Санкт-Петербургский Политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

**Отчёт о лабораторной работе №5**

**Дисциплина**: Базы данных

**Тема**: Триггеры

Выполнила студентка гр. 43501/1 В.А.Чернышева

(подпись)

Руководитель А.В. Мяснов

(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**

Был создан триггер для автоматического заполнения поля id в таблице жанров

SET NAMES WIN1251;

CONNECT 'C:\123\BIBLIOTEKA.FDB' USER 'SYSDBA' PASSWORD 'masterkey';

create trigger genre\_1 for genre before insert

as

begin

new.id = gen\_id(gen\_genre\_id,1);

end

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

SET NAMES WIN1251;

CONNECT 'C:\123\BIBLIOTEKA.FDB' USER 'SYSDBA' PASSWORD 'masterkey';

drop trigger book\_genre;

create exception error 'Error deleting genre book';

create trigger book\_genre for genre before delete or update

as

begin

if (OLD.id in (select book.id\_genre from book)) then

exception error;

end;

Индивидуальное задание:

Реализовать триггеры:

1. При выдаче книги проверять есть ли у читателя книги с задолженностью. Если есть - не выдавать.
2. При выдаче книги проверять не превышает ли количество книг, выданное читателю порогового значения. Если превышает - не выдавать.

Первый триггер:

SET NAMES WIN1251;

CONNECT 'C:\123\BIBLIOTEKA.FDB' USER 'SYSDBA' PASSWORD 'masterkey';

create exception E\_debts 'Debts!';

CREATE OR ALTER TRIGGER check\_debts FOR Operation

ACTIVE BEFORE INSERT POSITION 0

as

declare variable cnt integer;

begin

SELECT

sum(1) as cnt

from operation as operation

where operation.retern\_date is null and

dateadd(day, operation.period, operation.date\_of\_issue) < CAST('today' AS DATE) and

operation.id\_readership = new.id\_readership

into :cnt;

if(cnt>0)then

exception e\_debts;

end

Добавление новой операции:

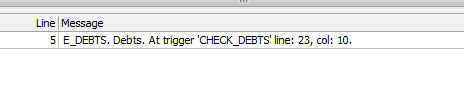
SET NAMES WIN1251;

CONNECT 'C:\123\BIBLIOTEKA.FDB' USER 'SYSDBA' PASSWORD 'masterkey';

insert into operation(id, id\_readership, id\_type\_of\_issue, period, date\_of\_issue, id\_exemp)

values (60, 2, 2, 1,'23.12.2015', 23);

Результат:



Второй триггер:

SET NAMES WIN1251;

CONNECT 'C:\123\BIBLIOTEKA.FDB' USER 'SYSDBA' PASSWORD 'masterkey';

create exception 'Limit book';

CREATE OR ALTER TRIGGER Check\_quantity\_book FOR operation

ACTIVE BEFORE UPDATE POSITION 0

as

declare variable new\_quantity\_book numeric(10);

begin

execute procedure quantity\_book(new.id\_readership) returning\_values (:new\_quantity\_book);

if (:new\_quantity\_book > 3) then exception limit;

end

Вызываемая процедура:

create procedure QUANTITY\_BOOK (id\_readership integer)

returns (QUANTITY\_BOOK NUMERIC(10))

as

begin

FOR select

count(operation.id) as QUANTITY\_BOOK

from operation as operation

where operation.retern\_date is null and

operation.id\_readership = :id\_readership

into

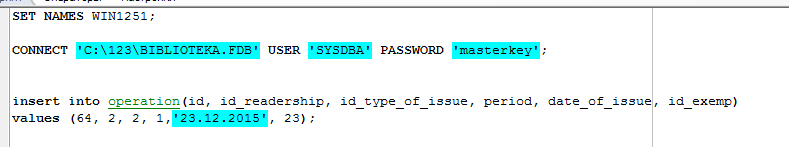
:QUANTITY\_BOOK

do

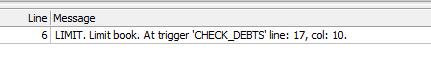
suspend;

end

Запрос:



Результат:



1. **Вывод**

В данной работе были созданы триггеры. Триггер — это хранимая процедура особого типа, которая не вызывается непосредственно, а исполнение которой обусловлено наступлением одного из событий, относящегося к одной конкретной таблице (представлению), или наступлению одного из событий базы данных.

Триггеры полезно использовать для проверки корректности вносимых в БД данных и их целостности. С помощью триггеров можно выдавать сообщения (предупреждения) о том, что необходимо выполнить некоторые действия при изменении таблиц. Также триггеры удобно использовать для оповещения об изменении данных в таблицах. С помощью триггеров можно накладывать ограничения на вносимые данные согласно требованиям предметной области БД.