|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **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**

**«Использование триггеров»**

**ДИСЦИПЛИНА: «Базы данных»**

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИУК4-52Б | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Карельский М.К. )  (Подпись) |
| Проверил: | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Глебов С.А. )  (Подпись) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |

Калуга, 2022

**Цель:** сформировать практические навыки разработки триггеров.

**Задачи:**

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

**Предметная область** – социальная сеть.



Рис. 1. Связи таблиц

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

**Результат:**

Триггеры автоинкремента:

create generator gen\_id\_users;

set generator gen\_id\_users to 6;

create or alter trigger take\_id\_users for users

active before insert position 0

AS

Begin

if (NEW.id IS NULL) then

Begin

NEW.id = gen\_id(gen\_id\_users, 1);

End

End;

create generator gen\_id\_friends;

set generator gen\_id\_friends to 12;

create or alter trigger take\_id\_friends for friends

active before insert position 0

AS

Begin

if (NEW.id IS NULL) then

Begin

NEW.id = gen\_id(gen\_id\_friends, 1);

End

End;

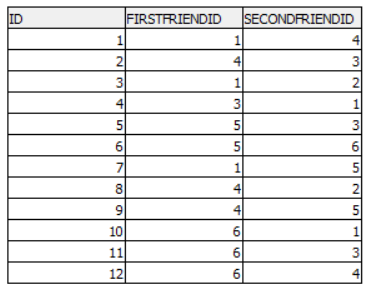
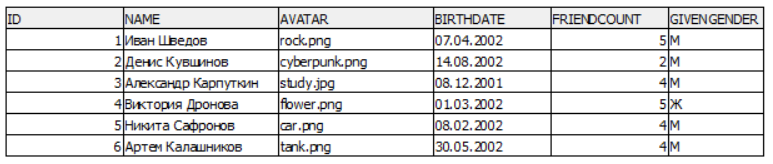


Рис. 2.1 Состояние таблиц до добавления пользователя и дружеской связи

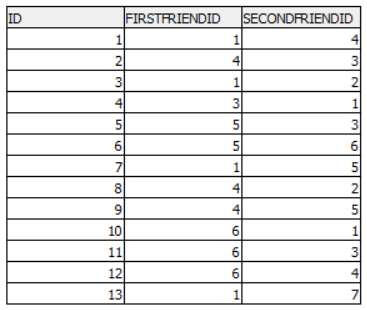
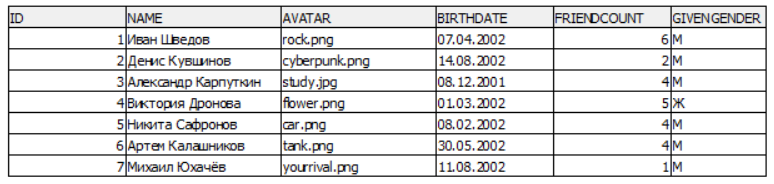


Рис. 3.1. Состояние таблиц после добавления пользователя и дружеской связи

Триггер каскадных воздействий:

create or alter trigger delete\_friend for users

active after delete position 0

AS

Begin

DELETE from friends

Where friends.firstfriendid = OLD.id or friends.secondfriendid = OLD.id;

End;

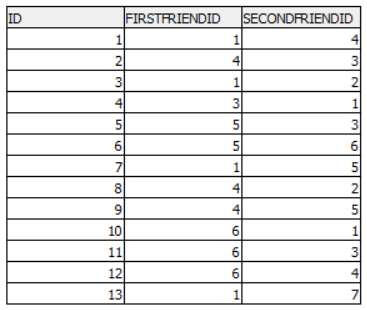
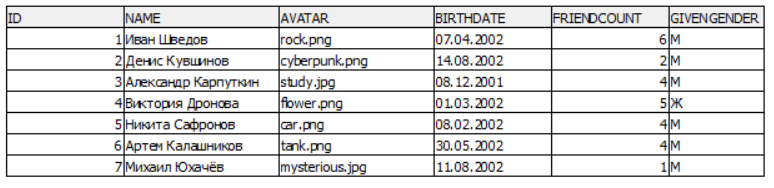


Рис. 3.1. Состояние таблиц до удаления пользователя

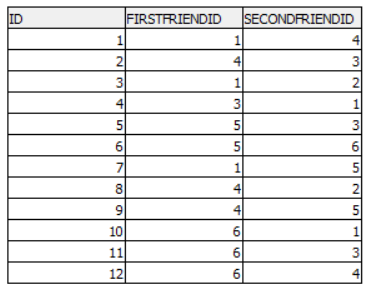
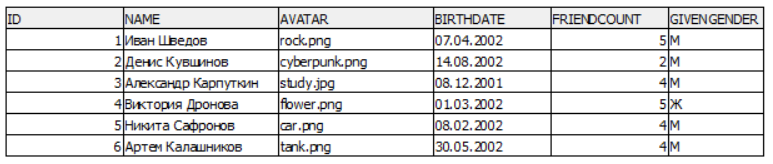


Рис. 3.1 Состояние таблиц после удаления пользователя

Триггеры журнала изменений:

CREATE TABLE users\_log (

id bigint NOT NULL PRIMARY KEY,

log\_time TIMESTAMP NOT NULL,

editor VARCHAR(32) NOT NULL,

changed\_column VARCHAR(11) NOT NULL,

old\_value VARCHAR(20),

new\_value VARCHAR(20),

action\_type VARCHAR(6) NOT NULL

);

create generator gen\_id\_users\_log;

create or alter trigger take\_id\_users\_log for users\_log

active before insert position 0

AS

Begin

if (NEW.id IS NULL) then

Begin

NEW.id = gen\_id(gen\_id\_users\_log, 1);

End

End;

create or alter trigger users\_insert\_log FOR users

active after insert position 1

AS

Begin

INSERT into users\_log (log\_time, editor, changed\_column, old\_value, new\_value, action\_type)

VALUES (current\_timestamp, current\_user, 'id',

NULL, NEW.id, 'insert');

End;

create or alter trigger users\_delete\_log FOR users

active after delete position 1

AS

Begin

INSERT into users\_log (log\_time, editor, changed\_column, old\_value, new\_value, action\_type)

VALUES (current\_timestamp, current\_user, 'id',

OLD.id, NULL, 'delete');

End;

create or alter trigger users\_update\_log FOR users

active after update position 1

AS

Begin

if (OLD.id <> NEW.id) then

INSERT into users\_log (log\_time, editor, changed\_column, old\_value, new\_value, action\_type)

VALUES (current\_timestamp, current\_user, 'id',

OLD.id, NEW.id, 'update');

if (OLD.name <> NEW.name) then

INSERT into users\_log (log\_time, editor, changed\_column, old\_value, new\_value, action\_type)

VALUES (current\_timestamp, current\_user, 'name',

OLD.name, NEW.name, 'update');

if (OLD.avatar <> NEW.avatar) then

INSERT into users\_log (log\_time, editor, changed\_column, old\_value, new\_value, action\_type)

VALUES (current\_timestamp, current\_user, 'avatar',

OLD.avatar, NEW.avatar, 'update');

if (OLD.birthdate <> NEW.birthdate) then

INSERT into users\_log (log\_time, editor, changed\_column, old\_value, new\_value, action\_type)

VALUES (current\_timestamp, current\_user, 'birthdate',

OLD.birthdate, NEW.birthdate, 'update');

if (OLD.givengender <> NEW.givengender) then

INSERT into users\_log (log\_time, editor, changed\_column, old\_value, new\_value, action\_type)

VALUES (current\_timestamp, current\_user, 'givengender',

OLD.givengender, NEW.givengender, 'update');

End;

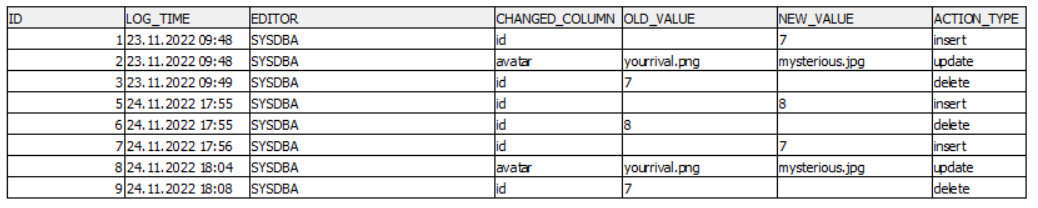


Рис. 4. Журнал таблицы пользователей

Триггер бизнес-правила (проверка на отсутствие повторения дружеской связи):

create EXCEPTION users\_are\_already\_friends 'Users are already friends';

create or alter trigger check\_friendship FOR friends

active before insert position 0

AS

Begin

if (exists(

select \*

from friends

where firstfriendid = NEW.firstfriendid and secondfriendid = NEW.secondfriendid

or firstfriendid = NEW.secondfriendid and secondfriendid = NEW.firstfriendid

)

) then EXCEPTION users\_are\_already\_friends;

End;

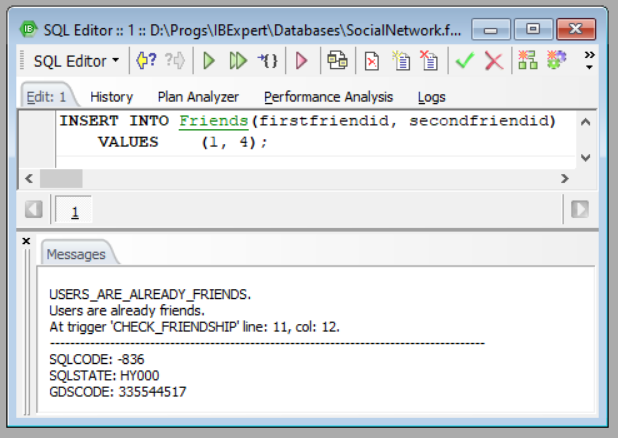
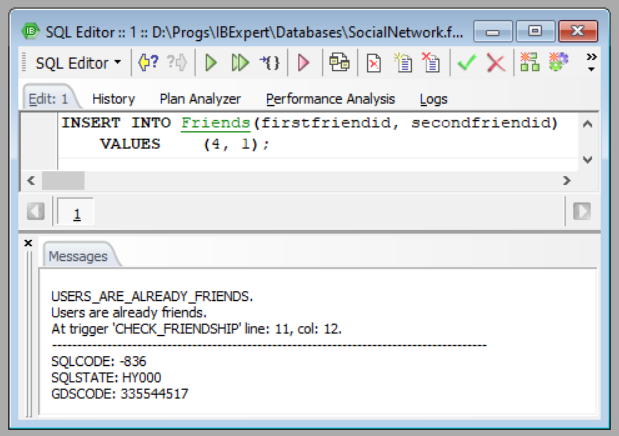
 

Рис. 5. Попытки добавить уже установленную дружескую связь

**Вывод:** в ходе выполнения лабораторной работы были получены практические навыки разработки триггеров.