

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 баз даних та знань»

Виконав:

студент групи ПМ-22

Янів П.В.

Перевірив:

Пабірівський В.В.

Львів-2024

Варіант №11 “Оплата праці”

На підприємстві працюють працівники, кожен з яких належить до певного структурного підрозділу. Кожен структурний підрозділ має свою назву, код, керівника. Кожен працівник працює на певній посаді та характеризується табельним номером, прізвищем, ім'ям, по-батькові, паспортними даними, датою та місцем народження, домашньою адресою.

На підприємстві встановлені такі форми оплати праці

- основна заробітна плата у вигляді погодинної оплати за кожну відпрацьовану годину, причому для кожної посади існує свій тариф за годину,
- додаткова заробітна плата – оплата за роботу у понад встановлені норми, яка полягає в подвійній оплаті за кожну додатково відпрацьовану годину та потрійній – за кожну відпрацьовану годину у вихідний (субота, неділя) та святковий день.

Працівник кожного року має відпустку. Система повинна зберігати дані про всі відпустки працівників. Одночасно перебувати в відпустці можуть не більше 15% працівників одного відділу. Оплата відпусток проводиться так: $S = M / (365 - C) * N$, де S – сума відпускних, M – сумарний заробіток працівника за останні 12 місяців, 365 – кількість днів у році, C – число святкових днів, N – тривалість відпустки в календарних днях.

Система повинна вести облік робочого часу (фактично відпрацьовані години працівником за кожен день), нараховувати місячну зарплату всім працівникам, розраховувати для кожного працівника заробітну плату до видачі на руки (заробітна плата до видачі на руки – це нарахована заробітна плата мінус соціальний податок 20%).

Система повинна вести облік виплати зарплати працівника.

В системі повинні діяти такі додаткові обмеження: працівник не може працювати в декількох відділах та на декількох посадах одночасно; основна зарплата працівника не може бути меншою від встановленої мінімальної зарплати, працівнику забороняється нараховувати зарплату після його звільнення; виплата зарплати працівникам повинна здійснюватись не пізніше 10 числа наступного місяця, який слідує за відпрацьованим, при порушенні цього правила працівнику додається до зарплати за поточний місяць пеня в розмірі 0,1% від суми заборгованості за кожен прострочений день.

- розрахункову відомість заробітної плати для відділу,
- відомість на отримання заробітної плати для відділу,
- відомість про роботу працівника впродовж певного періоду,
- відомість про борг підприємства перед працівниками.

Завдання:

Лабораторна робота № 4. “Тригери в 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 11. Працівник не може працювати в декількох відділах та на декількох посадах одночасно. Основна зарплата працівника не може бути меншою від встановленої мінімальної зарплати, працівнику забороняється нараховувати зарплату після його звільнення; виплата зарплати працівникам повинна здійснюватись не пізніше 10 числа

наступного місяця, який слідує за відпрацьованим, при порушенні цього правила працівнику додається до зарплати за поточний місяць пеня в розмірі 0,1% від суми заборгованості за кожен прострочений день.

Хід роботи:

Для виконання першої частини лабораторної роботи, додав до кожної таблиці поля: 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.

При створенні цих полів, задав для полів UCR, DCR значення DEFAULT, Тобто при створенні запису у таблицю, ці поля записуються автоматично.

Для кожної таблиці написав по два тригери 'table_name'+UpdateUDCR та 'table_name'+UpdateUDLC ,('table_name'-назва даної таблиці)

Призначення тригерів:

1'table_name'+UpdateUDCR : цей тригер забороняє змінювати поля UCR та DCR, які записуються автоматично під час запису, ця заборона потрібна для того щоб забезпечити коректність інформації про створений запис та захистити її, тобто не можна змінювати інформацію про створений запис.

Тригер написаний для INSTEAD OF UPDATE та перевіряє які поля оновлюються, якщо користувач намагається змінити поля UCR і DCR, виведеться відповідне повідомлення, якщо змінюються інші поля в таблиці то спрацює наступний тригер:

2'table_name'+UpdateUDLC : цей тригер оновлює поля ULC та DLC, при зміні будь-якого поля у записі. Записує ім'я користувача що змінює дані та дату зміни.

Для виконання наступної частини лабораторної роботи, написав тригер для таблиці EmployeeINFO, назва тригера EmployeeLimit

Цей тригер спрацьовує при записі у дану таблицю. Він бере ім'я та паспортні дані, працівника якого записують у таблицю, і порівнює з усіма наявними записами у таблиці, якщо паспортні дані та ім'я співпадають а посада чи код структурного підрозділу ні, то виводиться відповідне повідомлення. В протилежному випадку дані запишуться у таблицю.

Для таблиці SalaryGive створив тригер з назвою MinSalaryGive

Цей тригер також спрацьовує при записі у таблицю. Тригер виконує дві функції:

- 1) Тригер бере мінімальну зарплату з таблиці SalaryPerHours для посади на якій працює працівник для якого записується зарплата, і порівнює

мінімальну зарплату з зарплатою, яку нараховує процедура. Якщо мінімальна зарплата є більшою за зарплату яку нараховували, то у таблицю запишеться значення мінімальної зарплати для цього працівника. В протилежному випадку, запишеться нарахована зарплата.

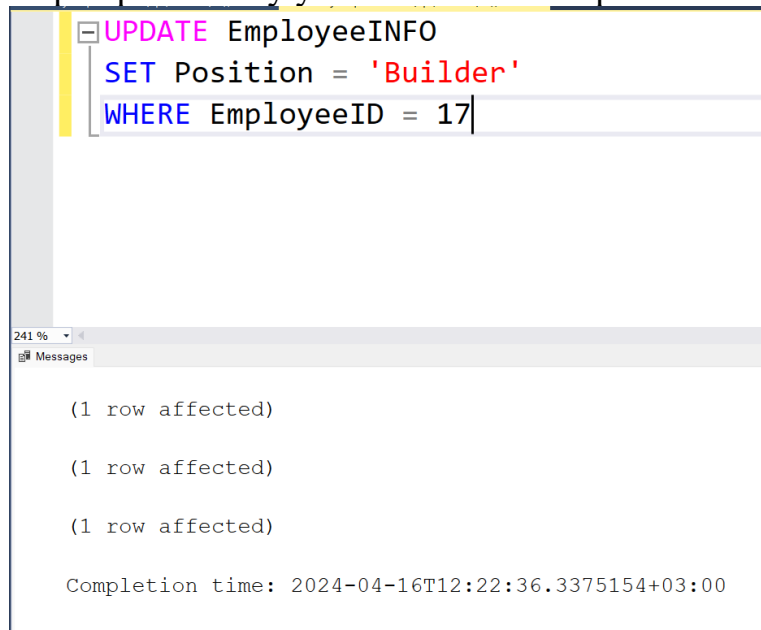
- 2) Друга частина роботи тригера - це нарахування пені для працівника. Якщо нарахування зарплати проходить пізніше 10 числа наступного місяця, то йому до зарплати додається 0.1% від зарплати. Тригер бере місяць за який нараховується зарплата та збільшує місяць на наступний і додає ще день, так щоб вийшло 10 число наступного місяця після якого надається зарплата. За допомогою функції DATEDIF() порівнює цю створену дату з датою запису. Функція повертає ціле число яке відповідає різниці у днях між двома датами. Якщо повернуте значення є менше нуля то значить, що зарплата нараховується ще до 10 числа, якщо більше 0 то запускається цикл, який нараховує до зарплати пеню.

В кінці роботи тригера у таблицю записуються наступні значення:

- 1) Мінімальна зарплата для працівника, якщо нарахована зарплата є меншою за мінімальну та значення записуються не пізніше 10 числа.
- 2) Нарахована зарплата, якщо вона є більшою за мінімальну та значення записуються не пізніше 10 числа.
- 3) Зарплата з врахуванням пені, якщо вона є більшою за мінімальну та значення записуються пізніше 10 числа.

Результат роботи тригерів:

Перевірка запису у поля DLC ULC при оновленні даних:



```
UPDATE EmployeeINFO
SET Position = 'Builder'
WHERE EmployeeID = 17
```

241 %

Messages

(1 row affected)

(1 row affected)

(1 row affected)

Completion time: 2024-04-16T12:22:36.3375154+03:00

Запит успішно виконано і при перегляді таблиці можна переконатись що поле змінено і поля DLC ULC є заповненні коректними записами:

SQL Query Editor - [P:\MS\SQL\MS\SSMS\11\sql>Admin>DB> - Microsoft SQL Server Enterprise 11.0.5000.11

```

SELECT TOP (1000) [Position]
, [EmployeeID]
, [Pasport]
, [HomeAddress]
, [PlaceBirth]
, [DateBirth]
, [DepartmentID]
, [Name]

```

Results Messages

Position	EmployeeID	Pasport	HomeAddress	PlaceBirth	DateBirth	DepartmentID	Name	Surname	FatherName	UCR	DCR	ULC	DLC
Builder	2	238754	Bandery 01	Sambir	01.01.1996	2	Oks	Underwood	Arthur	dbo	2024-04-11 17:50:00	NULL	NULL
Electrician	3	119944	Shevchenko 55	Kyiv	09.07.1989	2	Ricky	Lowe	Shaun	dbo	2024-04-11 17:50:00	NULL	NULL
Builder	4	259789	Sht 05	Lviv	29.07.2001	1	Valma	Stafford	Robert	dbo	2024-04-11 17:50:00	NULL	NULL
Electrician	5	567895	Rollins 34	Kharkiv	18.03.1996	3	Alex	Rollins	Deacon	dbo	2024-04-11 17:50:00	NULL	NULL
Electrician	6	123456	Zelena 45	Sambir	30.07.2002	4	Abdullah	Wiley	Alexis	dbo	2024-04-11 17:50:00	NULL	NULL
Builder	7	678757	Rollins 21	Lviv	21.11.1979	3	Arthur	Hill	Mohammad	dbo	2024-04-11 17:50:00	NULL	NULL
Builder	8	232123	Bandery 33	Dobromyl	12.12.2000	4	Nathan	Singh	Deacon	dbo	2024-04-11 17:50:00	NULL	NULL
Accountant	9	345632	Sht 90	Lviv	29.01.2000	1	Deacon	Pack	Oks	dbo	2024-04-11 17:50:00	NULL	NULL
Accountant	10	678904	Zelena 89	Dnipro	01.03.2002	2	Markus	Mccall	Abdullah	dbo	2024-04-11 17:50:00	NULL	NULL
Accountant	11	997885	Rollins 22	Sambir	23.02.1999	3	Shaun	Pollard	Jared	dbo	2024-04-11 17:50:00	NULL	NULL
Accountant	12	231235	Bandery 23	Kyiv	02.04.2001	4	Mohammad	Simpson	Jared	dbo	2024-04-11 17:50:00	NULL	NULL
Architect	13	888435	Zelena 43	Lviv	09.07.2000	1	Brett	Whisker	Robert	dbo	2024-04-11 17:50:00	NULL	NULL
Architect	14	675357	Sht 66	Dnipro	21.09.1976	2	Alexis	Camacho	Nathan	dbo	2024-04-11 17:50:00	NULL	NULL
Architect	15	567384	Rollins 21	Lviv	12.11.2002	3	Deacon	Lara	Jared	dbo	2024-04-11 17:50:00	NULL	NULL
Architect	16	754546	Zelena 76	Lviv	30.12.1999	4	Jared	Pittman	Ricky	dbo	2024-04-11 17:50:00	NULL	NULL
Builder	17	5533...	Zelena 02	Lviv	08.03.2000	2	Rey	Smith	Alex	dbo	2024-04-16 08:59:00	dbo	2024-04-16 12:23:00

Спроба змінення полів DCR UCR:

SQL Query Editor - [P:\MS\SQL\MS\SSMS\11\sql>Admin>DB> - Microsoft SQL Server Enterprise 11.0.5000.11

```

UPDATE EmployeeINFO
SET DCR = GETDATE(), UCR = CURRENT_USER
WHERE EmployeeID = 17

```

Results Messages

Msg
1 You can't update this column

Виводиться відповідне повідомлення і значення залишаються незмінними.

Перевірка тригера EmployeeLimit:

Я намагаюся записати у таблицю інформацію про працівника який уже є у таблиці, але я записую його під іншим табельним номером та на іншу посаду у інший структурний підрозділ:

SQL Query Editor - [P:\MS\SQL\MS\SSMS\11\sql>Admin>DB> - Microsoft SQL Server Enterprise 11.0.5000.11

```

INSERT [dbo].[EmployeeINFO] ([Position], [EmployeeID], [Pasport], [HomeAddress], [PlaceBirth], [DateBirth], [DepartmentID], [Name], [Surname], [FatherName])
VALUES (N'Builder', 17, 754546, N'Zelena 76', N'Lviv', N'30.12.1999', 4, N'Jared', N'Pittman', N'Ricky')

```

Results Messages

Msg
1 An employee cannot work at several departments at the same time

Виводиться відповідне повідомлення і дані не записуються.

Але якщо я зміню паспортні дані та ім'я то це очевидно буде зовсім інша людина, дані про яку запишуться у таблицю:

```

INSERT [dbo].[EmployeeINFO] ([Position], [EmployeeID], [Pasport], [HomeAddress], [PlaceBirth], [DateBirth], [DepartmentID], [Name], [Surname], [FatherName])
VALUES (N'Builder', 17, 7543446, N'Zelena 76', N'Lviv', N'30.12.1999', 4, N'Rey', N'Pittman', N'Ricky')

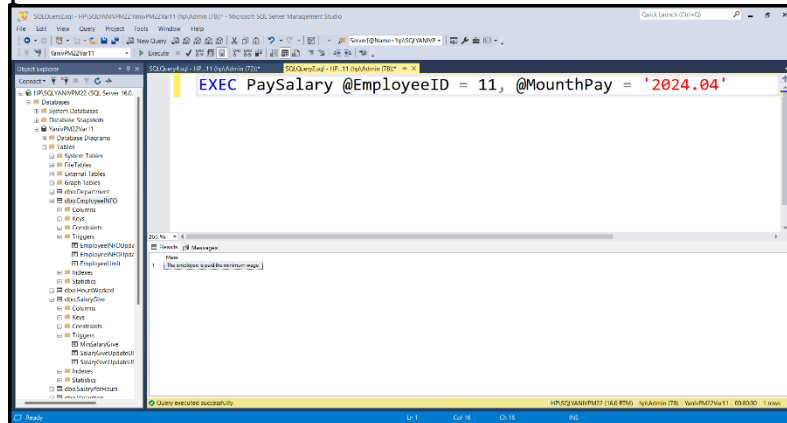
```

16	Architect	16	754546	Zelena 76	Lviv	30.12.1999	4	Jared	Pittman	Ricky	dbo	2024-04-11 17:50:00	NULL	NULL
17	Builder	17	7534...	Zelena 76	Lviv	30.12.1999	4	Rey	Pittman	Ricky	dbo	2024-04-16 12:36:00	NULL	NULL

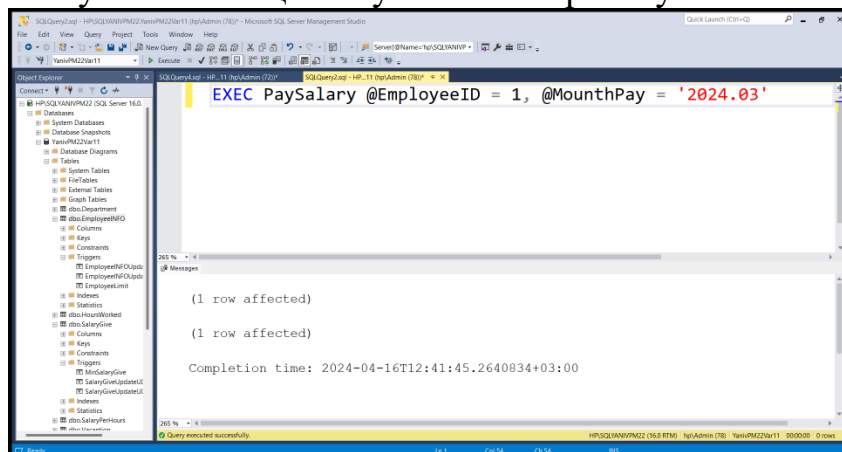
Дані записані.

Перевірка тригера MinSalaryGive:

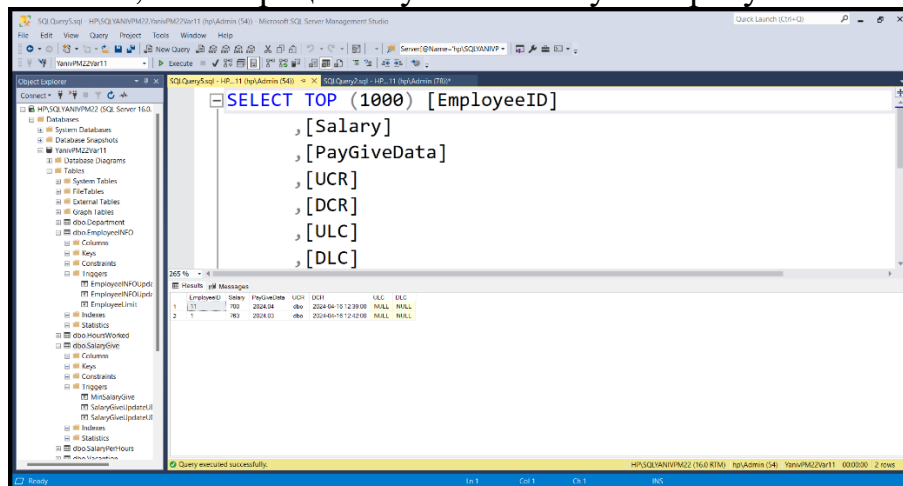
Для перевірки використаю процедуру написану у попередній лабораторній роботі:



Працівнику нарахована мінімальна зарплата, тому що за квітень він заробив менше мінімальної зарплати і нарахування зарплати проводиться до 10 числа наступного місяця тому пеня не нараховується



Дані записані. Для працівника з номером 1 за березень, але зарплата нараховується за березень, а дата коли записується зарплата є 16.04 що більше за 10.04, отже працівнику повинна була нарахуватися пеня:



Дані записано правильна для працівника з номером 11 нарахована мінімальна зарплата без пені, а для працівника з номером 1 нарахована пеня.

Коди для тригерів можна знайти за посиланням:

<https://github.com/PetroYaniv/DBSQLYanivPetroPM22/blob/main/Lab4/lab4.sql>

GitHub репозиторій:

<https://github.com/PetroYaniv/DBSQLYanivPetroPM22.git>

Висновок: Для виконання лабораторної роботи №4 створив тригери для кожної таблиці, згідно завданням.