

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 № 2**

по теме:

*«Запросы на выборку и модификацию данных, представления и
индексы в PostgreSQL»*

по дисциплине: Проектирование и реализация баз данных

Специальность:

09.03.03 Мобильная и сетевая разработка

Проверила: Говорова

М.М. Дата: «...» ...

2023 г.

Оценка _____

Выполнила:

студентка группы

K32392

Барталевич Е.В.

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 посмотреть 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 работы:

Предметная область – Автомастерская (вариант 11)

Наименование БД – Garage

Схемы логической модели базы данных

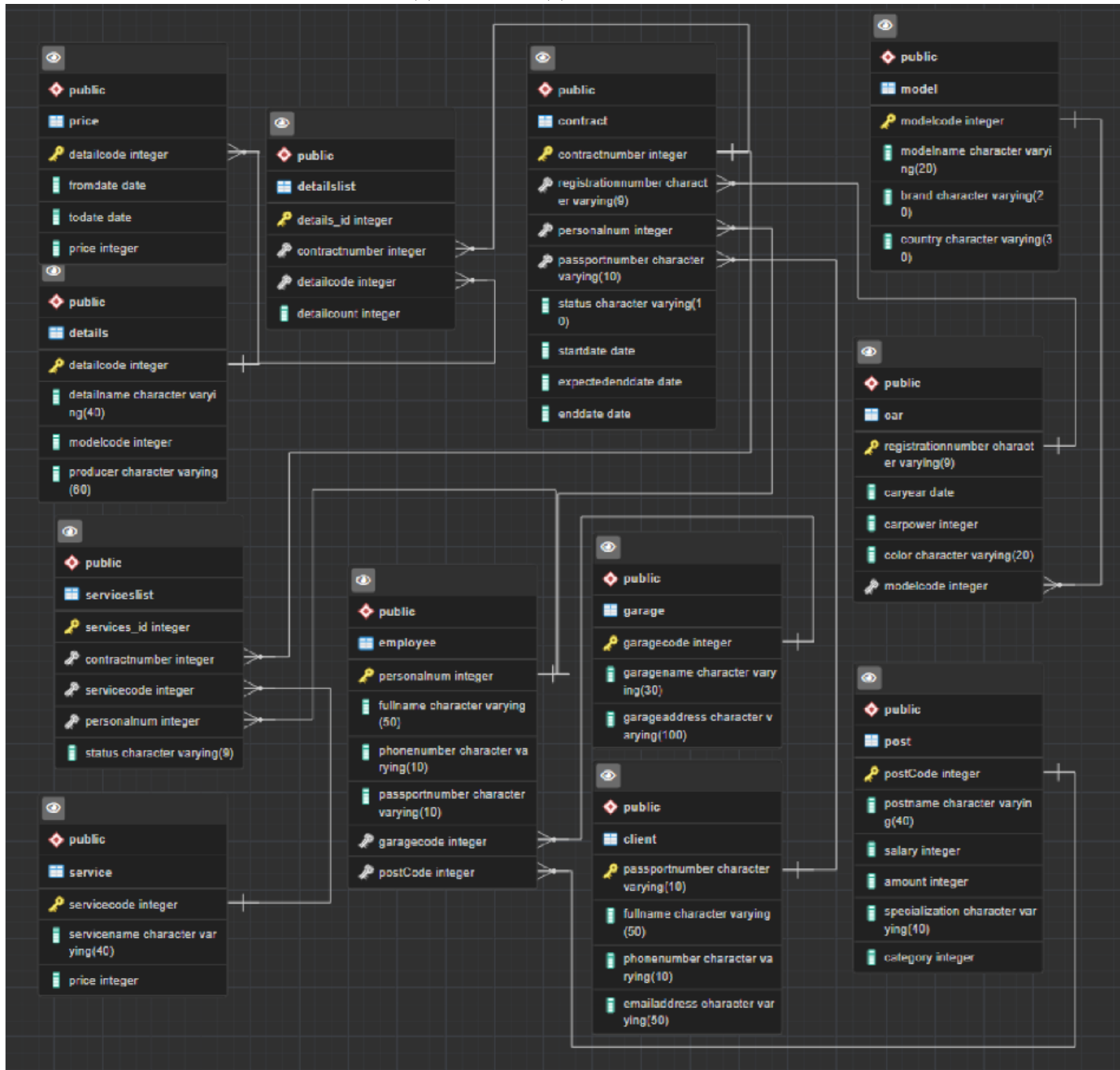



Рисунок 1 – ERD базы данных 1

Запросы на выборку.

- 1) Выбрать фамилию того механика, который чаще всех работает с автомобилями марки "Тойота".

```
SELECT e.fullname
FROM employee e
JOIN serviceslist sl ON e.personalnum = sl.personalnum
JOIN contract c ON sl.contractnumber = c.contractnumber
JOIN car car ON c.registrationnumber = car.registrationnumber
JOIN model m ON car.modelcode = m.modelcode
WHERE m.brand = 'Toyota'
GROUP BY e.fullname
ORDER BY COUNT(*) DESC
LIMIT 1;
```

Вывод:

	fullname character varying (50) 
1	Петров Петр Петров...

- 2) Определить тех владельцев автомобилей, которых всегда обслуживает один и тот же механик. Вывести фамилии механика и его постоянного клиента.

```
WITH client_mechanic_pairs AS (
    SELECT c.fullname AS client_fullname, e.fullname AS mechanic_fullname,
    COUNT(DISTINCT sl.contractnumber) AS contracts_count
    FROM client c
    JOIN contract ct ON c.passportnumber = ct.passportnumber
    JOIN serviceslist sl ON ct.contractnumber = sl.contractnumber
    JOIN employee e ON sl.personalnum = e.personalnum
    GROUP BY c.fullname, e.fullname
),
client_total_contracts AS (
    SELECT fullname, COUNT(DISTINCT contractnumber) AS total_contracts
    FROM client
    JOIN contract ON client.passportnumber = contract.passportnumber
    GROUP BY fullname
)
SELECT cmp.client_fullname, cmp.mechanic_fullname
FROM client_mechanic_pairs cmp
JOIN client_total_contracts ctc ON cmp.client_fullname = ctc.fullname
WHERE cmp.contracts_count = ctc.total_contracts
ORDER BY cmp.client_fullname, cmp.mechanic_fullname;
```

Вывод:

	client_fullname character varying (50)	mechanic_fullname character varying (50)
1	Иванов Иван Ивано...	Петров Петр Петров...

3) Вывести фамилии механиков, которые не выполняли работы в срок и количество дней просрочки выполнения заказа.

```
SELECT e.fullname, SUM(c.enddate - c.expectedenddate) AS delay_days
FROM employee e
JOIN serviceslist sl ON e.personalnum = sl.personalnum
JOIN contract c ON sl.contractnumber = c.contractnumber
WHERE c.enddate > c.expectedenddate
GROUP BY e.fullname;
```

Вывод:

	fullname character varying (50)	delay_days bigint
1	Миронов Евгений Д...	1
2	Сидоров Антон Серг...	31
3	Петров Петр Петров...	28

4) Вывести данные механика, который выполнял все виды ремонта за прошедшую неделю.

```
SELECT e.*
FROM employee e
JOIN serviceslist sl ON e.personalnum = sl.personalnum
WHERE sl.status = 'Выполнено'
AND sl.contractnumber IN (
    SELECT contractnumber
    FROM contract
    WHERE enddate >= current_date - INTERVAL '7 days'
)
GROUP BY e.personalnum
HAVING COUNT(DISTINCT sl.servicecode) = (
    SELECT COUNT(*)
    FROM service
);
```

Вывод:

	personalnum [PK] integer	fullname character varying (50)	phonenumber character varying (10)	passportnumber character varying (10)	garagecode integer	postCode integer
1	8	Миронов Евгений Д...	1111111114	1111111114	1	8

- 5) Сколько заработал каждый мастер за прошедший месяц? Оклад мастера = половина стоимости выполненных услуг.

```
SELECT e.fullname, SUM(s.price) / 2 AS salary
FROM employee e
JOIN serviceslist sl ON e.personalnum = sl.personalnum
JOIN service s ON sl.servicecode = s.servicecode
WHERE sl.status = 'Выполнено'
AND sl.contractnumber IN (
    SELECT contractnumber
    FROM contract
    WHERE enddate >= current_date - INTERVAL '1 month'
)
GROUP BY e.fullname;
```

Вывод:

	fullname character varying (50)	salary bigint
1	Сидоров Антон Сергеевич	500

- 6) Вывести данные владельцев автомобилей, которые обращались в ремонт больше одного раза.

```
SELECT cl.*
FROM client cl
JOIN contract c ON cl.passportnumber = c.passportnumber
GROUP BY cl.passportnumber
HAVING COUNT(*) > 1;
```

Вывод:

	passportnumber [PK] character varying (10)	fullname character varying (50)	phonenumber character varying (10)	emailaddress character varying (50)
1	1111111111	Бубусов Артем Николае...	1111111111	bubusov@example.c...
2	2222222222	Бебрина Арина Артемов...	2222222222	bebrina@example.com
3	3124567890	Иванов Сергей Геннадь...	9213631509	ivanov@example1.com

- 7) За каждый день просрочки выполнения заказа механику назначается штраф в размере 5%. Рассчитать штраф каждого механика за прошедший месяц.

```
SELECT e.fullname, SUM((c.enddate - c.expectedenddate) * s.price * 0.05) AS
```

```

fine
FROM employee e
JOIN serviceslist sl ON e.personalnum = sl.personalnum
JOIN service s ON sl.servicecode = s.servicecode
JOIN contract c ON sl.contractnumber = c.contractnumber
WHERE c.enddate > c.expectedenddate
      AND c.enddate >= current_date - INTERVAL '1 month'
GROUP BY e.fullname;

```

Вывод:

	fullname character varying (50)	fine numeric
1	Сидоров Антон Сергеевич	1550.00

Создание Представлений.

1. Для заказчиков (фамилию механика и модель автомобиля, которую он ремонтирует чаще всего).

Запрос:

```

CREATE VIEW mechanic_favorite_models AS
SELECT e.fullname, m.modelname, COUNT(*) AS repair_count
FROM employee e
JOIN serviceslist sl ON e.personalnum = sl.personalnum
JOIN contract c ON sl.contractnumber = c.contractnumber
JOIN car car ON c.registrationnumber = car.registrationnumber
JOIN model m ON car.modelcode = m.modelcode
WHERE sl.status = 'Выполнено'
GROUP BY e.fullname, m.modelname
ORDER BY e.fullname, repair_count DESC;

```

Вывод:

```
SELECT * FROM mechanic_favorite_models;
```

	fullname character varying (50)	modelname character varying (20)	repair_count bigint
1	Ершова Мария Ивановна	Cerato	6
2	Ершова Мария Ивановна	M2	1
3	Ершова Мария Ивановна	RAV 4	1
4	Коржов Никита Моисее...	M2	1
5	Миронов Евгений Денис...	M2	1
6	Петров Петр Петрович	RAV 4	5
7	Петров Петр Петрович	Cerato	1
8	Сидоров Антон Сергеев...	RAV 4	2



2. Для менеджеров (рассчитать премию все механикам, которые за прошедший месяц все свои заказы выполнили своевременно - 10% от зарплаты).

Запрос:

```
CREATE VIEW timely_mechanics_bonus AS
SELECT e.fullname, p.salary * 0.1 AS bonus
FROM employee e
JOIN post p ON e."postCode" = p."postCode"
WHERE NOT EXISTS (
  SELECT 1
  FROM serviceslist sl
  JOIN contract c ON sl.contractnumber = c.contractnumber
  WHERE e.personalnum = sl.personalnum
    AND c.enddate > c.expectedenddate
    AND c.enddate >= current_date - INTERVAL '1 month'
)
AND EXISTS (
  SELECT 1
  FROM serviceslist sl
  JOIN contract c ON sl.contractnumber = c.contractnumber
  WHERE e.personalnum = sl.personalnum
    AND c.enddate <= c.expectedenddate
    AND c.enddate >= current_date - INTERVAL '1 month'
);
```

Вывод:

```
SELECT * FROM timely_mechanics_bonus;
```

	fullname character varying (50) 	bonus numeric 
1	Петров Петр Петров...	3000.0

Запросы на модификацию данных.

- 1) INSERT: добавить новые детали для BMW.

Запрос:

```
INSERT INTO details ( detailname, modelcode, producer)
SELECT 'Новая деталь', modelcode, 'Новый производитель'
FROM model
WHERE brand = 'BMW';
```

Вывод:

```
select * from details
```


	detailcode [PK] integer	detailname character varying (40)	modelcode integer	producer character varying (60)
1	1	Тормозные колодки	1	Brembo
2	8	Свечи	2	Bremdhr
3	9	Фара	3	ActiveHybrid
4	10	Подшипник	4	FEBI
5	11	Новая деталь	16	Новый производител...

2) UPDATE: Уменьшить на 10% цену на детали для BMW.

Запрос:

```
UPDATE price
SET price = price * 0.9
WHERE detailcode IN (
  SELECT detailcode
  FROM details d
  JOIN model m ON d.modelcode = m.modelcode
  WHERE m.brand = 'BMW'
);
```

Вывод:

select * from price

	detailcode [PK] integer	fromdate date	todate date	price integer
1	8	2023-01-01	2023-12-31	1000
2	9	2020-09-09	2021-09-09	1300
3	10	2021-09-09	2022-09-09	1300
4	13	2020-09-09	2021-09-09	1600
5	1	2023-01-01	2023-12-31	900
6	11	2022-09-09	2023-09-09	936
7	12	2021-02-03	2021-09-09	1170

3) DELETE: Удалить записи о заказах, выполненных более 5 лет назад.

Запрос:

```
DELETE FROM serviceslist
WHERE status='Выполнено' and contractnumber=(select contractnumber from
contract c where c.enddate >= current_date - INTERVAL '5 years')
```

Вывод:

select * from serviceslist

	services_id [PK] integer	contractnumber integer	servicecode integer	personalnum integer	status character varying (9)
1	1	1	1	1	В работе

Сравнение запросов с использованием индексирования.

Для демонстрации создания простого и составного индексов и сравнения времени выполнения запросов я выберу два запроса:

- 1) Запрос на получение данных с определенным номером машины.
- 2) Запрос на получение всех машин с указанной моделью и цветом.

Запросы без индексов:

EXPLAIN SELECT * FROM contract WHERE registrationnumber = 'A111BC112';

EXPLAIN SELECT * FROM car WHERE modelcode = 1 AND color = 'Черный';

Вывод к запросам без индексов:

	QUERY PLAN text
1	Seq Scan on contract (cost=0.00..16.38 rows=3 width=132)
2	Filter: ((registrationnumber)::text = 'A111BC112'::text)
	QUERY PLAN text
1	Seq Scan on car (cost=0.00..19.00 rows=1 width=1...)
2	Filter: (((modelcode = 1) AND ((color)::text = 'Черны...

Создание индексов:

Создание простого индекса для столбца registrationnumber в таблице contract.

CREATE INDEX idx_contract_registrationnumber ON contract (registrationnumber);

Создание составного индекса для столбцов modelcode и color в таблице car.

CREATE INDEX idx_car_modelcode_color ON car (modelcode, color);

Запросы с использованием индексов:

EXPLAIN SELECT * FROM contract WHERE registrationnumber = 'A111BC112';

EXPLAIN SELECT * FROM car WHERE modelcode = 1 AND color = 'Черный';

Выводы к запросам с использованием индексов:

	QUERY PLAN text	🔒
	QUERY PLAN text	🔒
1	Seq Scan on car (cost=0.00..1.07 rows=1 width=106)	
2	Filter: ((modelcode = 1) AND ((color)::text = 'Черный':tex...	

После выполнения запросов бази с индексами стало понятно, что правильно использованные индексы заметно улучшают производительность. Еще заметней разница будет на больших объемах данных. Также нужно учесть, что создание и поддержка индексов может замедлить операции записи в базе данных, поэтому следует использовать их с умом.

Выводы:

В ходе выполнения данной лабораторной работе были реализованы запросы на выборку данных и представления к базе данных на PostgreSQL согласно индивидуальному заданию. Более того, были смоделированы различные штатные ситуации и имплементированы 3 запроса на модификацию данных. Также был проведен анализ графического представления всех запросов. Созданы простые и составные индексы для различных запросов и проанализировано их время выполнения с использованием индексов.