

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

\_\_\_\_\_  
(подпись) Нгуен Тиен Ву

Руководитель

\_\_\_\_\_  
(подпись) А.В. Мяснов

“\_\_” \_\_\_\_\_ 2016 г.

Санкт-Петербург

2016

## 1. Цель

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

## 2. Программа работы

1. 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
2. Создать триггер в соответствии с **индивидуальным заданием**, полученным у преподавателя
3. Создать триггер в соответствии с **индивидуальным заданием**, вызывающий хранимую процедуру
4. Выложить скрипт с созданными сущностями в svn
5. Продемонстрировать результаты преподавателю

## 3. Выполнение работы

Был создан триггер для автоматического заполнения поля `id_client` в таблице созданной для `client` данного триггера, чтобы не испортить одну из таблиц БД(Для избежания ошибок, вместо получения максимального значения из `ID` таблицы будем использовать генератор):

```
--create generator incr;
set term ^;
create or alter trigger auto_incr for CLIENT before insert
as
begin
  new.id = id_client(incr,1);
end^

set term ;^
```

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

```
set term ^;
CREATE OR ALTER trigger Control for ADRESS
active before delete or update
AS
```

```

begin
    if (OLD.id in (select address.city_id from address)) then
        exception;
    end^
set term ;^

```

### Индивидуальное задание:

Реализовать триггеры:

1. При выплате по страховке проверять есть ли действующий договор на данный тип страхового случая. Если нет - не добавлять.
2. При покупке нового полиса проверять нет ли действующего полиса аналогичного типа. Если есть - не добавлять

### Первый триггер:

```

set term ^;
CREATE OR ALTER trigger checkcase for insurance_case_con
active before insert position 0
as begin
    if (new."CASE" not in (select id_case from "CASE"
    where "CASE".id_case = new."CASE"))
    then
        exception errcase;
    end
set term ;^

```

Работа скрипта:

```
select "CASE".id_case from "CASE";
```

ID_CASE
1
2
3
4
5

```
insert into insurance_case_con values (6,10,31);
```

#### Messages

```

ERRCASE.
cannot add contract .
At trigger 'CHECKCASE' line: 6, col: 3.

```

## Второй триггер:

```
set term ^;  
CREATE OR ALTER trigger polic_bi0 for polic  
active before insert position 0  
as begin  
    if (new.name_polic like (select name_polic from "POLIC"  
        where "POLIC".id_polic <> new.id_polic))  
    then  
        exception ;  
  
end  
  
set term ;^
```

## Работа скрипта:

```
select name_polic from "POLIC";
```

NAME_POLIC
ALL
CAR+PRO
PRO+MED
TRA+ACC
CAR
PROPERTY
MEDICINE
TRAVELS
ACCIDENT

```
insert into "POLIC" values (10, 'ACCIDENT',15000,'12.03.2015') ;
```

### Messages

```
multiple rows in singleton select.  
multiple rows in singleton select.  
At trigger 'POLIC_BI0' line: 4, col: 3.
```

#### **4. Вывод**

В результате работы было проведено знакомство с триггерами Firebird. Было создано несколько стандартных триггеров, а так же реализованы триггеры в соответствии с индивидуальным заданием. Триггеры полезно использовать для проверки корректности вносимых в БД данных и их целостности. При данном контроле, или при добавлении записей в БД, при определенных условиях(условиях вызова триггера) выводятся диагностические сообщения(с помощью использования исключений).

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