

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по лабораторной работе №4 «Запросы на выборку и модификацию данных. Представления. Работа с индексами»

по дисциплине **«Проектирование и реализация баз данных»**

Автор: Пузенко А.А.

Факультет: ИКТ

Группа: К3240

Преподаватель: Говорова М.М.



Санкт-Петербург 2023

Оглавление

| | |
|---------------------------|----|
| Цель работы..... | 3 |
| Практическое задание..... | 3 |
| Выполнение..... | 3 |
| Вывод:..... | 18 |

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.

Выполнение

Создать запросы:

1. Сколько раз заправлял автомобиль каждый из клиентов за заданный период.

Query Query History

```
1 SELECT c.full_name, COUNT(*) AS purchase_count
2 FROM purchases p
3 JOIN cards ca ON p.id_card = ca.id_card
4 JOIN clients c ON ca.id_client = c.id_client
5 WHERE p.purchase_date BETWEEN '2024-01-01' AND '2024-01-31'
6 GROUP BY c.full_name;
```

| | full_name | purchase_count |
|----|-------------------------|----------------|
| | character varying (100) | bigint |
| 1 | Ivan Ivanov | 8 |
| 2 | Ira Zhukova | 3 |
| 3 | Dmitry Savin | 2 |
| 4 | Alexandra Soboleva | 9 |
| 5 | Alexander Sobolev | 11 |
| 6 | Varvara Frolova | 1 |
| 7 | Elena Morozova | 5 |
| 8 | Andrey Bobrov | 1 |
| 9 | Maxim Yefimov | 1 |
| 10 | Petr Popov | 6 |
| 11 | Andrey Puzenko | 3 |
| 12 | Polina Korneva | 2 |
| 13 | Sergey Chernov | 2 |
| 14 | Gleb Krylov | 4 |

2. Кто из клиентов не приобретал топливо в январе текущего года?

Query Query History

```

1 SELECT c.full_name
2 FROM clients c
3 LEFT JOIN cards ca ON c.id_client = ca.id_client
4 LEFT JOIN purchases p ON ca.id_card = p.id_card AND p.purchase_date BETWEEN '2024-01-01' AND '2024-01-31'
5 WHERE p.id_card IS NULL;

```

| | client_name |
|---|-------------------------|
| | character varying (100) |
| 1 | Kristina Petrova |

3.1. Найти клиента, купившего наибольший объем топлива по всей сети. (в общем)

Query
Query History

```

1 SELECT c.full_name, SUM(p.amount_fuel) AS total_fuel_amount
2 FROM purchases p
3 JOIN cards ca ON p.id_card = ca.id_card
4 JOIN clients c ON ca.id_client = c.id_client
5 GROUP BY c.full_name
6 ORDER BY total_fuel_amount DESC
7 LIMIT 1;

```

Data Output
Messages
Notifications

| | full_name character varying (100) | total_fuel_amount double precision |
|---|--------------------------------------|---------------------------------------|
| 1 | Alexander Sobolev | 326 |

3.2. Найти клиента, купившего наибольший объем топлива по всей сети. (единоразово)

Query
Query History

```

1 SELECT c.full_name, MAX(p.amount_fuel) AS max_fuel_amount_in_single_purchase
2 FROM purchases p
3 JOIN cards ca ON p.id_card = ca.id_card
4 JOIN clients c ON ca.id_client = c.id_client
5 GROUP BY c.full_name
6 ORDER BY max_fuel_amount_in_single_purchase DESC
7 LIMIT 1;



```

Data Output
Messages
Notifications

| | full_name character varying (100) | max_fuel_amount_in_single_purchase double precision |
|---|--------------------------------------|--|
| 1 | Gleb Krylov | 42 |

4. Вывести данные клиента, купившего топлива на наибольшую сумму в заданный день.

| Query | Query History |
|-------|--|
| 1 | SELECT |
| 2 | c.full_name , |
| 3 | SUM(p.amount_fuel * fs.price) AS total_spent |
| 4 | FROM |
| 5 | purchases p |
| 6 | JOIN |
| 7 | cards ca ON p.id_card = ca.id_card |
| 8 | JOIN |
| 9 | clients c ON ca.id_client = c.id_client |
| 10 | JOIN |
| 11 | fuels_sold fs ON p.id_fuel_sold = fs.id_fuel_sold |
| 12 | WHERE |
| 13 | p.purchase_date = '2024-01-08' |
| 14 | GROUP BY |
| 15 | c.full_name |
| 16 | ORDER BY |
| 17 | total_spent DESC |
| 18 | LIMIT 1; |

| | full_name character varying (100)  | total_spent double precision  |
|---|---|--|
| 1 | Alexandra Soboleva | 2114.20000000000003 |

5. Какое топливо пользуется наибольшим спросом в прошедшем году на АЗС конкретного поставщика?

Query

Query History

```

1 SELECT
2     f.mark,
3     SUM(p.amount_fuel) AS total_sold_amount
4 FROM
5     purchases p
6 JOIN
7     fuels_sold fs ON p.id_fuel_sold = fs.id_fuel_sold
8 JOIN
9     fuels f ON fs.id_fuel = f.id_fuel
10 JOIN
11     gas_stations gs ON fs.id_station = gs.id_station
12 JOIN
13     supplier_firms sf ON gs.id_firm = sf.id_firm
14 WHERE
15     sf.name = 'ЛУКОЙЛ'
16 GROUP BY
17     f.mark
18 ORDER BY
19     total_sold_amount DESC
20 LIMIT 1;

```

Data Output

Messages

Notifications

≡+

📄

▼

📋

▼

🗑️

🗄️

⬇️

📈

| | mark character varying (40) 🔒 | total_sold_amount double precision 🔒 |
|---|----------------------------------|---|
| 1 | Пропат-бутан | 433 |

6. Сколько топлива каждого вида было продано за прошедший месяц по каждому поставщику на каждой АЗС.

| Query | Query History |
|-------|---|
| 1 | SELECT |
| 2 | gs.name AS gas_station_name, |
| 3 | sf.name AS supplier_name, |
| 4 | f.type, |
| 5 | SUM(p.amount_fuel) AS total_amount_sold |
| 6 | FROM |
| 7 | purchases p |
| 8 | JOIN |
| 9 | fuels_sold fs ON p.id_fuel_sold = fs.id_fuel_sold |
| 10 | JOIN |
| 11 | gas_stations gs ON fs.id_station = gs.id_station |
| 12 | JOIN |
| 13 | fuels f ON fs.id_fuel = f.id_fuel |
| 14 | JOIN |
| 15 | supplier_firms sf ON gs.id_firm = sf.id_firm |
| 16 | WHERE |
| 17 | p.purchase_date BETWEEN '2024-02-01' AND '2024-02-29' |
| 18 | GROUP BY |
| 19 | gs.name, sf.name, f.type |
| 20 | ORDER BY |
| 21 | gs.name, sf.name, f.type; |

Data Output

Messages

Notifications

≡+

📄

▼

📋

▼

🗑️

🗄️

⬇️












📈

| | gas_station_name character varying (100) 🔒 | supplier_name character varying (100) 🔒 | type character varying (40) 🔒 | total_amount_sold double precision 🔒 |
|---|---|--|----------------------------------|---|
| 1 | Заправка 1 | РОСНЕФТЬ | бензин | 62 |
| 2 | Заправка 1 | РОСНЕФТЬ | дизель | 96 |
| 3 | Заправка 2 | ЛУКОЙЛ | газ | 101 |
| 4 | Заправка 3 | Газпром нефть | бензин | 301 |
| 5 | Заправка 4 | РОСНЕФТЬ | бензин | 87 |
| 6 | Заправка 5 | ЛУКОЙЛ | газ | 211 |

7. Какая из заправок продала топлива на наибольшую сумму по всем автозаправкам за последний год?


```
1 WITH total_sales_per_gas_station AS (  
2     SELECT  
3         fs.id_station,  
4         SUM(p.amount_fuel * fs.price) AS total_sales  
5     FROM  
6         purchases p  
7     JOIN  
8         fuels_sold fs ON p.id_fuel_sold = fs.id_fuel_sold  
9     JOIN  
10        fuels f ON fs.id_fuel = f.id_fuel  
11    GROUP BY  
12        fs.id_station  
13 )  
14 SELECT  
15     gs.name AS gas_station_name,  
16     t.total_sales AS total_sales_amount  
17 FROM  
18     total_sales_per_gas_station t  
19 JOIN  
20     gas_stations gs ON t.id_station = gs.id_station  
21 ORDER BY  
22     t.total_sales DESC  
23 LIMIT 1;
```

Data Output Messages Notifications

|          | | |
|---|---|--|
| | gas_station_name character varying (100)  | total_sales_amount double precision  |
| 1 | Заправка 3 | 36141.1 |

Создать представления:

1. Содержащее сведения обо всех АЗС и всех видах топлива, которые они продают;

```
1 CREATE VIEW gas_station_fuels AS
2 SELECT DISTINCT
3     gs.name AS gas_station_name,
4     f.mark,
5     f.type
6 FROM
7     gas_stations gs
8 JOIN
9     fuels_sold fs ON gs.id_station = fs.id_station
10 JOIN
11     fuels f ON fs.id_fuel = f.id_fuel;
12
```

```
1 SELECT * FROM public.gas_station_fuels
```

```
2
```

| | gas_station_name character varying (100) | mark character varying (40) | type character varying (40) |
|----|---|--------------------------------|--------------------------------|
| 1 | Заправка 3 | Пропат-бутан | газ |
| 2 | Заправка 3 | АИ-98 | бензин |
| 3 | Заправка 4 | АИ-92 | бензин |
| 4 | Заправка 1 | АИ-92 | бензин |
| 5 | Заправка 1 | АИ-95 | бензин |
| 6 | Заправка 4 | АИ-95 | бензин |
| 7 | Заправка 2 | Пропат-бутан | газ |
| 8 | Заправка 1 | ДТЗ | дизель |
| 9 | Заправка 5 | Пропат-бутан | газ |
| 10 | Заправка 3 | АИ-92 | бензин |
| 11 | Заправка 1 | АИ-98 | бензин |
| 12 | Заправка 1 | ДТЛ | дизель |
| 13 | Заправка 2 | Метан | газ |
| 14 | Заправка 5 | Метан | газ |
| 15 | Заправка 4 | АИ-98 | бензин |
| 16 | Заправка 2 | М-100 | мазут |
| 17 | Заправка 3 | АИ-95 | бензин |

2. Самая прибыльная АЗС за истекший месяц для каждого производителя.

Query Query History

```

1 CREATE VIEW most_profitable_gas_stations AS
2 SELECT DISTINCT ON (sf.name)
3     sf.name AS supplier_name,
4     gs.name AS gas_station_name,
5     SUM(p.amount_fuel * fs.price) AS total_profit
6 FROM
7     supplier_firms sf
8 JOIN
9     gas_stations gs ON sf.id_firm = gs.id_firm
10 JOIN
11     fuels_sold fs ON gs.id_station = fs.id_station
12 JOIN
13     fuels f ON fs.id_fuel = f.id_fuel
14 JOIN
15     purchases p ON fs.id_fuel_sold = p.id_fuel_sold
16 WHERE
17     DATE_TRUNC('month', p.purchase_date) = DATE_TRUNC('month', CURRENT_DATE) - INTERVAL '1 month'
18 GROUP BY
19     sf.name,
20     gs.name
21 ORDER BY
22     sf.name,
23     SUM(p.amount_fuel * fs.price) DESC;

```

Query Query History

1 SELECT * FROM public.most_profitable_gas_stations

2 |

Data Output Messages Notifications

| | supplier_name character varying (100) | gas_station_name character varying (100) | total_profit double precision |
|---|--|---|----------------------------------|
| 1 | Газпром нефть | Заправка 3 | 18284.6 |
| 2 | ЛУКОЙЛ | Заправка 5 | 5795.0999999999999 |
| 3 | РОСНЕФТЬ | Заправка 1 | 9715.3 |

Создать запросы на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов:

INSERT

Добавляет сотрудника в таблицу employees на должность Cashier на Заправку 2

Query

Query History

1

```
INSERT INTO employees (full_name, phone, passport_details, id_position, id_station, passport_number)
VALUES ('Ivan Terekhov', '89027350597', 'МВД России Москва',
(SELECT id_position FROM positions WHERE title = 'Cashier'),
(SELECT id_station FROM gas_stations WHERE name = 'Заправка 2'),
'6020001100');
```

Data Output

Messages

Notifications

INSERT 0 1

Query returned successfully in 91 msec.

| | | | | | | | |
|----|----|---------------|-------------|-------------------|---|---|------------|
| 18 | 33 | Ivan Terekhov | 89027350597 | МВД России Москва | 2 | 6 | 6020001100 |
|----|----|---------------|-------------|-------------------|---|---|------------|

UPDATE

Увеличивает баланс на карте, которой владеет клиент с именем Andrey Puzenko

Query

Query History

1

```
UPDATE cards
SET balance = balance + 100
WHERE id_client = (
SELECT id_client
FROM clients
WHERE full_name = 'Andrey Puzenko'
);
```

Data Output

Messages

Notifications

UPDATE 1

Query returned successfully in 101 msec.

До

| | id_card [PK] bigint | id_client bigint | balance double precision |
|---|------------------------|---------------------|-----------------------------|
| 1 | 9 | 4 | 998 |

После

| | id_card [PK] bigint | id_client bigint | balance double precision |
|---|------------------------|---------------------|-----------------------------|
| 1 | 9 | 4 | 998 |

DELETE

Удаляет карту клиента, если не было совершено ни одной покупки.

Query
Query History

```

1 DELETE FROM cards
2 WHERE id_card NOT IN (
3     SELECT id_card
4     FROM purchases
5 );

```

Data Output
Messages
Notifications

DELETE 1

Query returned successfully in 56 msec.

До

| | id_card [PK] bigint | id_client bigint | balance double precision |
|---|------------------------|---------------------|-----------------------------|
| 1 | 9 | 4 | 1098 |
| 2 | 10 | 5 | 1200 |
| 3 | 11 | 6 | 10 |
| 4 | 12 | 7 | 120.09 |

После

Query
Query History

1
EXPLAIN ANALYZE SELECT * FROM clients WHERE phone = '8950777777';

Data Output
Messages
Notifications

+

📄

▼

📋

▼

🗑️

🗄️

⬇️

📈

QUERY PLAN

text

🔒

| | |
|---|--|
| 1 | Seq Scan on clients (cost=0.00..1.19 rows=1 width=44) (actual time=0.018..0.020 rows=1 loops=... |
| 2 | Filter: ((phone)::text = '8950777777'::text) |
| 3 | Rows Removed by Filter: 14 |
| 4 | Planning Time: 1.896 ms |
| 5 | Execution Time: 0.172 ms |

Graphical
Analysis
Statistics

🔍

🔄

🔍

⬇️

↓

📊

clients

Второй индекс

Query
Query History

1
CREATE INDEX idx_gas_stations_sales ON gas_stations(id_station);

Data Output
Messages
Notifications

CREATE INDEX

Query returned successfully in 40 msec.

Без индекса


```

1 EXPLAIN ANALYZE SELECT gs.name, COUNT(*)
2 FROM gas_stations gs
3 JOIN fuels_sold fs ON gs.id_station = fs.id_station GROUP BY gs.name;
4

```



QUERY PLAN

text



| | |
|----|--|
| 1 | HashAggregate (cost=3.50..3.55 rows=5 width=226) (actual time=0.104..0.106 rows=5 loops=1) |
| 2 | Group Key: gs.name |
| 3 | Batches: 1 Memory Usage: 24kB |
| 4 | -> Hash Join (cost=1.11..3.15 rows=70 width=218) (actual time=0.040..0.070 rows=70 loops=1) |
| 5 | Hash Cond: (fs.id_station = gs.id_station) |
| 6 | -> Seq Scan on fuels_sold fs (cost=0.00..1.70 rows=70 width=8) (actual time=0.019..0.025 rows=70 loops=1) |
| 7 | -> Hash (cost=1.05..1.05 rows=5 width=226) (actual time=0.014..0.014 rows=5 loops=1) |
| 8 | Buckets: 1024 Batches: 1 Memory Usage: 9kB |
| 9 | -> Seq Scan on gas_stations gs (cost=0.00..1.05 rows=5 width=226) (actual time=0.008..0.009 rows=5 loop... |
| 10 | Planning Time: 0.198 ms |
| 11 | Execution Time: 0.151 ms |

С индексом

Query
Query History

```

1 EXPLAIN ANALYZE SELECT gs.name, COUNT(*)
2 FROM gas_stations gs
3 JOIN fuels_sold fs ON gs.id_station = fs.id_station GROUP BY gs.name;
4

```

Data Output
Messages
Notifications

+

📄

▼

📋

▼

🗑️

🗄️

⬇️

📈

| | QUERY PLAN | 🔒 |
|----|--|---|
| | text | |
| 1 | HashAggregate (cost=3.50..3.55 rows=5 width=226) (actual time=0.064..0.066 rows=5 loops=1) | |
| 2 | Group Key: gs.name | |
| 3 | Batches: 1 Memory Usage: 24kB | |
| 4 | -> Hash Join (cost=1.11..3.15 rows=70 width=218) (actual time=0.030..0.046 rows=70 loops=1) | |
| 5 | Hash Cond: (fs.id_station = gs.id_station) | |
| 6 | -> Seq Scan on fuels_sold fs (cost=0.00..1.70 rows=70 width=8) (actual time=0.010..0.012 rows=70 loops=1) | |
| 7 | -> Hash (cost=1.05..1.05 rows=5 width=226) (actual time=0.013..0.013 rows=5 loops=1) | |
| 8 | Buckets: 1024 Batches: 1 Memory Usage: 9kB | |
| 9 | -> Seq Scan on gas_stations gs (cost=0.00..1.05 rows=5 width=226) (actual time=0.008..0.009 rows=5 loop... | |
| 10 | Planning Time: 2.800 ms | |
| 11 | Execution Time: 0.105 ms | |

Вывод:

В данной лабораторной работе запросы на выборку данных в базе данных PostgreSQL. А также были созданы представления, запросы на удаление, обновление и добавление данных. Были созданы простые и составные индексы и сравнено время выполнения запросов с индексами и без.