Санкт-Петербургский Политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

**Отчёт о лабораторной работе №6**

**Дисциплина**: Базы данных

**Тема**: Триггеры

Выполнил студент гр. 43501/1 Данг Хань

(подпись)

Руководитель Мяснов А.В.

(подпись)

“ ” 2015 г.

Санкт-Петербург

2015

1. **Цель**

Познакомить студентов с возможностями реализации более сложной обработки данных на стороне сервера с помощью хранимых процедур и триггеров.

1. **Программа работы**
2. Создать два триггера: один триггер для автоматического заполнения ключевого поля, второй триггер для контроля целостности данных в подчиненной таблице при удалении/изменении записей в главной таблице
3. Создать триггер в соответствии с **индивидуальным заданием**, полученным у преподавателя
4. Создать триггер в соответствии с **индивидуальным заданием**, вызывающий хранимую процедуру
5. Выложить скрипт с созданными сущностями в svn
6. Продемонстрировать результаты преподавателю
7. **Выполнение работы**

Вначале была создана тестовая таблица *AUTOTRIG* с полями *INCR* и *ZNACH* типа integer. Далее был написан скрипт для создания триггера для автоматического заполнения поля *INCR* в таблице *AUTOTRIG.*

set term ^;

CREATE OR ALTER trigger AUTOINCR for AUTOTRIG active before insert AS

begin

select max(AUTOTRIG.INCR) from AUTOTRIG into NEW.INCR;

NEW.INCR = NEW.INCR + 1;

end^

set term ;^

Работа триггера:

SQL> select \* from AUTOTRIG;

INCR ZNACH

============ ============

1 20

SQL> insert into AUTOTRIG values (100, 50);

SQL> select \* from AUTOTRIG;

INCR ZNACH

============ ============

1 20

2 50

SQL> insert into AUTOTRIG values (1, 45);

SQL> select \* from AUTOTRIG;

INCR ZNACH

============ ============

1 20

2 50

3 45

По результатам работы триггера видно, что при добавлении значения в таблицу сначала проверяется последнее (максимальное) значение поля *INCR*, оно инкрементируется, а затем подставляется в добавляемую величину.

Затем был создан триггер для контроля целостности данных в подчиненной таблице при удалении/изменении записей в главной таблице *MODELS;*

set term ^;

create or alter exception DEL\_UPD\_FAILED ‘FAILED TO CREATE OR UPDATE THE INFORMATION’^

CREATE OR ALTER trigger CONTRZEL for MODELS

active before delete or update

AS

begin

if (OLD.ID in (select Models\_Code from CARS)) then

exception DEL\_UPD\_FAILED;

end^

set term ;^

**Индивидуальное задание:**

Реализовать триггеры:

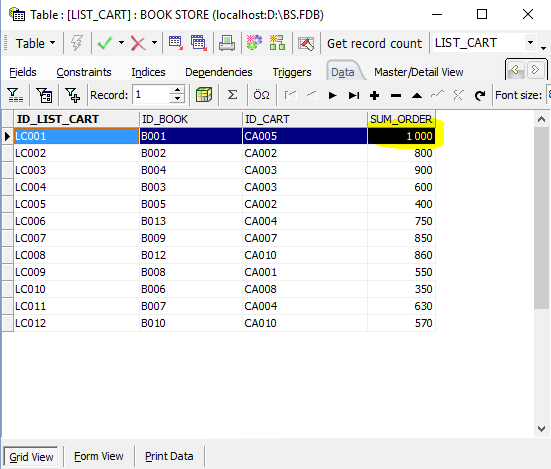
1. При добавлении книги в заказ проверять есть ли такая книга уже в заказе. Если есть - увеличивать количество этой книги в заказе, а повторно одну и ту же книгу не добавлять.
2. Обновлять рейтинг автора с четом продаж его книг за последний год.

**Первый триггер:**

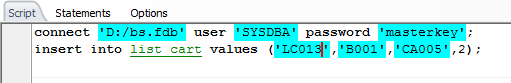
|  |
| --- |
| **connect 'D:/bs.fdb' user 'SYSDBA' password 'masterkey';**  **create or alter trigger lab61 before insert on list\_cart**  **as**  **begin**  **if (new.id\_book in (select id\_book from list\_cart)) then**  **update list\_cart set list\_cart.sum\_order=new.sum\_order+list\_cart.sum\_order where list\_cart.id\_book=new.id\_book;**  **end**  **connect 'D:/bs.fdb' user 'SYSDBA' password 'masterkey';**  **CREATE trigger lab62 after insert on list\_cart**  **as**  **declare variable idd varchar(10);**  **declare variable cc integer;**  **begin**  **idd= new.id\_book;**  **select count(\*) from list\_cart where list\_cart.id\_book =:idd into :cc;**  **if (:cc >1) then**  **DELETE FROM list\_cart WHERE list\_cart.id\_list\_cart=new.id\_list\_cart;**  **end** |

Результат выполнения

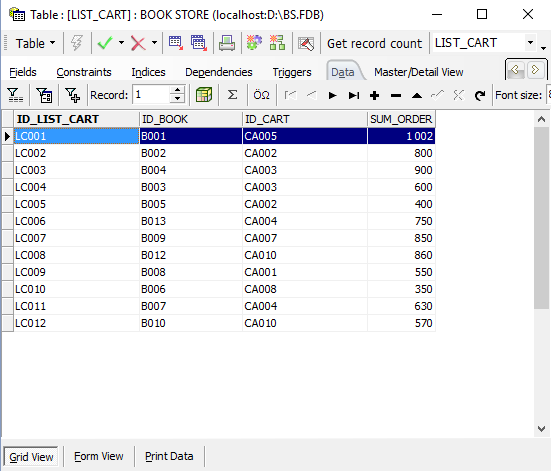
Данные в таблице заказе до добавления



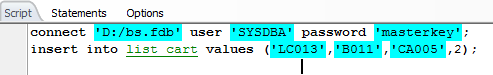
Выполнил команды добавить



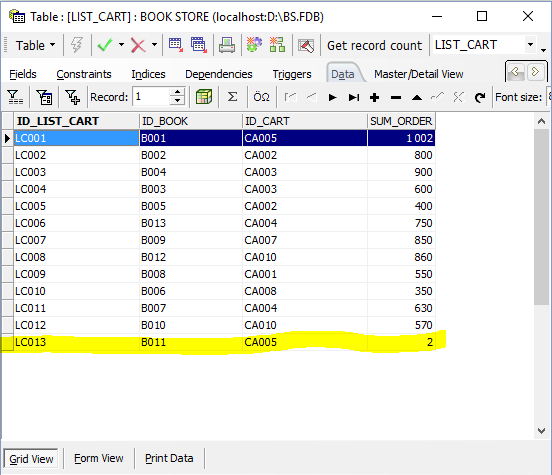
И результат получил



При добавлении другое значение



И реультат получил



**Второй триггер:**

1. **Вывод**

В данной работе мы познакомились с реализацией триггеров. Было создано несколько стандартных триггеров, а так же реализованы триггеры в соответствие с индивидуальным заданием.

С помощью триггеров можно накладывать ограничения на вносимые данные согласно требованиям предметной области БД.

Триггеры полезно использовать для проверки корректности вносимых в БД данных и их целостности.

Также триггеры удобно использовать для оповещения об изменении данных в таблицах.