Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут прикладної математики та фундаментальних наук

Кафедра прикладної математики

**Звіт**

про виконання лабораторної роботи №4

з курсу «Організація без даних та знань»

**Виконав :**   
студент групи ПМ-23   
Музика Денис

**Перевірив:**

кандидат наук   
 Пабирівський Віктор Володимирович

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата) (підпис викладача)

**Лабораторна робота №4. «Тригери СУБД Oracle»**

Завдання лабораторної роботи виконувати в призначеній предметній області. Обов’язково врахувати проблему “mutating table”. Змістовне призначення тригерів є наступним:

***Для всіх варіантів***:

1. У всіх таблицях створити поля UCR, DCR, ULC, DLC. Написати тригери які будуть заповнювати дані поля наступним чином: UCR – ім’я користувача, що створив даний запис; DCR – дата та час створення даного запису; ULC – ім’я користувача, що останнім змінив даний запис; DLC – дата та час останньої модифікації даного запису.

2. Створити сурогатний ключ для деякої таблиці, та написати тригер для обов’язкового заповнення цього поля послідовними значеннями.

3. Написати тригери для перевірки наступних обмежень цілісності:

**Варіант 12**. Клієнт не може отримувати нові кредити у випадках, визначених у предметній області. Автоматично при будь-якій операції (одержання, сплата чи погашення кредиту) повинен обчислюватися рейтинг клієнта.

**Індивідуальне завдання:**

Обчислення рейтингу клієнта

use CREDIT

GO -- CALCULATE RATING

CREATE TRIGGER UpdateRating

ON PaymentAction

AFTER INSERT

AS

BEGIN

UPDATE Rating

SET actualRating = ((C.amount + ISNULL(P.currentPayment, 0)) / (DATEDIFF(day, C.startDate, P.paymentDate) + 1)) - (C.amount / (C.term \* 365 - DATEDIFF(day, C.startDate, P.paymentDate)))

FROM inserted I

JOIN CreditContract C ON I.clientID = C.clientID

JOIN PaymentAction P ON I.id = P.id

WHERE Rating.clientID = I.clientID;

END

GO

**Завдання для усіх варіантів:**

use CREDIT

-- Додати поля UCR, DCR, ULC, DLC до всіх таблиць

ALTER TABLE Clients ADD UCR VARCHAR(100);

ALTER TABLE Clients ADD DCR DATETIME;

ALTER TABLE Clients ADD ULC VARCHAR(100);

ALTER TABLE Clients ADD DLC DATETIME;

ALTER TABLE Rating ADD UCR VARCHAR(100);

ALTER TABLE Rating ADD DCR DATETIME;

ALTER TABLE Rating ADD ULC VARCHAR(100);

ALTER TABLE Rating ADD DLC DATETIME;

ALTER TABLE CreditContract ADD UCR VARCHAR(100);

ALTER TABLE CreditContract ADD DCR DATETIME;

ALTER TABLE CreditContract ADD ULC VARCHAR(100);

ALTER TABLE CreditContract ADD DLC DATETIME;

ALTER TABLE CreditContracts ADD UCR VARCHAR(100);

ALTER TABLE CreditContracts ADD DCR DATETIME;

ALTER TABLE CreditContracts ADD ULC VARCHAR(100);

ALTER TABLE CreditContracts ADD DLC DATETIME;

ALTER TABLE PaymentAction ADD UCR VARCHAR(100);

ALTER TABLE PaymentAction ADD DCR DATETIME;

ALTER TABLE PaymentAction ADD ULC VARCHAR(100);

ALTER TABLE PaymentAction ADD DLC DATETIME;

ALTER TABLE PaymentSchedule ADD UCR VARCHAR(100);

ALTER TABLE PaymentSchedule ADD DCR DATETIME;

ALTER TABLE PaymentSchedule ADD ULC VARCHAR(100);

ALTER TABLE PaymentSchedule ADD DLC DATETIME;

GO

CREATE TRIGGER tr\_Clients\_Insert

ON Clients

AFTER INSERT

AS

BEGIN

UPDATE Clients

SET UCR = SUSER\_SNAME(),

DCR = GETDATE(),

ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO

CREATE TRIGGER tr\_Clients\_Update

ON Clients

AFTER UPDATE

AS

BEGIN

UPDATE Clients

SET ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO ------------------------------------------------

CREATE TRIGGER tr\_Rating\_Insert

ON Rating

AFTER INSERT

AS

BEGIN

UPDATE Rating

SET UCR = SUSER\_SNAME(),

DCR = GETDATE(),

ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO

CREATE TRIGGER tr\_Rating\_Update

ON Rating

AFTER UPDATE

AS

BEGIN

UPDATE Rating

SET ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO ------------------------------------------------

CREATE TRIGGER tr\_CreditContract\_Insert

ON CreditContract

AFTER INSERT

AS

BEGIN

UPDATE CreditContract

SET UCR = SUSER\_SNAME(),

DCR = GETDATE(),

ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO

CREATE TRIGGER tr\_CreditContract\_Update

ON CreditContract

AFTER UPDATE

AS

BEGIN

UPDATE CreditContract

SET ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO ------------------------------------------------

CREATE TRIGGER tr\_CreditContracts\_Insert

ON CreditContracts

AFTER INSERT

AS

BEGIN

UPDATE CreditContracts

SET UCR = SUSER\_SNAME(),

DCR = GETDATE(),

ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO

CREATE TRIGGER tr\_CreditContracts\_Update

ON CreditContracts

AFTER UPDATE

AS

BEGIN

UPDATE CreditContracts

SET ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO ------------------------------------------------

CREATE TRIGGER tr\_PaymentAction\_Insert

ON PaymentAction

AFTER INSERT

AS

BEGIN

UPDATE PaymentAction

SET UCR = SUSER\_SNAME(),

DCR = GETDATE(),

ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO

CREATE TRIGGER tr\_PaymentAction\_Update

ON PaymentAction

AFTER UPDATE

AS

BEGIN

UPDATE PaymentAction

SET ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO ------------------------------------------------

CREATE TRIGGER tr\_PaymentSchedule\_Insert

ON PaymentSchedule

AFTER INSERT

AS

BEGIN

UPDATE PaymentSchedule

SET UCR = SUSER\_SNAME(),

DCR = GETDATE(),

ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO

CREATE TRIGGER tr\_PaymentSchedule\_Update

ON PaymentSchedule

AFTER UPDATE

AS

BEGIN

UPDATE PaymentSchedule

SET ULC = SUSER\_SNAME(),

DLC = GETDATE()

WHERE id IN (SELECT id FROM inserted);

END;

GO ------------------------------------------------