

***Федеральное агентство по рыболовству***

***Федеральное государственное бюджетное образовательное***

***учреждение высшего образования***

***«Астраханский государственный технический университет»***

**Система менеджмента качества в области образования, воспитания, науки и инноваций сертифицирована**

**ООО «ДКС РУС» по международному стандарту ISO 9001:2015**

|  |  |
| --- | --- |
| Институт | Информационных технологий и коммуникаций |
| Направление | 09.03.01 Информатика и вычислительная техника |
| Профиль | «Автоматизированные системы обработки информации и управления» |
| Кафедра | «Автоматизированные системы обработки информации и управления» |

**Лабораторная работа № 15**

**«Перегружаемые функции, триггеры в PostgreSQL**»

по дисциплине «СУБД PostgreSQL»

|  |  |  |
| --- | --- | --- |
|  |  | Работа выполнена студентом группы ДИНРБ-31  Кузургалиев Р.А. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Фамилия И.О.) подпись |
|  |  | Проверил работу:  ст. преподаватель Мамлеева А.Р,\_\_\_\_\_\_\_\_\_\_\_\_  (ученая степень, ученое звание, Фамилия И.О.) |

Работа защищена

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

**АСТРАХАНЬ – 2024**

**СОДЕРЖАНИЕ ОТЧЕТА**

Цель работы: Изучение перегружаемых функций и триггеров в PostgreSQL. Изучить синтаксис команд. Получение навыков работы с триггерами.

***Задание. Общая часть:*** Ознакомиться с теоретическими сведениями о возможностях создания перегружаемых функций и триггеров в PostgreSQL. ***Согласно варианту из лабораторной работы №14*** создать перегружаемые функции и триггеры. Продемонстрировать работу триггеров на примерах вставки и удаления записей из таблицы. Если в базе нет подходящих данных, то изменить условия, либо добавить подходящие данные. Удалить триггер. По заданию преподавателя создать триггер и проверить его работоспособность. Просмотреть и проанализировать полученную в результате выполнения операций информацию.

**Вариант 9:** Триггер выполняется перед удалением записи из таблицы мастеров. Триггер проверяет наличие в другой таблице записей, относящихся к удаляемому мастеру, и, если такие записи есть, удаляет их.

Триггер выполняется перед вставкой новой записи в таблицу мастеров. Триггер проверяет значения, которые должна содержать новая запись и может их изменить:

• eсли не указано имя мастера – оно генерируется по схеме – Master + уникальный номер из последовательности(создай эту последовательность);

• eсли не указан номер мастера – он генерируется по схеме – S+ уникальный номер из последовательности(создай эту последовательность);

• eсли не указан город мастера– ставится значение по умолчанию – “Irkutsk” ;

• если не указан стаж мастера или стаж <=1 – устанавливается стаж, равный 1 для мастера из города “ Irkutsk ” и 0 для всех остальных.

CREATE SEQUENCE master\_id\_seq;

CREATE OR REPLACE FUNCTION delete\_related\_records()

RETURNS TRIGGER AS $$

BEGIN

DELETE FROM other\_table WHERE master\_id = OLD.master\_id;

RETURN OLD;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER before\_master\_delete

BEFORE DELETE ON Master

FOR EACH ROW

EXECUTE FUNCTION delete\_related\_records();

CREATE OR REPLACE FUNCTION check\_and\_modify\_values()

RETURNS TRIGGER AS $$

BEGIN

IF NEW.name IS NULL THEN

NEW.name := 'Master' || nextval('master\_id\_seq');

END IF;

IF NEW.phone\_number IS NULL THEN

NEW.phone\_number := 'S' || nextval('master\_id\_seq');

END IF;

IF NEW.city IS NULL THEN

NEW.city := 'Irkutsk';

END IF;

IF NEW.experience IS NULL OR NEW.experience <= 1 THEN

IF NEW.city = 'Irkutsk' THEN

NEW.experience := 1;

ELSE

NEW.experience := 0;

END IF;

END IF;

RETURN NEW;

END;

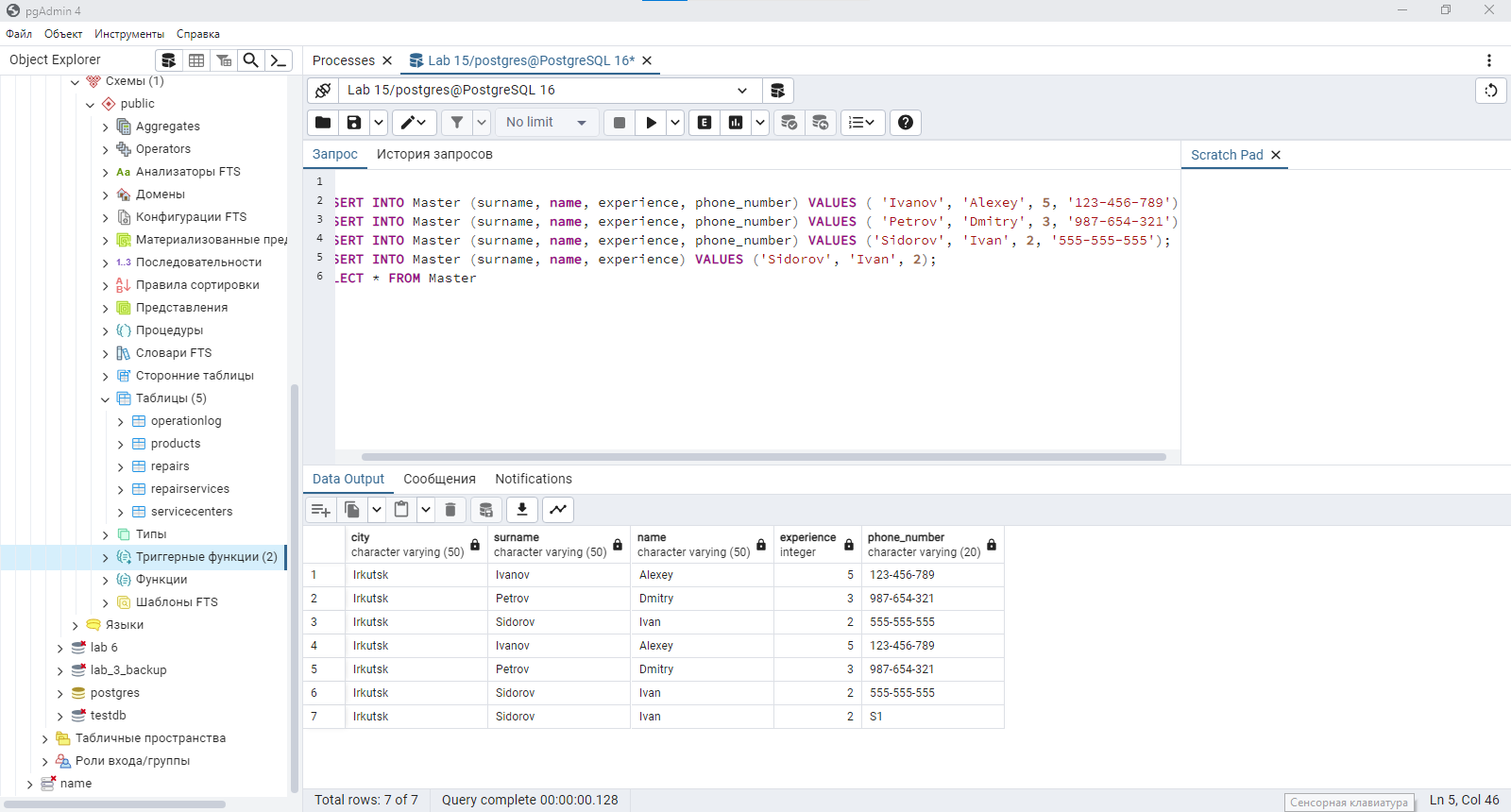
$$ LANGUAGE plpgsql;

CREATE TRIGGER before\_master\_insert

BEFORE INSERT ON Master

FOR EACH ROW

EXECUTE FUNCTION check\_and\_modify\_values();



Выводы: в ходе лабораторной работы я изучил перегружаемые функции и триггеры в PostgreSQL, изучил синтаксис команд, а также получил навыки работы с триггерами.