**Отчёт по лабораторной работе №5**

Выполнил студент группы ИВТ(ВМК)-17-2:

Ковалёв Владимир Сергеевич

**Цель работы**

Реализация процедурных ограничений целостности и создание триггеров в СУБД PostgreSQL

**Содержание работы**

* Выявление процедурных ограничений целостности предметной области;
* Реализация триггеров и хранимых процедур в СУБД PostgreSQL.

**Ход работы**

Предметная область:

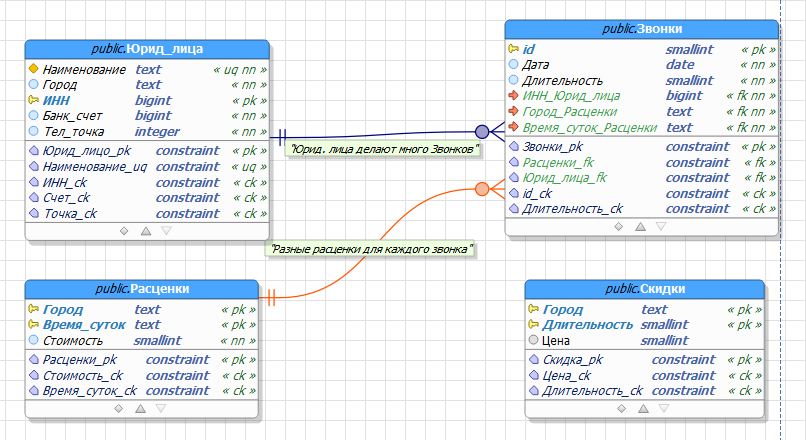
Вы работаете в коммерческой службе телефонной компании. Компания предоставляет абонентам телефонные линии для междугородних переговоров. Вашей задачей является отслеживание стоимости междугородних телефонных переговоров. Абонентами компании являются юридические лица, имеющие телефонную точку, ИНН, расчетный счет в банке. Стоимость переговоров зависит от города, в который осуществляется звонок, и времени суток (день, ночь). Каждый звонок абонента автоматически фиксируется в базе данных. При этом запоминаются город, дата, длительность разговора и время суток. Фирма вводит гибкую систему скидок. Так, стоимость минуты теперь уменьшается в зависимости от длительности разговора. Размер скидки для каждого города разный.

Рисунок 1. Диаграмма ER

|  |
| --- |
| Таблица 1. Юридические лица |
| Таблица 2. Расценки |
| Таблица 3. Звонки |
| Таблица 4. Скидки |

Проанализировав данные и бизнес-правила предметной области в прошлой лабораторной работе №3, помимо декларативных ограничений целостности было выявлено следующее процедурное ограничение:

* Значения поля «Цена» из таблицы «Скидки» должно быть больше значения поля «Стоимость» из таблицы «Расценки» и отношение между полем «Цена» и полем «Стоимость» должно быть больше 0.5

Примеры ситуаций, при которых возможно появление вышеописанного нарушения ограничения целостности:

1. При вводе данных в таблицу «Скидки» может получится, что для одного города цена скидки может быть больше стоимости расценки, и при расчете месячной абонентской оплаты в сумме могут появится отрицательные слагаемые.
2. При вводе данных в таблицу «Скидки» может получится, что соответствующего города в таблице «Расценки» нет.

Поскольку нарушения возникают при вводе неправильных данных в таблицу «Скидки», то для реализации ограничения следует использовать триггерную функцию.

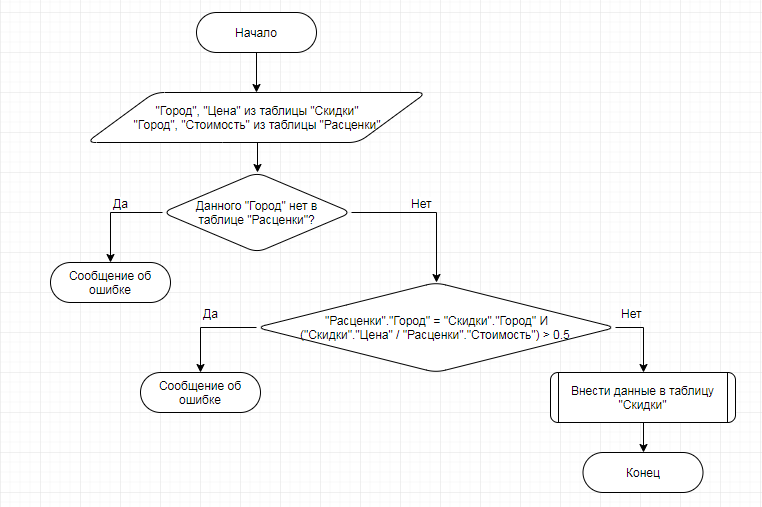
Блок-схема триггерной функции для данного ограничения:

Рисунок 2. Блок-схема

После определения процедурного ограничения создаем триггерную процедуру в программе pgAdmin версии 4 через следующий SQL-скрипт:

create or replace function insert\_new\_discount()

returns trigger

as $$

begin

if not exists (select \* from "Расценки" where "Город" = new."Город") then

raise exception 'Значение "Город" отсутствует в таблице!'

using hint = 'Заполните необходимые данные в таблицу "Расценки"';

end if;

if exists (select \* from "Расценки" where "Город" = new."Город" and (new."Цена"::float / "Стоимость"::float) > 0.5) then

raise exception 'Неправильное значение для "Цены скидки"!'

using hint = 'Цена скидки должна быть строго меньше стоимости расценки';

else

return new;

end if;

end;

$$ language 'plpgsql';

create trigger ins\_tr

before insert on "Скидки"

for row execute procedure insert\_new\_discount();

Сначала мы описываем функцию **insert\_new\_discount**, в которой описываем реализацию алгоритма из вышеуказанной блок-схемы. Важно подчеркнуть, что в реализации второго условия типы полей «Цена» и «Стоимость» должны быть приведены в вещественные типы, так как результат деления между значениями этих полей далее сравнивается с вещественным числом.

После описания функции создаем триггер ins\_tr, который вызывает функцию insert\_new\_discount перед запросом INSERT в таблицу “Скидки».

Далее заносим триггер и функцию в СУБД и описываем следующие запросы для проверки работы процедурного ограничения:

1. insert into "Скидки" values ('Омск', 5, 10);
2. insert into "Скидки" values ('Хабаровск', 10, 10).
3. insert into "Скидки" values ('Хабаровск', 10, 5).

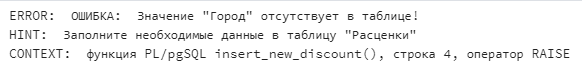
При выполнении первого запроса программа не занесет информацию в таблицу и выведет сообщение об ошибке, что поле «Город» из строки ('Омск', 5, 10) в «Расценках» не существует:

Рисунок 3. Результат выполнения первого запроса

При выполнении второго запроса программа не занесет информацию в таблицу и выведет сообщение об ошибке, что отношение между полями «Цена» и «Стоимость» больше 0.5 и, следовательно, значение поля «Цена» для таблицы «Скидки» недопустимо:

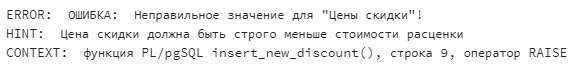


Рисунок 4. Результат выполнения второго запроса

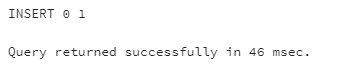
При выполнении второго запроса программа занесет информацию. Это означает, что веденные данные не нарушают процедурное ограничение целостности:

Рисунок 5. Результат выполнения третьего запроса