

**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 «**Базы данных**»

Автор: Самаров И

Факультет: ИКТ

Группа: K32402

Преподаватель: Говорова М. М.



Санкт-Петербург 2023

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

**Оборудование:** компьютерный класс.

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

**Практическое задание:**

Вариант 1

- скрипты кода разработанных объектов (процедур/функций и триггера на логирование действий) и подтверждающие скриншоты работы и результатов в psql согласно индивидуальному заданию (часть 4 и 5).

**Ход работы:**



Создать хранимые процедуры:

Вывести список свободных аудиторий для проведения практических занятий заданной группы в заданное время.

```
CREATE OR REPLACE FUNCTION get_free_rooms(  
    group_id varchar,  
    start_time timestamp,  
    end_time timestamp  
) RETURNS TABLE (nomberaud int) AS $$  
BEGIN  
    RETURN QUERY  
        SELECT audience.nomberaud  
        FROM audience  
        WHERE audience.nomberaud NOT IN (  
            SELECT clase.audiance  
            FROM clase  
            WHERE clase.numgroup = get_free_rooms.group_id  
            AND clase.begig <= CAST(get_free_rooms.end_time AS time)  
            AND clase.ends >= CAST(get_free_rooms.start_time AS time)  
        );  
END;  
$$ LANGUAGE PLpgSQL;
```

- Вывести расписание занятий для заданного преподавателя.

```
1 CREATE OR REPLACE FUNCTION get_schedule_for_teacher(teacher_name VARCHAR)
2 RETURNS TABLE (dated DATE, begig TIME, ends TIME, type_class VARCHAR(100), audience INT)
3 AS $$
4 BEGIN
5     RETURN QUERY
6         SELECT clase.dated, clase.begig, clase.ends, clase.type_class, clase.audience
7         FROM clase
8         WHERE clase.teacher = teacher_name;
9 END;
10 $$ LANGUAGE plpgsql;
11 |
```

Вывод:

```
1 SELECT * FROM get_schedule_for_teacher(2311);
```

Data Output   Сообщения   Notifications

	dated date	begig time without time zone	ends time without time zone	type_class character varying	audience integer
1	2024-04-24	10:00:00	11:30:00	Лаба	11
2	2024-04-24	11:50:00	13:20:00	Лекция	12
3	2024-04-24	13:30:00	14:50:00	Практика	12
4	2024-04-25	10:00:00	11:30:00	Практика	10
5	2024-04-24	10:00:00	11:30:00	Лаба	11

- Вывести список аудиторий, в которых может разместиться заданная группа.

```
1 CREATE OR REPLACE FUNCTION get_available_classrooms(group_name VARCHAR(100), group_size INTEGER)
2 RETURNS TABLE (audience_num int)
3 AS $$
4 BEGIN
5     RETURN QUERY
6         SELECT audience.numberaud
7         FROM audience
8         WHERE audience.capacity >= group_size;
9 END;
10 $$ LANGUAGE plpgsql;
```

Вывод

```
1 SELECT * FROM get_available_classrooms('к1210', 10);
```

Data Output   Сообщения   Notifications



	audience_num integer
1	11
2	12
3	113
4	212
5	10

## Триггер для логирования событий вставки, удаления и обновления данных в таблице

### Вывод

```
1 CREATE OR REPLACE FUNCTION LogTriggerFunction()
2 RETURNS TRIGGER AS $$
3 BEGIN
4     IF (TG_OP = 'INSERT') THEN
5         INSERT INTO LogTable (TableName, ActionType)
6         VALUES (TG_TABLE_NAME, 'Insert');
7     ELSIF (TG_OP = 'UPDATE') THEN
8         INSERT INTO LogTable (TableName, ActionType)
9         VALUES (TG_TABLE_NAME, 'Update');
10    ELSIF (TG_OP = 'DELETE') THEN
11        INSERT INTO LogTable (TableName, ActionType)
12        VALUES (TG_TABLE_NAME, 'Delete');
13    END IF;
14
15    RETURN NEW;
16 END;
17 $$ LANGUAGE plpgsql;
18
19 -- Привязка триггера к таблице
20 CREATE TRIGGER LogTrigger
21 AFTER INSERT OR UPDATE OR DELETE ON student
22 FOR EACH ROW
23 EXECUTE FUNCTION LogTriggerFunction();
```

Проверим работу:

```
1 insert into student
2 values (1092, 'Kolya')
```

	logid [PK] integer	tablename character varying (100)	actiontype character varying (50)	actiondate timestamp without time zone
1	1	student	Insert	2023-05-27 16:30:29.90095



В ходе лабораторной работы я научился создавать и использовать процедуры, функции и триггеры в базе данных PostgreSQL. Также, я понял, что функции и процедуры в SQL недостаточно гибкие, как в ЯП.