB/postgres@PostgreSQL 14'. Below the connection bar, there are tabs for 'Query Editor' and 'Query History'. The 'Query Editor' tab is active, displaying a SQL query: `select * from public.employee`. Below the query editor, there are tabs for 'Data Output', 'Explain', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table with 7 columns: `person_number` (integer), `last_name` (character varying (20)), `name` (character varying (20)), `middle_name` (character varying (20)), `id_division` (integer), and `id_post` (integer). The table contains 3 rows of data.

	<code>person_number</code> [PK] integer	<code>last_name</code> character varying (20)	<code>name</code> character varying (20)	<code>middle_name</code> character varying (20)	<code>id_division</code> integer	<code>id_post</code> integer
1	1	Ivanov	Katya	Eduadovna	11	123
2	2	Petrov	Alena	Yuryevna	11	234
3	3	Smirnov	Ivan	Eduardovich	22	456

Запрос UPDATE:

Повысить зарплату всем, у кого она меньше 80000, на 20000 До update:


ProjectDB/postgres@PostgreSQL 14

Query Editor
Query History

```

1  select * from public.post
2
3
4
5
6

```

Data Output
Explain
Messages
Notifications

	id_post [PK] integer	salary integer	name_post character varying (20)
1	123	30000	Cleaner
2	234	35000	Commercial cleaner
3	456	45000	Full time cleaner
4	567	60000	Manager
5	890	70000	Director

Запрос:

update public.post set salary = salary + 20000 WHERE salary IN (  
SELECT salary FROM public.post WHERE salary < 80000)После запроса:

ProjectDB/postgres@PostgreSQL 14

Query Editor

Query History

1

2

3

4

5

6

7

```
select * from public.post
```

Data Output

Explain

Messages

Notifications

	<div>id_post</div> <div>[PK] integer</div>	<div>salary</div> <div>integer</div>	<div>name_post</div> <div>character varying (20)</div>
1	123	50000	Cleaner
2	234	55000	Commercial cleaner
3	456	65000	Full time cleaner
4	567	80000	Manager
5	890	90000	Director

16



Запрос INSERT:

Добавление новых сотрудников

До запроса:

The screenshot shows a PostgreSQL Query Editor interface. At the top, the database connection is 'ProjectDB/postgres@PostgreSQL 14'. Below the connection bar, there are tabs for 'Query Editor' and 'Query History'. The 'Query Editor' tab is active, displaying a SQL query: `select * from public.employee`. The query is numbered 1 through 6 on the left margin. Below the query editor, there are tabs for 'Data Output', 'Explain', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table with 6 columns: `person_number` (integer, PK), `last_name` (character varying (20)), `name` (character varying (20)), `middle_name` (character varying (20)), `id_division` (integer), and `id_post` (integer). The table contains 3 rows of data.

person_number [PK] integer	last_name character varying (20)	name character varying (20)	middle_name character varying (20)	id_division integer	id_post integer
1	Ivanov	Katya	Eduardovna	11	123
2	Petrov	Alena	Yuryevna	11	234
3	Smirnov	Ivan	Eduardovich	22	456

```
insert into public.employee (person_number, last_name, name, middle_name, id_division, id_post)
values (4, 'camara', 'Bandu', 'Janneh', 33, 890);
```

После запроса:

 ProjectDB/postgres@PostgreSQL 14 ▾

Query Editor

Query History

1

2

3

4

5

6

7

8

9

10

11

12

```
1 insert into public.employee (person_number, last_name, name, middle_name, id_division, id_post)
2 values ( 4, 'camara', 'Bandu', 'Janneh', 33, 890);
3
4
5
6
7
8
9
10
11
12
```

Data Output

Explain

Messages

Notifications

```
INSERT 0 1
```

Query returned successfully in 408 msec.

ProjectDB/postgres@PostgreSQL 14

Query Editor    Query History

```

1  select * from public.employee
2
3
4
5
6
7
8
9
10
11
12

```

Data Output    Explain    Messages    Notifications

	person_number <small>[PK] integer</small>	last_name <small>character varying (20)</small>	name <small>character varying (20)</small>	middle_name <small>character varying (20)</small>	id_division <small>integer</small>	id_post <small>integer</small>
1	1	Ivanov	Katya	Eduadovna	11	123
2	2	Petrov	Alena	Yuryevna	11	234
3	3	Smirnov	Ivan	Eduardovich	22	456
4	4	camara	Bandu	Janneh	33	890



ProjectDB/postgres@PostgreSQL 14

Query Editor

Query History

Show queries generated internally by pgAdmin? Yes

SELECT \* FROM public.employee ORDER BY person\_number ASC

01:24:12

▶ insert into public.employee (person\_number, last\_name, name, m...

01:22:36

▶ insert into public.employee (person\_number, last\_name, name, m...

01:22:00

▶ insert into public.employee (person\_number, last\_name, name, m...

01:21:49

SELECT \* FROM public.employee ORDER BY person\_number ASC

01:20:25

▶ insert into public.employee (person\_number, last\_name, name, m...

01:19:00

▶ insert into public.employee (person\_number, last\_name, name, m...

01:17:30

▶ insert into public.employee (person\_number, last\_name, name, m...

01:09:19

▶ insert into public.employee (person\_number, last\_name, name, m...

01:07:59

▶ insert into public.employee (person\_number, last\_name, name, m...

01:07:16

▶ insert into public.employee (person\_number, last\_name, name, m...

01:06:18

▶ insert into public.employee (person\_number, last\_name, name, m...

01:05:12

▶ insert into public.employee (person\_number, last\_name, name, m...

01:03:39

▶ insert into public.employee (person\_number, last\_name, name, m...

01:02:12

▶ insert into public.employee (person\_number, last\_name, name, m...

01:00:35

Error Message

INSERT has more expressions than target columns

5/23/2022 1:09:19 AM

0

145 msec

Date

Rows Affected

Duration

Copy

Copy to Query Editor

insert into public.employee (person\_number, last\_name, name, m...

values ( 4, 'camara', 'Bandu', 'Janneh', 'Administrative', 890,

Mes < >

Messages

ERROR: INSERT has more expressions than target columns

LINE 2: ...camara', 'Bandu', 'Janneh', 'Administrative', 890, (select

SQL state: 42601

Character: 160

Use

Activate Windows

Go to Settings to activate Windows

## **Задание 5 Индексы**

Создание простого индекса для 2ого запроса

```
create index basic ON public.project(code_project)
```

```
1 create index basic ON public.project(code_project)
```

Data Output Explain Messages Notifications

CREATE INDEX

Query returned successfully in 463 msec.

Создание составного индекса для 7ого запроса:

```
create unique index obtain_index on public.project(code_project,  
name_project)
```

The screenshot shows the Query Editor interface with the following components:

- Top Bar:** Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, ProjectDB/postgres@PostgreSQL 14.
- Toolbar:** Contains icons for file operations, search, and execution.
- Query Editor:** Displays the SQL query:

```
1 create unique index obtain_index on public.project(code_project,  
2 name_project)
```
- Bottom Bar:** Data Output, Explain, Messages, Notifications.
- Messages Panel:** Shows the output:

```
CREATE INDEX  
  
Query returned successfully in 555 msec.
```

**Запросы с индексами:**

2ой запрос



## Удаление индексов:

ProjectDB/postgres@PostgreSQL 14 ▾

Query Editor Query History

1 drop index public.basic

Data Output Explain Messages Notifications

DROP INDEX

Query returned successfully in 334 msec.

Query Editor Query History

1 drop index public.obtain\_index

Data Output Explain Messages Notifications

DROP INDEX

Query returned successfully in 221 msec.

## **Вывода:**

SQL — очень удобный способ быстро настроить ваши записи в базе данных. Вы можете чередовать, добавлять или удалять информацию, делать подборки в соответствии с нужными вам стандартами и вычислять важную информацию по важным характеристикам.

Сравнивая время выполнения запросов с индексами и без, можно сделать вывод, что запросы с индексами выполнялись немного медленнее из-за малого количества информации внутри стола.