



Министерство науки и высшего образования Российской Федерации  
Калужский филиал  
федерального государственного бюджетного  
образовательного учреждения высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и Управление»

КАФЕДРА ИУК4 «Программное обеспечение ЭВМ, информационные технологии»

## ЛАБОРАТОРНАЯ РАБОТА №5

ДИСЦИПЛИНА: «Базы данных»

Выполнил: студент гр. ИУК4-52Б \_\_\_\_\_ (Калашников А. С.)  
(Подпись) (Ф.И.О.)

Проверил: \_\_\_\_\_ (Глебов С. А.)  
(Подпись) (Ф.И.О.)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Калуга, 2022

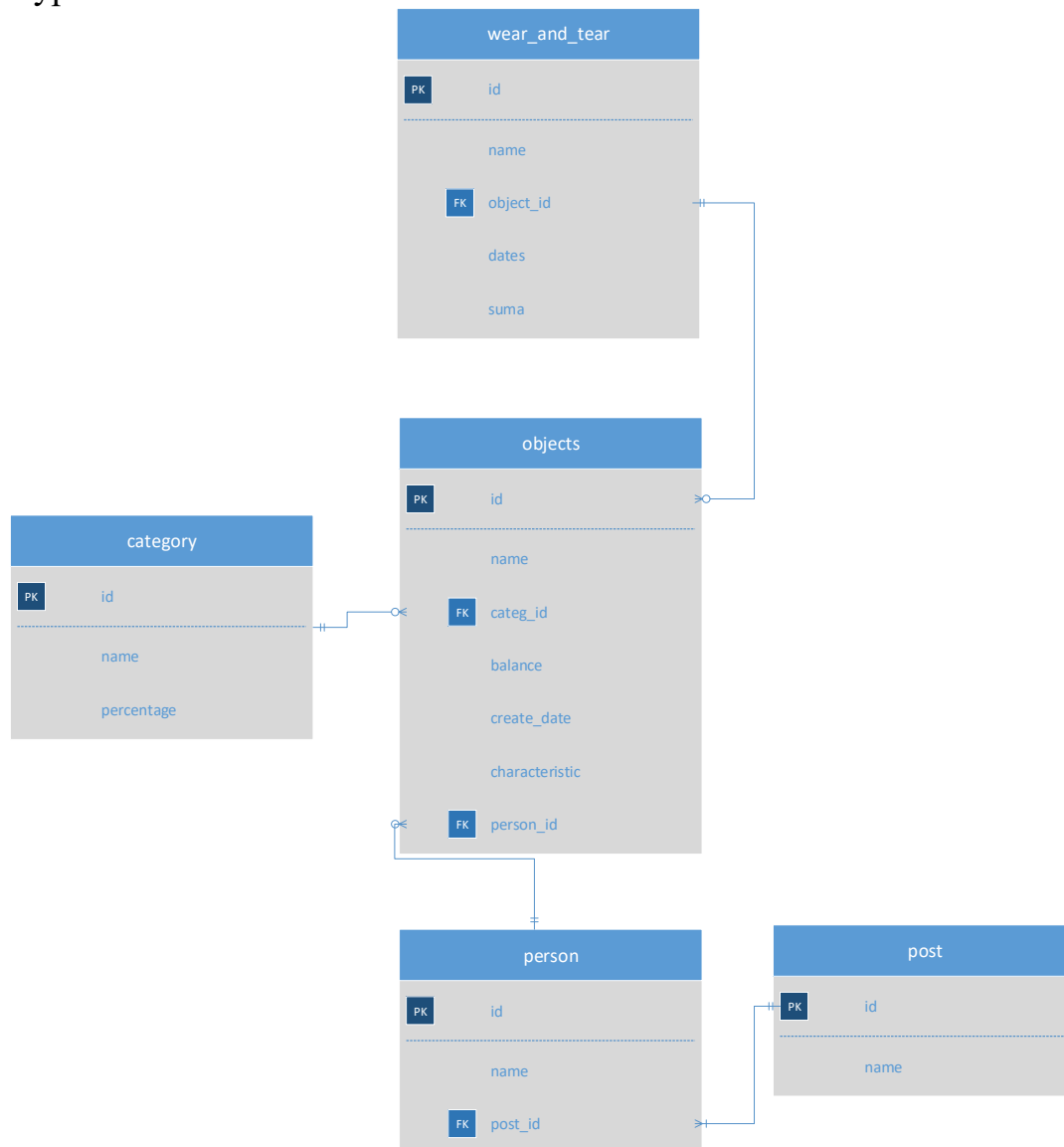
**Целью** выполнения лабораторной работы является сформировать практические навыки разработки триггеров.

**Основными задачами** выполнения лабораторной работы являются:

- разработать триггер для реализации автоинкремента
- разработать триггер для обеспечения каскадных воздействий
- разработать триггер для обеспечения бизнес-правил
- разработать триггер для журнализации

### Вариант 10

В базе данных, созданной в предыдущей лабораторной работе, разработать триггеры для автоинкремента, бизнес-правил, каскадных воздействий и журнала изменений.



### Автоинкремент

```
create trigger id_object
for objects
active
before
insert position 0
as
```

```

begin
    if(new.id is null)
    then
        new.id=gen_id("GENERATORID",1);
    end
end

```

## Бизнес-правило

```

CREATE OR ALTER TRIGGER " No_Update_For_Parent " FOR OBJECTS
ACTIVE AFTER INSERT POSITION 0
AS
DECLARE VARIABLE delta INTEGER;
DECLARE VARIABLE dates date;
DECLARE VARIABLE percentage INTEGER;
DECLARE VARIABLE SUMMA INTEGER;
begin
    select "SUMA"
    From wear_and_tear
    Where OBJECT_ID=NEW."ID"
    into :SUMMA;
    select "PERCENTAGE"
    From category
    Where ID=NEW."CATEG_ID"
    into :percentage;
    select "DATES"
    From wear_and_tear
    Where OBJECT_ID=NEW."ID"
    into :percentage;
    delta=(extract(month from new.create_date)-extract(month from
dates))*percentage/100;
    insert into objects ("BALANCE") values(:summa-:delta);
    if (new.balance >= 0) then
        exception BALANCENULL;
    end
end

```

**Выполнение бизнес-правила делает все за человека и изменяет цену триггер, а не сам человек**

## Каскадные воздействия

```

CREATE TRIGGER "Change_Price" FOR objects
ACTIVE AFTER UPDATE POSITION 0
AS
DECLARE VARIABLE delta INTEGER;
DECLARE VARIABLE tmp INTEGER;
begin
    delta=NEW."BALANCE"-OLD."BALANCE";
    if (delta<>0) then
        begin
            select "BALANCE"
            From objects
            Where ID=OLD."ID_PARENT"
            into :tmp;
            update objects set "BALANCE"=:tmp+:delta
            where "ID"=OLD."ID_PARENT";
        end
    end
end

```

## Журнализация

### Добавление

```

CREATE TABLE objects_Log (
    "Time" TIMESTAMP,
    "User" VARCHAR(31),
    "Column" VARCHAR(15),

```

```

"Old_Value" VARCHAR(100),
"New_Value" VARCHAR(100),
"Action" VARCHAR(3) CHECK ("Action" IN ('INS', 'UPD', 'DEL')),
"ID" INTEGER,
CONSTRAINT "FK_Log" FOREIGN KEY (ID)
REFERENCES objects (ID) ON DELETE CASCADE
ON UPDATE CASCADE
);

```

## Изменение

```

CREATE TRIGGER objects_log_insert FOR objects
ACTIVE AFTER INSERT POSITION 0
AS
BEGIN
    INSERT INTO objects_log
    VALUES (current_timestamp, current_user, 'Objects', null, null, 'INS',
NEW."ID");
END

```

```

CREATE TRIGGER objects_log_update FOR objects
ACTIVE AFTER UPDATE POSITION 0
AS
begin
    if (OLD."ID"<>NEW."ID") then
        INSERT INTO objects_log
        VALUES (current_timestamp, current_user, 'ID',
        OLD."ID", NEW."ID", 'UPD', NEW."ID");

    if (OLD."NAME"<>NEW."NAME") then
        INSERT INTO objects_log
        VALUES (current_timestamp, current_user, 'NAME',
        OLD."NAME", NEW."NAME", 'UPD', NEW."ID");

    if (OLD."CATEG_ID"<>NEW."CATEG_ID") then
        INSERT INTO objects_log
        VALUES (current_timestamp, current_user, 'CATEG_ID',
        OLD."CATEG_ID", NEW."CATEG_ID", 'UPD', NEW."ID");

    if (OLD."BALANCE"<>NEW."BALANCE") then
        INSERT INTO objects_log
        VALUES (current_timestamp, current_user, 'BALANCE',
        OLD."BALANCE", NEW."BALANCE", 'UPD', NEW."ID");

    if (OLD."CREATE_DATE"<>NEW."CREATE_DATE") then
        INSERT INTO objects_log
        VALUES (current_timestamp, current_user, 'CREATE_DATE',
        OLD."CREATE_DATE", NEW."CREATE_DATE", 'UPD', NEW."ID");

    if (OLD."CHARACTERISTICS"<>NEW."CHARACTERISTICS") then
        INSERT INTO objects_log
        VALUES (current_timestamp, current_user, 'CHARACTERISTICS',
        OLD."CHARACTERISTICS", NEW."CHARACTERISTICS", 'UPD', NEW."ID");

    if (OLD."PERSON_ID"<>NEW."PERSON_ID") then
        INSERT INTO objects_log
        VALUES (current_timestamp, current_user, 'PERSON_ID',
        OLD."PERSON_ID", NEW."PERSON_ID", 'UPD', NEW."ID");
END

```

## Удаление

```

CREATE TRIGGER objects_log_delete FOR objects

```

```

ACTIVE
AFTER UPDATE
AS
BEGIN
    INSERT INTO objects_log
    VALUES (current_timestamp, current_user, 'Objects', null, null, 'DEL',
old."ID") ;
END

```

Time	User	Column	Old_Value	New_Value	Action	ID
28.09.2022 15:31	SYSDBA	Objects	<null>	<null>	DEL	1
28.09.2022 15:31	SYSDBA	Objects	<null>	<null>	DEL	2
28.09.2022 15:31	SYSDBA	Objects	<null>	<null>	DEL	3
28.09.2022 15:31	SYSDBA	Objects	<null>	<null>	DEL	4
28.09.2022 15:32	SYSDBA	Objects	<null>	<null>	DEL	5

### Рис.1 Журнализация

**Вывод:** в ходе выполнения лабораторной работы является сформировать практические навыки разработки триггеров.