

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

## **ОТЧЕТ**

по лабораторной работе «Процедуры, функции, триггеры в PostgreSQL»

по дисциплине «Проектирование и реализация баз данных»

Отчет выполнил:

Студент: Никоноров Максим

Группа: K32422

Факультет: ИКТ

Преподаватель: Говорова Марина Михайловна



Санкт-Петербург 2023

## Цель работы

Овладеть практическими создания и использования процедур, функций и триггеров в базе данных PostgreSQL.

## Практическое задание

Ссылка на папку со скриншотами:

[https://drive.google.com/drive/folders/15XGJRpPPerwimIJZHjNlMj\\_eROn\\_l6sSk?usp=share\\_link](https://drive.google.com/drive/folders/15XGJRpPPerwimIJZHjNlMj_eROn_l6sSk?usp=share_link)

## Часть I

Запрос 1. Вывести список свободных аудиторий для проведения практических занятий в заданное время.

```
CREATE OR REPLACE PROCEDURE auditory_group(year_num
INTEGER, week_num INTEGER, week_day INTEGER, b_time
TIME) LANGUAGE SQL AS $$
    SELECT
        auditory.auditory_no
    FROM
        structure.auditory
    WHERE
        auditory.auditory_no NOT IN
        (
            SELECT
                auditory_no
            FROM
                structure.schedule
            WHERE
                year = year_num AND week_no = week_num
                AND day_of_week_no = week_day AND begin_time =
                b_time
        )
        AND auditory_type = 'laboratory'
$$;

CALL auditory_group(2022, 45, 1, '10:00:00');
```

Запрос 2. Создать хранимую процедуру, которая содержит расписание занятий для заданного преподавателя.

```
CREATE OR REPLACE PROCEDURE auditory_tutor (tutor_num  
INTEGER, year_num INTEGER, week_num INTEGER)  
LANGUAGE SQL AS $$  
    SELECT  
        day_of_week_no, fixed_day, begin_time,  
        discipline_code, auditory_no  
    FROM  
        structure.schedule  
    WHERE  
        tutor_isu_no = tutor_num AND year = year_num  
AND week_no = week_num  
    $$;  
  
CALL auditory_tutor(233876, 2022, 45)
```

Запрос 3. Создать хранимую процедуру, которая содержит список аудиторий, в которых может разместиться заданная группа.

```
CREATE OR REPLACE PROCEDURE
```

```
auditory_list_for_group(group_num VARCHAR(7))
```

```
LANGUAGE SQL AS $$
```

```
    SELECT
```

```
        auditory_no
```

```
    FROM
```

```
        structure.auditory
```

```
    WHERE
```

```
        value_of_seats >=
```

```
        (
```

```
            SELECT
```

```
                COUNT(student_isu_no)
```

```
            FROM
```

```
                structure.student_status
```

```
            WHERE
```

```
                status = 'study' AND group_no =
```

```
group_num
```

```
        )
```

```
$$;
```

```
CALL auditory_list_for_group('K32422')
```

## Часть II

```
CREATE OR REPLACE FUNCTION add_log() RETURNS TRIGGER
AS $$
DECLARE
    mstr varchar(50);
    astr varchar(100);
    retstr varchar(254);
BEGIN
    IF
        TG_OP = 'INSERT' THEN
            astr = NEW.name;
            mstr := 'Добавлен новый студент ';
            retstr := mstr || astr;
            INSERT INTO structure.logs(text,added) values
(retstr,NOW());
            RETURN NEW;
        ELIF
            TG_OP = 'UPDATE' THEN
                astr = NEW.name;
                mstr := 'Изменение информации о студенте ';
                retstr := mstr || astr;
                INSERT INTO structure.logs(text,added) values
(retstr,NOW());
                RETURN NEW;
            ELIF
                TG_OP = 'DELETE' THEN
                    astr = OLD.name;
                    mstr := 'Удален студент ';
                    retstr := mstr || astr;
                    INSERT INTO structure.logs(text,added) values
(retstr,NOW());
                    RETURN OLD;
            END IF;
END;
$$ LANGUAGE PLPGSQL;
```

```
CREATE OR REPLACE TRIGGER ch_students AFTER  
INSERT  
OR  
UPDATE  
OR  
DELETE ON structure.student  
FOR EACH ROW EXECUTE PROCEDURE add_log ();
```

```
INSERT INTO structure.student  
VALUES ('+79633150363', 455500, 2021, 'Замятин  
Акакий');
```

```
UPDATE structure.student  
SET phone_no = '+77056019649'  
WHERE student_isu_no = 455500;
```

```
DELETE  
FROM structure.student  
WHERE student_isu_no = 455500;
```

```
SELECT *  
FROM structure.logs
```