**Практическая работа по теме «Табличные триггеры»**

*Цель работы* — формирование умений по разработке и использованию табличных триггеров в PL/SQL

**ПРИМЕР 1. необходимо автоматически заполнять поле NN таблицы Students\_Hobby и если пользователь не занес дату начала действия хобби, то указывать текущую дату.**

--Создание последовательности для заполнения поля NN таблицы Students\_Hobby

CREATE SEQUENCE student\_seq

START WITH 1 INCREMENT BY 1 NOCACHE;

---------Строковый триггер для таблицы Students\_Hobby . Имеется доступ к значениям полей через корреляционные имена :new и :old

CREATE OR REPLACE TRIGGER GenerateId

BEFORE INSERT OR UPDATE ON Students\_Hobby

--срабатывает перед вставкой или изменением строк

FOR EACH ROW

BEGIN

-- в поле NN через корреляционное имя :new заносится значение из последовательности вне зависимости от того, что ввел пользователь

SELECT student\_seq.nextval INTO :new.Nn FROM dual;

-- если пользователь не заполнил дату начала хобби, то подставляется текущая дата

if :new.START\_DATA is null then

:new.START\_DATA:=sysdate;

end if;

END GenerateId;

**ПРИМЕР 2. запретить изменять данные о судентах с 9 до 18 часов (по времени сервера)**

-----Операторный триггер, запрещает изменять данные о студентах с 9 до 18 часов (по времени сервера)

CREATE or replace TRIGGER UP\_DEL\_Time

BEFORE UPDATE OR DELETE ON students

BEGIN

IF to\_char(sysdate, 'HH24') BETWEEN 9 and 18 THEN

RAISE\_APPLICATION\_ERROR(-20101, 'НЕЛЬЗЯ ИЗМЕНЯТЬ ДАННЫЕ О СТУДЕНТАХ В ЭТО ВРЕМЯ');

END IF;

END UP\_DEL\_Time;

**Задание 2. необходимо отслеживать все изменения данных в таблице Студенты, записывая в таблицу STUD\_ARCHIV пользователя, который произвел изменения, дату изменения, действие (Insert/Update/Delete) и количество измененных данной командой записей;**

***Создадим таблицу, в которую будут записываться действия пользователей с таблицей Students.***

-----Создание таблицы, в которую будут протоколироваться действия

Create table STUD\_ARCHIV

(id number,

User\_Name varchar2(100),

Action varchar2(10),

DATA\_A Date,

Count\_Rows number);

--- Создание последовательности для нумерации в таблице STUD\_ARCHIV

CREATE SEQUENCE STUD\_ARCHIV\_seq

START WITH 1 INCREMENT BY 1 NOCACHE;

***Создадим 2 триггера After: первый будет строковым для подсчета измененных записей, а второй триггер будет операторным для записи нужных данных в архив.***

-----Строковый триггер для подсчета количества записей, попавших по действие команды

create or replace TRIGGER Stud\_Archive

AFTER INSERT or UPDATE or DELETE ON Students

FOR EACH ROW

BEGIN

Students\_.qq:=Students\_.qq+1; -- используем глобальную переменную qq, определенную в пакете Students\_. Если пакета нет, то создайте пакет только с этой переменной (см. внизу).

END;

----Операторный триггер для добавления записи в архив

create or replace TRIGGER Add\_Archive

AFTER INSERT or UPDATE or DELETE ON Students

DECLARE

act VARCHAR2(25);

dat date;

usernm VARCHAR2(25);

crows NUMBER(5);

BEGIN

usernm:= apex\_util.get\_username(apex\_util.get\_current\_user\_id);

IF INSERTING THEN

act:='Insert';

elsif UPDATING then

act:='Update';

else

act:='Delete';

END IF;

Select sysdate into dat from dual;

INSERT INTO STUD\_ARCHIV(id,User\_Name,Action,DATA\_A,Count\_Rows) VALUES (STUD\_ARCHIV\_seq.nextval,usernm,act,dat,Students\_.qq);

Students\_.qq:=0;

END;

----Создание пакета с глобальной переменной. Нужно создавать, если Вы не делали этого ранее

create package Students\_ as

--объявление глобальной переменной

qq number(4):=0;

end;

***Индивидуальные задания по теме «Табличные триггеры»***

***Вариант 1.***

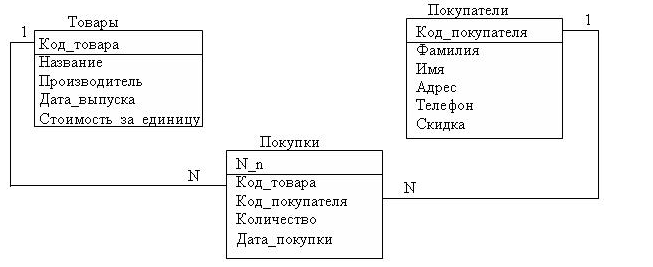
Для Вашей БД, схема которой представлена на рисунке (скрипт «Авторы\_картины\_музеи.sql») необходимо отслеживать все изменения данных в таблице **Авторы**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк.



***Вариант 2.***

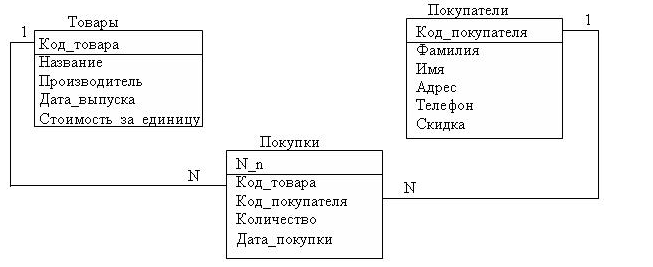
Для Вашей БД, схема которой представлена на рисунке (скрипт «Авторы\_картины\_музеи.sql») необходимо отслеживать все изменения данных в таблице **Музеи**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк

***Вариант 3.***

Для Вашей БД, схема которой представлена на рисунке (скрипт «товары\_покупки\_покупатели.sql») необходимо отслеживать все изменения данных в таблице **Товары**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк

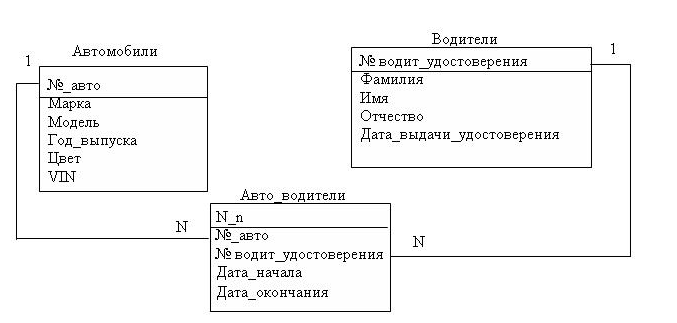
***Вариант 4.***

Для Вашей БД, схема которой представлена на рисунке (скрипт «товары\_покупки\_покупатели.sql») необходимо отслеживать все изменения данных в таблице **Покупатели**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк



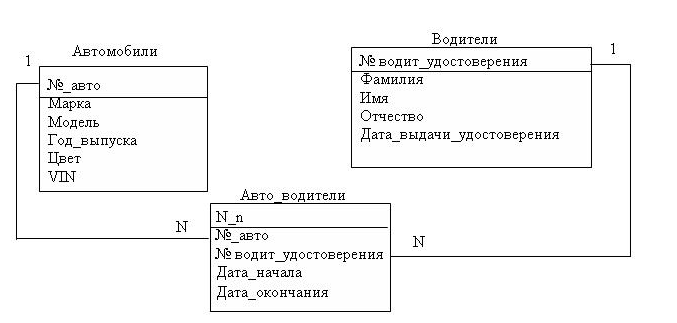
***Вариант 5.***

Для Вашей БД, схема которой представлена на рисунке (скрипт «автомобили\_водители.sql») необходимо отслеживать все изменения данных в таблице **Автомобили**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк



***Вариант 6.***

Для Вашей БД, схема которой представлена на рисунке (скрипт «автомобили\_водители.sql») необходимо отслеживать все изменения данных в таблице **Водители**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк



***Вариант 7.***

Для Вашей БД, схема которой представлена на рисунке (скрипт «издания\_авторы.sql») необходимо отслеживать все изменения данных в таблице **Издания**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк



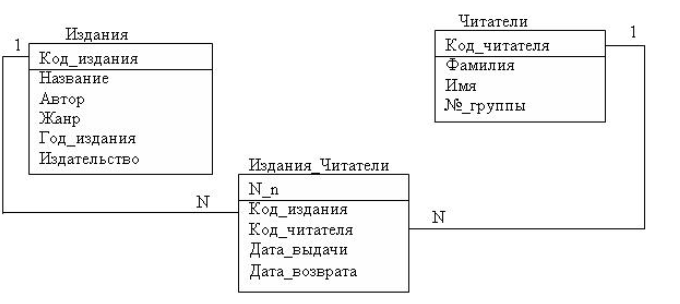
***Вариант 8.***

Для Вашей БД, схема которой представлена на рисунке (скрипт «издания\_авторы.sql») необходимо отслеживать все изменения данных в таблице **Авторы**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк



***Вариант 9.***

Для Вашей БД, схема которой представлена на рисунке (скрипт «**издания\_читатели.sql**») необходимо отслеживать все изменения данных в таблице **Издания**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк

. 

***Вариант 10.***

Для Вашей БД, схема которой представлена на рисунке (скрипт «**издания\_читатели.sql**») необходимо отслеживать все изменения данных в таблице **Читатели**, записывая в таблицу ARCHIVE пользователя, который произвел изменения, дату и время изменения, действие (Insert/Update/Delete) и количество измененных строк

