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

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

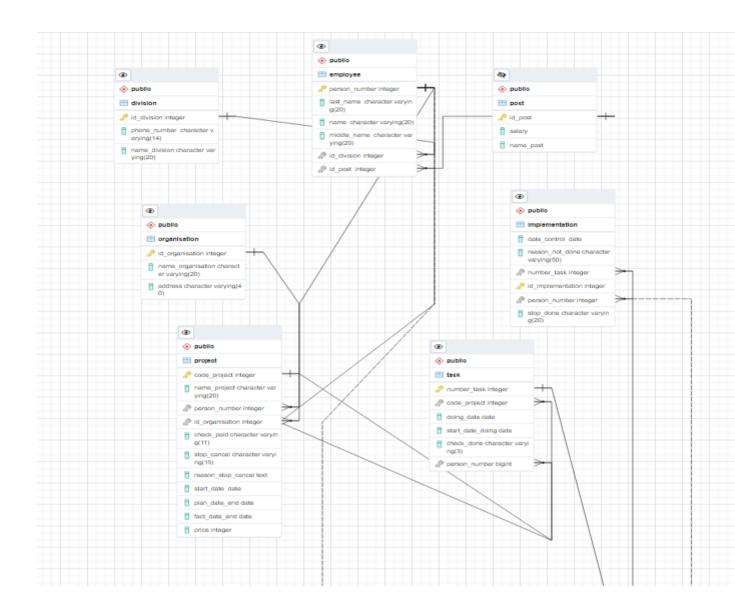
Практическое задание:

- **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 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:



Задание 1. Создайте запросы:

• Составить список всех заданий каждого проекта с указанием организаций, отделов и исполнителей, занятых в его выполнении.

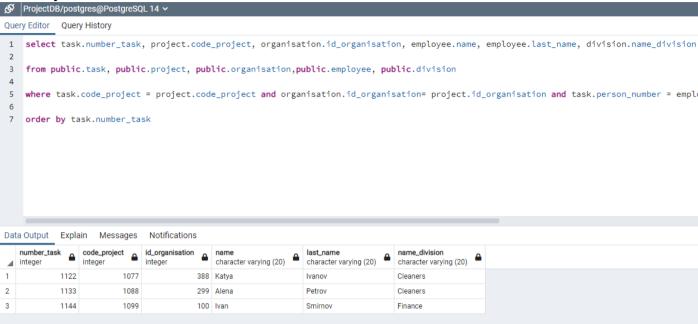
select task.number_task, project.code_project, organisation.id_organisation, employee.name, employee.last_name, employee.name_post, division.name_division

from public.task, public.project, public.organisation,public.employee, public.division

where task.code_project = project.code_project and organisation.id_organisation

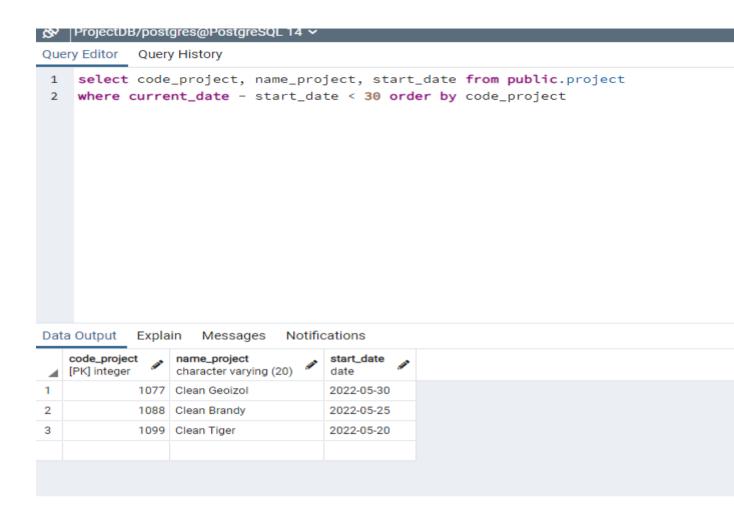
= project.id_organisation and task.person_number = employee.person_numberand employee.id_division = division.id_division

order by task.number_task



• 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



• 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)

```
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 = employee.id_division
```

Data Output Explain Messages Notifications			
4	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
^	D		00000

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

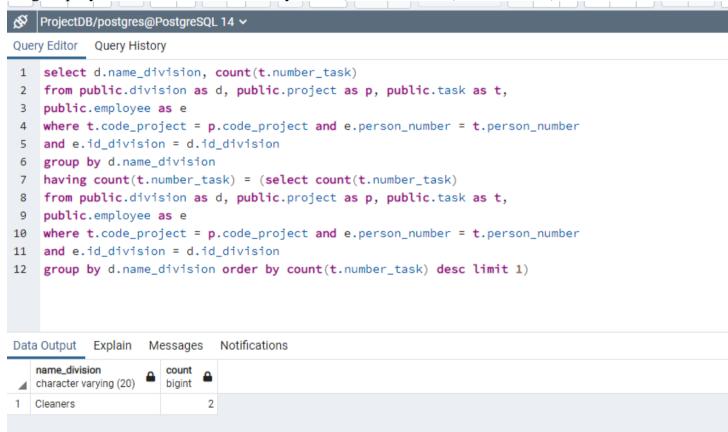
select d.name_division, count(t.number_task)

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)

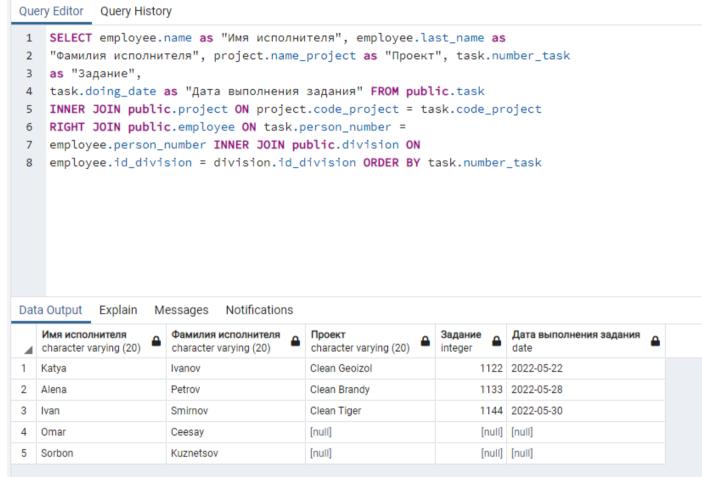
from public.division as d, public.project as p, public.task as t, public.employee as e where t.code_project = p.code_project and e.person_number = t.person_number and e.id_division = d.id_division

group by d.name_division order by count(t.number_task) desc limit 1)



• 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



• 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 >

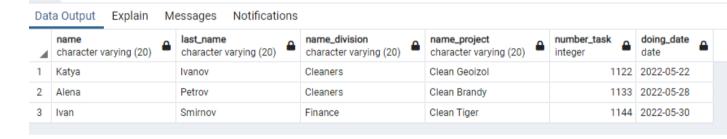
Query Editor Query History select project.name_project, count(employee.person_number) as cnt from 1 public.project, public.employee, public.task where 2 3 employee.person_number = task.person_number and task.code_project = project.code_project 4 group by project.name_project 5 having count(employee.person_number) > 0 order by 6 count(employee.person_number) DESC 7 Notifications Data Output Explain Messages name_project cnt character varying (20) bigint 1 Clean Tiger 1 2 Clean Geoizol 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;
```



• 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

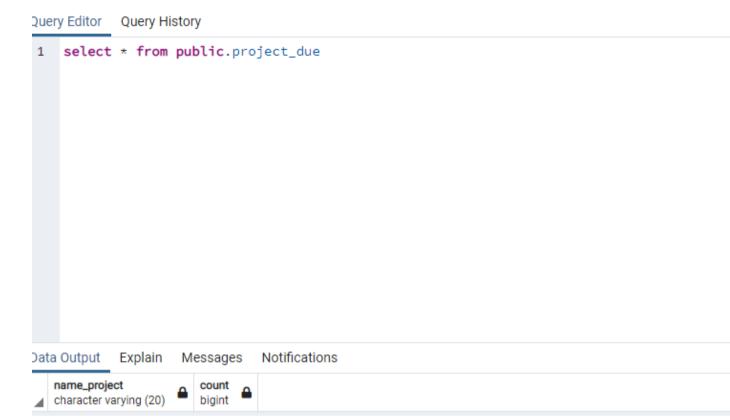
task.check_done::text = 'no'::text AND CURRENT_DATE =

task.doing_date

GROUP BY project.code_project

HAVING count(task.number_task) >= 3

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



Задание 3 Запросы DELETE UPDATE INSERT

Запрос DELETE:

Удалить сотрудника, который не имеет задания

До запроса:

```
Query Editor Query History

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)

Data Output Explain Messages Notifications

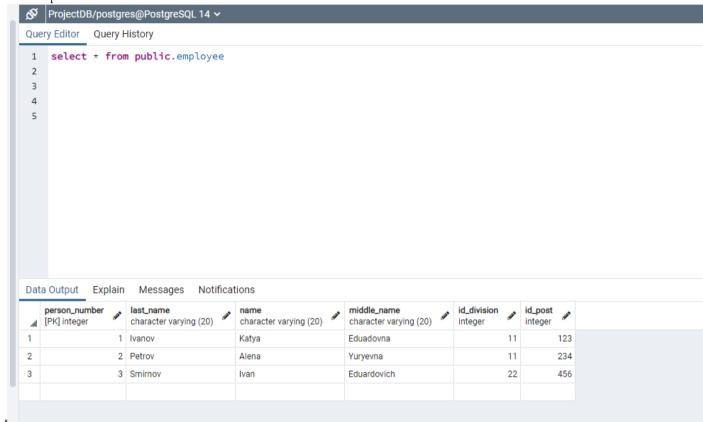
DELETE 2

Query returned successfully in 282 msec.
```

Запрос:

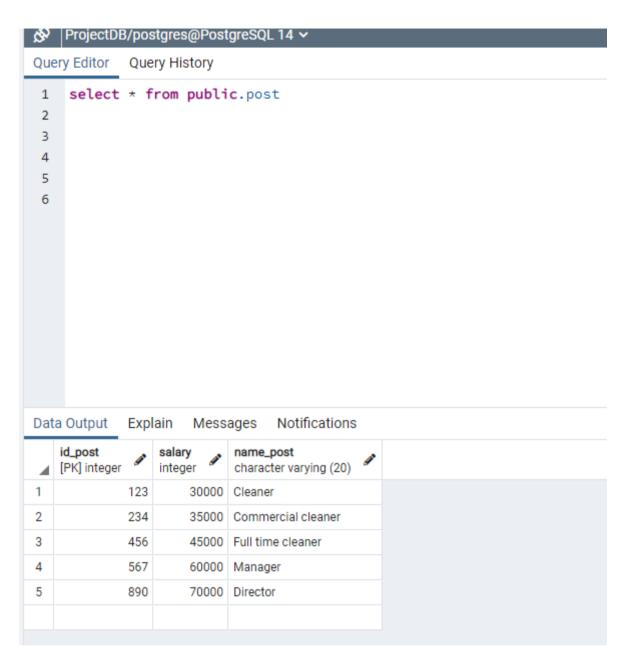
delete from public.employee where person_number not in (select task.person_number from public.employee, public.task where employee.person_number <> task.person_number)

После запроса:



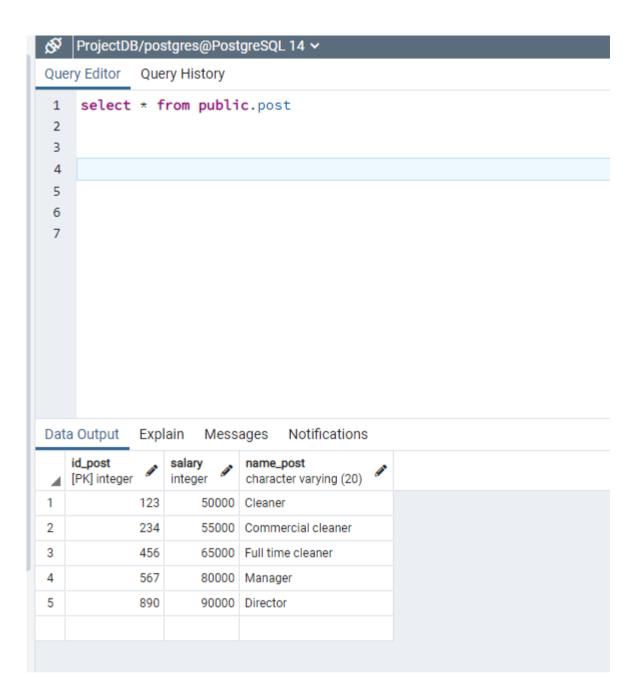
Запрос UPDATE:

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



Запрос:

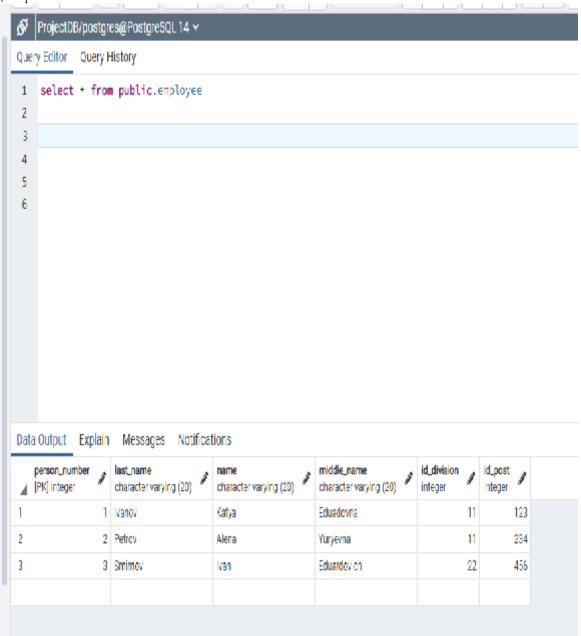
update public.post set salary = salary + 20000 WHERE salary IN (SELECT salary FROM public.post WHERE salary < 80000) После запроса:



Запрос INSERT:

Добавление новых сотрудников

До запроса:



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

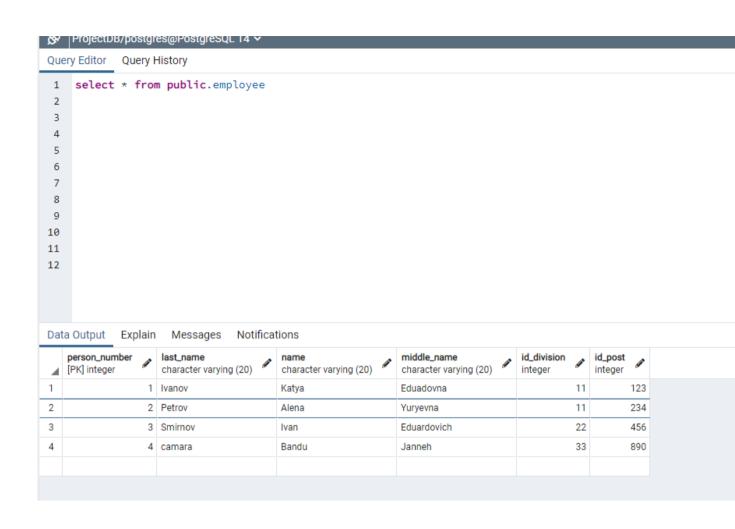
После запроса:

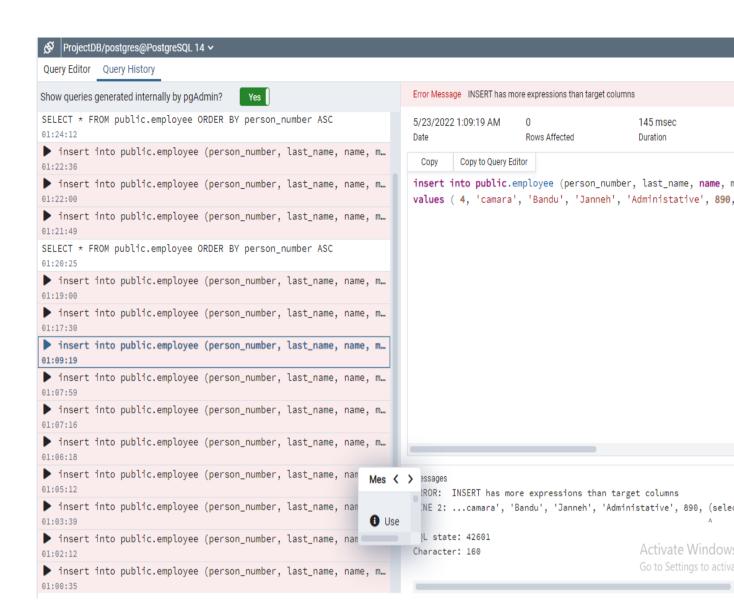
```
        Ø ProjectDB/postgres@PostgreSQL 14 ▼

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

INSERT 0 1

Query returned successfully in 408 msec.





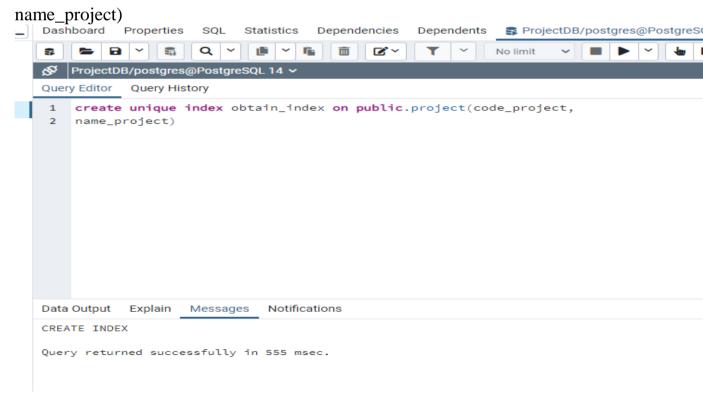
Задание 5 Индексы

Создание простого индекса для 20го запроса create index basic ON public.project(code_project)



Создание составного индекса для 7ого запроса:

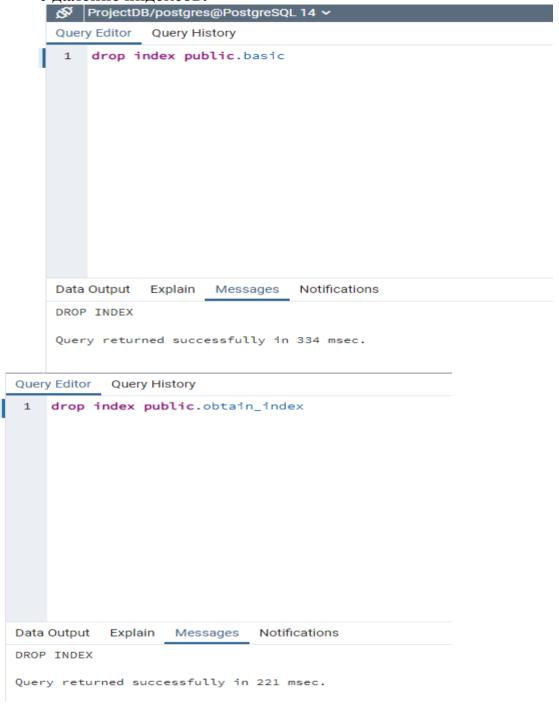
create unique index obtain_index on public.project(code_project,



Запросы с индексами:

2ой запрос

Удаление индексов:



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.