

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 «Проектирование и реализация баз данных»

Вариант №7. БД «Курсы»

Авторы: Антонова М. А.

Факультет: ФИКТ

Группа: К32422

Преподаватель: Говорова М.М.

Санкт-Петербург, 2023

1 Описание работы

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 1

1. Создать процедуры/функции согласно индивидуальному заданию и (согласно индивидуальному заданию, часть 4).
2. 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.

Вариант 2

1. Создать процедуры/функции согласно индивидуальному заданию и (согласно индивидуальному заданию, часть 4).

2.1. Модифицировать триггер (триггерную функцию) на проверку корректности входа и выхода сотрудника (см. Практическое задание 1 Лабораторного практикума (Приложение)) с максимальным учетом «узких» мест некорректных данных по входу и выходу.

- 2.2. Создать авторский триггер по варианту индивидуального задания.

2 Описание предметной области

Вариант 7. БД «Курсы»

Сеть учебных подразделений занимается организацией внебюджетного образования.

Имеется несколько образовательных программ краткосрочных курсов, предназначенных для определенных специальностей, связанных с программным обеспечением ИТ. Каждая программа имеет определенную длительность и свой перечень изучаемых дисциплин. Одна дисциплина может относиться к нескольким программам. На каждую программу может быть набрано несколько групп обучающихся. По каждой дисциплине могут

проводиться лекционные, лабораторные/практические занятия и практика определенном объеме часов. По каждой дисциплине и практике проводится аттестация в формате экзамен/дифзачет/зачет.

Подразделение обеспечивает следующие ресурсы: учебные классы, лекционные аудитории и преподавателей. Необходимо составить расписание занятий.

БД должна содержать следующий минимальный набор сведений: Фамилия слушателя. Имя слушателя. Паспортные данные. Контакты. Код программы. Программа. Тип программы. Объем часов. Номер группы. максимальное количество человек в группе (для набора). Дата начала обучения. Дата окончания обучения. Название дисциплины. Количество часов. Дата занятий. Номер пары. Номер аудитории. Тип аудитории. Адрес площадки. Вид занятий (лекционные, практические или лабораторные). Фамилия преподавателя. Имя и отчество преподавателя. Должность преподавателя. Дисциплины, которые может вести преподаватель.

Наименование БД – Timetable

Была установлена программа PostgreSQL версии 14, а также платформа администрирования и управления для баз данных – PGAdmin. В PGAdmin была спроектирована база данных «Timetable». Были также установлены ограничения на атрибуты, заданы связи между таблицами.

Схема логической модели базы данных, сгенерированная в Generate ERD

Результат выполнения:

select * from schedule_for_today('2023-02-02');

```
timetable=# select * from schedule_for_today('2023-02-02');
```

ID_lesson	L_group_number	L_ID_class	L_ID_passport	L_ID_discipline_of_pr	Lesson_type	Lesson_number	Date_of_lessons
140031	1112020	1002	4018785670		86003 практика	1	2023-02-02

(1 строка)

2) Запись на курс слушателя

CREATE OR REPLACE FUNCTION welcome_to_family2 (id_p character varying, fn character varying, c character varying, in_id integer, dn integer, ss “St_status”, gn integer) RETURNS VOID AS \$\$ INSERT INTO “timetable_courses”.”Student” (“ID_passport”, “Full_name”, “Contacts”) VALUES (id_p, fn, c);

INSERT INTO “timetable_courses”.”Is_in_group” (“ID_in_group”, “Document_number”, “Student_status”, “I_group_number”, “I_ID_passport”) VALUES (in_id, dn,ss, gn, id_p); UPDATE “timetable_courses”.”Group” set “Number_of_students” = “Number_of_students”+ 1 WHERE “Group_number” = gn; \$\$ LANGUAGE SQL;

Результат выполнения:

```
timetable$# SELECT timetable_courses.welcome_to_family2(
timetable$#      '4017878797',
timetable$#      'Петров Василий Иванович',
timetable$#      '89218746474',
timetable$#      '206642','11100240',
timetable$#      'учится',
timetable$#      '1262018'
timetable$# )
timetable$# ;
timetable$# SELECT timetable_courses.welcome_to_family2(
timetable$#      '4012186858',
timetable$#      'Иванов Василий Иванович',
timetable$#      '89214446474',
timetable$#      '206643','11100241',
timetable$#      'учится',
timetable$#      '1262018'
timetable$# );
```

Добавление в таблице

Data Output Messages Notifications

	ID_passport [PK] character varying (10)	Full_name character varying (50)	Contacts character varying (50)
1	4011984520	Сахаров Дмитрий Иванович	8-962-789-67-52
2	4012186858	Иванов Василий Иванович	89214446474
3	4012682518	Николаева Алла Викторовна	8-981-666-65-35
4	4014984518	Антонов Николай Петрович	8-911-837-45-33
5	4016984518	Максимова Ирина Владимировна	8-921-837-45-68
6	4016989999	Левина Ольга Александровна	8-955-997-45-90
7	4017878797	Петров Василий Иванович	89218746474

3) Получения перечня свободных лекционных аудиторий на любой день недели. Если свободных аудиторий не имеется, то выдать соответствующее сообщение

```
CREATE OR REPLACE FUNCTION free_lecture_rooms2(date_f date)

RETURNS TABLE(classroom_number integer) AS $$

DECLARE

kol INTEGER;

BEGIN

SELECT COUNT(DISTINCT "Class"."ID_class") INTO kol

FROM "Class" WHERE "Class"."ID_class" NOT IN

(SELECT "Lesson"."L_ID_class" FROM "Lesson" WHERE "Lesson"."Date_of_lessons"=date_f)

AND "Class"."Class_type" = 'лекционная';

IF kol > 0 THEN

RETURN QUERY(SELECT "Class"."ID_class"

FROM "Class" WHERE "Class"."ID_class" NOT IN

(SELECT "Lesson"."L_ID_class" FROM "Lesson" WHERE "Lesson"."Date_of_lessons"=date_f)

AND "Class"."Class_type" = 'лекционная');

ELSE RAISE NOTICE 'Свободных лекционных аудиторий нет!';

END IF;

END;

$$

LANGUAGE plpgsql;
```

```
timetable=# CREATE OR REPLACE FUNCTION free_lecture_rooms2(date_f date)
timetable=# RETURNS TABLE(classroom_number integer) AS $$
timetable$# DECLARE
timetable$# kol INTEGER;
timetable$# BEGIN
timetable$# SELECT COUNT(DISTINCT "Class"."ID_class") INTO kol
timetable$# FROM "Class" WHERE "Class"."ID_class" NOT IN (SELECT "Lesson"."L_ID_class" FROM "Lesson" WHERE "Lesson"."
timetable$# "Date_of_lessons"=date_f) AND "Class"."Class_type" = 'лекционная';
timetable$# IF kol > 0 THEN
timetable$# RETURN QUERY(SELECT "Class"."ID_class"
timetable$# FROM "Class" WHERE "Class"."ID_class" NOT IN (SELECT "Lesson"."L_ID_class" FROM "Lesson" WHERE "Lesson"."
timetable$# "Date_of_lessons"=date_f)AND "Class"."Class_type" = 'лекционная');
timetable$# ELSE RAISE NOTICE 'Свободных лекционных аудиторий нет!';
timetable$# END IF;
timetable$# END;
timetable$# $$
timetable=# LANGUAGE plpgsql;
CREATE FUNCTION
```

Результат выполнения:

```
select free_lecture_rooms2('2022-02-02');
```

```
timetable=# select free_lecture_rooms2('2022-02-02');
 free_lecture_rooms2
-----
                1003
(1 строка)
```

Создать триггер для логирования событий вставки, удаления, редактирования данных в базе данных PostgreSQL

Создание таблицы логов

```
CREATE TABLE log_table (target text,operation text,id_passport character varying, full_name character varying, position character varying, date_time timestamp);
```

```
timetable=# CREATE TABLE log_table (target text,operation text,id_passport character varying, full_name character varying, position character varying, date_time timestamp);
CREATE TABLE
```

Создание и настройка триггерной функции

```
CREATE OR REPLACE FUNCTION log_func3() RETURNS TRIGGER LANGUAGE plpgsql AS $$
```

```
BEGIN
```

```
INSERT INTO log_table VALUES (TG_NAME, TG_OP, NEW."ID_passport", NEW."Full_name", NEW."Position", NOW());
```

```
RETURN NEW;
```

```
End;
```

```
$$;
```

```
timetable=# CREATE OR REPLACE FUNCTION log_func3()
timetable-# RETURNS TRIGGER
timetable-# LANGUAGE plpgsql
timetable-# AS $$
timetable$# BEGIN
timetable$# INSERT INTO log_table VALUES (TG_NAME, TG_OP, NEW."ID_passport", NEW."Full_name", NEW."Position", NOW());
timetable$# RETURN NEW;
timetable$# End;
timetable$# $$;
CREATE FUNCTION
```

```
CREATE TRIGGER log_change_teacher3 AFTER
```

```
INSERT OR UPDATE OR DELETE
```

```
ON "Teacher"
```

```
FOR EACH ROW EXECUTE FUNCTION log_func3();
```

```
CREATE FUNCTION
timetable=# CREATE TRIGGER log_change_teacher3 AFTER
timetable=# INSERT OR UPDATE OR DELETE
timetable=# ON "Teacher"
timetable=# FOR EACH ROW EXECUTE FUNCTION log_func3();
CREATE TRIGGER
```

Проверка работы триггера

Вставка

```
INSERT INTO timetable_courses."Teacher" ("ID_passport","Full_name","Position") VALUES ('11223
34456', 'Буланкина Антонина Петровна', 'преподаватель');
```

```
timetable=# INSERT INTO timetable_courses."Teacher" ("ID_passport","Full_name","Position") VALUES ('11223
34456', 'Буланкина Антонина Петровна', 'преподаватель');
```

```
INSERT 0 1
```

```
timetable=# SELECT * FROM log_table;
```

target	operation	id_passport	full_name	position	date_time
Teacher	INSERT	1122334456	Буланкина Антонина Петровна	преподаватель	2023-05-01 16:15:01.86

6082
(1 строка)

```
timetable=# INSERT INTO timetable_courses."Teacher" ("ID_passport","Full_name","Position") VALUES ('11223
34456', 'Буланкина Антонина Петровна', 'преподаватель');
INSERT 0 1
```

Обновление

```
UPDATE timetable_courses."Teacher" SET "Full_name" = 'Козлова Мария Никитична' WHERE
"ID_passport"='1122334456';
```

```
timetable=# UPDATE timetable_courses."Teacher" SET "Full_name" = 'Козлова Мария Никитична' WHERE "ID_pass
port"='1122334456';
UPDATE 1
```

```
timetable=# SELECT * FROM log_table;
```

target	operation	id_passport	full_name	position	date_time
Teacher	INSERT	1122334456	Буланкина Антонина Петровна	преподаватель	2023-05-01 16:15:01.86
Teacher	UPDATE	1122334456	Козлова Мария Никитична	преподаватель	2023-05-01 16:22:00.01

6082
(3 строки)

Удаление

```
DELETE FROM "Teacher" WHERE "ID_passport"='1122334456';
```



```

timetable=# DELETE FROM "Teacher" WHERE "ID_passport"='1122334456'
timetable=# ;
DELETE 1
timetable=# SELECT * FROM log_table;
 target | operation | id_passport | full_name | position | date_time
-----+-----+-----+-----+-----+-----
Teacher | INSERT   | 1122334456 | Буланкина Антонина Петровна | преподаватель | 2023-05-01 16:15:01.86
6082
Teacher | UPDATE   | 1122334456 | Козлова Мария Никитична | преподаватель | 2023-05-01 16:22:00.01
6814
Teacher | DELETE   |             |             |             | 2023-05-01 16:26:48.83
8253

```

Выводы

При выполнении данной лабораторной работы были приобретены практические навыки по работе с таблицами в базе данных PostgreSQL, выполнением процедур и функций, а также использованием триггеров. Был создан триггер для вставки, удаления, редактирования данных.

Таким образом, был получен опыт по работе и использованию программы pgAdmin 4.