

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

**Отчет**  
по лабораторной работе № 3  
**«ПРОЦЕДУРЫ, ФУНКЦИИ, ТРИГГЕРЫ В POSTGRESQL»**  
по дисциплине **«Базы данных»**

Автор: Демша Евгения

Факультет: ИКТ

Группа: K32422

Преподаватель: Говорова М.М.



**УНИВЕРСИТЕТ ИТМО**

Санкт-Петербург, 2023

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

### Практическое задание:

1. Создать процедуры/функции согласно индивидуальному заданию
2. Создать триггер для логирования событий вставки, удаления, редактирования данных в базе данных PostgreSQL

### 1. Создать хранимые процедуры:

- О текущей сумме вклада и сумме начисленного за месяц процента для заданного клиента;

```

bank=# CREATE OR REPLACE PROCEDURE client_month_accrual(passport IN VARCHAR(11), ma OUT INT) AS $$
bank$# BEGIN
bank$#   ma:= (SELECT month_accrual
bank$#     FROM deposit_contract
bank$#    JOIN interest_accrual USING(contract_number)
bank$#   WHERE client_passport = passport
bank$#        AND date_part('year', date_accrual) = date_part('year', CURRENT_DATE)
bank$#        AND date_part('month', date_accrual) = 4);
bank$# END;
bank$# $$ LANGUAGE PLPGSQL;
CREATE PROCEDURE
bank=# CALL client_month_accrual('5906 376110', NULL);
      ma
-----
 257433
(1 record)

```

Так как база данных заполнялась месяц назад, последние начисления по вкладу указаны в апреле 2023 года. В запросе есть допущение, за текущий (последний) месяц считается апрель (4). В настоящем запросе месяц должен быть равен текущему.

```
bank=# CREATE OR REPLACE PROCEDURE client_current_sum(passport IN VARCHAR(11), sm OUT INT) AS $$  
bank$# BEGIN  
bank$#   sm:= (SELECT current_sum  
bank$#         FROM deposit_contract  
bank$#        JOIN interest_accrual  
bank$#        USING(contract_number)  
bank$#       WHERE client_passport = passport  
bank$#             AND date_part('year', date_accrual) = date_part('year', CURRENT_DATE)  
bank$#             AND date_part('month', date_accrual) = 4);  
bank$# END;  
bank$# $$ LANGUAGE PLPGSQL;  
CREATE PROCEDURE  
bank=# CALL client_current_sum('5906 376110', NULL);  
      sm  
-----  
21138145  
(1 ÷сЕктябрь)
```

- Найти клиента банка, имеющего максимальное количество кредитов на текущий день;

```

bank=# CREATE OR REPLACE FUNCTION max_credit_count() RETURNS TABLE(client_name fio) AS $$
bank$# BEGIN
bank$#     RETURN QUERY
bank$#     (SELECT full_name
bank$#     FROM client
bank$#     JOIN
bank$#     (SELECT client_passport
bank$#     FROM
bank$#     (SELECT client_passport,
bank$#             COUNT(credit_contract.contract_number)
bank$#     FROM credit_contract
bank$#     GROUP BY client_passport) AS cr
bank$#     WHERE COUNT =
bank$#             (SELECT MAX(COUNT)
bank$#             FROM
bank$#             (SELECT client_passport,
bank$#                     COUNT(credit_contract.contract_number)
bank$#             FROM credit_contract
bank$#             WHERE status = 'active'
bank$#             GROUP BY client_passport) AS cr)) AS mxcr ON mxcr.client_passport=client.passport);
bank$# END;
bank$# $$ LANGUAGE PLPGSQL;
CREATE FUNCTION
bank=# SELECT max_credit_count();
max_credit_count
-----
Цёёёютр Ын*сшэр Туюёютэр
(1 өёёютр)

```

Функция вернула полное имя человека с наибольшим количеством кредитов...

- найти клиентов банка, не имеющих задолженности по кредитам.

```

bank=# CREATE OR REPLACE FUNCTION no_debt() RETURNS TABLE(client_name fio) AS $$
bank$# BEGIN
bank$#     RETURN QUERY
bank$#     SELECT DISTINCT(full_name)
bank$#     FROM payment_schedule
bank$#     JOIN credit_contract USING(contract_number)
bank$#     JOIN client ON credit_contract.client_passport = client.passport
bank$#     WHERE real_date IS NOT NULL
bank$#     AND status = 'active';
bank$# END;
bank$# $$ LANGUAGE PLPGSQL;
CREATE FUNCTION
bank=# SELECT no_debt();
no_debt
-----
Ць эўхт Ыхс Црьшыхтшў
Цхьхэют Тшьёё Цртыютшў
ТхёVи Цыхё ЦхээрфNхтэр
Цюь°шыютр ЫN*тшёр Црьёшьютэр
Цхё°ёютр Цёшёсшэр Цёёёёютэр
Црэрё■ь Цыхьёрэфё Цюёшёютшў
ЦхёэV°хтёр Цюшэр Т ўхьертютэр
Цеўеёрхт Цхьхэ Цшёшыютшў
Цеь° ЦрьшыN ЦёN*хтшў
Цхёхьёёшшэр Тхёюэшэр ЦёрфNхтэр
Цшяшэр Ц■фшыр Цыхуютер
Цюэхэютр Цшёр Цыхуютер

```

## 2. Создать триггеры

- Триггер, который не позволит вставить в таблицу с начислениями по вкладу начисление, дата которого раньше последнего.

