Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Новосибирский государственный технический университет»

Кафедра вычислительной техники

**Отчет**

По лабораторной работе №8

По дисциплине: «Базы данных»

Перегружаемые функции, триггеры в PostgreSQL

Вариант 10

Выполнили: Павлюк А.С. Преподаватель: Харюткина С. А.

Казанцев К. О.

Группа: АВТ-008

Факультет: АВТ

Новосибирск, 2022

Оглавление

[Цель работы 3](#_Toc113728084)

[Задание 3](#_Toc113728085)

[Структура базы данных 5](#_Toc113728086)

[Ход работы 5](#_Toc113728087)

[Вывод 12](#_Toc113728088)

[Приложение 12](#_Toc113728089)

Цель работы

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

Задание

Ознакомиться с теоретическими сведениями о возможностях создания перегружаемых функций и триггеров в PostgreSQL. Создать перегружаемые функции и триггеры. Продемонстрировать работу триггеров на примерах вставки и удаления записей из таблицы. Если в базе нет подходящих данных, то добавить подходящие данные. Удалить триггер. По заданию преподавателя создать триггер и проверить его работоспособность. Просмотреть и проанализировать полученную в результате выполнения операций информацию. Обязательные требования к БД:

*Вариант 10:* Триггер выполняется перед удалением записи из таблицы данных о туре. Триггер проверяет наличие в другой таблице записей, относящихся к удаляемой записи, и, если такие записи есть, удаляет их. Триггер выполняется перед вставкой новой записи в таблицу данных о туре. Триггер проверяет значения, которые должна содержать новая запись и может их изменить:

* eсли не указано наименование тура – оно генерируется по схеме – Tur + уникальный номер из последовательности;
* eсли не указан город у тура – ставится значение по умолчанию –“Kaliningrad”;
* если не указана длительность тура или длительность тура <= 7 дней –   
  устанавливается длительность тура, равная 7 для туров в город   
  “Kaliningrad” и 0 для всех остальных.

Структура базы данных

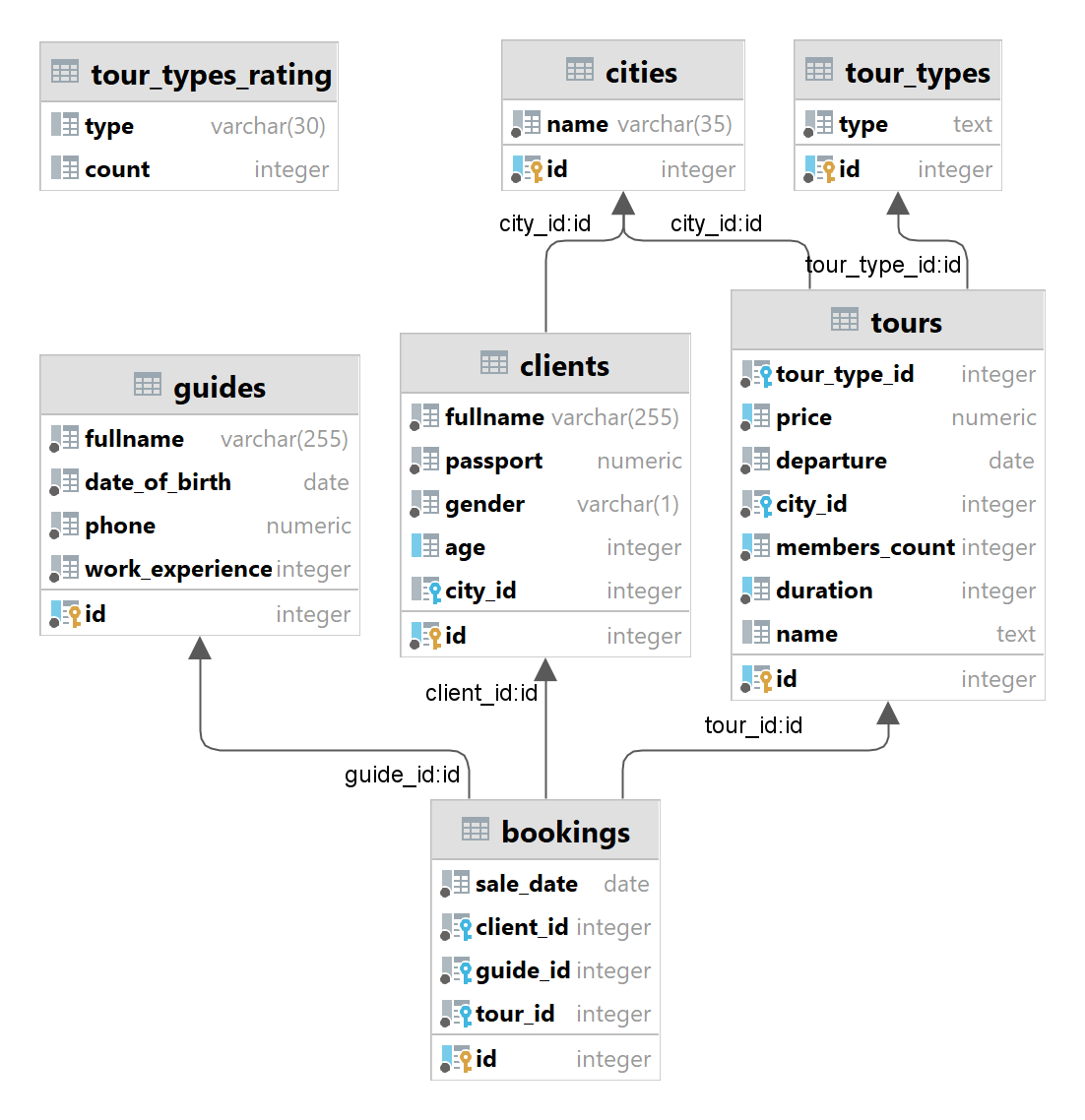


Рисунок 1. Структура базы данных

Ход работы

Создадим функцию для удаления туров из всех связанных таблиц.

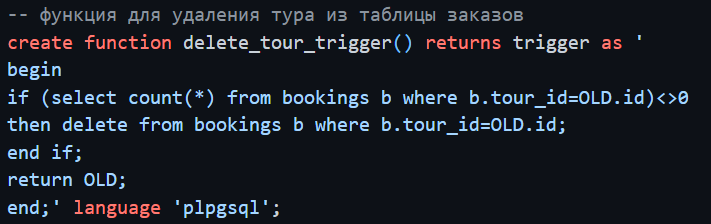


Рисунок 2. Функция удаления туров из всех связанных таблиц

Далее напишем для нее триггер.

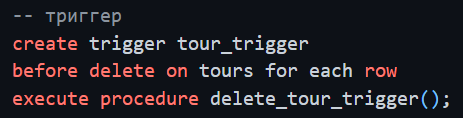


Рисунок 3. Триггер для функции удаления тура

Проверим.

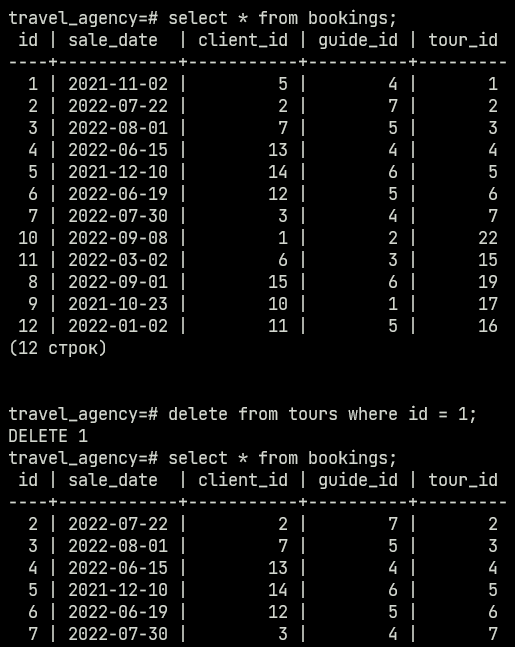


Рисунок 4. Проверка первого триггера

Перейдем к добавлению туров. Для этого добавим столбец **name text** в таблицу **tours**.

Далее создадим перегружаемые функции, принимающие в параметры либо integer, либо text.

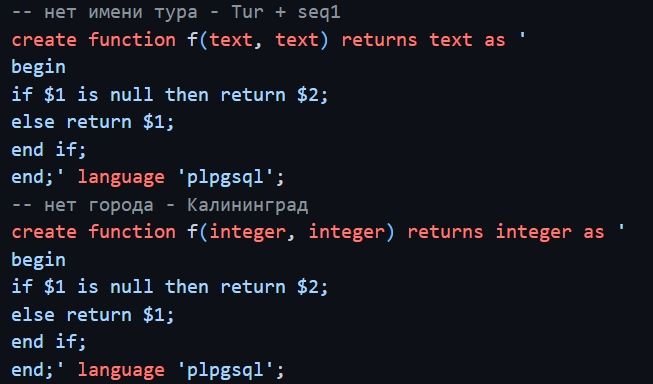


Рисунок 5. Перегружаемые функции

Теперь создадим функцию для второго триггера.

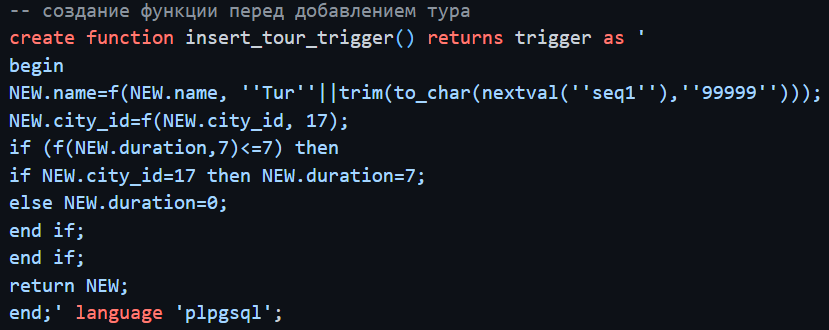


Рисунок 6. Функция для добавления тура

И сам триггер.

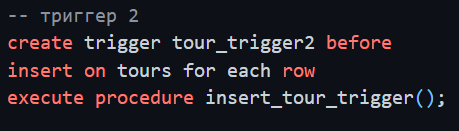


Рисунок 7. Триггер для добавления тура

Проверим.

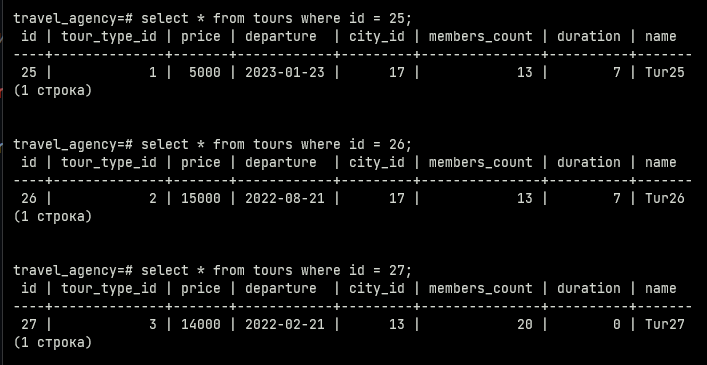


Рисунок 8. Проверка работоспособности триггера

Вывод

В результате выполнения лабораторной работы были изучены перегружаемые функции и триггеры в PostgreSQL, синтаксис команд. Также были получены навыки работы с триггерами.

Приложение

-- функция для удаления тура из таблицы заказов

create function delete\_tour\_trigger() returns trigger as '

begin

if (select count(\*) from bookings b where b.tour\_id=OLD.id)<>0

then delete from bookings b where b.tour\_id=OLD.id;

end if;

return OLD;

end;' language 'plpgsql';

-- триггер

create trigger tour\_trigger

before delete on tours for each row

execute procedure delete\_tour\_trigger();

-- создание последовательности

create sequence seq1 increment by 1 start with 25;

-- alter

alter table tours

add column name text;

-- перегрузки

-- нет имени тура - Tur + seq1

create function f(text, text) returns text as '

begin

if $1 is null then return $2;

else return $1;

end if;

end;' language 'plpgsql';

-- нет города - Калининград

create function f(integer, integer) returns integer as '

begin

if $1 is null then return $2;

else return $1;

end if;

end;' language 'plpgsql';

-- длительность тура - нет/<=7 -> 7 для Калининграда, 0 - ост

-- создание функции перед добавлением тура

create function insert\_tour\_trigger() returns trigger as '

begin

NEW.name=f(NEW.name, ''Tur''||trim(to\_char(nextval(''seq1''),''99999'')));

NEW.city\_id=f(NEW.city\_id, 17);

if (f(NEW.duration,7)<=7) then

if NEW.city\_id=17 then NEW.duration=7;

else NEW.duration=0;

end if;

end if;

return NEW;

end;' language 'plpgsql';

-- триггер 2

create trigger tour\_trigger2 before

insert on tours for each row

execute procedure insert\_tour\_trigger();