# Лабораторна робота 12. Користувачі, права і ролі

**Мета:** Визначення прав доступу до об'єктів БД за допомогою утиліт управління сервером і SQL.

Зміст

[Лабораторна робота 12. Користувачі, права і ролі 1](#_Toc400928511)

[Вправи та теоретична частина для MsSQL 1](#_Toc400928512)

[Доступ до MsSQL 1](#_Toc400928513)

[12.1.1. Створення ідентифікатора користувача (логіну) для сервера БД 2](#_Toc400928514)

[12.1.2. Включення нового логіну до серверної ролі sysadmin 3](#_Toc400928515)

[12.2.1. Створення користувача Alex БД IS01\_00\_TS 5](#_Toc400928516)

[12.2.2. Створення користувацької ролі БД 6](#_Toc400928517)

[12.2.3. Надання користувацькій ролі БД прав на роботу з певними таблицями БД (DELETE, INSERT, SELECT, UPDATE) 7](#_Toc400928518)

[12.2.4. Надання цій ролі прав на роботу з певною таблицею БД (DELETE, INSERT, SELECT, UPDATE) засобами Transact-SQL 9](#_Toc400928519)

[Контрольні питання 10](#_Toc400928520)

[Теоретична частина для Oracle 11](#_Toc400928521)

[Рекомендації по виконанню вправи 12.1. 11](#_Toc400928522)

[CREATE USER 11](#_Toc400928523)

[CREATE ROLE 11](#_Toc400928524)

[GRANT 11](#_Toc400928525)

[REVOKE 11](#_Toc400928526)

[Приклад можливих дій для виконання вправи 12.2. 12](#_Toc400928527)

[Рекомендації по виконанню роботи для MySQL 17](#_Toc400928528)

[Контрольні питання 17](#_Toc400928529)

# Вправи та теоретична частина для MsSQL

Див. [курс лекцій, тема 6](http://web.asu.ntu-kpi.kiev.ua/moodle/mod/resource/view.php?id=93), розділи Доступ до MsSQL, Доступ до Oracle

## Доступ до MsSQL

**Вправа 12.1**. Управління доступом на рівні сервера за допомогою логіна і серверної ролі

12.1.1. Створити логін для сервера БД.

12.1.2. Включити цей логін до серверної ролі sysadmin і переконатись у наданні відповідних прав.

12.1.3. Виключити цей логін з серверної ролі sysadmin і переконатись у зникненні відповідних прав.

Виконаємо ці дії на БД Технічної служби.

### 12.1.1. Створення ідентифікатора користувача (логіну) для сервера БД

Запускаємо Microsoft SQL Server Management Studio (далі – MS) і заходимо під логіном sa. Контекстне меню на папці Security\Logins надає можливість створити логін:

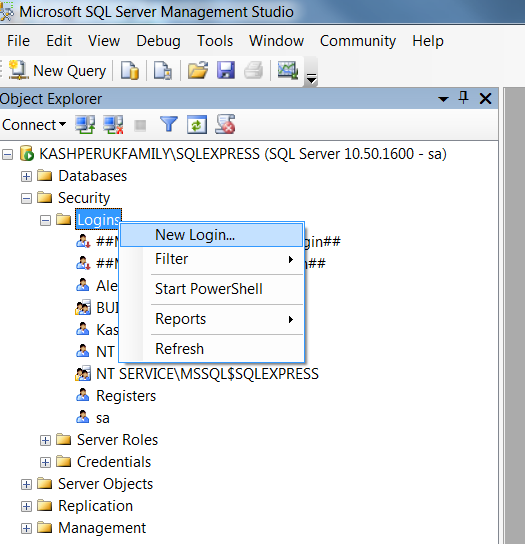


Рис.12.1. Запуск створення нового логіну

Створимо логін Alex з паролем alex . Для цього заповнимо форму логіну, вказавши ім’я, пароль, SQL Server Authentification:

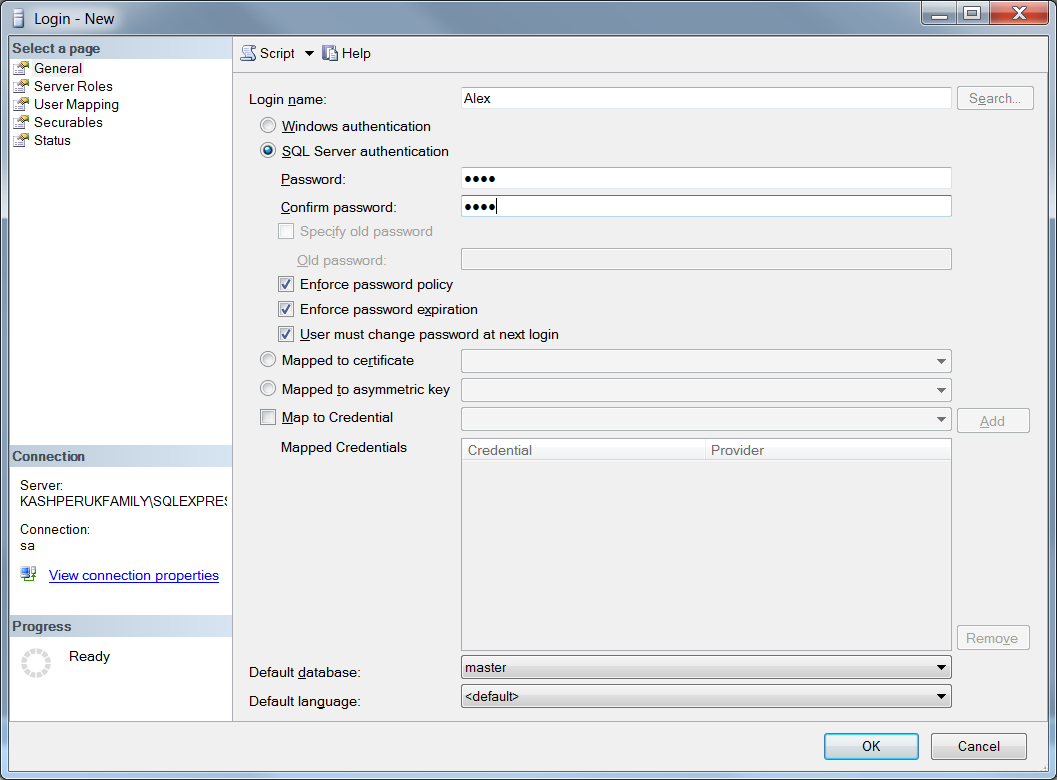


Рис.12.2. Форма властивостей логіну для сервера БД

Новий логін не має прав на користувацькі об’єкти. Пересвідчимся в цьому, змінивши поточний логін на Alex (меню File\Disconnect Object Explorer, File\Connect Object Explorer). Переглянути зміст користувацьких таблиць створеної в попередній роботі БД IS01\_00\_TS неможливо.

### 12.1.2. Включення нового логіну до серверної ролі sysadmin

Включивши логін до серверної ролі sysadmin, ми «одним махом» надаємо йому всі права. Цей спосіб є найпростішим, і водночас найменш гнучким, він діє по принципу «все або нічого».

Виконаємо конекцію як sa. Відчинивши контекстне меню на серверній ролі sysadmin і обравши пункт Properties, отримаємо каскад з трьох форм, які дозволяють додати логін Alex у члени ролі. Після встановлення галочки для Alex усі три форми зачиняються кнопками Ok.

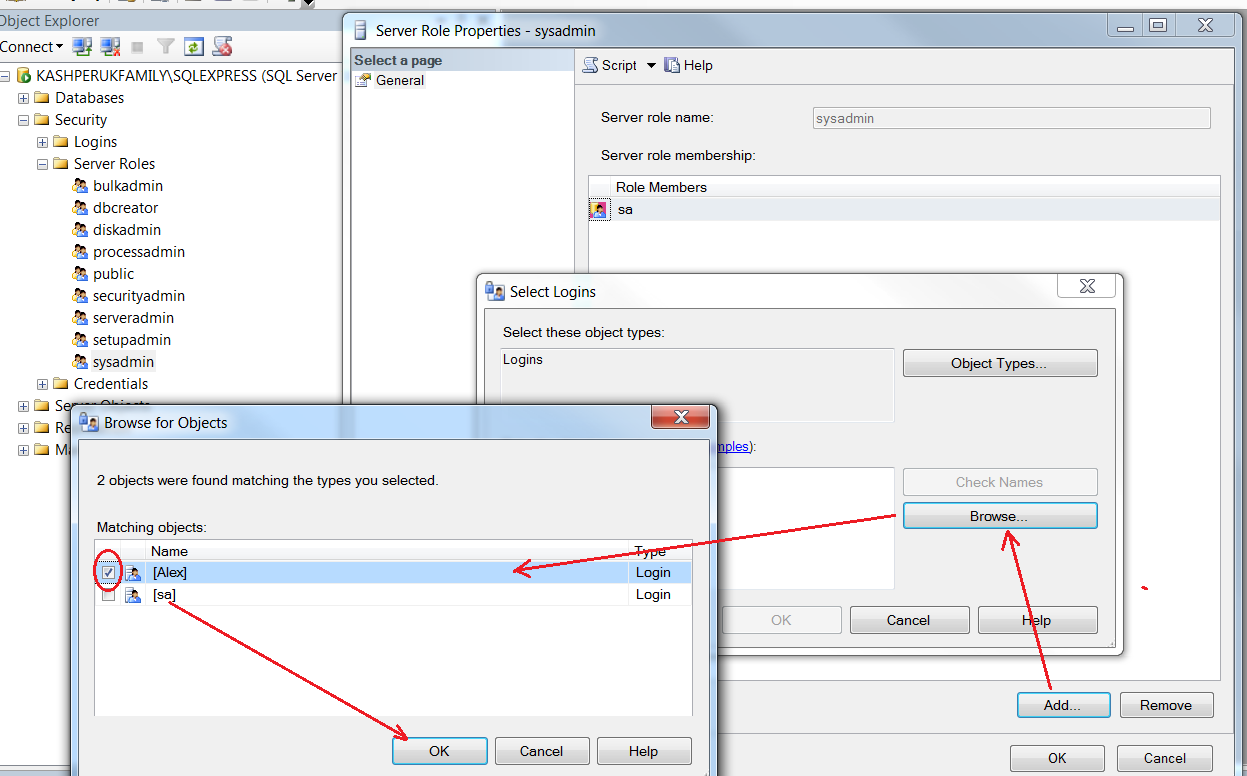


Рис.12.3. Додавання логіну Alex у члени ролі sysadmin

Після цього, виконавши конекцію як Alex, переконаймося в наявності доступу до таблиць БД IS01\_00\_TS.

Оскільки цей спосіб надання прав є недосконалим, виключимо Alex зі членів ролі sysadmin. Для цього виконуємо конекцію як sa, відчиняємо форму Properties і натискаємо кнопки Remove, Ok.

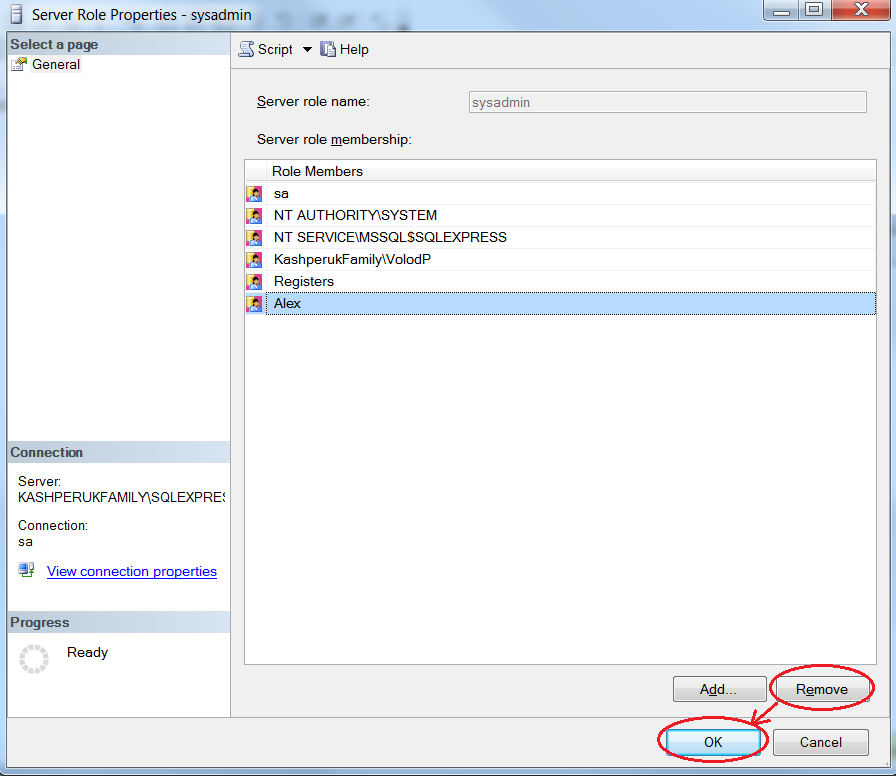


Рис.12.4. Вилучення логіну Alex зі членів ролі sysadmin

**Вправа 12.2**. Управління доступом на рівні БД за допомогою користувацької ролі БД

12.2.1. Створити користувача БД.

12.2.2. Створити користувацьку роль БД і включити користувача у цю роль.

12.2.3. Надати цій ролі прав на роботу з певними таблицями БД (DELETE, INSERT, SELECT, UPDATE) засобами MS. Перевірити, що у користувача є ці права.

12.2.4. Вилучити у ролі права на роботу з таблицями БД засобами Transact SQL. Пересвідчитись, що вилучені права зникли в інтерфейсі MS, і що вони зникли у користувача БД.

12.2.5. Надати цій ролі права на роботу з певною таблицею БД (DELETE, INSERT, SELECT, UPDATE) засобами Transact-SQL. Перевірити, що відповідні права з’явились у користувача БД. Зокрема, продемонструвати у звіті належність таблиці до користувацької предметної області.

Виконаємо ці дії на БД Технічної служби.

### 12.2.1. Створення користувача Alex БД IS01\_00\_TS

Користувач БД створюється на основі логіну. Відчинимо папки Object Explorer: БД IS01\_00\_TS\Security\Users, на останній відчинимо контекстне меню і оберемо New User. Каскад з трьох форм, що відчиняються після цього, зображений на наступному малюнку, дозволяє створити користувача БД на основі логіну. Після обрання логіну форми зачиняються кнопками Ok.

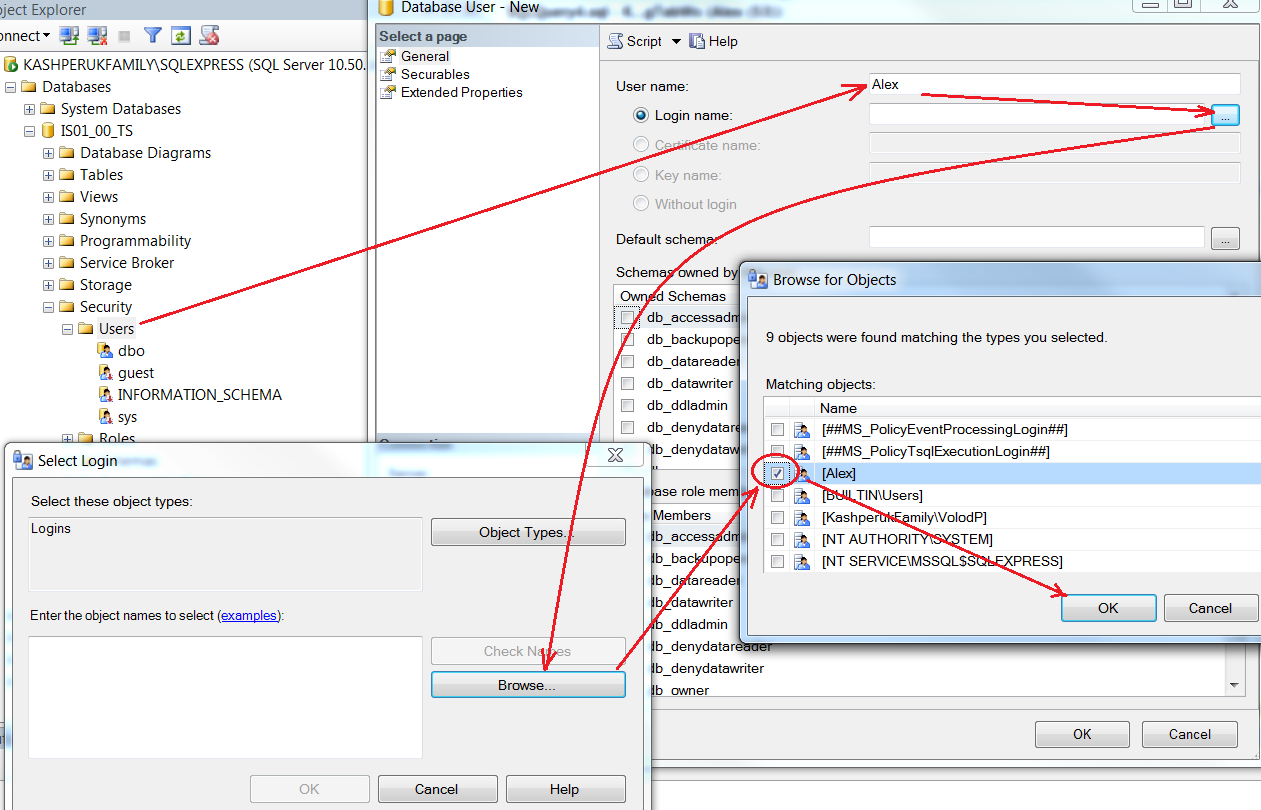


Рис.12.5. Створення користувача БД

### 12.2.2. Створення користувацької ролі БД

Крім десяти вбудованих ролей БД, користувач може додавати користувацькі ролі БД. Створимо таку роль з іменем CompManager (спеціаліст по номенклатурі деталей автомашин), включивши в неї користувача БД Alex, і надамо необхідні права ролі замість користувача.

Створення користувацької ролі БД так само починається з пункту контекстного меню New Database Role на папці Database Roles, затим відчиняється каскад трьох форм, у першій з них зверху вказуємо ім’я ролі; після встановлення галочки біля користувача БД Alex форми зачиняються кнопками Ok.

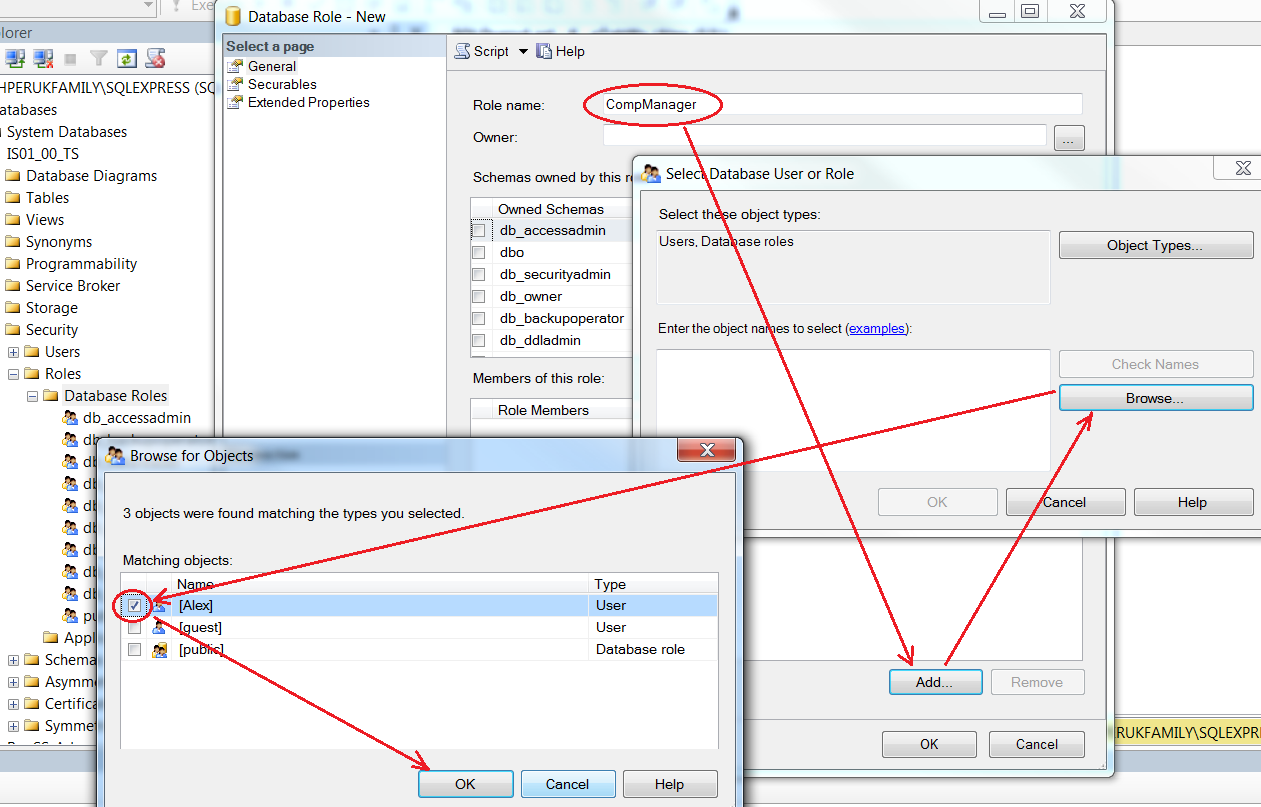


Рис.12.6. Створення користувацької ролі БД

### 12.2.3. Надання користувацькій ролі БД прав на роботу з певними таблицями БД (DELETE, INSERT, SELECT, UPDATE)

Для цього треба для кожної таблиці, на яку ролі CompManager надаються права, виконати наступне:

* За допомогою контекстного меню відчинити форму Table Properties таблиці.
* Виділити зверху закладку Permissions.
* Відчинивши ще дві форми, обрати роль CompManager для таблиці (рис.12.7).
* Зачинити останні дві форми. У формі Table Properties з’являться 3 колонки галочок для обрання прав ролі на таблицю.

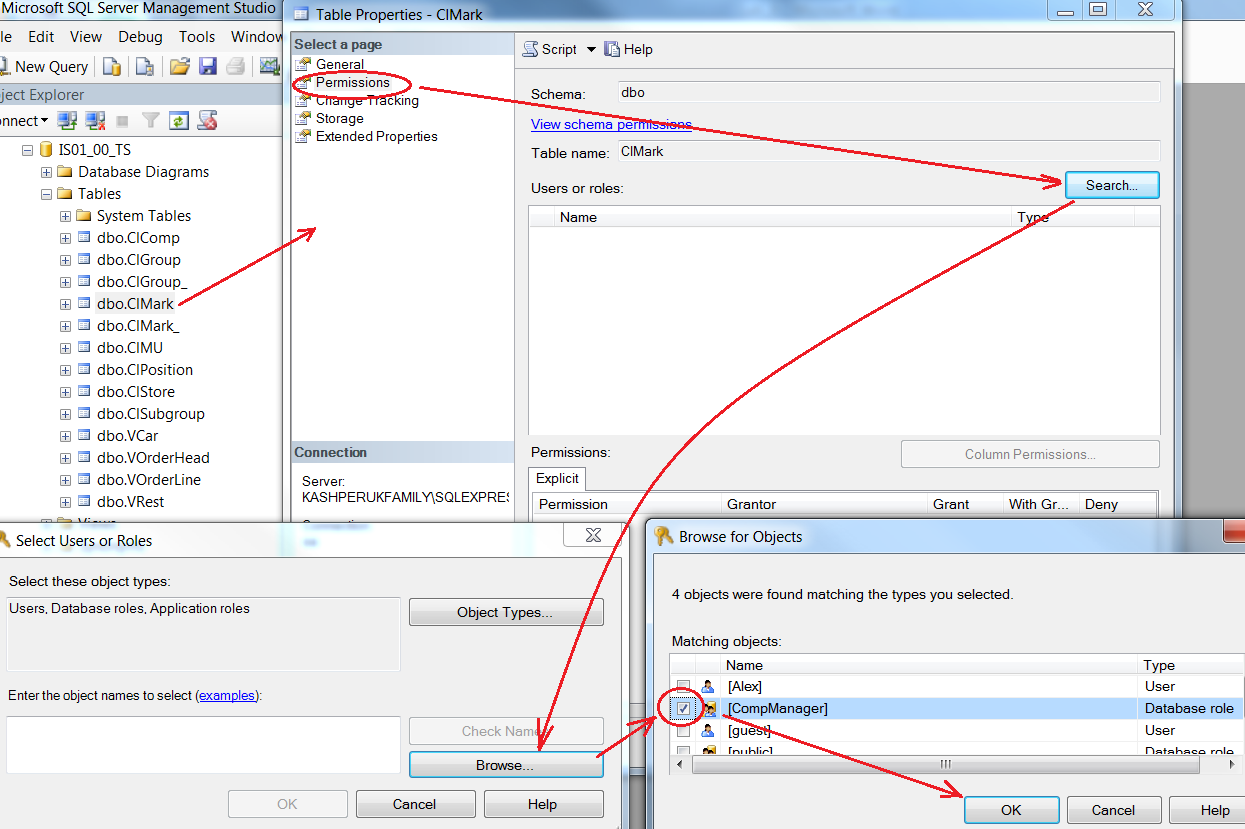
• Выдчинивши ще двы форми

Рис.12.7. Обрання користувацької ролі БД, яка має права на таблицю

* У колонці галочок Grant встановлюємо галочки для наступних прав (Permissions): DELETE, INSERT, SELECT, UPDATE (рис.12.8):

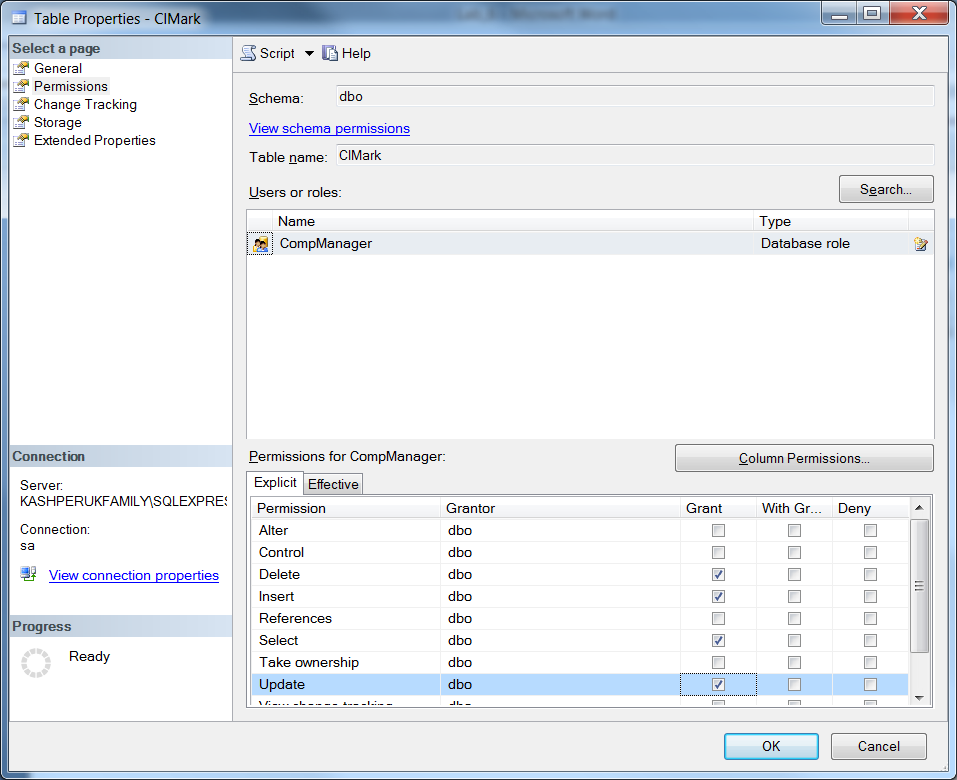


Рис.12.8. Обрання прав для користувацької ролі БД на таблицю

### 12.2.4. Надання цій ролі прав на роботу з певною таблицею БД (DELETE, INSERT, SELECT, UPDATE) засобами Transact-SQL

Маніпуляція правами на об’єкти БД виконується наступними операторами Transact-SQL:

|  |  |
| --- | --- |
| GRANT | Надати «позитивне» право, тобто дозвіл, на об’єкт БД |
| DENY | Надати «негативне» право, тобто заборону, на об’єкт БД |
| REVOKE | Скасувати право, надане раніше, в т.ч.операторами GRANT і DENY |

Для відчинення вікна консолі SQL треба виконати пункт меню File\New\Query with Current Connection або на панелі інструментів – New Query.

Запит на додавання прав ролі CompManager на маніпуляції з таблицею ClComp:

GRANT DELETE, INSERT, SELECT, UPDATE ON ClComp TO CompManager

Виконаємо цей вираз.

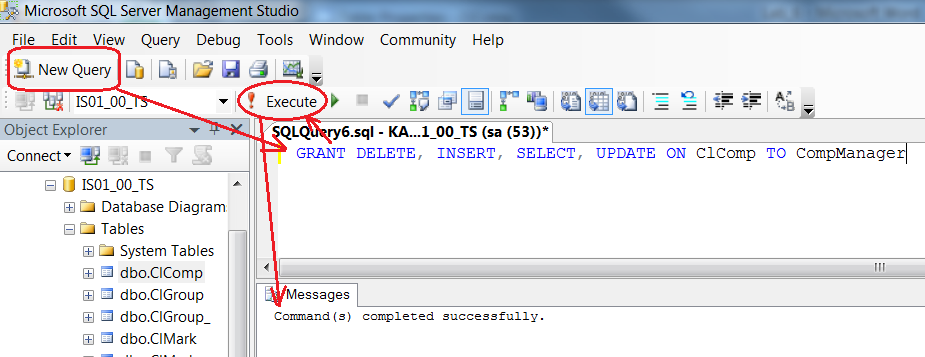


Рис.12.9. Виконання команди Transact-SQL на створення права

Після виконання команди негайно створюються відповідні права, що видно у формі Table Properties. Аби пересвідчитись в наявності прав у Alex на таблиці довідника деталей, треба виконати конекцію з логіном Alex і відчинити відповідні таблиці.

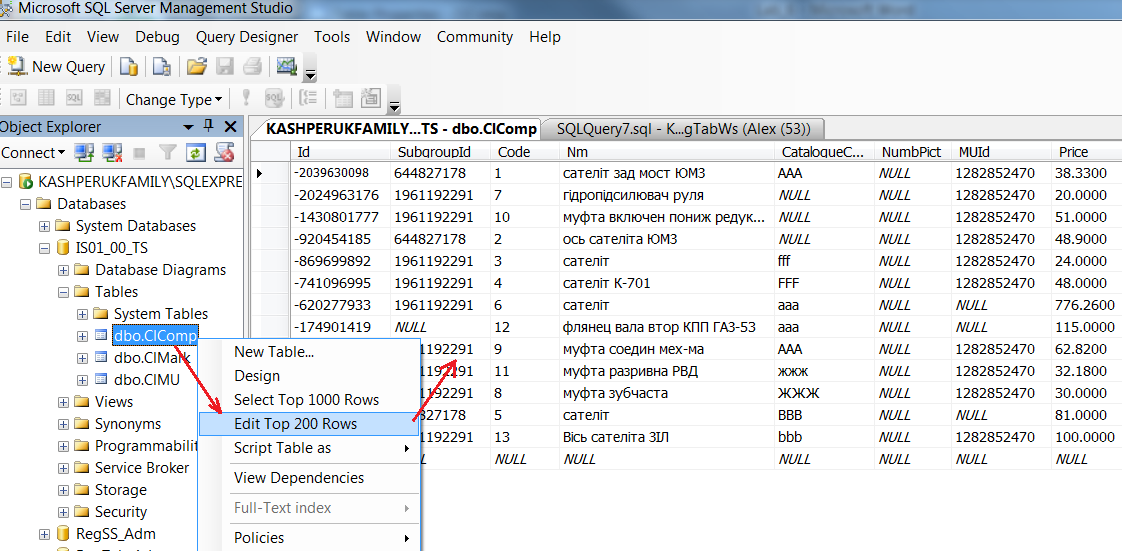


Рис.12.10. Переконаємось, що права Alex на таблиці довідника деталей існують.

## Контрольні питання

1. Поясніть 4 терміни, на яких тримається контроль прав доступу в Ms SQL 2008: логін, серверна роль, користувач БД, роль у БД.
2. Як найпростішим шляхом надати всі права логіну?
3. Яка послідовність дій по наданню прав користувачу БД на об’єкти БД за допомогою користувацької ролі БД?
4. Що таке Application Roles (ролі застосунку)?
5. Яка множинність зв’язку логіну і серверної ролі, користувача БД і ролі у БД?

# Теоретична частина для Oracle

Виконувати вправу потрібно тільки на домашньому комп’ютері, на кафедральних комп’ютерах є обмеження на вказані дії, для здачі достатньо показати відповідні скрипти та прокоментувати їх.

**Вправа 12.1**. Створення схеми та користувацької ролі БД

12.2.1. Створення схеми БД. Пересвідчитись, що він створена схема не має прав на таблиці індивідуальної предметної області.

12.2.2. Створення ролі БД.

12.2.3. Надання цій ролі прав на роботу з певними таблицями індивідуальної предметної області (DELETE, INSERT, SELECT, UPDATE).

**Вправа 12.2**. Управління доступом на рівні БД за допомогою користувацької ролі БД, в яку включена схема.

12.2.1. Включення схеми до користувацької ролі, створеної у вправі 12.1. Перевірити, що відповідні права на роботу з певними таблицями (DELETE, INSERT, SELECT, UPDATE) з’явились у схеми. Зокрема, продемонструвати у звіті належність по змісту таблиць до індивідуальної предметної області.

12.2.3. Вилучення у користувацької ролі прав на роботу з певними таблицями БД, визначених у п.12.2. Пересвідчитись, що схема не має прав на таблиці індивідуальної предметної області.

## Рекомендації по виконанню вправи 12.1.

Теоретична частина опису ролей та прав на них наведена в «[Методичні рекомендації по створенню структури БД в Oracle](http://web.asu.ntu-kpi.kiev.ua/moodle/mod/resource/view.php?id=120)» що знаходяться на кафедральній базі Moodle, а найсвіжіші (для розробників) на сайті Oracle, наприклад:

### CREATE USER

<http://docs.oracle.com/cd/B19306_01/server.102/b14200/statements_8003.htm>

### CREATE ROLE

<http://docs.oracle.com/cd/B19306_01/server.102/b14200/statements_6012.htm>

### GRANT

<http://docs.oracle.com/cd/B19306_01/server.102/b14200/statements_9013.htm#SQLRF01603>

### REVOKE

[http://docs.oracle.com/cd/B19306\_01/server.102/b14200/statements\_9020.htm#i2100253](http://docs.oracle.com/cd/B19306_01/server.102/b14200/statements_9020.htm%23i2100253)

## Приклад можливих дій для виконання вправи 12.2.

Створення "схеми" в БД Oracle. Під поняттям схема мається на увазі сам створений акаунт або кажучи прямо - користувач БД! Метою створення користувача і є одержання схеми БД, з певними правами і привілеями. Створення користувача в БД Oracle досить не складний, але в той же час досить концептуальний момент. Для створення користувача (схеми), застосовується команда DDL - CREATE USER. Вона має наступний синтаксис:

CREATE USER [користувач] IDENTIFIED BY [пароль]

DEFAULT TABLESPACE [tablespace] QUOTA целое число [K][M] ON [tablespace]

TEMPORARY TABLESPACE [tablespace] QUOTA целое число [K][M] ON [tablespace]

Цей синтаксис не зовсім повний, але для початку нам достатньо, далі ми розглянемо те, що було не вказано. Як правило, створення користувачів (схем) БД проводиться при підключенні до БД, користувачем SYS або SYSTEM. Так як ці дві схеми мають права адміністраторів БД. Розглянемо основні параметри команди - CREATE USER:

* **[**користувач**] [Username]** - Ім'я користувача (назву схеми).
* **[пароль] [Password]** - Пароль для облікового запису.
* **DEFAULT TABLESPACE** - Табличний простір в якому будуть знаходитися створювані в даній схемі об'єкти. Ця настройка не дає користувачеві права створювати об'єкти - тут встановлюється тільки значення за замовчуванням.
* **TEMPORARY TABLESPACE** - Табличний простір, в якому знаходяться тимчасові сегменти, використовувані в процесі сортування транзакцій.
* **QUOTA** - Дозволяє користувачеві зберігати об'єкти в зазначеному табличному просторі, займаючи там місце аж до визначеного в квоті загального розміру.

Для того, щоб запити користувачів могли створювати тимчасові сегменти в табличному просторі TEMP, їм не потрібні квоти на дисковий простір. Спробуємо створити користувача! Запускайте SQL \* Plus (чи у “Command Line” PLSQL Developer) з користувачем SYS або SYSTEM (паролі адміністратора для локального комп’ютера у вас мають бути власно установлені). З усього вище сказаного, запишемо ось таку конструкцію:

CREATE USER DUMMY IDENTIFIED BY DUMB

DEFAULT TABLESPACE USERS QUOTA 100M ON USERS

TEMPORARY TABLESPACE TEMP QUOTA 10M ON TEMP

/

Тут ми створюємо користувача (схему) DUMMY з паролем DUMB і дозволяємо йому квоту на на 100 Мб дискового простору USERS і ще трохи виділяємо з простору TEMP. Отримуємо в результаті:

SQL> CREATE USER DUMMY IDENTIFIED BY DUMB

2 DEFAULT TABLESPACE USERS QUOTA 100M ON USERS

3 TEMPORARY TABLESPACE TEMP QUOTA 10M ON TEMP

4 /

Пользователь создан.

Користувач (схема) створений. Можна вже підключиться і почати створювати об'єкти! Пробуємо!

CONNECT DUMMY/DUMB@PROBA

Такий синтаксис підключення можна використовувати, він ще називається рядок конекту і розписується ось так:

Username/Password@[//]Server[:Port][/Service]

де:

/ / - не обов'язково

Server - ім'я або IP-адреса комп'ютера, на якому працює Oracle Database

Port (не обов'язково) номер TCP-порту по якому приймає з'єднання слухач Oracle Net. Якщо порт не вказаний, то мається на увазі стандартний порт номер 1521.

Service (не обов'язково) - це ім'я сервісу бази даних, до якого треба підключитися. Для Oracle Database XE ім'я сервісу - XE. Якщо імя\_сервіса не вказано, то клієнт Oracle Database XE додає запит до сервісу по-замовчуванню бази даних, який налаштовується в процесі установки під ім'ям XE.

Отримаємо:

SQL> CONNECT DUMMY/DUMB

ERROR:

ORA-01045: user DUMMY lacks CREATE SESSION privilege; logon denied

Предупреждение: Вы больше не соединены с ORACLE.

Але створення користувача - це ще не все! Тепер йому потрібно дозволити саме основне - створювати сесію з сервером. Зробити це можна командою GRANT. Вона досить об'ємна і ми їй займемося трохи пізніше, а поки відновимо підключення:

SQL> CONNECT SYS/MANAGER@PROBA

Соединено.

Даємо користувачеві право створювати сесію з сервером:

SQL> GRANT CREATE SESSION TO DUMMY

2 /

Привилегии предоставлены.

пробуємо підключитися:

SQL> CONNECT DUMMY/DUMB

Соединено.

Ось тепер підключення успішне. Отже, ми створили користувача, визначили йому табличні простору, призначили квоти на них. І навіть дозволили створювати сесію з сервером. Давайте переконаємося, що користувач створений і все нормально. Виробляємо переконнект на адміна БД:

SQL> CONNECT SYS/MANAGER@PROBA

Соединено.

Дамо такий запит до подання DBA\_USERS:

SELECT USERNAME, USER\_ID, PASSWORD, ACCOUNT\_STATUS, DEFAULT\_TABLESPACE,

TEMPORARY\_TABLESPACE, PROFILE

FROM DBA\_USERS

WHERE USERNAME = 'DUMMY'

/

отримуємо:

SQL> SELECT USERNAME, USER\_ID, PASSWORD, ACCOUNT\_STATUS, DEFAULT\_TABLESPACE,

2 TEMPORARY\_TABLESPACE, PROFILE

3 FROM DBA\_USERS

4 WHERE USERNAME = 'DUMMY'

5 /

USERNAME USER\_ID PASSWORD ACCOUNT\_STATUS DEFAULT\_TABLESPACE TEMPORARY\_TABLESPACE PROFILE

-------- -------- ---------------- --------------- ------------------ -------------------- -------

DUMMY 64 E888ADB4D5FFE1B2 OPEN USERS TEMP DEFAULT

Хто знайомий з криптографією, може на дозвіллі розкусити - E888ADB4D5FFE1B2! Отже, всі з нашої схемою в порядку. Залишилося тільки дозволити користувачеві створювати об'єкти БД.

дозволяємо:

GRANT CREATE TABLE TO DUMMY

/

GRANT CREATE PROCEDURE TO DUMMY

/

GRANT CREATE TRIGGER TO DUMMY

/

GRANT CREATE VIEW TO DUMMY

/

GRANT CREATE SEQUENCE TO DUMMY

/

отримуємо:

SQL> GRANT CREATE TABLE TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT CREATE PROCEDURE TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT CREATE TRIGGER TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT CREATE VIEW TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT CREATE SEQUENCE TO DUMMY

2 /

Привилегии предоставлены.

Оператор GRANT це DDL, то COMMIT викликається не явно. В даному випадку ми дозволили користувачеві, створювати такі основні об'єкти БД як - таблиці, процедури, тригери, VIEW, SEQUENCE. Для початку цього достатньо. А що робити, якщо користувачеві буде необхідно змінювати ці об'єкти? Тоді потрібно додати ще небагато прав, на зміну (ALTER) ось так:

GRANT ALTER ANY TABLE TO DUMMY

/

GRANT ALTER ANY PROCEDURE TO DUMMY

/

GRANT ALTER ANY TRIGGER TO DUMMY

/

GRANT ALTER PROFILE TO DUMMY

/

отримуємо:

SQL> GRANT ALTER ANY TABLE TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT ALTER ANY PROCEDURE TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT ALTER ANY TRIGGER TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT ALTER PROFILE TO DUMMY

2 /

Привилегии предоставлены.

Ось тепер він може не тільки створювати ці об'єкти, але і змінювати їх. А, що якщо користувачеві необхідно буде видалити який-небудь об'єкт або видалити записи з таблиць? Тоді потрібно додати права на видалення об'єктів БД ось так:

GRANT DELETE ANY TABLE TO DUMMY

/

GRANT DROP ANY TABLE TO DUMMY

/

GRANT DROP ANY PROCEDURE TO DUMMY

/

GRANT DROP ANY TRIGGER TO DUMMY

/

GRANT DROP ANY VIEW TO DUMMY

/

GRANT DROP PROFILE TO DUMMY

/

отримуємо:

SQL> GRANT DELETE ANY TABLE TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT DROP ANY TABLE TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT DROP ANY PROCEDURE TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT DROP ANY TRIGGER TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT DROP ANY VIEW TO DUMMY

2 /

Привилегии предоставлены.

SQL> GRANT DROP PROFILE TO DUMMY

2 /

Привилегии предоставлены.

Зберіть всі рядки, в файл, скажімо make.sql, помістіть його в каталог, наприклад, Temp, і введіть наступне:

SQL> @C:\TEMP\MAKE.SQL

Після цього всі команди та запити, що наведені в файлі будуть виконані.

В SQL Plus є ще багато внутрішніх команд, наприклад, дуже корисною може виявитися **SET TIME ON** та запрошення який прийме ось такий вигляд:

14:09:52 SQL>

Наприклад, можна оцінювати час на запит з таблиці.

Якщо ввести SET TIME OFF, то все стане по колишньому. Наприклад, якщо написати SHOW USER, то побачимо приблизно таке:

SQL> SHOW USER

USER имеет значение "DUMMY"

# Рекомендації по виконанню роботи для MySQL

На сьогодні в MySQL ролі не передбачені. Тому завдання дещо спрощене:

**Вправа 12.1**. Управління доступом на рівні користувача БД

12.1.1. Створення користувача БД. Пересвідчитись, що він не має прав на таблиці індивідуальної предметної області.

12.1.2. Надання користувачу прав на роботу з певними таблицями індивідуальної предметної області (DELETE, INSERT, SELECT, UPDATE). Перевірити, що відповідні права з’явились у користувача БД. Зокрема, продемонструвати у звіті належність по змісту таблиць до індивідуальної предметної області.

12.1.3. Вилучення у користувача прав на роботу з певними таблицями БД, визначених вище у п. 12.1.2. Пересвідчитись, що він не має прав на таблиці індивідуальної предметної області.

## Контрольні питання

1. Поясніть 3 терміни, що використовуються у контролі прав доступу в Oracle: схема, роль, користувач БД.
2. Як найпростішим шляхом надати всі права логіну?
3. Яка послідовність дій по наданню прав користувачу БД на об’єкти БД за допомогою ролі БД?
4. Яка множинність зв’язку користувача і ролі?