

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 № 5**  
по теме: процедуры, функции, триггеры в PostgreSQL.  
по дисциплине: Проектирование и реализация баз данных

**Специальность:**  
09.03.03 Мобильные и сетевые технологии

**Проверил:**  
Говорова М.М. \_\_\_\_\_  
**Дата:** «\_\_» \_\_\_\_\_ 2021 г.  
**Оценка** \_\_\_\_\_

**Выполнил:**  
студентка группы К3240  
Кулагина С.В.

Санкт-Петербург 2022 г.

## ЦЕЛЬ РАБОТЫ

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

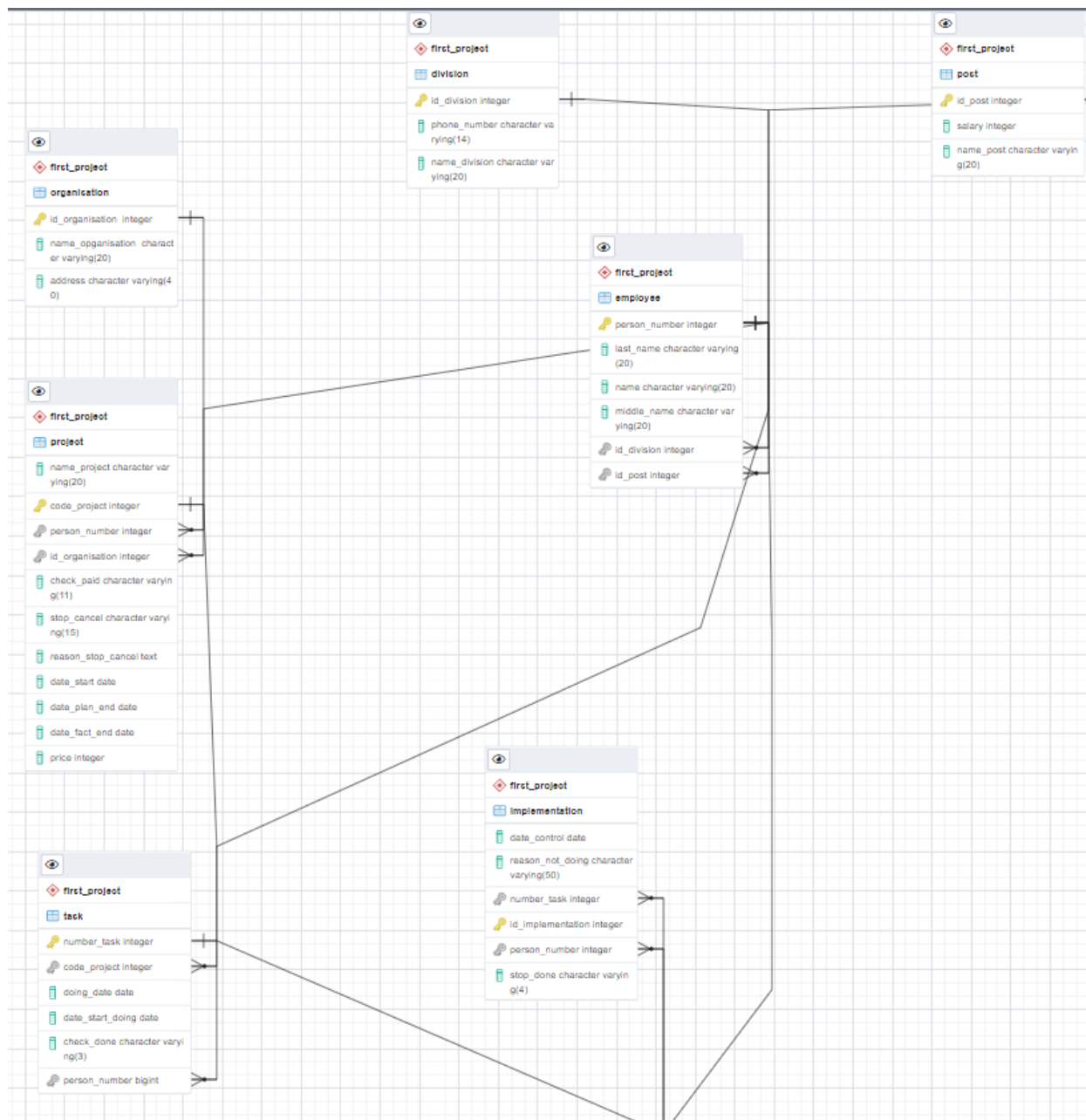
**Оборудование:** компьютерный класс.

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

### Вариант 1. Практическое задание:

- I. Создать процедуры/функции согласно индивидуальному заданию и (согласно индивидуальному заданию, часть 4).
- II. Создать триггер для логирования событий вставки, удаления, редактирования данных в базе данных PostgreSQL (согласно индивидуальному заданию, часть 5). Допустимо создать универсальный триггер или отдельные триггеры на логирование действий.

## СХЕМА БАЗЫ ДАННЫХ



## Выполнение:

### Задание 1. Создайте хранимые процедуры.

- 1) Для повышения оклада сотрудников, выполнивших задания с трехдневным опережением графика на заданный процент(3%).

Результат План выполнения Сообщения Notifications

	number_task [PK] integer	code_project integer	doing_date date	date_start_doing date	check_done character varying (3)	person_number bigint	date_planned_end date
1	1	1220	2022-03-05	2022-02-28	no	4	2022-02-25
2	2	1221	2022-03-10	2022-02-26	yes	2	2022-02-27
3	3	1222	2025-02-10	2022-03-01	no	1	2022-03-20
4	4	1220	2022-03-16	2022-02-25	yes	3	2022-03-19
5	5	1220	2022-05-05	2022-03-03	yes	6	2022-05-08

Результат План выполнения Сообщения Notifications

	person_number [PK] integer	last_name character varying (20)	name character varying (20)	middle_name character varying (20)	id_division integer	id_post integer	salary integer
1	1	Petrov	Nikita	Sergeevich	12	789	50000
2	2	Kulagina	Svetlana	Victorovna	10	123	30000
3	3	Komarova	Julia	Mihailovna	12	456	40000
4	4	Ivanov	Ivan	[null]	11	963	20000
5	5	Kozlov	Ivan	Petrovich	12	123	30000
6	6	Goslov	Konstantin	Ivanov	12	963	20000
7	7	Aloha	Karl	[null]	10	456	40000

```

1  create procedure salary_increase_3()
2  as
3  $$
4  begin
5  update first_project.employee
6  set salary = salary*1.03
7  where salary in (
8      select employee.salary from first_project.employee
9  join first_project.task on task.person_number = employee.person_number where (
10 task.date_planned_end - task.doing_date) = 3);
11 end;
12 $$ language plpgsql;
13

```

Результат План выполнения Сообщения Notifications

CREATE PROCEDURE

Запрос завершён успешно, время выполнения: 372 мсек.

[Результат](#)
[План выполнения](#)
[Сообщения](#)
[Notifications](#)

	person_number [PK] integer	last_name character varying (20)	name character varying (20)	middle_name character varying (20)	id_division integer	id_post integer	salary integer
1	1	Petrov	Nikita	Sergeevich	12	789	50000
2	2	Kulagina	Svetlana	Victorovna	10	123	30000
3	5	Kozlov	Ivan	Petrovich	12	123	30000
4	3	Komarova	Julia	Mihailovna	12	456	41200
5	4	Ivanov	Ivan	[null]	11	963	20600
6	6	Goslov	Konstantin	Ivanov	12	963	20600
7	7	Aloha	Karl	[null]	10	456	41200

2) Для вычисления количества проектов, в выполнении которых участвует сотрудник

```

1 create function emp_pr(emp int) returns table (person_number int, project_number bigint)
2 as
3 $$
4 begin
5 return query
6 select project.person_number, count (project.code_project) from first_project.project where project.person_number = emp
7 group by project.person_number;
8 end;
9 $$ language plpgsql;
10

```

[Результат](#)
[План выполнения](#)
[Сообщения](#)
[Notifications](#)

CREATE FUNCTION

Запрос завершён успешно, время выполнения: 403 мсек.

```

1 select * from emp_pr(3)
2

```

[Результат](#)
[План выполнения](#)
[Сообщения](#)
[Notifications](#)

	person_number integer	project_number bigint
1	3	1

### 3) Для поиска номера телефона сотрудника (телефон установлен в каждом отделе)

```
1 create function person_phone(emp int) returns table (per_number int, phone_per text)
2 as
3 $$
4 begin
5 return query
6 select employee.person_number, division.phone_number from first_project.employee join first_project.d
7 on division.id_division = employee.id_division where employee.person_number = emp;
8 end;
9 $$ language plpgsql;
```

Результат План выполнения Сообщения Notifications

CREATE FUNCTION

Запрос завершён успешно, время выполнения: 82 msec.

```
1 select * from person_phone_num(5)
```

Результат План выполнения Сообщения Not

	per_number integer	phone_per character varying	
1	5	+79272687401	

## Задание 2. Создать триггер для логирования событий вставки, удаления, редактирования

```
create or replace function add_to() returns trigger
as
$$
declare
    mstr varchar(30);
    astr varchar(100);
    retstr varchar(254);
begin
if TG_OP = 'INSERT' then
    astr = new;
    mstr := 'Add data';
    retstr := mstr||astr;
    insert into first_project.logs(text, added, table_name) values (retstr, now(), TG_TABLE_NAME);
    return new;
elsif TG_OP = 'UPDATE' then
    astr = new;
    mstr := 'Update data';
    retstr := mstr||astr;
    insert into first_project.logs(text, added, table_name) values (retstr, now(), TG_TABLE_NAME);
    return new;
elsif TG_OP = 'DELETE' then
    astr = old;
    mstr := 'Remove data';
    retstr := mstr||astr;
    insert into first_project.logs(text, added, table_name) values (retstr, now(), TG_TABLE_NAME);
    return old;
end if;
end;
$$ language plpgsql;
```

Результат План выполнения Сообщения Notifications

CREATE FUNCTION

Запрос завершён успешно, время выполнения: 118 msec.

```
1 create trigger divv after insert or update or delete on first_project.division for each row execute procedure add_to()
```

Результат План выполнения Сообщения Notifications

CREATE TRIGGER

Запрос завершён успешно, время выполнения: 138 msec.

```
1 insert into first_project.division (id_division, phone_number, name_division) values (9, '+79888888888', 'DataBase')
```

Результат	План выполнения	Сообщения	Notifications
-----------	-----------------	-----------	---------------

INSERT 0 1

Запрос завершён успешно, время выполнения: 39 msec.

```
1 update first_project.division set phone_number = '+79212121212' where id_division = 9
```

Результат	План выполнения	Сообщения	Notifications
-----------	-----------------	-----------	---------------

UPDATE 1

Запрос завершён успешно, время выполнения: 36 msec.

```
1 delete from first_project.division where id_division = 9
```

Результат	План выполнения	Сообщения	Notifications
-----------	-----------------	-----------	---------------

DELETE 1

Запрос завершён успешно, время выполнения: 35 msec.



```
1 select * from first_project.logs
```

Результат    План выполнения    Сообщения    Notifications

	<b>text</b> text	<b>added</b> timestamp without time zone	<b>table_name</b> text
1	Add data(9,+798888888888,DataBase)	2022-05-13 01:24:20.468891	division
2	Update data(9,+79212121212,DataBase)	2022-05-13 01:26:23.077923	division
3	Remove data(9,+79212121212,DataBase)	2022-05-13 01:39:26.360529	division

## ВЫВОДЫ

В данной работе я изучила функции и процедуры, создала их под разные задачи и условия. Также я узнала многое про триггеры и использовала их для хранения информации об изменениях в данных своей базы

