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

Проверил:
Говорова М.М. _____
Дата: «__» _____20__г.
Оценка _____

Выполнил:
студент группы К3240 Кулагина С.В.

Санкт-Петербург 2021/2022

Цель работы: овладеть практическими навыками создания представлений 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 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.

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 first_project.task, first_project.project, first_project.organisation,
first_project.employee, first_project.division
```

where task.code_project = project.code_project and organisation.id_organisation = project.id_organisation and task.person_number = employee.person_number and employee.id_division = division.id_division

order by task.number_task

postgres/postgres@PostgreSQL 13

История запросов Scratch Pad Query Editor

```
1 select task.number_task, project.code_project, organisation.id_organisation, employee.name, employee.last_name, employee.name_post, 
2
3 from first_project.task, first_project.project, first_project.organisation, first_project.employee, first_project.division
4
5 where task.code_project = project.code_project and organisation.id_organisation = project.id_organisation and task.person_number = e
6
7 order by task.number_task
8
```

Результат План выполнения Сообщения Notifications

	number_task integer	code_project integer	id_organisation integer	name character varying (20)	last_name character varying (20)	name_post character varying (20)	name_division character varying (20)
1	10	7410	12210	Julia	Komarova	analyst	analitical division
2	11	963	12211	Anastasia	Koshkarova	economist	economic division
3	12	789	12211	Svetlana	Kulagina	main economist	economic division

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

select code_project, name_project, date_start from first_project.project

where current_date - date_start > 30 order by code_project

```
select code_project, name_project, date_start from first_project.project

where current_date - date_start > 30 order by code_project
```

Результат | План выполнения | Сообщения | Notifications

	code_project [PK] integer	name_project character varying (20)	date_start date
1	789	Queens' main money	2022-02-24
2	963	Queens' money	2022-02-26
3	7410	Kings' Bid Data	2022-02-25

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

  select employee.name, employee.last_name, post.salary from first_project.employee, first_project.post

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

postgres/postgres@PostgreSQL 13

История запросов Scratch Pad Query Editor

```
1 select employee.name, employee.last_name, post.salary from first_project.employee, first_project.post
2
3 where post.salary > (select avg(post.salary) from first_project.post where employee.id_division = employee.id_division)
4
5
6
```

Результат План выполнения Сообщения Notifications

	name character varying (20)	last_name character varying (20)	salary integer
1	Julia	Komarova	50000
2	Anastasia	Koshkarova	50000
3	Svetlana	Kulagina	50000

- Найти отдел, работающий над максимальным количеством проектов.

Т.к 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 first_project.division as d, first_project.project as p, first_project.task as t,

first_project.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)

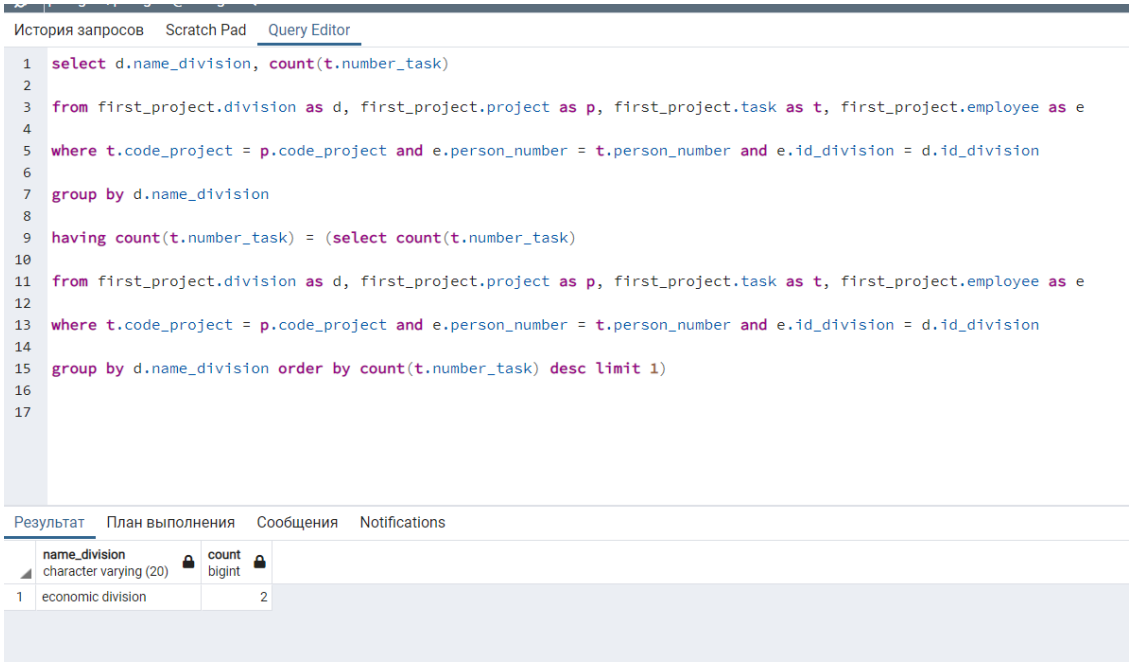
from first_project.division as d, first_project.project as p, first_project.task as t,

first_project.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 first_project.task INNER JOIN first_project.project ON project.code_project = task.code_project RIGHT JOIN first_project.employee ON task.person_number = employee.person_number INNER JOIN first_project.division ON employee.id_division = division.id_division ORDER BY task.number_task

```
1 SELECT employee.name as "Имя исполнителя", employee.last_name as "Фамилия исполнителя", project.name
2 task.doing_date as "Дата выполнения задания" FROM first_project.task
3 INNER JOIN first_project.project ON project.code_project = task.code_project
4 RIGHT JOIN first_project.employee ON task.person_number = employee.person_number INNER JOIN first_pr
5
```

Результат | План выполнения | Сообщения | Notifications

	Имя исполнителя character varying (20)	Фамилия исполнителя character varying (20)	Проект character varying (20)	Задание integer	Дата выполнения задания date
1	Julia	Komarova	Kings' Bid Data	10	2022-03-05
2	Anastasia	Koshkarova	Queens' money	11	2022-03-05
3	Svetlana	Kulagina	Queens' main money	12	2022-03-10
4	Ivan	Ivanov	[null]	[null]	[null]
5	Anna	Ivanova	[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 first_project.employee, first_project.project, first_project.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

```
select employee.name, employee.last_name, project.name_project, task.number_task, (current_dat

from first_project.employee, first_project.project, first_project.task

where task.check_done not in ('yes') and task.doing_date < current_date and employee.person_nu
```

Результат | План выполнения | Сообщения | Notifications

	name character varying (20)	last_name character varying (20)	name_project character varying (20)	number_task integer	days_overdue integer
1	Julia	Komarova	Kings' Bid Data	10	40
2	Anastasia	Koshkarova	Queens' money	11	40

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

select project.name_project, count(employee.person_number) as cnt from first_project.project, first_project.employee, first_project.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) > 3 order by count(employee.person_number) DESC

```
select project.name_project, count(employee.person_number) as cnt from first_project.project, first_project.employee, first_project.
group by project.name_project
having count(employee.person_number) > 3 order by count(employee.person_number) DESC
```

Результат | План выполнения | Сообщения | Notifications

	name_project character varying (20)	cnt bigint
1	Queens' main money	4
2	Queens' money	4

Задание 2. Создать представления:

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

create view first_project.management as select employee.name, employee.last_name,
division.name_division, project.name_project,
task.number_task, task.doing_date

from first_project.employee, first_project.division, first_project.project, first_project.task

where employee.id_division = division.id_division and employee.person_number = task.person_number and task.code_project = project.code_project

postgres/postgres@PostgreSQL 13

История запросов Scratch Pad Query Editor

```
1  select * from first_project.management
```

Результат План выполнения Сообщения 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	Julia	Komarova	analitical division	Kings' Bid Data	10	2022-03-05
2	Anastasia	Koshkarova	economic division	Queens' money	11	2022-03-05
3	Svetlana	Kulagina	economic division	Queens' main money	12	2022-03-10

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

  create view first_project.project_view

  as select project.name_project, count(task.number_task) from

  first_project.project, first_project.task

  where project.code_project = task.code_project and task.check_done = 'no' and

  current_date = task.doing_date

  group by project.code_project

  having count(task.number_task) >= 3 order by count(task.number_task) DESC

postgres/postgres@PostgreSQL 13

История запросов | Scratch Pad | Query Editor

```
1  select * from first_project.project_view
```

Результат | План выполнения | Сообщения | Notifications

	name_project character varying (20)	count bigint
1	Queens' main money	3
2	Queens' money	3

Задание 3 Запросы DELETE UPDATE INSERT

Запрос DELETE:

Удалить сотрудника, который не имеет задания

До запроса:

```
1 select * from first_project.employee
```

Результат План выполнения Сообщения Notifications

	person_number [PK] integer	last_name character varying (20)	name character varying (20)	middle_name character varying (20)	name_division character varying (20)	id_division integer	id_post integer	name_post character varying (20)
1	122	Komarova	Julia	Mihailovna	analitical division	12	1	analyst
2	121	Koshkarova	Anastasia	Pavlovna	economic division	11	2	economist
3	120	Kulagina	Svetlana	Victorovna	economic division	11	3	main economist
4	111	Ivanov	Ivan	Ivanovich	design	13	7	project_manager
5	222	Ivanova	Anna	Ivanovna	analitical division	12	8	supervisor
6	1	Mers	Goga	[null]	economic division	11	2	economist
7	2	Sera	Serafim	[null]	economic division	11	2	economist
8	3	Kors	Maximilian	[null]	analitical division	12	1	analyst
9	4	Ponomarev	Dmitrii	Alexandrovich	analitical division	12	1	analyst
10	5	Kovalev	Alexandr	[null]	design	13	3	main economist

Запрос:

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

После запроса:

postgres/postgres@PostgreSQL 13

История запросов Scratch Pad Query Editor

```
1  select * from first_project.employee
```

Результат План выполнения Сообщения Notifications

	person_number [PK] integer	last_name character varying (20)	name character varying (20)	middle_name character varying (20)	name_division character varying (20)	id_division integer	id_post integer	name_post character varying (20)
1	122	Komarova	Julia	Mihailovna	analitical division	12	1	analyst
2	121	Koshkarova	Anastasia	Pavlovna	economic division	11	2	economist
3	120	Kulagina	Svetlana	Victorovna	economic division	11	3	main economist
4	111	Ivanov	Ivan	Ivanovich	design	13	7	project_manager
5	222	Ivanova	Anna	Ivanovna	analitical division	12	8	supervisor

Запрос UPDATE:

Повысить зарплату всем, у кого она меньше 80000, на 20000

До update:

postgres/postgres@PostgreSQL 13

История запросов | Scratch Pad | Query Editor

```
1 select * from first_project.post
```

Результат | План выполнения | Сообщения | Notifications

	id_post [PK] integer	name_post [PK] character varying (20)	salary integer
1	1	analyst	50000
2	2	economist	30000
3	3	main economist	40000
4	7	project_manager	20000
5	8	supervisor	30000

Запрос:

update first_project.post set salary = salary + 20000 WHERE salary IN (SELECT salary FROM first_project.post WHERE salary < 80000)

После запроса:

```
1  select * from first_project.post
```

Результат | План выполнения | Сообщения | Notifications

	id_post [PK] integer	name_post [PK] character varying (20)	salary integer
1	1	analyst	70000
2	2	economist	50000
3	3	main economist	60000
4	7	project_manager	40000
5	8	supervisor	50000

Запрос INSERT:

Добавление новых сотрудников

До запроса:

postgres/postgres@PostgreSQL 13

История запросов Scratch Pad Query Editor

```
1 select * from first_project.employee
```

Результат План выполнения Сообщения Notifications

	person_number [PK] integer	last_name character varying (20)	name character varying (20)	middle_name character varying (20)	name_division character varying (20)	id_division integer	id_post integer	name_post character varying (20)
1	122	Komarova	Julia	Mihailovna	analitical division	12	1	analyst
2	121	Koshkarova	Anastasia	Pavlovna	economic division	11	2	economist
3	120	Kulagina	Svetlana	Victorovna	economic division	11	3	main economist
4	111	Ivanov	Ivan	Ivanovich	design	13	7	project_manager
5	222	Ivanova	Anna	Ivanovna	analitical division	12	8	supervisor

После запроса:

postgres/postgres@PostgreSQL 13

История запросов Scratch Pad Query Editor

```
1 insert into first_project.employee (person_number, last_name, name, middle_name, name_division, id_division, id_post, name_post)
2 VALUES ( 777, 'Ponomarev', 'Alexandr', 'Dmitrievich', 'design', 13, 3, 'main economist')
3
```

Результат План выполнения Сообщения Notifications

```
INSERT 0 1

Запрос завершён успешно, время выполнения: 109 msec.
```

```
1  select * from first_project.employee
```

Результат | План выполнения | Сообщения | Notifications

	person_number [PK] integer	last_name character varying (20)	name character varying (20)	middle_name character varying (20)	name_division character varying (20)	id_division integer	id_post integer	name_post character varying (20)
1	122	Komarova	Julia	Mihailovna	analitical division	12	1	analyst
2	121	Koshkarova	Anastasia	Pavlovna	economic division	11	2	economist
3	120	Kulagina	Svetlana	Victorovna	economic division	11	3	main economist
4	111	Ivanov	Ivan	Ivanovich	design	13	7	project_manager
5	222	Ivanova	Anna	Ivanovna	analitical division	12	8	supervisor
6	777	Ponomarev	Alexandr	Dmitrievich	design	13	3	main economist

Задание 4 История запросов

postgres/postgres@PostgreSQL 13

История запросов | Scratch Pad | Query Editor

Show queries generated internally by pgAdmin? Да

Today - 14.04.2022

```
select * from first_project.employee
09:47:53
insert into first_project.employee (person_number, last_name…
09:47:20
insert into first_project.employee (person_number, last_name…
09:46:49
select * from first_project.employee
09:39:36
delete from first_project.employee where person_number not i…
09:38:46
delete from first_project.employee where person_number not i…
09:38:39
delete from first_project.employee where person_number not i…
09:38:29
BEGIN;
09:32:21
INSERT INTO first_project.employee ( person_number, last_nam…
09:32:21
COMMIT;
09:32:21
select * from first_project.employee
09:31:16
select * from first_project.post
09:28:11
```

14.04.2022 09:47:53	6	62 msec
Дата	Rows Affected	Продолжительность

Copy | Copy to Query Editor

```
select * from first_project.employee
```

Сообщения

```
Запрос выполнен успешно. Общее время выполнения: 62 msec.
обработано строк: 6.
```

История запросов | Scratch Pad | Query Editor

Show queries generated internally by pgAdmin? Да

```
08:40:08
INSERT INTO first_project.project ( name_project, name_organ…
08:40:08
ROLLBACK;
08:40:08
select * from first_project.project
08:37:31
select employee.name, employee.last_name, project.name_proje…
01:38:47
select employee.name, employee.last_name, project.name_proje…
01:37:03
select employee.name, employee.last_name, project.name_proje…
01:36:37
select employee.name, employee.last_name, project.name_proje…
01:36:32
select employee.name, employee.last_name, project.name_proje…
01:36:25
select employee.name, employee.last_name, project.name_proje…
01:36:05
select * from first_project.task
01:35:40
select employee.name, employee.last_name, project.name_proje…
01:34:34
SELECT employee.name as "Имя исполнителя", employee.last_nam…
01:27:18
```

14.04.2022 09:47:53	6	62 msec
Дата	Rows Affected	Продолжительность

Copy | Copy to Query Editor

```
select * from first_project.employee
```

Сообщения

```
Запрос выполнен успешно. Общее время выполнения: 62 msec.
обработано строк: 6.
```

Результат | План выполнения | Сообщения | Notifications

Задание 6 EXPLAIN

Без индексов 2ой запрос

План запроса:

postgres/postgres@PostgreSQL 13

История запросов | Scratch Pad | Query Editor

```
1 explain select code_project, name_project, date_start from first_project.project
2
3 where current_date - date_start > 30 order by code_project
4
5
```

Результат | План выполнения | Сообщения | Notifications

	QUERY PLAN text
1	Sort (cost=1.06..1.07 rows=1 width=66)
2	[...] Sort Key: code_project
3	[...] -> Seq Scan on project (cost=0.00..1.05 rows=1 width=66)
4	[...] Filter: ((CURRENT_DATE - date_start) > 30)

Время:

14.04.2022 09:56:27	2	59 msec
Дата	Rows Affected	Продолжительность

Без индекса 3ий запрос
План запроса:

История запросов Scratch Pad Query Editor

```
1 explain select employee.name, employee.last_name, post.salary
2 from first_project.employee, first_project.post
3 where post.salary > (select avg(post.salary) from first_project.post where employee.id_division = employee.id_division)
4
5
```

Результат План выполнения Сообщения Notifications

	QUERY PLAN text
1	Nested Loop (cost=0.00..11.70 rows=3 width=120)
2	[...] Join Filter: ((post.salary)::numeric > (SubPlan 1))
3	[...] -> Seq Scan on employee (cost=0.00..1.03 rows=3 width=120)
4	[...] -> Materialize (cost=0.00..1.04 rows=3 width=4)
5	[...] -> Seq Scan on post (cost=0.00..1.03 rows=3 width=4)
6	[...] SubPlan 1
7	[...] -> Aggregate (cost=1.04..1.05 rows=1 width=32)
8	[...] -> Result (cost=0.00..1.03 rows=3 width=4)

Время:

14.04.2022 10:24:02	10	35 msec
Дата	Rows Affected	Продолжительность

Copy Copy to Query Editor

Задание 7 Индексы
Создание простого индекса для 2ого запроса
create index my_pr ON first_project.project(code_project)

```
1 create index my_pr ON first_project.project(code_project)
2
```

Результат План выполнения Сообщения Notifications

```
CREATE INDEX

Запрос завершён успешно, время выполнения: 127 msec.
```

Создание составного индекса для 7ого запроса:

create unique index dou_index on first_project.project(code_project, name_project)

postgres/postgres@PostgreSQL 13

История запросов Scratch Pad Query Editor

```
1 create unique index dou_index on first_project.project(code_project, name_project)
2
```

Результат План выполнения Сообщения Notifications

```
CREATE INDEX

Запрос завершён успешно, время выполнения: 39 msec.
```

Запросы с индексами:

2ой запрос

План запроса:

Результат | План выполнения | Сообщения | Notifications

	QUERY PLAN text
1	Sort (cost=1.06..1.07 rows=1 width=66)
2	[...] Sort Key: code_project
3	[...] -> Seq Scan on project (cost=0.00..1.05 rows=1 width=66)
4	[...] Filter: ((CURRENT_DATE - date_start) > 30)

Время:

14.04.2022 10:43:59	4	51 msec
Дата	Rows Affected	Продолжительность

Copy | Copy to Query Editor

3ий запрос
План запроса:

Результат | План выполнения | Сообщения | Notifications

	QUERY PLAN text
1	Nested Loop (cost=0.00..11.70 rows=3 width=120)
2	[...] Join Filter: ((post.salary)::numeric > (SubPlan 1))
3	[...] -> Seq Scan on employee (cost=0.00..1.03 rows=3 width=120)
4	[...] -> Materialize (cost=0.00..1.04 rows=3 width=4)
5	[...] -> Seq Scan on post (cost=0.00..1.03 rows=3 width=4)
6	[...] SubPlan 1
7	[...] -> Aggregate (cost=1.04..1.05 rows=1 width=32)
8	[...] -> Result (cost=0.00..1.03 rows=3 width=4)
9	[...] One-Time Filter: (employee.id_division = employee.id_division)
10	[...] -> Seq Scan on post post_1 (cost=0.00..1.03 rows=3 width=4)

Время:

14.04.2022 10:47:49	10	67 msec
Дата	Rows Affected	Продолжительность

Удаление индексов:

История запросов Scratch Pad Query Editor

```
1 drop index first_project.my_pr
2
3
4
5
6
7
```

Результат План выполнения Сообщения Notifications

```
DROP INDEX

Запрос завершён успешно, время выполнения: 87 msec.
```

История запросов Scratch Pad Query Editor

```
1 drop index first_project.dou_index
2
3
4
5
6
7
```

Результат План выполнения Сообщения Notifications

```
DROP INDEX

Запрос завершён успешно, время выполнения: 67 msec.
```

Проверка удаления:

Результат План выполнения Сообщения Notifications

	schemaname name	tablename name	indexname name	tablespace name	indexdef text
1	first_project	project	project_pkey	[null]	CREATE UNIQUE INDEX project_pkey ON first_project.project USING btree (code_project)

Вывод:

SQL это очень удобная возможность быстро видоизменять

наши данные в базе данных. Мы можем изменять, добавлять или удалять данные, делать выборки по нужным нам критериям и считать нужные данные по необходимым характеристикам

Сравнив время выполнения запросов с индексами и без, можно сделать вывод, что с индексами запросы выполнялись немного медленнее из-за небольшого количества данных в таблице