Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

**Дисциплина**

Базы данных

**Отчёт по лабораторной работе**

# SQL-программирование: триггеры, вызовы процедур

Работу выполнил студент группы № 43501/1 Трофимова Н.С.

Работу принял преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мяснов А.В.

Санкт-Петербург

2016

Триггер является программой, которая хранится в области метаданных базы данных и выполняется на стороне сервера. Напрямую обращение к триггеру невозможно. Он вызывается автоматически при наступлении одного или нескольких событий, относящихся к одной конкретной таблице (к представлению), или при наступлении одного из событий базы данных. Триггер, вызываемый при наступлении события таблицы, связан с одной таблицей или представлением, с одним или более событиями для этой таблицы или представления (INSERT, UPDATE, DELETE) и ровно с одной фазой такого события (BEFORE или AFTER).

Триггер выполняется в той транзакции, в контексте которой выполнялась программа, вызвавшая соответствующее событие. Исключением являются триггеры, реагирующие на события базы данных. Для некоторых из них запускается транзакция по умолчанию. Для каждой комбинации фаза-событие может быть определено более одного триггера. Порядок, в котором они выполняются, может быть указан явно с помощью дополнительного аргумента POSITION в определении триггера. Максимальная позиция равна 32767. Триггеры с меньшей позицией вызываются первыми.

Если предложение POSITION опущено или несколько триггеров с одинаковыми фазой и событием имеют одну и ту же позицию, то такие триггеры будут выполняться в алфавитном порядке их имен.

В DML триггерах Firebird обеспечивает доступ к множеству контекстных переменных NEW.\* и OLD.\* Каждое множество является массивом всей строки: OLD.\* — значение строки до изменения данных и NEW.\* — требуемое ("новое") значение строки. Операторы могут ссылаться на них использую следующие формы NEW.*columname* и OLD.*columnname*. *columnname* может быть любым столбцом определённым в таблице(представлении). а не только тем что был изменён.

Контекстные переменные NEW и OLD подчиняются следующим правилам:

• Во всех триггерах контекстные переменные OLD доступны только для чтения;

• В триггерах BEFORE UPDATE и BEFORE INSERT переменные NEW доступны для чтения и записи, за исключением COMPUTED BY столбцов;

• В INSERT триггерах ссылка на переменные OLD не допускается и вызовет исключение;

• В DELETE триггерах ссылка на переменные NEW не допускается и вызовет исключение;

• Во всех AFTER триггерах переменные NEW доступны только для чтения.

Индивидуальные задания:

1. При внесении записи в расписание проверять, что нет пересечений по времени с уже созданными записями для данной комнаты.

Скрипт:

set exception wrongdate 'Error: wrong date';

CREATE OR ALTER TRIGGER ROOMLIST\_BI1 FOR ROOMLIST

ACTIVE BEFORE INSERT POSITION 1

AS

begin

if(new.firsttime < (select lasttime from roomlist where roomid=new.roomid and lastday=new.firstday)) then exception wrongdate;

if(new.firstday <(select lastday from roomlist where roomid=new.roomid)) then exception wrongdate;

end

^

SET TERM ; ^

1. После проведения заданного типа работ увеличивать стоимость комнаты на 3%.

Скрипт:

create procedure newcost(newid int)

as

begin

update rooms set roomcost=1.03 \* roomcost where roomid=:newid;

end

^

create or alter trigger tserving for roomlist before insert

as

begin

if(new.tservid=2) then execute procedure tserving(new.roomid);

end

^

**Выводы:**

Я получила опыт написания триггеров на событие таблицы(ввод новых данных). Были использованы условные операторы и исключения.