

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 студент:

Алексеев Павел Алексеевич

Группа №K3243

Проверила:

Говорова Марина Михайловна

Санкт-Петербург
2022

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:

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 и просмотреть историю запросов.
4. 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.

Схема базы данных (Отели):

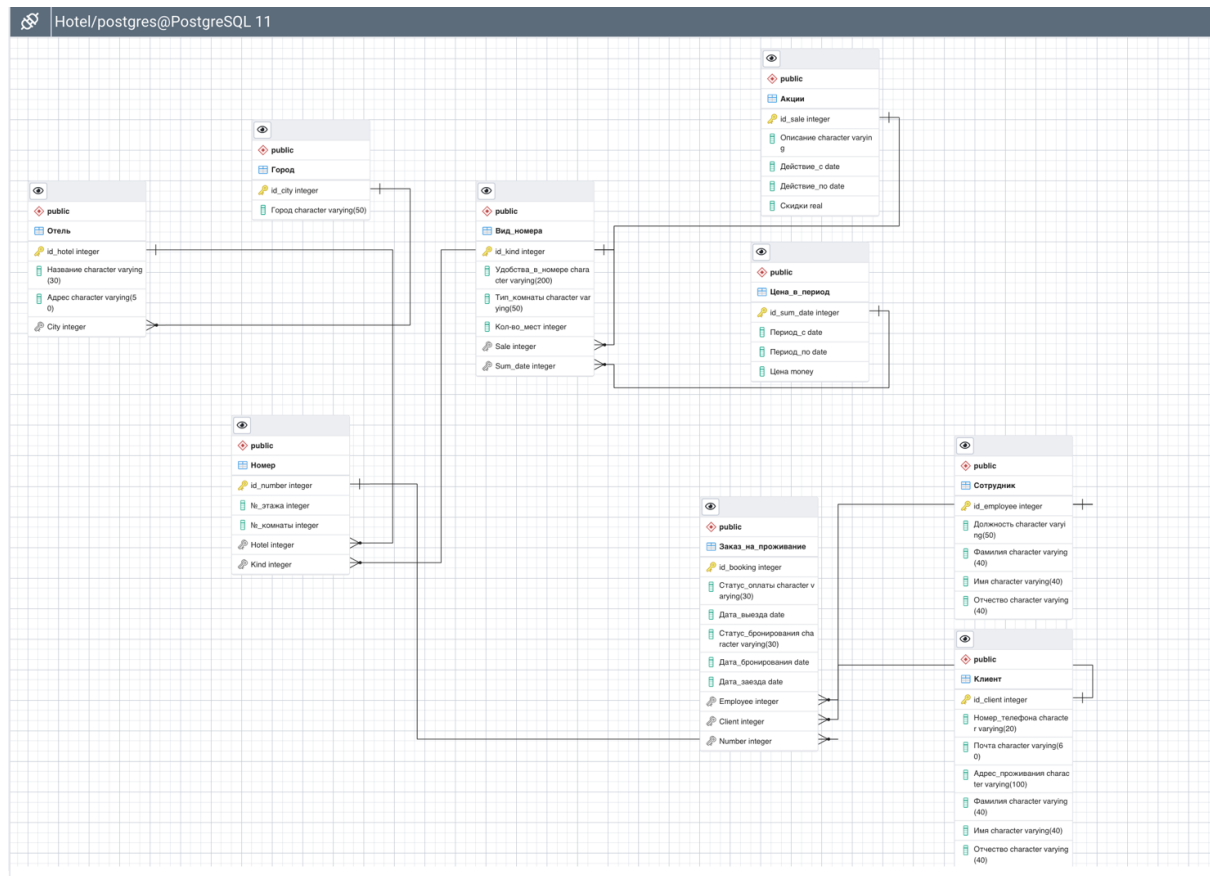


Рисунок 1. Схема базы данных.

Выполнение:

1. Запросы к базе данных. Выполнить запросы согласно индивидуальному заданию, часть 2.

- Составить список всех 2-местных номеров отелей, с ценой менее 2000\$, упорядочив данные в порядке уменьшения стоимости.

Hotel/postgres@PostgreSQL 11

Query Editor

История запросов

1

SELECT

"Вид_номера"."Тип_комнаты", "Цена_в_период"."Цена"

FROM

"Вид_номера", "Цена_в_период"

2

WHERE

"Цена_в_период".number_kind = "Вид_номера".id_kind AND "Цена_в_период"."Цена" < '2000' AND "Вид_номера"."Кол-во_мест" = 2

3

AND

"Цена_в_период"."Период_с" = '2022-05-01'

4

ORDER BY

"Цена_в_период"."Цена" DESC

Результат

План выполнения

Сообщения

Notifications

	Тип_комнаты character varying (50)	Цена money
1	двухместный полулюкс	\$1,000.00
2	Люкс	\$800.00
3	полу-люкс	\$600.00
4	стандарт	\$400.00

- Выбрать все записи регистрации постояльцев, которые выехали из отелей в течение двух последних недель.

Hotel/postgres@PostgreSQL 11

Query Editor

История запросов

```
1 SELECT * FROM "Заказ_на_проживание" WHERE "Статус_бронирования" = 'Выехал' AND "Дата_выезда" BETWEEN '2022-03-16' AND '2022-03-30'
```

Результат

План выполнения

Сообщения

Notifications

	id_booking [PK] integer	Статус_оплаты character varying (30)	Дата_выезда date	Статус_бронирования character varying (30)	Дата_бронирования date	Дата_звезда date	Employee integer	Client integer	Number integer
1	9	Оплачено	2022-03-20	Выехал	2022-03-10	2022-03-13		4	1
2	10	Оплачено	2022-03-23	Выехал	2022-03-14	2022-03-16		4	2
3	11	Оплачено	2022-03-25	Выехал	2022-03-18	2022-03-20		5	3
4	17	Оплачено	2022-03-27	Выехал	2022-03-11	2022-03-20		5	7
5	18	Оплачено	2022-03-27	Выехал	2022-03-13	2022-03-20		6	7
6	19	Оплачено	2022-03-25	Выехал	2022-03-12	2022-03-19		6	8
7	20	Оплачено	2022-03-26	Выехал	2022-03-15	2022-03-17		6	9
8	21	Оплачено	2022-03-28	Выехал	2022-03-10	2022-03-20		7	10

- Чему равен общий доход с каждого отеля за последний день?

Query EditorИстория запросов

```
1 SELECT "Отель"."Название", SUM("Цена_в_период"."Цена") FROM "Отель"
2 INNER JOIN "Номер" ON "Отель"."id_hotel" = "Номер"."Hotel"
3 LEFT JOIN "Цена_в_период" ON "Номер"."kind" = "Цена_в_период"."number_kind"
4 LEFT JOIN "Заказ_на_проживание" ON "Заказ_на_проживание"."Number" = "Номер"."id_number"
5 WHERE '2022-03-22' BETWEEN "Заказ_на_проживание"."Дата_заезда" AND "Заказ_на_проживание"."Дата_выезда"
6 GROUP BY "Отель"."Название"
7
8
```

РезультатПлан выполненияСообщенияNotifications

	Название character varying (30)	sum money
1	Redisson	\$3,400.00
2	Four Seasons	\$3,600.00

- Составить список свободных номеров одного из отелей на текущий день.

Hotel/postgres@PostgreSQL 11Query EditorИстория запросов

```
1 SELECT "№_комнаты"
2 FROM "Номер" LEFT JOIN "Заказ_на_проживание"
3 ON "Номер"."id_number" = "Заказ_на_проживание"."Number"
4 WHERE "Номер"."Hotel" = 1 AND ("Заказ_на_проживание"."Дата_заезда" IS NULL OR '2022-03-22'
5 NOT BETWEEN "Заказ_на_проживание"."Дата_заезда" AND "Заказ_на_проживание"."Дата_выезда")
6
7
```

РезультатПлан выполненияСообщенияNotifications

	№_комнаты integer
1	301
2	403
3	501
4	503
5	602
6	401
7	402
8	701
9	601
10	302
11	502
12	404

- Найти общие потери от незанятых номеров за текущий день по всей сети.

Hotel/postgres@PostgreSQL 11	
Query Editor	История запросов
<pre> 1 SELECT SUM("Цена_в_период"."Цена") FROM "Номер" 2 LEFT JOIN "Заказ_на_проживание" ON "Номер"."id_number" = "Заказ_на_проживание"."Number" 3 LEFT JOIN "Цена_в_период" ON "Цена_в_период"."number_kind" = "Номер"."Kind" 4 5 WHERE "Заказ_на_проживание"."Дата_заезда" IS NULL OR '2022-03-22' 6 NOT BETWEEN "Заказ_на_проживание"."Дата_заезда" AND "Заказ_на_проживание"."Дата_выезда" </pre>	
Query Tool - Hotel/postgres@PostgreSQL 11	
Результат План выполнения Сообщения Notifications	
sum	money
1	\$31,950.00

- Определить, в каком отеле имеется наибольшее количество незанятых номеров на текущие сутки.

Hotel/postgres@PostgreSQL 11	
Query Editor	История запросов
<pre> 1 SELECT "Название", COUNT(*) c FROM "Отель" 2 INNER JOIN "Номер" ON "Отель"."id_hotel" = "Номер"."Hotel" 3 LEFT JOIN "Заказ_на_проживание" ON "Заказ_на_проживание"."Number" = "Номер"."id_number" 4 WHERE ("Заказ_на_проживание"."Дата_заезда" IS NULL OR '2022-03-22' 5 NOT BETWEEN "Заказ_на_проживание"."Дата_заезда" AND "Заказ_на_проживание"."Дата_выезда") 6 GROUP BY "Отель"."Название" 7 ORDER BY c DESC LIMIT 1; 8 </pre>	
Результат План выполнения Сообщения Notifications	
Название	c
character varying (30)	bigint
1 Hilton	15

- Определить самый популярный тип номеров.

Hotel/postgres@PostgreSQL 11			
Query Editor История запросов			
<pre> 1 SELECT "Вид_номера"."Тип_комнаты", COUNT(*) c FROM "Заказ_на_проживание", "Вид_номера", "Номер" 2 WHERE "Заказ_на_проживание"."Number" = "Номер"."id_number" 3 AND "Номер"."Kind" = "Вид_номера"."id_kind" 4 GROUP BY "Вид_номера"."Тип_комнаты" 5 ORDER BY c DESC LIMIT 1; 6 </pre>			
Результат План выполнения Сообщения Notifications			
Тип_комнаты	c		
character varying (50)	bigint		
1 Семейный стандарт	4		

- вывести данные клиента, купившего топлива на наибольшую сумму в заданный день;

Query Editor История запросов			
<pre> 1 select full_name, sum(sales_quantity*final_price_for_liter) 2 from gss.bill, gss.fuel_sales 3 where fuel_sales.bill_number=bill.bill_number 4 and sales_date='2022-11-11' 5 group by full_name 6 having sum(sales_quantity*final_price_for_liter)>= all (select sum(sales_quantity*final_price_for_liter) 7 from gss.fuel_sales, gss.bill 8 where bill.bill_number=fuel_sales.bill_number 9 and sales_date='2022-11-11' 10 group by full_name) </pre>			
Результат План выполнения Сообщения Notifications			
full_name	sum		
text	bigint		
1 Иванов Иван Иванович	2079		

- какое топливо пользуется наибольшим спросом в прошедшем году на АЗС конкретного поставщика;

```

1 select fuel_name
2 from gss.fuel_sales, gss.fuel, gss.fuel_guide, gss.gas_station
3 where fuel_sales.fuel_code_supplied=fuel.fuel_code_supplied
4     and fuel.fuel_code=fuel_guide.fuel_code
5     and fuel_sales.gas_station_code=gas_station.gas_station_code
6     and (sales_date between '2022-01-01' and '2022-12-31')
7     and company_name='ПТК'
8 group by fuel_name
9 having count(fuel_name)>= all (select count(fuel_name)
10                             from gss.fuel_sales, gss.fuel, gss.fuel_guide, gss.gas_station
11                             where fuel_sales.fuel_code_supplied=fuel.fuel_code_supplied
12                             and fuel.fuel_code=fuel_guide.fuel_code
13                             and fuel_sales.gas_station_code=gas_station.gas_station_code
14                             and (sales_date between '2022-01-01' and '2022-12-31')
15                             and company_name='ПТК'
16                             group by fuel_name)

```

Результат План выполнения Сообщения Notifications

	fuel_name text	
1	АИ-92	

- сколько топлива каждого вида было продано за прошедший месяц по каждому поставщику на каждой АЗС;

Query Editor История запросов

```

1 select company_name, gas_station_address, fuel_type, sum(sales_quantity)
2 from gss.fuel_sales, gss.fuel, gss.fuel_guide, gss.gas_station
3 where fuel_sales.fuel_code_supplied=fuel.fuel_code_supplied
4     and fuel.fuel_code=fuel_guide.fuel_code
5     and fuel_sales.gas_station_code=gas_station.gas_station_code
6     and (sales_date between '2022-01-01' and '2022-12-31')
7 group by company_name, gas_station_address, fuel_type

```

Результат План выполнения Сообщения Notifications

	company_name text	gas_station_address text	fuel_type text	sum bigint	
1	Neste	Иркутская 21	дизельное топливо	9	
2	Газпромнефть	Рощинская 24	дизельное топливо	12	
3	Киришиавтосервис	Витебская 13	газ	22	
4	Лукойл	Бухарестская 4	бензин	21	
5	Лукойл	Волковский проспект 14	дизельное топливо	8	
6	Лукойл	Переулк декабристов 9	газ	5	
7	Лукойл	Стрельбищенская 3	газ	5	
8	ПТК	Кондратенко 6	газ	20	
9	Роснефть	Цветочная 2	бензин	8	
10	Роснефть	Цветочная 2	газ	13	

- какая из заправок продала топлива на наибольшую сумму по всем автозаправкам за последний год?

Query Editor История запросов

```

1 select company_name
2 from gss.fuel_sales, gss.gas_station
3 where fuel_sales.gas_station_code=gas_station.gas_station_code
4     and (sales_date between '2022-01-01' and '2022-12-31')
5 group by company_name
6 having sum(sales_quantity*final_price_for_liter)>= all (select sum(sales_quantity*final_price_for_liter)
7                                                         from gss.fuel_sales, gss.gas_station
8                                                         where fuel_sales.gas_station_code=gas_station.gas_station_code
9                                                         and (sales_date between '2022-01-01' and '2022-12-31')
10                                                         group by company_name)

```

Результат План выполнения Сообщения Notifications

	company_name text	
1	Лукойл	

2. Представления. Выполнить запросы на создание представлений согласно индивидуальному заданию, часть 3. В отчете привести формулировку запроса, команду, скриншот выполнения запроса и просмотр содержимого представления;

- содержащее сведения обо всех АЗС и всех видах топлива, которые они продают;

[Query Editor](#)
[История запросов](#)

```

1 select gas_station_address, fuel_code_supplied
2 from gss.gas_station_company, gss.gas_station, gss.fuel_price
3 where gas_station_company.company_name=gas_station.company_name
4 and gas_station_company.company_name=fuel_price.company_name
5 order by gas_station_address

```

[Результат](#)
[План выполнения](#)
[Сообщения](#)
[Notifications](#)

	gas_station_address text	fuel_code_supplied integer
1	Бухарестская 4	54273122
2	Бухарестская 4	46453291
3	Бухарестская 4	24978312
4	Витебская 13	54273122
5	Витебская 13	10104832
6	Волковский проспект 14	24978312
7	Волковский проспект 14	46453291
8	Волковский проспект 14	54273122
9	Иркутская 21	37453853
10	Кондратенко 6	54273122
11	Переулок декабристов 9	24978312
12	Переулок декабристов 9	54273122
13	Переулок декабристов 9	46453291
14	Рощинская 24	37453853
15	Стрельбищенская 3	24978312
16	Стрельбищенская 3	46453291
17	Стрельбищенская 3	54273122
18	Театральная площадь 7	98612442
19	Театральная площадь 7	38172467
20	Цветочная 2	98612442
21	Цветочная 2	38172467

[Результат](#)
[План выполнения](#)
[Сообщения](#)
[Notifications](#)

[Graphical](#)
[Analysis](#)
[Статистика](#)

gas_station_company

Hash

gas_station

Hash Inner Join

fuel_price

Hash Inner Join

Hash

Sort

- самая прибыльная АЗС за истекший месяц для каждого производителя.

Query Editor История запросов

```

1 select f.company_name, s.gas_station_address
2 from (select company_name, max(sum_sales) max_sales
3       from (select company_name, gas_station_address, sum(sales_quantity*final_price_for_liter) sum_sales
4             from gss.fuel_sales, gss.gas_station
5             where fuel_sales.gas_station_code=gas_station.gas_station_code
6                   and (sales_date between '2022-01-01' and '2022-12-31')
7             group by company_name, gas_station_address) s
8       group by company_name) f,
9 (select company_name, gas_station_address, sum(sales_quantity*final_price_for_liter) sum_sales
10    from gss.fuel_sales, gss.gas_station
11    where fuel_sales.gas_station_code=gas_station.gas_station_code
12          and (sales_date between '2022-01-01' and '2022-12-31')
13    group by company_name, gas_station_address) s
14 where f.max_sales=s.sum_sales

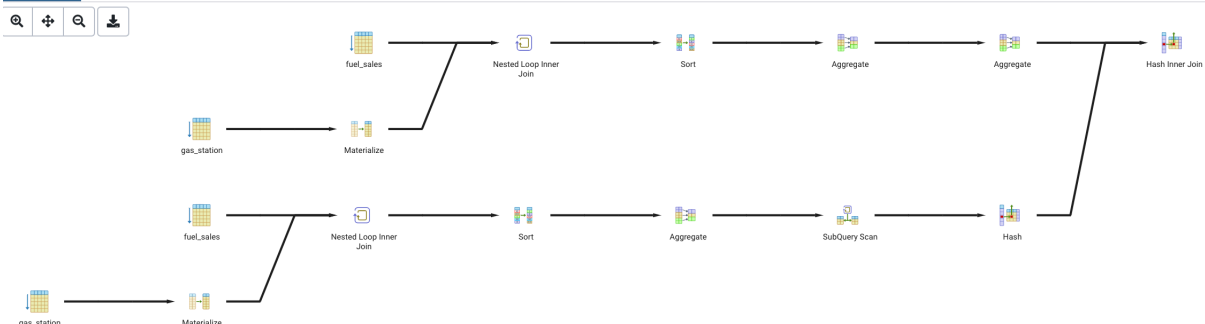
```

Результат План выполнения Сообщения Notifications

	company_name text	gas_station_address text
1	Neste	Иркутская 21
2	Газпромнефть	Рощинская 24
3	Киришиавтосервис	Витебская 13
4	Лукойл	Бухарестская 4
5	ПТК	Кондратенко 6
6	Роснефть	Цветочная 2

Результат План выполнения Сообщения Notifications

Graphical Analysis Статистика



- Запросы на модификацию данных. Выполнить запросы на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов (составить самостоятельно). В отчете привести формулировку запроса, команду, скриншот до и после выполнения запроса. Учитывается сложность запроса;
 - INSERT с использованием подзапросов;

```

1 insert into gss.bill (bill_number, bill_sum, company_name, full_name)
2 select 44001238, 10000, (select company_name
3                        from gss.gas_station_company
4                        where company_phone_number=89129872367), full_name
5                        from gss.customer
6                        where customer_phone_number=89812538474

```

Результат План выполнения Сообщения Notifications

INSERT 0 1

Запрос завершён успешно, время выполнения: 50 msec.

Результат План выполнения Сообщения Notifications

	bill_number [PK] integer	bill_sum integer	company_name text	full_name text
1	44005673	1500	Лукойл	Иванов Иван Иванович
2	44008788	5000	ПТК	Иванов Иван Иванович
3	44008236	3000	Роснефть	Петров Петр Петрович
4	44006523	8000	Лукойл	Наумов Петр Семенович
5	44009234	4500	Роснефть	Павлов Павел Павлович
6	44008762	2000	ПТК	Бобров Глеб Антонович
7	44002317	1500	Лукойл	Смирнов Владимир Максимович
8	44008912	2000	Киришиавтосервис	Тимофеев Егор Андреевич
9	44002387	5300	Neste	Николаев Николай Николаевич
10	44001351	1000	Газпромнефть	Слепцов Евгений Николаевич

Результат План выполнения Сообщения Notifications

	bill_number [PK] integer	bill_sum integer	company_name text	full_name text
1	44005673	1500	Лукойл	Иванов Иван Иванович
2	44008788	5000	ПТК	Иванов Иван Иванович
3	44008236	3000	Роснефть	Петров Петр Петрович
4	44006523	8000	Лукойл	Наумов Петр Семенович
5	44009234	4500	Роснефть	Павлов Павел Павлович
6	44008762	2000	ПТК	Бобров Глеб Антонович
7	44002317	1500	Лукойл	Смирнов Владимир Максимович
8	44008912	2000	Киришиавтосервис	Тимофеев Егор Андреевич
9	44002387	5300	Neste	Николаев Николай Николаевич
10	44001351	1000	Газпромнефть	Слепцов Евгений Николаевич
11	44001238	10000	Neste	Дмиртиенко Дмитрий Дмитриевич

– UPDATE с использованием подзапросов;

```
1  update gss.fuel_price
2  set price = 85
3  where fuel_code_supplied = (select fuel_code_supplied
4                               from gss.fuel
5                               where fuel_code=94127322 and supplier_code=4)
6  and company_name = (select company_name
7                       from gss.gas_station_company
8                       where company_address='Выборгская набережная 18')
```

Результат План выполнения Сообщения Notifications

UPDATE 1

Запрос завершён успешно, время выполнения: 58 msec.

Результат План выполнения Сообщения Notifications

	price_code [PK] integer	validity_period_from date	validity_period_until date	price integer	fuel_code_supplied integer	company_name text
1	646721	2022-04-01	2022-10-01	76	24978312	Лукойл
2	728462	2022-04-01	2022-10-01	82	98612442	Роснефть
3	732644	2022-05-01	2022-11-01	79	46453291	Лукойл
4	230943	2022-05-22	2022-12-22	82	38172467	Роснефть
5	202344	2022-01-23	2022-08-23	79	54273122	ПТК
6	129839	2022-04-01	2022-11-01	83	54273122	Лукойл
7	904124	2022-05-01	2022-12-01	90	54273122	Киришиавтосервис
8	129492	2022-03-01	2022-10-01	89	10104832	Киришиавтосервис
9	492911	2022-02-01	2022-09-01	75	37453853	Газпромнефть
10	102392	2022-07-01	2022-12-01	79	37453853	Neste

Результат План выполнения Сообщения Notifications

	price_code [PK] integer	validity_period_from date	validity_period_until date	price integer	fuel_code_supplied integer	company_name text
1	646721	2022-04-01	2022-10-01	76	24978312	Лукойл
2	728462	2022-04-01	2022-10-01	82	98612442	Роснефть
3	732644	2022-05-01	2022-11-01	79	46453291	Лукойл
4	230943	2022-05-22	2022-12-22	82	38172467	Роснефть
5	202344	2022-01-23	2022-08-23	79	54273122	ПТК
6	904124	2022-05-01	2022-12-01	90	54273122	Киришиавтосервис
7	129492	2022-03-01	2022-10-01	89	10104832	Киришиавтосервис
8	492911	2022-02-01	2022-09-01	75	37453853	Газпромнефть
9	102392	2022-07-01	2022-12-01	79	37453853	Neste
10	129839	2022-04-01	2022-11-01	85	54273122	Лукойл

– DELETE с использованием подзапросов.

Query Editor История запросов

```
1 delete from gss.bill
2 where full_name = (select full_name
3                     from gss.customer
4                     where customer_phone_number=89811298473)
5 and company_name = (select company_name
6                     from gss.gas_station_company
7                     where company_phone_number=89124591645)
```

Результат План выполнения Сообщения Notifications

DELETE 1

Запрос завершён успешно, время выполнения: 40 msec.

Результат План выполнения Сообщения Notifications

	bill_number [PK] integer	bill_sum integer	company_name text	full_name text
1	44005673	1500	Лукойл	Иванов Иван Иванович
2	44008788	5000	ПТК	Иванов Иван Иванович
3	44008236	3000	Роснефть	Петров Петр Петрович
4	44006523	8000	Лукойл	Наумов Петр Семенович
5	44009234	4500	Роснефть	Павлов Павел Павлович
6	44008762	2000	ПТК	Бобров Глеб Антонович
7	44002317	1500	Лукойл	Смирнов Владимир Максимович
8	44008912	2000	Киришиавтосервис	Тимофеев Егор Андреевич
9	44002387	5300	Neste	Николаев Николай Николаевич
10	44001351	1000	Газпромнефть	Слепцов Евгений Николаевич
11	44001238	10000	Neste	Дмиртиенко Дмитрий Дмитриевич

Результат План выполнения Сообщения Notifications

	bill_number [PK] integer	bill_sum integer	company_name text	full_name text
1	44005673	1500	Лукойл	Иванов Иван Иванович
2	44008788	5000	ПТК	Иванов Иван Иванович
3	44008236	3000	Роснефть	Петров Петр Петрович
4	44009234	4500	Роснефть	Павлов Павел Павлович
5	44008762	2000	ПТК	Бобров Глеб Антонович
6	44002317	1500	Лукойл	Смирнов Владимир Максимович
7	44008912	2000	Киришиавтосервис	Тимофеев Егор Андреевич
8	44002387	5300	Neste	Николаев Николай Николаевич
9	44001351	1000	Газпромнефть	Слепцов Евгений Николаевич
10	44001238	10000	Neste	Дмиртиенко Дмитрий Дмитриевич

4. Создание индексов. Выполнить запросы без индекса и создать планы запросов. Выполнить создание индексов. Выполнить запросы с индексами и создать планы запросов. Сравнить время выполнения запросов. Удалить индексы.

Время выполнения запроса в таблицу без индекса – 37 мсек:

Query Editor История запросов

```
1 select sales_code
2 from gss.fuel_sales
3 where sales_quantity > 25
```

Результат Сообщения План выпо.

Graphical Analysis Статистика

🔍 🔄 🔍 📄



gss.fuel_sales

Создание индекса:

Query Editor История запросов

```
1 create index sales_idx
2 on gss.fuel_sales(sales_quantity)
3 where sales_quantity > 25
```

Результат Сообщения План выполнения Notifications

CREATE INDEX

Запрос завершён успешно, время выполнения: 36 msec.

Время выполнения запроса в таблицу с индексом – 39 мсек:


Query Editor История запросов

```
1 select sales_code
2 from gss.fuel_sales
3 where sales_quantity > 25
```

Результат Сообщения План выпо.

Graphical Analysis Статистика

🔍 🔄 🔍 📄



gss.fuel_sales

Удаление индекса:

```
1 drop index gss.sales_idx
```

Результат Сообщения План выполнения Notifications

DROP INDEX

Запрос завершён успешно, время выполнения: 37 msec.

Запрос не использует индекс, поэтому время с индексом и без индекса ±равны. Думаю, проблема в том, что у меня мало данных в таблице, и из-за этого запросу не выгодно работать с индексом. Еще я создал составной индекс, но нет никаких отличий, поэтому я его не добавил в отчет.

Выводы:

В результате выполненной работы:

- были выполнены запросы и представления
- были изучены и использованы запросы на модификацию данных
- были изучены и реализованы индексы, хотя мне не удалось сделать так, чтобы запрос работал через индекс, а не таблицу. Но самое главное усвоено.