

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

Отчет

по лабораторной работе №5 «Процедуры, функции, триггеры в PostgreSQL»

по дисциплине «Проектирование и реализация баз данных»

Автор: Леманов А.А.

Факультет: ИКТ

Группа: K3241

Преподаватель: Говорова М.М.



Санкт-Петербург 2023

Оглавление

Цель работы:	3
Практическое задание	3
Выполнение	3
Вывод	8

Цель работы: овладеть практическими создания и использования процедур, функций и триггеров в базе данных PostgreSQL.

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL, SQL Shell (psql).

Практическое задание:

Вариант 2 (max - 8 баллов)

1. Создать процедуры/функции согласно индивидуальному заданию (часть 4).
2. Создать авторский триггер по варианту индивидуального задания.

Выполнение

Создайте хранимые процедуры:

1. **Вывести сведения обо всех покупках одного из клиентов за заданную дату (данные клиента, дата, объем топлива, уплаченная сумма).**

```
CREATE OR REPLACE FUNCTION lab3.view_client_info(client_id_input INTEGER,  
date_input DATE)
```

```
RETURNS TABLE (
```

```
client_name CHARACTER VARYING(50),
```

```
client_address CHARACTER VARYING(50),
```

```
client_phone_number CHARACTER VARYING(50),
```

```
purchase_date DATE,
```

```
fuel_amount DOUBLE PRECISION,
```

```
total_amount DOUBLE PRECISION
```

```
) AS
```

```
$$
```

```
BEGIN
```

```
RETURN QUERY
```

```
SELECT client.*, date, SUM(amount), SUM(amount * fuel.price_per_liter)  
FROM lab3.fuel_sales
```

```
JOIN lab3.fuel ON fuel_sales.id_fuel = fuel.code
```



```

JOIN lab3.client_card ON fuel_sales.card_number = client_card.card_number

JOIN lab3.client ON client_card.client_phone_number = client.phone_number

WHERE client_card.card_number = client_id_input AND date = date_input

GROUP BY client.phone_number, client.full_name, client.address, fuel_sales.date;

END;

$$

LANGUAGE plpgsql;

```

```

lab3=# CREATE OR REPLACE FUNCTION lab3.view_client_info(client_id_input INTEGER, date_input DATE) RETURNS TABLE (
lab3(#   client_name CHARACTER VARYING(50),
lab3(#   client_address CHARACTER VARYING(50),
lab3(#   client_phone_number CHARACTER VARYING(50),
lab3(#   purchase_date DATE,
lab3(#   fuel_amount DOUBLE PRECISION,
lab3(#   total_amount DOUBLE PRECISION
lab3(# ) AS
lab3-# $BODY$
lab3$# BEGIN
lab3$#     RETURN QUERY
lab3$#     SELECT client.*, date, SUM(amount), SUM(amount * fuel.price_per_liter) FROM lab3.fuel_sales
lab3$# JOIN lab3.fuel ON fuel_sales.id_fuel = fuel.code
lab3$# JOIN lab3.client_card ON fuel_sales.card_number = client_card.card_number
lab3$# JOIN lab3.client ON client_card.client_phone_number = client.phone_number
lab3$# WHERE client_card.card_number = client_id_input AND date = date_input
lab3$# GROUP BY client.phone_number, client.full_name, client.address, fuel_sales.date;
lab3$# END;
lab3-# $BODY$
lab3-# LANGUAGE plpgsql;
CREATE FUNCTION

```

```

lab3=# SELECT * FROM lab3.view_client_info(2, date '2022-11-30');

```

client_name	client_address	client_phone_number	purchase_date	fuel_amount	total_amount
к. Октябрьское (Хант.), пр. Коммунальный, д. 82 к. 3/6, 217150	Афанасий Жоресович Федотов	+7 (476) 797-7926	2022-11-30	145	8540.06

(1 строка)

- **Количество видов топлива, поставляемых каждой фирмой-поставщиком.**

```

CREATE OR REPLACE FUNCTION lab3.firm_fuel_types() RETURNS TABLE (
title_name CHARACTER VARYING(50),
fuel_types_count BIGINT
) AS
$$
BEGIN
RETURN QUERY
SELECT firm.title, COUNT(DISTINCT fuels) AS fuel_types FROM lab3.gas_station

```



```
INNER JOIN lab3.firm ON gas_station.id_firm = firm.id_firm
```

```
GROUP BY firm.title;
```

```
END;
```

```
$$
```

```
LANGUAGE plpgsql;
```

```
lab3=# CREATE OR REPLACE FUNCTION lab3.firm_fuel_types() RETURNS TABLE (  
lab3(#      title_name CHARACTER VARYING(50),  
lab3(# fuel_types_count INTEGER  
lab3(# ) AS  
lab3-# $$  
lab3$# BEGIN  
lab3$#     RETURN QUERY  
lab3$#     SELECT firm.title, COUNT(DISTINCT fuels) AS fuel_types FROM lab3.gas_station  
lab3$#     INNER JOIN lab3.firm ON gas_station.id_firm = firm.id_firm  
lab3$#     GROUP BY firm.title;  
lab3$# END;  
lab3$# $$  
lab3-# LANGUAGE plpgsql;  
CREATE FUNCTION
```

```
lab3=# SELECT * FROM lab3.firm_fuel_types();  
  title_name  | fuel_types_count  
-----+-----  
ЗАО «Ермакова» |                1  
ИП «Куликов»   |                1  
ОАО «Крылова»  |                2  
ОАО «Морозова» |                3  
Соболев Лимитед |                3  
(5 строк)
```

- **Самый непопулярный вид топлива за прошедшую неделю**

```
CREATE OR REPLACE FUNCTION lab3.non_popular_week_fuel() RETURNS TABLE (
```

```
    fuel_name CHARACTER VARYING(50)
```

```
) AS
```

```
$$
```

```
BEGIN
```

```
    RETURN QUERY
```

```
    SELECT fuel.type FROM lab3.fuel_sales
```

```
    INNER JOIN lab3.fuel ON fuel_sales.id_fuel = fuel.code
```

```
    WHERE fuel_sales.date >= date - 7
```

```
    GROUP BY fuel.type
```



```
ORDER BY SUM(fuel_sales.amount) ASC
```

```
LIMIT 1;
```

```
END;
```

```
$$
```

```
LANGUAGE plpgsql;
```

```
lab3=# CREATE OR REPLACE FUNCTION lab3.non_popular_week_fuel() RETURNS TABLE (  
lab3(# title_name CHARACTER VARYING(50),  
lab3(# fuel_types_count BIGINT  
lab3(# ) AS  
lab3-# $$  
lab3$# BEGIN  
lab3$# RETURN QUERY  
lab3$# SELECT fuel.type FROM lab3.fuel_sales  
lab3$# INNER JOIN lab3.fuel ON fuel_sales.id_fuel = fuel.code  
lab3$# WHERE fuel_sales.date >= date - 7  
lab3$# GROUP BY fuel.type  
lab3$# ORDER BY SUM(fuel_sales.amount) ASC  
lab3$# LIMIT 1;  
lab3$# END;  
lab3$# $$  
lab3-# LANGUAGE plpgsql;  
CREATE FUNCTION
```

```
lab3=# SELECT * FROM lab3.non_popular_week_fuel();  
fuel_name  
-----  
АИ-98  
(1 строка)
```

Создайте необходимые триггеры:

- Триггер для автоматического обновления баланса карты клиента
BEGIN

```
CREATE OR REPLACE FUNCTION update_balance_after_fuel_purchase()
```

```
RETURNS TRIGGER AS $$
```

```
DECLARE
```

```
fuel_price DOUBLE PRECISION;
```

```
BEGIN
```

```
SELECT price_per_liter INTO fuel_price FROM lab3.fuel WHERE fuel_id =  
NEW.fuel_id;
```

```
UPDATE lab3.client_card
```

```
SET balance = balance - (NEW.amount * fuel_price)
```


WHERE card_number = NEW.card_number;

RETURN NEW;

END;

\$\$ LANGUAGE plpgsql;

CREATE TRIGGER new_update_balance_after_fuel_purchase

AFTER INSERT ON lab3.fuel_sales

FOR EACH ROW EXECUTE FUNCTION update_balance_after_fuel_purchase();

Data Output Messages Notifications				
	card_number [PK] integer	validity character varying	client_phone_number text	balance double precision
1	2	year	+7 (476) 797-7926	10000
2	3	year	+7 (484) 371-22-85	0
3	4	year	+7 (501) 568-82-80	0

Data Output Messages Notifications				
	card_number [PK] integer	validity character varying	client_phone_number text	balance double precision
1	2	year	+7 (476) 797-7926	3889
2	3	year	+7 (484) 371-22-85	0
3	4	year	+7 (501) 568-82-80	0
4	5	year	+7 (515) 891-85-91	0
5	6	year	+7 (904) 468-00-93	0
6	7	month	+7 (961) 963-94-24	0
7	8	month	+7 (209) 537-90-34	0
8	9	month	+7 (769) 777-28-94	0
9	10	half-year	+7 (928) 150-66-35	0
10	11	half-year	+7 (158) 084-93-83	0
11	12	half-year	+7 (360) 158-96-14	0
12	13	half-year	+7 (672) 845-24-76	0
13	14	half-year	+7 (719) 459-10-64	0
14	15	month	+7 (024) 409-32-60	0

Вывод

В ходе лабораторной работы была освоена работа с процедурами и триггерами.