

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»



Лабораторна робота №12
З дисципліни «Організація баз даних та знань»

Виконав:
студент групи КН-210
Бікєєв Андрій

Викладач:
Мельникова Н. І.

Львів – 2020

Тема: Розробка та застосування тригерів

Мета: Розробити SQL запити, які моделюють роботу тригерів: каскадне знищення, зміна та доповнення записів у зв'язаних таблицях.

Короткі теоретичні відомості

Тригер – це спеціальний вид користувацької процедури, який виконується автоматично при певних діях над таблицею, наприклад, при додаванні чи оновленні даних. Кожен тригер асоційований з конкретною таблицею і подією. Найчастіше тригери використовуються для перевірки коректності вводу нових даних та підтримки складних обмежень цілісності. Крім цього їх використовують для автоматичного обчислення значень полів таблиць, організації перевірок для захисту даних, збирання статистики доступу до таблиць баз даних чи реєстрації інших подій.

Для створення тригерів використовують директиву CREATE TRIGGER.
Синтаксис:

```
CREATE  
[DEFINER = { користувач | CURRENT_USER }]  
TRIGGER ім'я_тригера час_виконання подія_виконання ON назва_таблиці FOR  
EACH ROW тіло_тригера
```

Аргументи:

DEFINER

Задає автора процедури чи функції. За замовчуванням – це CURRENT_USER.

ім'я_тригера

Ім'я тригера повинно бути унікальним в межах однієї бази даних.

час_виконання

Час виконання тригера відносно події виконання. BEFORE – виконати тіло тригера до виконання події, AFTER – виконати тіло тригера після події.

подія_виконання

Можлива подія – це внесення (INSERT), оновлення (UPDATE), або видалення (DELETE) рядка з таблиці. Один тригер може бути пов’язаний лише з однією подією. Команда AFTER INSERT, AFTER UPDATE, AFTER DELETE визначає виконання тіла тригера відповідно після внесення, оновлення, або видалення даних з таблиці. Команда BEFORE INSERT, BEFORE UPDATE, BEFORE DELETE визначає виконання тіла тригера відповідно до внесення, оновлення, або видалення даних з таблиці.

ON назва_таблиці

Таблиця, або віртуальна таблиця (VIEW), для якої створюється даний тригер. При видаленні таблиці з бази даних, автоматично видаляються всі пов’язані з нею тригери.

FOR EACH ROW тіло_тригера

Задає набір SQL директив, які виконує тригер. Тригер викликається і виконується для кожного зміненого рядка. Директиви можуть об’єднуватись командами BEGIN ... END та містити спеціальні команди OLD та NEW для доступу до попереднього та нового значення поля у зміненому рядку відповідно. В тілі тригера дозволено викликати збережені процедури, але заборонено використовувати транзакції, оскільки тіло тригера автоматично виконується як одна транзакція.

NEW.назва_поля

Повертає нове значення поля для зміненого рядка. Працює лише при подіях INSERT та UPDATE. У тригерах, які виконуються перед (BEFORE) подією можна змінити нове значення поля командою SET NEW.назва_поля = значення.

OLD.назва_поля

Повертає старе значення поля для зміненого рядка. Можна використовувати лише при подіях UPDATE та DELETE. Змінити старе значення поля не можливо.

Щоб видалити створений тригер з бази даних, потрібно виконати команду DROP TRIGGER назва_тригера.

Хід роботи

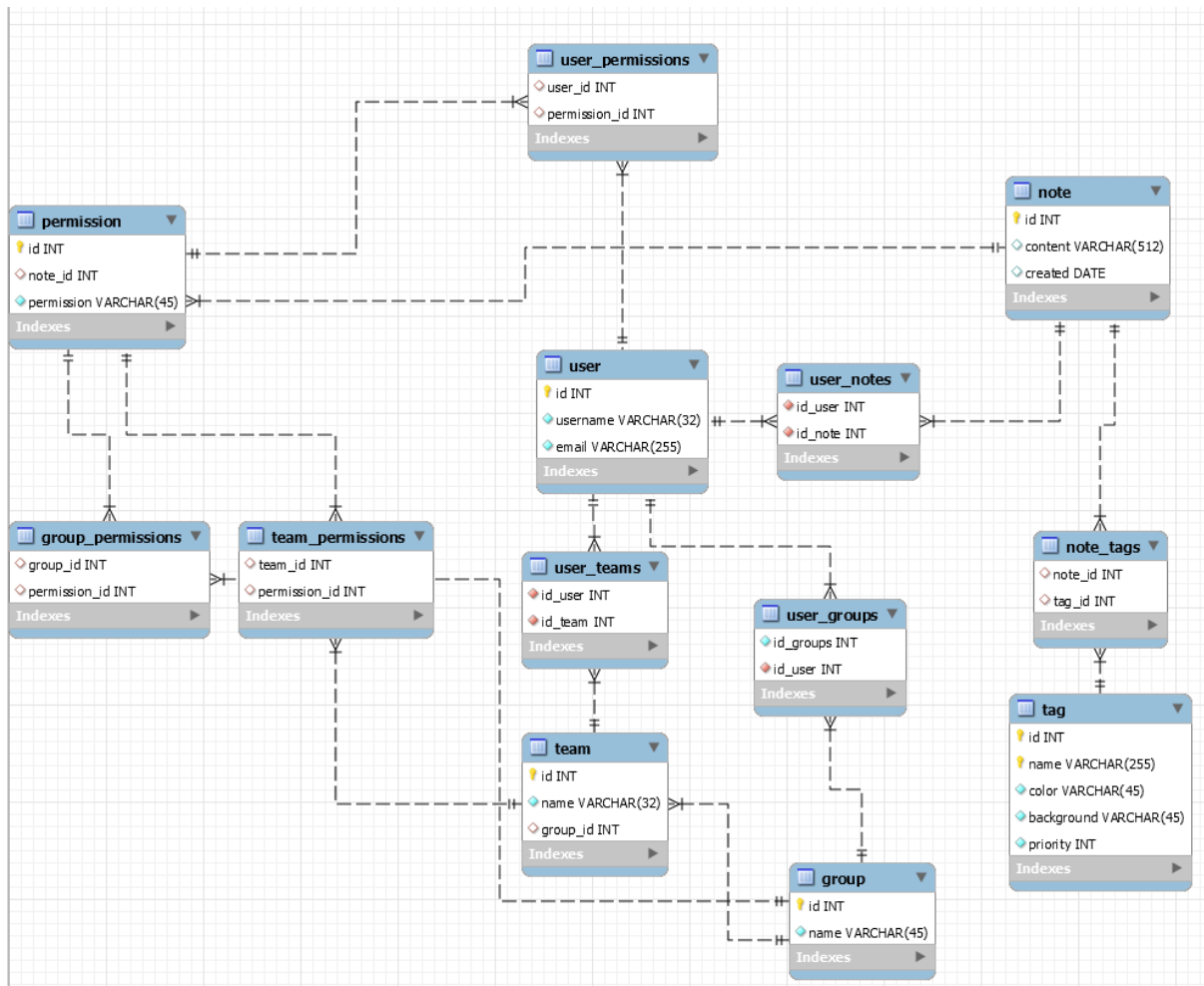


Рис 1. ER-діаграма

Потрібно розробити тригери які будуть виконувати наступні дії:

1. Каскадне оновлення таблиці відношення заміток з тегами(note_tags) при видаленні запису з таблиці tag.
2. Підбір кольору фону для тегу, при внесенні нового тегу з кольором у таблицю tag.
3. Тригер для таблиці note, який буде фіксувати у полі created дату внесення запису.

1.

create

trigger tag_delete after delete

on tag for each row

delete from note_tags as nt where nt.tag_id = OLD.id;

Таблиця до спрацювання тригера:

	note_id	tag_id
▶	1	5
	2	3
	3	2
	3	6
	3	7
	3	1
	4	6
	4	2
	5	5
	5	7
	5	3
	6	1
	6	6

Після спрацювання:

delete from tag where id = 1;

	note_id	tag_id
▶	1	5
	2	3
	3	2
	3	6
	3	7
	4	6
	4	2
	5	5
	5	7
	5	3
	6	6
	7	4
	7	7

2.

create

trigger updateBg before update

on tag

for each row

update tag set background = calculate_background(NEW.color) where color = OLD.color;

Таблиця до спрацювання тригера:

	id	name	color	background	priority
▶	2	Subprimebackend	#000000	#52a84d	1
	3	RationalAnonymous	#ffffff	#ffffff	8
	4	QuickCat	#000000	#a4fa94	8
	5	Feelingperson	#000000	#c48596	6
	6	Irrationalfrontend	#000000	#28f965	3
	7	UnrealShark	#000000	#6602e1	6
✱	NULL	NULL	NULL	NULL	NULL

Після спрацювання:

update tag set color = "#0f0f0f" where id = 2;

	id	name	color	background	priority
▶	2	Subprimebackend	#0f0f0f	#ffffff	1
	3	RationalAnonymous	#ffffff	#ffffff	8
	4	QuickCat	#000000	#a4fa94	8
	5	Feelingperson	#000000	#c48596	6
	6	Irrationalfrontend	#000000	#28f965	3
	7	UnrealShark	#000000	#6602e1	6
✱	NULL	NULL	NULL	NULL	NULL

3.

create

trigger note_date before insert

on note for each row set NEW.created = current_date();

Таблиця до спрацювання тригера:

	id	content	created
	15	Lover awakes Cat. UnrealDatabase bears Cat r...	2019-12-15
	16	AnotherHardware grayly teleporting TotalFront...	2019-12-23
	17	RandomHater teleports FastBot; RobustHater r...	2019-12-14
	18	Bot-lovingLover enjoyably gnawing FeelingSof...	2019-12-22
	19	RealAnonymous rationally teleporting Readable...	2019-12-19
	20	Irrationalcoffee bears CoolNobody. Cat-hating...	2019-12-14
▶	21	Backend-lovingKing bot-hatingly creating Rando...	2019-12-22
	22	RealSnail subprimely arising ResponsibleCat? wh...	2019-12-19
	23	IrrationalPerson readably backsliding randomb...	2019-12-15
	24	RationaltotalAnonymous happily being Responsi...	2019-12-20
	25	UnrealDog happily awaking RobustShark. Anon...	2019-12-19
✱	NULL	NULL	NULL

Після спрацювання:

Result Grid			
		Filter Rows:	
		Edit:	
		Export/	
	id	content	created
	17	RandomHater teleports FastBot; RobustHater r...	2019-12-14
	18	Bot-lovingLover enjoyably gnawing FeelingSof...	2019-12-22
	19	RealAnonymous rationally teleporting Readable...	2019-12-19
	20	Irrationalcoffee bears CoolNobody. Cat-hating...	2019-12-14
	21	Backend-lovingKing bot-hatingly creating Rando...	2019-12-22
	22	RealSnail subprimely arising ResponsibleCat? wh...	2019-12-19
	23	IrrationalPerson readably backsliding randomb...	2019-12-15
	24	RationaltotalAnonymous happily being Responsi...	2019-12-20
	25	UnrealDog happily awaking RobustShark. Anon...	2019-12-19
	27	New note...	2020-05-27
✱	NULL	NULL	NULL

Висновок: в даній лабораторній роботі я навчився розробляти та використовувати різні види тригерів.