

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
(Университет ИТМО)

Факультет **Инфокоммуникационных технологий**

Образовательная программа **Интеллектуальные системы в гуманитарной сфере**

Направление подготовки (специальность) **45.03.04 Интеллектуальные системы в гуманитарной сфере**

Лабораторная работа №2

по дисциплине «Проектирование и реализация баз данных»

тема: «Запросы на выборку данных к БД PostgreSQL»

Обучающийся: Шикалова Софья Сергеевна, К3242

Работа выполнена с оценкой _____

Преподаватель (и): Говорова М.М.

(подпись)

Дата 15.04.2022

Цель работы: Овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL и 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. Посмотреть историю запросов.

Схема, полученная с помощью Generate ERD.

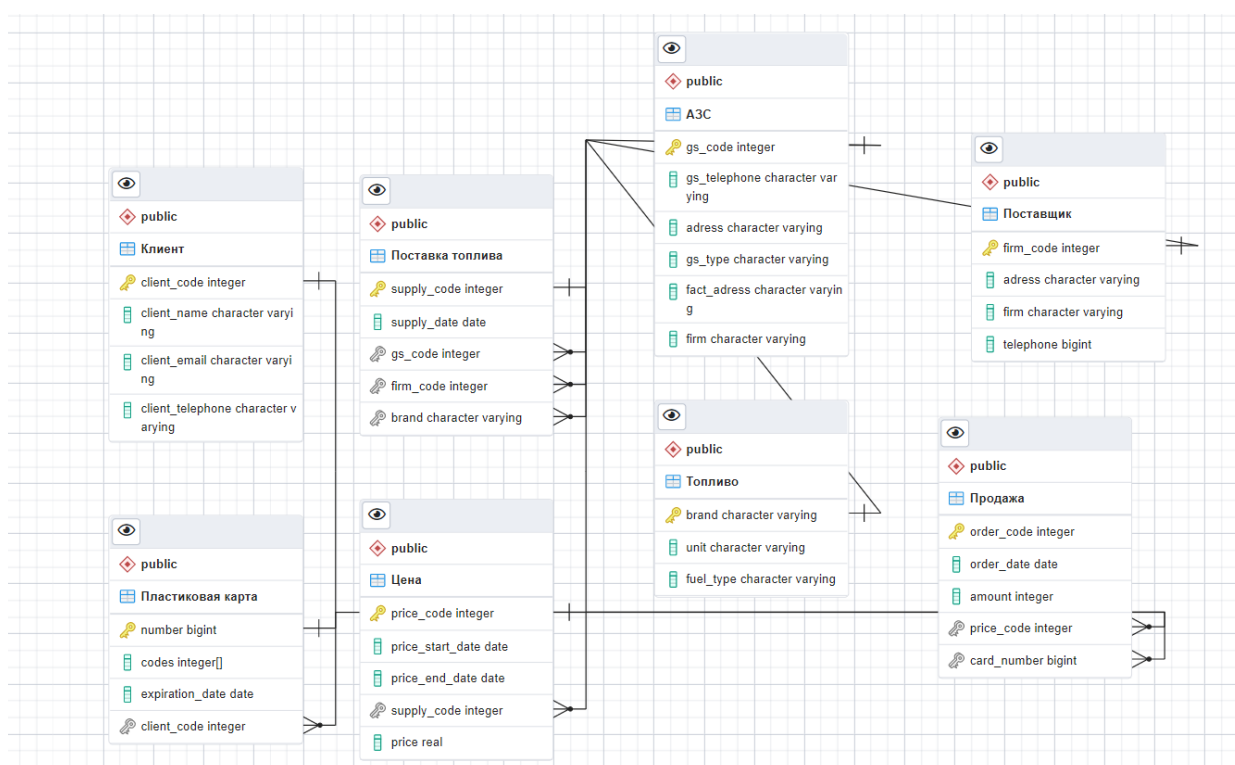


Рисунок 1 — Схема логической модели БД

Ход выполнения работы:

1) Запросы к базе данных:

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

```

SELECT
    CLIENT.client_name AS "Имя клиента",
    COUNT(SELL.order_date) AS "Количество"
FROM public.Клиент CLIENT
INNER JOIN public."Пластиковая карта" CARD ON CLIENT.client_code =
CARD.client_code
INNER JOIN public.Продажа SELL ON SELL.card_number = CARD.number
WHERE SELL.order_date > '2021-12-15' AND SELL.order_date < '2022-02-01'
GROUP BY CLIENT.client_name
    
```

	Имя клиента character varying	Количество bigint
1	Arkadiy Maliy	2
2	Oleg Kotov	2
3	Ulyana Terehova	2
4	Zulhiza Galiakberdina	1

Рисунок 2 — Вывод первого запроса

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

```
SELECT CLIENT.client_name AS "Имя клиента"
FROM public.Клиент CLIENT
INNER JOIN public."Пластиковая карта" CARD ON CLIENT.client_code =
CARD.client_code
INNER JOIN public.Продажа SELL ON SELL.card_number = CARD.number
WHERE ((EXTRACT(YEAR from SELL.order_date) != EXTRACT(YEAR from NOW())) OR
(((EXTRACT(YEAR from SELL.order_date) = EXTRACT(YEAR from NOW()) AND
(EXTRACT(MONTH from SELL.order_date) = 1))
GROUP BY CLIENT.client_name
```

	Имя клиента character varying
1	Arkadiy Maliy
2	Maksim Volkov
3	Masha Korobkina

Рисунок 3 — Вывод второго запроса

3. Найти клиента, купившего наибольший объем топлива по всей сети.

```
SELECT
    CLIENT.client_name AS "Имя клиента",
    SUM(SELL.amount) AS "Объём"
FROM public.Клиент CLIENT
INNER JOIN public."Пластиковая карта" CARD ON CLIENT.client_code =
CARD.client_code
INNER JOIN public.Продажа SELL ON SELL.card_number = CARD.number
GROUP BY CLIENT.client_name
HAVING SUM(SELL.amount) = (
    SELECT MAX(MAX_TABLE.sum) FROM(
        SELECT SUM(SUM_TABLE.amount), SUM_TABLE.client_name FROM (
            public.Клиент CLIENT
            INNER JOIN public."Пластиковая карта" CARD ON
CLIENT.client_code = CARD.client_code
```

```

INNER JOIN public.Продажа SELL ON SELL.card_number =
CARD.number
) SUM_TABLE
GROUP BY SUM_TABLE.client_name
) MAX_TABLE
)

```

	Объём bigint	Имя клиента character varying
1	40	Masha Korobkina

Рисунок 4 — Вывод третьего запроса

4. Вывести данные клиента, купившего топлива на наибольшую сумму в январе 2022.

```

SELECT *
FROM (SELECT
    CLIENT.client_code AS "Код клиента",
    CLIENT.client_name AS "Имя клиента",
    CLIENT.client_email AS "Почта клиента",
    CLIENT.client_telephone AS "Телефон клиента",
    SUM(PRICE.price * SELL.amount) AS "Траты"
FROM public.Клиент CLIENT
INNER JOIN public."Пластиковая карта" CARD ON CLIENT.client_code =
CARD.client_code
INNER JOIN public.Продажа SELL ON SELL.card_number = CARD.number
INNER JOIN public.Цена PRICE ON PRICE.price_code = SELL.price_code
GROUP BY CLIENT.client_name, CLIENT.client_code,
    CLIENT.client_email, CLIENT.client_telephone) TAB
WHERE TAB.Траты IN (
    SELECT MAX(MAX_TABLE.sum) FROM(
        SELECT SUM(SUM_TABLE.price * SUM_TABLE.amount),
SUM_TABLE.client_name FROM (
            public.Клиент CLIENT
            INNER JOIN public."Пластиковая карта" CARD ON
CLIENT.client_code = CARD.client_code
            INNER JOIN public.Продажа SELL ON SELL.card_number =
CARD.number
            INNER JOIN public.Цена PRICE ON PRICE.price_code =
SELL.price_code
        ) SUM_TABLE
        WHERE (EXTRACT(YEAR from SUM_TABLE.order_date) == 2022 and
EXTRACT(MONTH from SUM_TABLE.order_date) == 1)
        GROUP BY SUM_TABLE.client_name
    ) MAX_TABLE
)

```

	Код клиента integer	Имя клиента character varying	Почта клиента character varying	Телефон клиента character varying	Траты double precision
1	7	Ulyana Terehova	kiss@mail.ru	79528493847	1369.45000076294

Рисунок 5 — Вывод четвёртого запроса

5. Какое топливо пользуется наибольшим спросом в январе на АЗС поставщика «Лукойл»?

```
SELECT "Поставщик", "Код поставщика", "Топливо", "Тип топлива"
FROM (SELECT
    SUPPLIER.firm AS "Поставщик",
    SUPPLIER.firm_code AS "Код поставщика",
    FUEL.brand AS "Топливо",
    FUEL.fuel_type AS "Тип топлива",
    SELL.order_date AS "Дата покупки"
FROM public."Поставка топлива" SUPPLY
INNER JOIN public.Поставщик SUPPLIER ON SUPPLIER.firm_code = SUPPLY.firm_code
INNER JOIN public.Топливо FUEL ON FUEL.brand = SUPPLY.brand
INNER JOIN public.Цена PRICE ON PRICE.supply_code = SUPPLY.supply_code
INNER JOIN public.Продажа SELL ON PRICE.price_code = SELL.price_code
GROUP BY SUPPLIER.firm, SUPPLIER.firm_code, SELL.order_date,
    FUEL.brand, FUEL.fuel_type) TAB
WHERE TAB."Код поставщика" = 200 AND (EXTRACT(YEAR from TAB. "Дата покупки")
= 2022 AND EXTRACT(MONTH from TAB. "Дата покупки") == 1)
GROUP BY TAB.Поставщик, TAB."Код поставщика", TAB.Топливо, TAB."Тип топлива",
TAB."Дата покупки"
ORDER BY COUNT(TAB.Топливо) DESC
LIMIT 1
```

	Поставщик character varying	Код поставщика integer	Топливо character varying	Тип топлива character varying
1	Lukoil	200	ai-100	ecto

Рисунок 6 — Вывод пятого запроса

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

```
SELECT TAB.Поставщик, TAB."Код АЗС", TAB.Топливо, SUM("Объём")
FROM (SELECT
    SUPPLIER.firm AS "Поставщик",
    GS.gs_code AS "Код АЗС",
    FUEL.brand AS "Топливо",
    SELL.amount AS "Объём",
    SELL.order_date AS "Дата покупки"
FROM public."Поставка топлива" SUPPLY
INNER JOIN public.АЗС GS ON GS.gs_code = SUPPLY.gs_code
INNER JOIN public.Поставщик SUPPLIER ON SUPPLIER.firm_code = SUPPLY.firm_code
INNER JOIN public.Топливо FUEL ON FUEL.brand = SUPPLY.brand
INNER JOIN public.Цена PRICE ON PRICE.supply_code = SUPPLY.supply_code
INNER JOIN public.Продажа SELL ON PRICE.price_code = SELL.price_code
GROUP BY SUPPLIER.firm, SUPPLIER.firm_code, SELL.order_date, SELL.amount,
    FUEL.brand, FUEL.fuel_type, GS.gs_code) TAB
WHERE (EXTRACT(YEAR from TAB. "Дата покупки") = 2022 and EXTRACT(MONTH
from TAB. "Дата покупки") == 1)
GROUP BY TAB.Поставщик, TAB."Код АЗС", TAB.Топливо
```

	Поставщик character varying	Код АЗС integer	Топливо character varying	sum bigint
1	Gasprom	1	ai-92	12
2	Lukoil	2	ai-100	32
3	Lukoil	2	ai-92	12

Рисунок 7 — Вывод шестого запроса

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

```

SELECT TAB.Поставщик, TAB."Код АЗС", SUM(TAB.Выручка) AS "Выручка"
FROM (SELECT
    SUPPLIER.firm AS "Поставщик",
    GS.gs_code AS "Код АЗС",
    FUEL.brand AS "Топливо",
    SELL.amount AS "Объём",
    SELL.order_date AS "Дата покупки",
    SUM(PRICE.price * SELL.amount) AS "Выручка"
FROM public."Поставка топлива" SUPPLY
INNER JOIN public.АЗС GS ON GS.gs_code = SUPPLY.gs_code
INNER JOIN public.Поставщик SUPPLIER ON SUPPLIER.firm_code = SUPPLY.firm_code
INNER JOIN public.Топливо FUEL ON FUEL.brand = SUPPLY.brand
INNER JOIN public.Цена PRICE ON PRICE.supply_code = SUPPLY.supply_code
INNER JOIN public.Продажа SELL ON PRICE.price_code = SELL.price_code
GROUP BY SUPPLIER.firm, SUPPLIER.firm_code, SELL.order_date, SELL.amount,
    FUEL.brand, FUEL.fuel_type, GS.gs_code) TAB
GROUP BY TAB.Поставщик, TAB."Код АЗС"
ORDER BY SUM(TAB.Выручка) DESC
LIMIT 1

```

	Поставщик character varying	Код АЗС integer	Выручка double precision
1	Lukoil	2	3902.92002105713

Рисунок 8 — Вывод седьмого запроса

2) Модификация данных:

1. Добавить новую запись о поставке на ту же заправку, как и для поставки с кодом 193.

```

INSERT INTO public."Поставка топлива" (supply_code, supply_date, gs_code, firm_code,
brand)
VALUES (362, '2022-04-01',
    (SELECT gs_code FROM public."Поставка топлива" WHERE supply_code = 193),
    (SELECT firm_code FROM public."Поставка топлива" WHERE supply_code = 193),
    'ai-98')

```

	supply_code [PK] integer	supply_date date	gs_code integer	firm_code integer	brand character varying
1	101	2021-12-18	2	200	ai-92
2	111	2021-11-30	2	200	ai-80
3	150	2022-01-07	4	300	ai-92
4	193	2021-12-02	3	300	ai-80
5	244	2021-12-11	3	300	ai-95
6	291	2022-03-02	1	100	ai-92
7	331	2022-03-01	2	200	ai-100
8	362	2022-04-01	3	300	ai-98
9	756	2022-03-01	4	300	ai-98

Рисунок 9 — запрос INSERT

1) Заменить номер карты в покупке с кодом 62993 на номер из покупки с кодом 49303.

```
UPDATE public."Продажа"
SET card_number = (
    SELECT card_number FROM public."Продажа"
    WHERE order_code = 49303)
WHERE order_code = 62993
```

	order_code [PK] integer	order_date date	amount integer	price_code integer	card_number bigint
1	14527	2022-03-02	10	1	4650220237804858
2	28393	2022-01-17	13	4	3850263737001604
3	28930	2022-04-08	15	5	4650220237804858
4	28942	2022-02-28	15	2	4650220237804858
5	30288	2021-12-27	15	7	3909234506709236
6	35644	2022-01-18	12	7	3850263737001604
7	47828	2022-01-26	5	3	4056924537589405
8	49303	2022-01-13	12	4	3678023450198645
9	53488	2021-12-01	8	6	8503748501845367
10	55256	2022-01-11	7	3	4056924537589405
11	62553	2022-02-27	5	4	8503748501845367
12	62993	2022-01-04	7	4	3678023450198645
13	63859	2022-02-28	10	3	8503748501845367

Рисунок 10 — запрос UPDATE

2) Удалить клиентов, к аккаунтам которых не привязаны карты.

```
DELETE FROM public."Клиент"
WHERE client_code NOT IN (
```

```
SELECT client_code FROM public."Пластиковая карта")
RETURNING *
```

	client_code [PK] integer	client_name character varying	client_email character varying	client_telephone character varying
1	1	Masha Korobkina	korobk@gmail.com	79556439283
2	2	Gena Alyoshin	bigboss@rambler.ru	79633934051
3	3	Arkadiy Maliy	malenkiy@mail.ru	79992303120
4	4	Zulhiza Galiakberdina	tatarochka@mail.ru	79133425463
5	5	Maksim Volkov	aufauf@gmail.com	79992035263
6	6	Oleg Kotov	meowmeow@gmail.com	79576403629
7	7	Ulyana Terehova	kiss@mail.ru	79528493847
8	8	Oleg Ogurtsov	loveogurchiki@yandex.ru	79993756164

Рисунок 11 — запрос DELETE (до)

	client_code [PK] integer	client_name character varying	client_email character varying	client_telephone character varying
1	2	Gena Alyoshin	bigboss@rambler.ru	79633934051
2	8	Oleg Ogurtsov	loveogurchiki@yandex.ru	79993756164

Рисунок 12 — запрос DELETE (удалённые строки)

3) Представления:

1. Сведения обо всех АЗС и всех видах топлива, которые они продают.

```
CREATE OR REPLACE VIEW public."АЗС и топливо" AS
SELECT
  GS.gs_code AS "Код АЗС",
  GS.gs_telephone AS "Телефон",
  GS.adress AS "Юр. адрес",
  GS.fact_adress AS "Факт. адрес",
  GS.firm AS "Фирма",
  FUEL.brand AS "Бренд топлива",
  FUEL.unit AS "Единица измерения",
  FUEL.fuel_type AS "Тип топлива"
FROM public."АЗС" GS
JOIN public."Поставка топлива" SUPPLY ON GS.gs_code = SUPPLY.gs_code
JOIN public."Топливо" FUEL ON SUPPLY.brand = FUEL.brand
ORDER BY GS.gs_code
```

	Код АЗС integer	Телефон character varying	Юр. адрес character varying	Факт. адрес character varying	Фирма character varying	Бренд топлива character varying	Единица измерения character varying	Тип топлива character varying
1	1	79748132655	Gas ave., 13/1	Pushkina st., 19	Tatneft	ai-92	liters	regular
2	2	79994056575	Morgenshterna lane, 15, office 3	Rossiyskaya nab., 115	Lukoil	ai-100	liters	ecto
3	2	79994056575	Morgenshterna lane, 15, office 3	Rossiyskaya nab., 115	Lukoil	ai-80	liters	normal
4	2	79994056575	Morgenshterna lane, 15, office 3	Rossiyskaya nab., 115	Lukoil	ai-92	liters	regular
5	3	79033541622	Salavata Yulayeva ave., 19, block 4	Sagita Agisha st., 20	Rosneft	ai-80	liters	normal
6	3	79033541622	Salavata Yulayeva ave., 19, block 4	Sagita Agisha st., 20	Rosneft	ai-98	liters	extra
7	3	79033541622	Salavata Yulayeva ave., 19, block 4	Sagita Agisha st., 20	Rosneft	ai-95	liters	premium
8	4	78049467343	Morgenshterna lane, 15, office 3	Solnechnaya st., 100	Lukoil	ai-98	liters	extra
9	4	78049467343	Morgenshterna lane, 15, office 3	Solnechnaya st., 100	Lukoil	ai-92	liters	regular

Рисунок 13 — первое представление

2. Самая прибыльная АЗС за январь 2022 для каждого производителя.

```
CREATE OR REPLACE VIEW public."Заправочные станции" AS
SELECT
  SUPPLIER.firm AS "Производитель",
  SUM(SELL.amount * PRICE.price) AS "Выручка",
  GS.gs_code AS "Код АЗС",
  GS.gs_telephone AS "Телефон",
  GS.adress AS "Юр. адрес",
  GS.fact_address AS "Факт. адрес"
FROM public."АЗС" GS
JOIN public."Поставка топлива" SUPPLY ON GS.gs_code = SUPPLY.gs_code
JOIN public."Поставщик" SUPPLIER ON SUPPLY.firm_code = SUPPLIER.firm_code
JOIN public."Цена" PRICE ON SUPPLY.supply_code = PRICE.supply_code
JOIN public."Продажа" SELL ON PRICE.price_code = SELL.price_code
GROUP BY SUPPLIER.firm, GS.gs_code, SELL.order_date
HAVING SUM(SELL.amount*PRICE.price) IN (
  SELECT sum FROM(
    SELECT SUM(SUM_TABLE.amount*SUM_TABLE.price), SUM_TABLE.fact_address
  FROM (
    public.АЗС GS
    JOIN public."Поставка топлива" SUPPLY ON GS.gs_code = SUPPLY.gs_code
    JOIN public."Поставщик" SUPPLIER ON SUPPLY.firm_code =
SUPPLIER.firm_code
    JOIN public."Цена" PRICE ON SUPPLY.supply_code = PRICE.supply_code
    JOIN public."Продажа" SELL ON PRICE.price_code = SELL.price_code
  ) SUM_TABLE
  GROUP BY SUM_TABLE.fact_address
  WHERE (EXTRACT(YEAR from SUM_TABLE.order_date) = 2022 and
EXTRACT(MONTH from SUM_TABLE.order_date) = 1)
  ) MAX_TABLE
)
ORDER BY GS.gs_code
```

	Производитель character varying	Выручка double precision	Код АЗС integer	Телефон character varying	Юр. адрес character varying	Факт. адрес character varying
1	Rosneft	991.94995880127	4	78049467343	Morgenshterna lane, 15, office 3	Solnechnaya st., 100

Рисунок 14 — второе представление

4) Индексы:

1. Без индекса

```
EXPLAIN SELECT * FROM public.Продажа WHERE order_date < (NOW() - INTERVAL '4
month')
```

	QUERY PLAN
	text
1	Seq Scan on "Продажа" (cost=0.00..1.23 rows=4 width=24)
2	[...] Filter: (order_date < (now() - '4 mons'::interval))

Рисунок 15 — запрос 1, без индекса, время: 941 м/с

EXPLAIN SELECT * FROM public.A3C WHERE firm =
 (SELECT firm FROM public.Поставщик WHERE firm_code = 200)

	QUERY PLAN
	text
1	Seq Scan on "A3C" (cost=8.17..23.42 rows=2 width=164)
2	[...] Filter: ((firm)::text = (\$0)::text)
3	[...] InitPlan 1 (returns \$0)
4	[...] -> Index Scan using "Supplier_pkey" on "Поставщик" (с...
5	[...] Index Cond: (firm_code = 200)

Рисунок 15 — запрос 2, без индекса, время: 321 м/с

2. Создание индекса

i. CREATE INDEX id_order ON public."Продажа"(order_code)

ii. CREATE INDEX id_firm ON public."Поставщик"(firm_code)

3. С индексом:

Первый запрос был выполнен за 201 м/с, а второй за 173 м/с.

Вывод:

В ходе выполнения данной лабораторной работы я получила данные из предварительно подготовленной базы данных согласно запросам, указанным в условиях.