

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 теме: Запросы на выборку данных к БД PostgreSQL.  
Представления в PostgreSQL.  
по дисциплине: Проектирование и реализация баз  
данных

Специальность:

09.03.03 Мобильные и сетевые технологии

Проверил:

Говорова М.М. \_\_\_\_\_

Дата: «13» мая 2021г.

Оценка \_\_\_\_\_

Выполнил:

студент  
группы К3241

Шутов Д. Э.

Санкт-Петербург 2021 г.

**Цель работы:** овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL и использования подзапросов при модификации данных.

**Оборудование:** компьютерный класс.

**Программное обеспечение:** СУБД PostgreSQL, pgadmin 4.

**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.

**Индивидуальное задание:**

Вариант №9 “Оптовая база”

## ВЫПОЛНЕНИЕ

### 1. Название БД «Wholesale base»

### 2. Схема инфологической модели данных БД

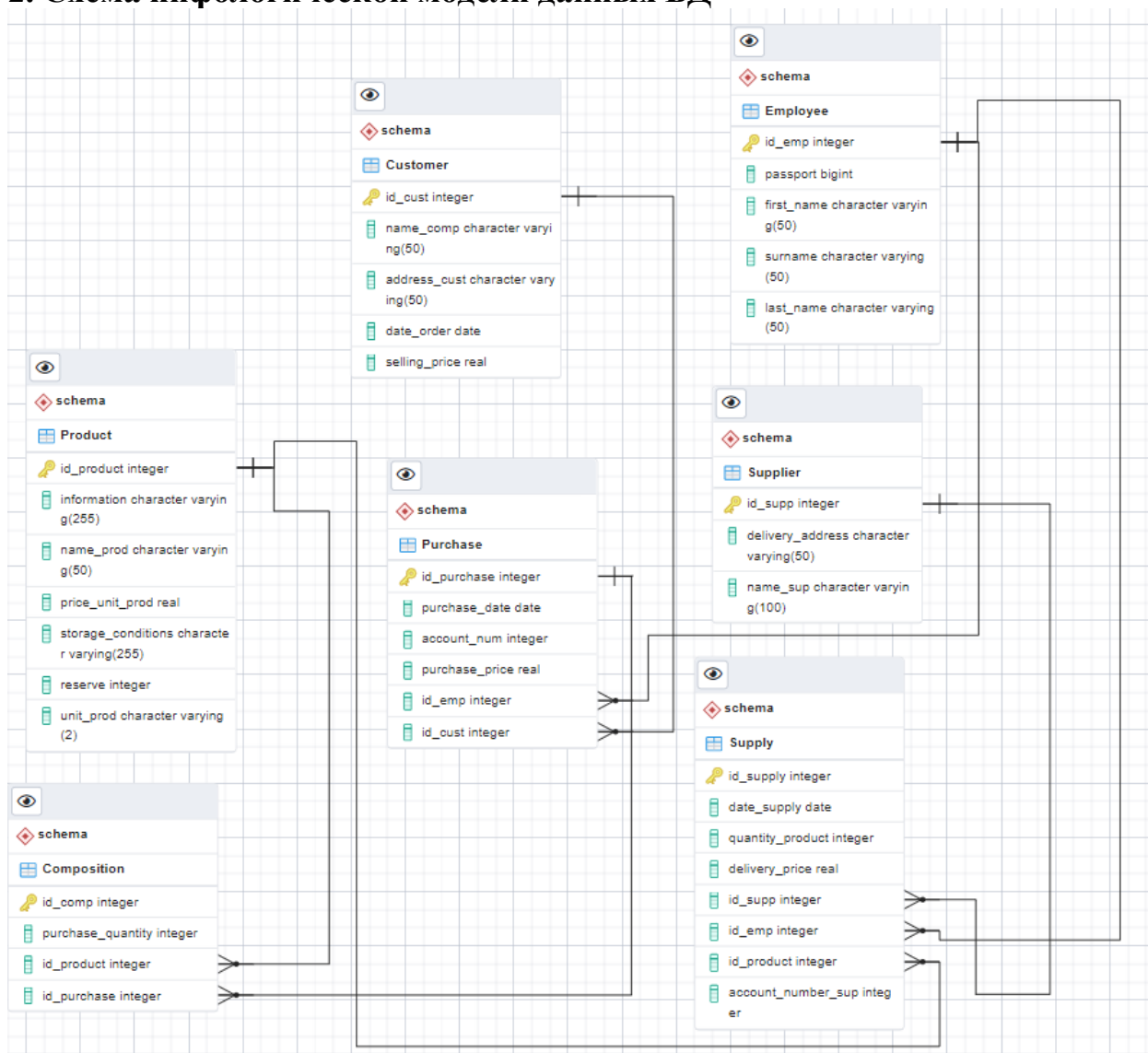


Рисунок 1 – Схема инфологической модели БД, сгенерированная в Generate ERD

### 3. Запросы к базе данных

- Вывести список поставщиков, которые поставляют все товары.

Wholesale base/postgres@PostgreSQL 11

Query Editor История запросов

```
1 SELECT schema."Supplier".name_sup FROM
2     (SELECT id_supp FROM (SELECT id_supp, COUNT (DISTINCT id_product)AS count_products
3     FROM schema."Supply" GROUP BY id_supp) AS tab
4     WHERE (count_products = (SELECT COUNT(id_product) FROM schema."Product"))AS tab1
5 INNER JOIN schema."Supplier" ON tab1.id_supp = schema."Supplier".id_supp;
```

Результат План выполнения Сообщения Notifications

	name_sup	
	character varying (100)	
1	Аврора	

- Определить поставщика, который поставляет каждый из товаров по самой низкой цене.

Wholesale base/postgres@PostgreSQL 11

Query Editor История запросов

```
1 SELECT schema."Supplier".name_sup FROM
2     (SELECT id_supp, COUNT(id_product)
3     FROM (SELECT id_product, id_supp, MIN(delivery_price) AS min_price
4     FROM schema."Supply" GROUP BY id_product, id_supp)as foo GROUP BY id_supp
5     HAVING COUNT(id_product) = (SELECT COUNT(DISTINCT id_product) FROM schema."Supply")) AS tab
6 INNER JOIN schema."Supplier" ON tab.id_supp = schema."Supplier".id_supp;
```

Результат План выполнения Сообщения Notifications

	name_sup	
	character varying (100)	
1	Аврора	

- Вывести названия товаров, цены на которые у всех поставщиков одинаковы.

Wholesale base/postgres@PostgreSQL 11

Query Editor История запросов

```
1 SELECT schema."Product".name_prod FROM
2     (SELECT id_product, COUNT(DISTINCT delivery_price) AS sum FROM schema."Supply" GROUP BY id_product) f_result, schema."Product"
3 where schema."Product".id_product = f_result.id_product
```

Результат План выполнения Сообщения Notifications

	name_prod	
	character varying (50)	
1	горбыль	
2	необрезная доска	
3	обрезная доска	
4	четвертина	
5	пластина	

- Чему равен общий суточный доход оптового склада за прошедший день

Wholesale base/postgres@PostgreSQL 11 ▾	
Query Editor История запросов	
1	SELECT SUM(purchase_price) AS result
2	FROM schema."Purchase"
3	WHERE purchase_date = TIMESTAMP 'yesterday';
Результат План выполнения Сообщения Notifications	
	<div>result</div> <div>real</div>
1	55465

- Вычислить стоимость товаров, оставшихся на складе

Wholesale base/postgres@PostgreSQL 11 ▾	
Query Editor История запросов	
1	SELECT name_prod, price_unit_prod*reserve AS result
2	FROM schema."Product"
3	WHERE reserve > 0;
Результат План выполнения Сообщения Notifications	
	<div>name_prod</div> <div>character varying (50)</div> <div>result</div> <div>double precision</div>
1	пластина 150000
2	четвертина 200000
3	горбыль 250000
4	обрезная доска 230000
5	необрезная доска 300000

- В какой день было вывезено минимальное количество товара

Wholesale base/postgres@PostgreSQL 11 ▾	
Query Editor История запросов	
1	SELECT date_supply, quantity_product
2	FROM schema."Supply"
3	WHERE quantity_product = (SELECT min(quantity_product) FROM schema."Supply");
Результат План выполнения Сообщения Notifications	
	<div>date_supply</div> <div>date</div> <div>quantity_product</div> <div>integer</div>
1	2020-01-15 10
2	2020-01-15 10
3	2020-01-15 10
4	2020-01-16 10

- Сколько различных видов товара имеется на базе

Wholesale base/postgres@PostgreSQL 11

Query Editor

```

1 SELECT SUM(qty) AS quantity
2 FROM (SELECT COUNT(DISTINCT name_prod) AS qty FROM schema."Product" GROUP BY id_product) AS tab;
3
4

```

Query Editor История запросов План выполнения

Результат Сообщения Notifications

	quantity numeric
1	5

## 4. Представления

### 1) Количество заказов фирм-покупателей за прошедший год

Wholesale base/postgres@PostgreSQL 11

Query Editor История запросов

```

1 CREATE VIEW schema.comp AS
2 SELECT sum(count_purchase) AS quantity_purchase
3     FROM (SELECT schema."Customer".name_comp, tab.count_purchase
4           FROM (SELECT id_cust, COUNT (id_cust)AS count_purchase
5                 FROM schema."Purchase"
6                 WHERE purchase_date BETWEEN '2020-01-01' AND '2020-12-31'
7                 GROUP BY id_cust) as tab
8     INNER JOIN schema."Customer" ON tab.id_cust = schema."Customer".id_cust) as result;

```

Результат План выполнения Сообщения Notifications

CREATE VIEW

Запрос завершён успешно, время выполнения: 121 msec.

Вид:

Wholesale base/postgres@PostgreSQL 11

Query Editor История запросов

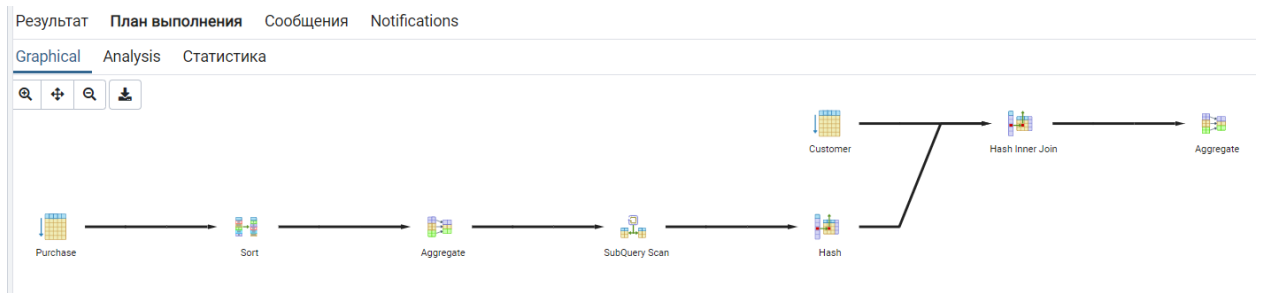
```

1 SELECT quantity_purchase
2     FROM schema.comp;

```

Результат План выполнения Сообщения Notifications

	quantity_purchase numeric
1	8



## 2) Доход базы за конкретный период.

Wholesale base/postgres@PostgreSQL 11 ▾

Query Editor История запросов

```

1 CREATE VIEW schema.warehouse AS
2 SELECT sum(purchase_price) AS proceeds
3 FROM schema."Purchase"
4 where purchase_date BETWEEN '2020-01-01' AND '2020-12-31';
  
```

Результат План выполнения Сообщения Notifications

CREATE VIEW

Запрос завершён успешно, время выполнения: 93 msec.

Вид:

Wholesale base/postgres@PostgreSQL 11 ▾

Query Editor История запросов

```

1 SELECT proceeds
2 FROM schema.warehouse;
  
```

Результат План выполнения Сообщения Notifications

	proceeds	
	real	
1	69145	

Wholesale base/postgres@PostgreSQL 11 ▾

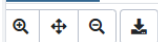
Query Editor История запросов

```

1 SELECT proceeds
2 FROM schema.warehouse;
  
```

Результат План выполнения Сообщения Notifications

Graphical Analysis Статистика



Purchase

Aggregate

## 5. Запросы на модификацию данных

### 1) INSERT

Wholesale base/postgres@PostgreSQL 11 ▾




Query Editor История запросов

```
1 INSERT INTO schema."Employee"(  
2     id_emp, passport, first_name, surname, last_name)  
3     VALUES (5, 1243432123, 'Егор', 'Зубов', 'Андреевич');
```






Результат План выполнения Сообщения Notifications

INSERT 0 1

До:

Результат	План выполнения		Сообщения	Notifications		
	id_emp [PK] integer	 passport bigint	first_name character varying (50)	 surname character varying (50)	last_name character varying (50)	
1	1	1000000009	Артём	Власов	Александрович	
2	2	2000000009	Егор	Зубов	Витальевич	
3	3	3000000009	Никита	Тихоненков	Алексеевич	
4	4	4000000009	Глеб	Петров	Никитович	

После:

Результат	План выполнения		Сообщения	Notifications	
	 id_emp [PK] integer	 passport bigint	 first_name character varying (50)	 surname character varying (50)	 last_name character varying (50)
1	1	1000000009	Артём	Власов	Александрович
2	2	2000000009	Егор	Зубов	Витальевич
3	3	3000000009	Никита	Тихоненков	Алексеевич
4	4	4000000009	Глеб	Петров	Никитович
5	5	1243432123	Егор	Зубов	Андреевич

### 2) UPDATE

Wholesale base/postgres@PostgreSQL 11 ▾

Query Editor История запросов

```
1 UPDATE schema."Composition"  
2     SET purchase_quantity = 131  
3     WHERE purchase_quantity = 10 ;
```

Результат План выполнения Сообщения Notifications





UPDATE 1

Запрос завершён успешно, время выполнения: 159 msec.







До:

Результат   План выполнения   Сообщения   Notifications

	 id_comp [PK] integer	 purchase_quantity integer	 id_product integer	 id_purchase integer
1	1	10	1	1
2	2	20	2	2
3	3	30	3	3
4	4	40	2	1

После:

Результат   План выполнения   Сообщения   Notifications

	 id_comp [PK] integer	 purchase_quantity integer	 id_product integer	 id_purchase integer
1	2	20	2	2
2	3	30	3	3
3	4	40	2	1
4	1	131	1	1

### 3) DELETE

Wholesale base/postgres@PostgreSQL 11 ▾

Query Editor   История запросов









```
1 DELETE FROM schema."Supply"
2 WHERE delivery_price = 150
```

Результат   План выполнения   Сообщения   Notifications










DELETE 1

Запрос завершён успешно, время выполнения: 65 msec.

До:

	 id_supply [PK] integer	 date_supply date	 quantity_product integer	 delivery_price real	 id_supp integer	 id_emp integer	 id_product integer	 account_number_sup integer
1	8	2020-01-15	10	100	2	2	5	67766
2	5	2020-01-15	10	150	2	2	1	23436
3	7	2020-01-15	10	110	2	2	4	46744
4	6	2020-01-16	10	500.5	2	2	3	38974
5	4	2020-10-01	34	130	1	4	1	40123
6	2	2020-01-15	30	590	2	2	2	20123
7	3	2020-01-15	32	550	3	3	3	30123
8	9	2020-02-13	13	480	2	3	3	13432
9	1	2020-01-10	30	490	2	1	3	10123
10	10	2020-02-02	12	90	2	1	1	32343
11	11	2020-02-12	12	54	1	3	2	32422

После:

Результат	План выполнения		Сообщения	Notifications				
	 id_supply [PK] integer	 date_supply date	 quantity_product integer	 delivery_price real	 id_supp integer	 id_emp integer	 id_product integer	 account_number_sup integer
1	8	2020-01-15	10	100	2	2	5	67766
2	7	2020-01-15	10	110	2	2	4	46744
3	6	2020-01-16	10	500.5	2	2	3	38974
4	4	2020-10-01	34	130	1	4	1	40123
5	2	2020-01-15	30	590	2	2	2	20123
6	3	2020-01-15	32	550	3	3	3	30123
7	9	2020-02-13	13	480	2	3	3	13432
8	1	2020-01-10	30	490	2	1	3	10123
9	10	2020-02-02	12	90	2	1	1	32343
10	11	2020-02-12	12	54	1	3	2	32422

Вывод:

Созданы запросы и представления на выборку данных к базе данных Wholesale base (согласно индивидуальному заданию, часть 2 и 3). Составлены 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов. Изучить графическое представление запросов. Просмотреть историю запросов.