

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Інститут прикладної математики та фундаментальних наук
Кафедра прикладної математики

Звіт

про виконання лабораторної роботи №5
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 № 5. “Адміністрування БД”

1) Створити 3-4 різнотипних користувача БД та створити текстовий опис типових задач, що виникають перед користувачами (відповідно до обраної предметної області).

2) Надати створеним користувачам привілеї відповідно до їх типових задач.

3) Створити 2-3 типові ролі для користувачів БД та створити текстовий опис типових задач, що асоціюються з цими ролями.

4) Надати необхідні привілеї створеним ролям.

5) Призначити користувачам ролі.

6) Відкликати у користувача привілеї, що також призначений через роль. Переконавшись у збереженні здатності користувача користуватися привілеєм через роль.

7) Відкликати роль у користувача. Переконавшись у збереженні у користувача привілеїв, що були призначені йому персонально (не через роль), та відсутності у нього привілеїв, що були йому надані лише через роль.

8) Видалити роль. Видалити користувача.

Хід роботи:

Для виконання пункту 1 з завдання до лабораторної роботи №5, створив 4 користувачів для бази даних, створеної у попередніх лабораторних роботах. Для створення використав команду CREATE USER.

Призначення користувачів:

Користувач DB_ADMIN – буде мати доступ до усіх таблиць та до усієї бази даних, виконуватиме роль адміністратора та буде мати можливість вносити будь-які зміни у таблиці та функціонал бази.

Користувач DB_SimpleUSER – матиме змогу лише переглядати вміст таблиць, без права вносити будь-які зміни у функціонал бази даних.

Користувач DB_Accountant – буде мати змогу робити записи у таблицю про видачу зарплати для робітників та викликати відповідні процедури.

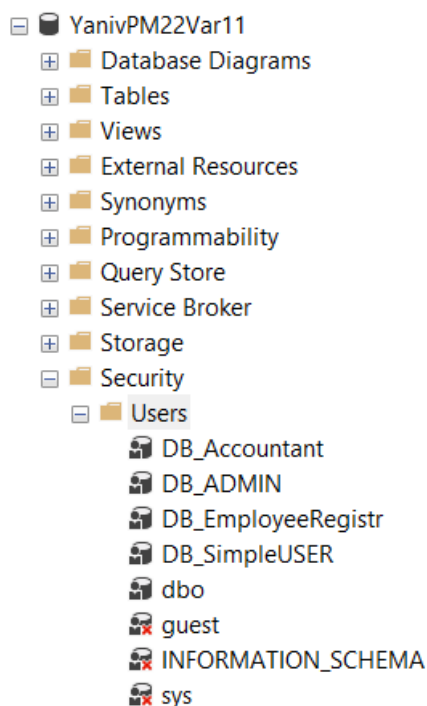
Користувач DB_EmployeeRegistr – буде мати доступ до змін у таблиці з інформацією про усіх працівників.

Для користувачів DB_ADMIN, DB_Accountant та DB_EmployeeRegistr створив логін та паролі:

DB_ADMIN пароль – ADMIN логін - DB_ADMIN

DB_Accountant пароль – Accountant логін - DB_Accountant

DB_EmployeeRegistr пароль – EmployeeRegistr логін - DB_EmployeeRegistr



Для виконання пункту 2 використав команду GRANT та надав створеним користувачам відповідні привілеї:

Користувачу DB_EmployeeRegistr надав доступ до змін у таблиці EmployeeINFO, наступною командою

```
GRANT INSERT, SELECT, DELETE, UPDATE ON OBJECT::EmployeeINFO TO DB_EmployeeRegistr
```

Користувачу DB_Accountant надав доступ до змін у таблиці SalaryGive та дав можливість викликати процедури для нарахування зарплати, наступною командою

```
GRANT INSERT, SELECT, DELETE, UPDATE ON OBJECT::SalaryGive TO DB_Accountant  
GRANT EXECUTE ON OBJECT::PaySalary TO DB_Accountant  
GRANT EXECUTE ON OBJECT::PaySalaryForAllEmployee TO DB_Accountant
```

Користувачу DB_SimpleUSER надав доступ до перегляду вмісту усіх таблиць, наступною командою

```
GRANT SELECT TO DB_SimpleUSER
```

Користувачу DB_ADMIN надав повний доступ до бази даних, наступною командою

```
GRANT CONTROL ON DATABASE::YanivPM22Var11 TO DB_ADMIN
```

Далі створив 2 ролі користувачам для виконання пункту 3. Використав команду CREATE ROLE:

Опис та призначення ролей:

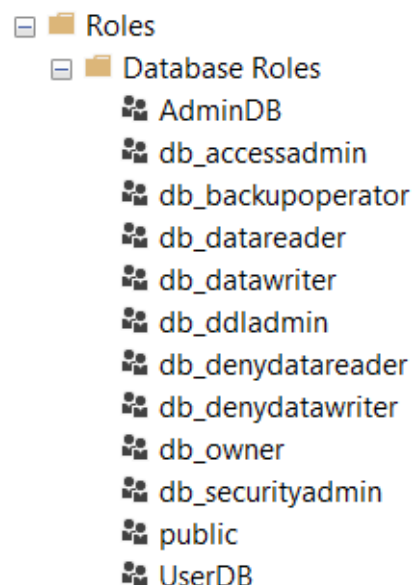
Роль UserDB – призначена для користувачів які користуються базою даних для перегляду інформації у таблицях чи внесені змін у таблиці.

Роль AdminDB – призначена для користувачів які мають можливість змінювати функціонал бази даних.

Створення ролей:

```
CREATE ROLE UserDB
```

```
CREATE ROLE AdminDB
```



Надав ролям, відповідні до їх типових задач, привілеї:

Користувачу UserDB надав дозвіл на перегляд, оновлення та видалення вмісту таблиць:

```
GRANT SELECT, UPDATE, DELETE TO UserDB
```

Користувачу AdminDB надав усі дозволи для бази даних:

```
GRANT CONTROL TO AdminDB
```

Призначив користувачам ролі:

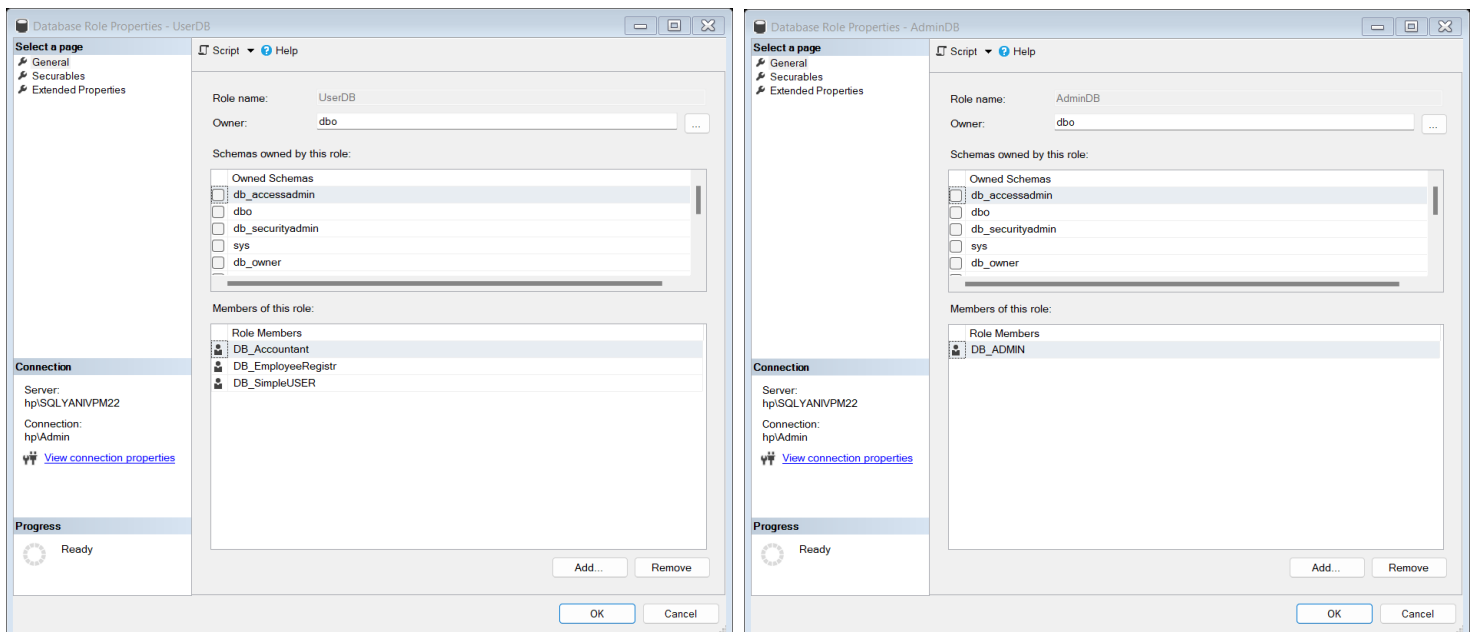
Користувачу DB_ADMIN призначив роль AdminDB командою:

```
ALTER ROLE AdminDB ADD MEMBER DB_ADMIN
```

Користувачам DB_Accountant, DB_SimpleUSER, DB_EmployeeRegistr надав роль UserDB командою:

```
ALTER ROLE UserDB ADD MEMBER DB_Accountant  
ALTER ROLE UserDB ADD MEMBER DB_EmployeeRegistr  
ALTER ROLE UserDB ADD MEMBER DB_SimpleUSER
```

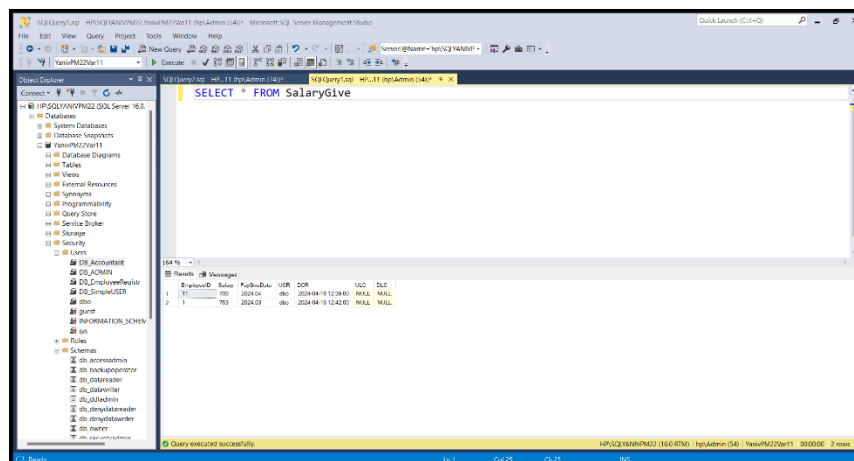
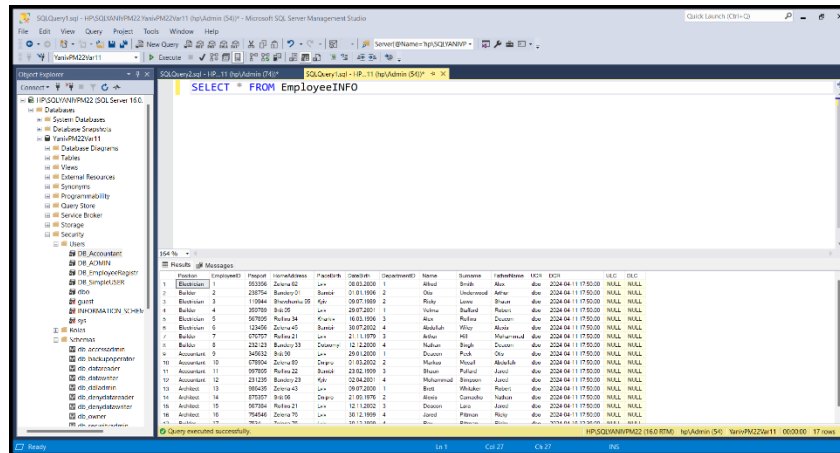
Перевірів чи користувачам були надані ролі:



Для виконання наступних пунктів завдання, буду використовувати користувача DB_Accountant

```
EXECUTE AS USER = 'DB_Accountant'  
SELECT CURRENT_USER
```

Перевірим здатність користувача користуватися запитом SELECT, який був наданий йому напряму та через роль:



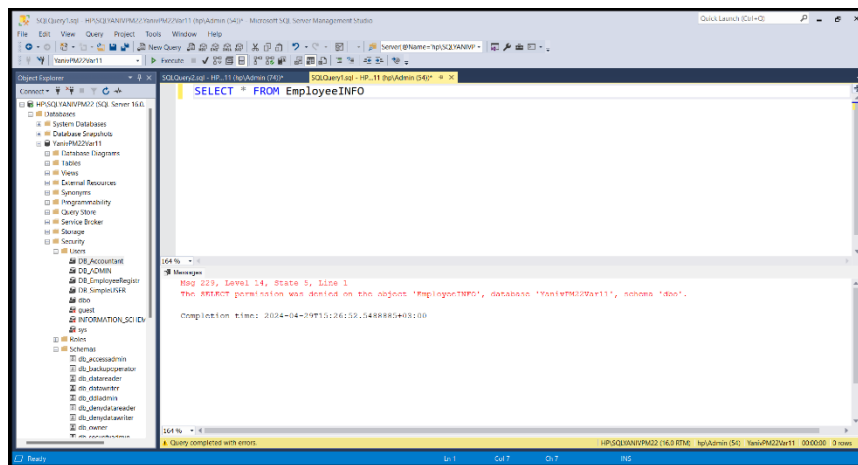
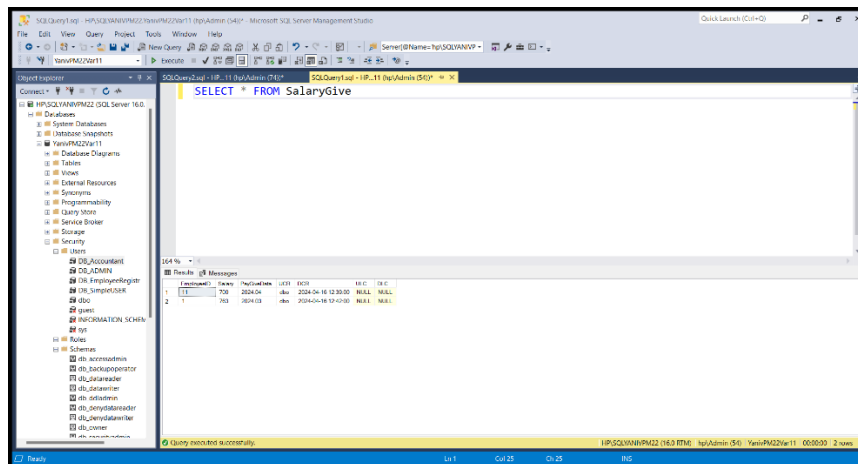
Як можна побачити SELECT-запит працює для таблиці SalaryGive для якої користувачу був наданий дозвіл, також цей запит працює для будь-якої іншої таблиці, тому що такий дозвіл надає роль.

За допомогою команди:

```
REVOKE SELECT ON OBJECT::EmployeeInfo TO DB_Accountant
```

Заберу у користувача можливість використовувати SELECT-запит для таблиці EmployeeInfo:

Перевірка:



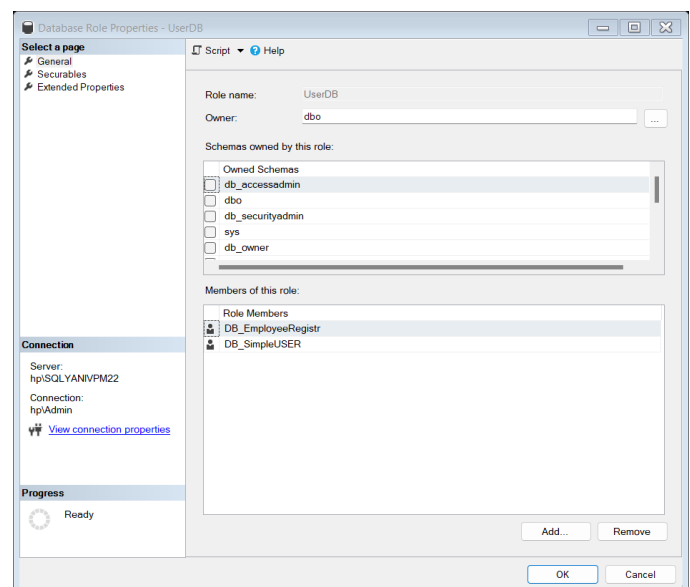
Тепер можна побачити що SELECT-запит працює для таблиці SalaryGive, а для таблиці EmployeeINFO не працює.

Це значить що привілей який був наданий користувачу персонально зберігся, а той що був наданий через роль тепер не належить користувачу.

Далі заберу роль, яка була надана для цього користувача, наступною командою:

```
ALTER ROLE UserDB  
DROP MEMBER DB_Accountant
```

Тепер перевірю чи виконалася команда:



Команда виконалася успішно та у користувача пропала роль.

Тепер командами DROP видаляю роль UserDB та користувача DB_Accountant:

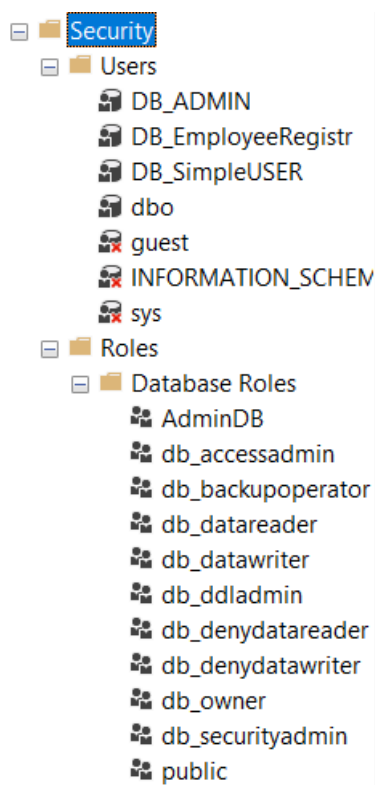
Перед видаленням ролі потрібно щоб у користувачів не було цієї ролі, тож я забрав цю роль у усіх користувачів та видалив її:

```
DROP ROLE UserDB
```

Та видалив користувача:

```
DROP USER DB_Accountant
```

Тепер у вікні Object Explorer можна побачити що роль UserDB та користувача DB_Accountant відсутні:



GitHub репозиторій:

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

Висновок: В процесі виконання лабораторної роботи №5, створив 4 користувачів для бази даних, яким надав дозволи відповідно для їх типових завдань, також створив 2 ролі для яких також надав відповідні дозволи. Виконав надання ролей користувачам, відкликав надані дозволи, відкликав у користувача ролі, видалив роль та користувача.