

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**

Інститут комп'ютерних наук та інформаційних технологій

Кафедра систем штучного інтелекту



**Звіт до лабораторної роботи №12
з дисципліни
“ОБДЗ”**

Виконав:
ст. гр. КН-211
Ткачук Орест

Викладач:
Якимишин Х.М.

Львів – 2020

Лабораторна робота №12

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

Хід роботи.

1. Каскадне оновлення таблиці квитків при видаленні покупця. Діюче обмеження зовнішнього ключа при видаленні покупця видаляє квиток. Натомість, за допомогою тригера, квитку будемо присвоювати в поле isReserved 0 за замовчуванням.

```
DELIMITER $$
```

```
CREATE
```

```
TRIGGER customer_delete BEFORE DELETE
```

```
ON lab12.customer FOR EACH ROW
```

```
BEGIN
```

```
UPDATE lab12.ticket SET isReserved=0 where
```

```
customerID=OLD.id_customer;
```

```
END$$
```

```
DELIMITER ;
```

```
+-----+-----+-----+-----+-----+-----+-----+
| id_ticket | seanceID | customerID | type      | place | price_ticket | isReserved |
+-----+-----+-----+-----+-----+-----+-----+
|          1 |          1 |          1 | standart | 40    | NULL        | 1          |
|          2 |          1 |          1 | standart | 41    | NULL        | 1          |
|          3 |          2 |          2 | standart | 20    | NULL        | 1          |
|          4 |          3 |          3 | standart | 23    | NULL        | 1          |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> delete from customer where id_customer=1;
Query OK, 1 row affected (0.01 sec)

mysql> select * from ticket;
+-----+-----+-----+-----+-----+-----+-----+
| id_ticket | seanceID | customerID | type      | place | price_ticket | isReserved |
+-----+-----+-----+-----+-----+-----+-----+
|          1 |          1 |          1 | standart | 40    | NULL        | 0          |
|          2 |          1 |          1 | standart | 41    | NULL        | 0          |
|          3 |          2 |          2 | standart | 20    | NULL        | 1          |
|          4 |          3 |          3 | standart | 23    | NULL        | 1          |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

2. Створимо тригер, який буде шифрувати пароль користувача функцією

```
DELIMITER $$
```

```
CREATE
```

```
TRIGGER author_password BEFORE INSERT
```

```
ON lab12.staff FOR EACH ROW
```

```
BEGIN
```

```
SET NEW.password = AES_ENCRYPT(NEW.password, 'key');
```

```
END$$
```

```
DELIMITER ;
```

```
mysql> INSERT INTO staff
-> VALUES
-> (15,"pos2","name15","777","1997-04-30","mail.name15@gmail.com","+380677778899",8,"passname77857");
Query OK, 1 row affected (0.01 sec)
```

```
mysql>
mysql> select * from staff;
```

id_staff	position	staff_name	salary	birthday	email	phone	theaterID	password
1	pos1	name1	200	1991-08-10	mail.name1@gmail.com	+380671112233	8	УJтf\$ДeBhaDэТ
2	pos1	name2	300	1995-09-12	mail.name2@gmail.com	+380672223344	9	лAE2=8BтBТYP
3	pos0	name3	322	1995-07-09	mail.name3@gmail.com	+380673334455	10	ME#b43wBgBиwm\$
4	pos2	name4	330	1997-10-21	mail.name4@gmail.com	+380674445566	8	9BaeH4%A00ExtnZE
5	pos2	name5	47	1991-10-15	mail.name5@gmail.com	+380675556677	11	zh
6	pos2	name6	74	1996-04-30	mail.name6@gmail.com	+380676667788	13	q(ig[-h-γB]o&γ-Я
7	pos2	name7	474	1996-04-30	mail.name7@gmail.com	+380677778899	8	iMyCleR ИjBhB
8	pos1	name8	200	1990-08-10	mail.name8@gmail.com	+380671112230	8	УJтf\$ДeBhaDэТ
9	pos1	name9	300	1995-09-12	mail.name9@gmail.com	+380672223341	9	"Ю+Q,От-YpVъyИM
10	pos0	name3	322	1995-07-09	mail.name31@gmail.com	+380673334452	10	;Cт rЕb
11	pos2	name4	330	1997-10-21	mail.name411@gmail.com	+380674445564	8	"B-SrEGwDQJton'
12	pos2	name5	47	1991-10-15	mail.name5@gmail.com	+380675556678	11	C Bv°CybX=vy@dЙ
13	pos2	name13	74	1996-04-30	mail.name13@gmail.com	+380676667786	13	ШьBbA
14	pos2	name14	474	1996-04-30	mail.name14@gmail.com	+380677778897	8	z E:BeAюB R1B+GB
15	pos2	name15	777	1997-04-30	mail.name15@gmail.com	+380677778899	8	z E:BeAюB R1B+GB

15 rows in set (0.00 sec)

3. Створимо тригер, який буде оновлювати ціну квитка в залежності від типу квитка та вартості самого сеансу.

```
DELIMITER $$
```

```
CREATE TRIGGER price_update AFTER
```

```
UPDATE ON lab12.seance FOR EACH ROW
```

```
BEGIN
```

```
UPDATE lab12.ticket SET ticket.price_ticket = (NEW.price)*1.1
```

```
WHERE ticket.seanceID = NEW.id_seance;
```

```
END$$
```

```
DELIMITER ;
```

id_ticket	seanceID	customerID	type	place	price_ticket	isReserved
1	1	1	standart	40	275	0
2	1	1	standart	41	275	0
3	2	2	standart	20	NULL	1
4	3	3	standart	23	NULL	1

4 rows in set (0.00 sec)

```
mysql> select * from Seance;
```

id_seance	date	time	hallID	repertoireID	price
1	2020-04-10	15:40:00	1	2	250
2	2020-04-10	18:00:00	1	3	150
3	2020-05-15	14:20:00	3	1	180
4	2020-06-01	13:45:00	4	4	200
5	2020-06-01	16:30:00	5	4	200
6	2020-06-10	15:40:00	1	5	210
7	2020-06-10	16:30:00	2	4	200
8	2020-07-02	14:20:00	3	1	180
9	2020-07-02	13:45:00	4	4	200
10	2020-07-15	16:30:00	2	1	180

10 rows in set (0.00 sec)

Висновок: на цій лабораторній роботі було розглянуто тригери, їх призначення, створення та використання.