



CHƯƠNG 4

BẢO MẬT VÀ AN TOÀN DỮ LIỆU

NỘI DUNG



SAO LƯU VÀ PHỤC HỒI CƠ SỞ DỮ LIỆU

PHỤC HỒI CƠ SỞ DỮ LIỆU

QUẢN LÝ NGƯỜI DÙNG VÀ BẢO MẬT HỆ THỐNG



SAO LƯU VÀ PHỤC HỒI CSDL

GIỚI THIỆU



Khi làm việc với CSDL, đặc biệt là những tác vụ dễ xảy ra sự cố như:

- Thử nghiệm tính năng mới
- Sửa hoặc xóa dữ liệu quan trọng

⇒ Tạo một bản sao CSDL để phục hồi khi có sự cố xảy ra.

1. CÁC MÔ HÌNH PHỤC HỒI



Có 3 mô hình phục hồi:

- ☐ Full Recovery Model
- ☐ Bulk-Logged Recovery Model
- ☐ Simple Recovery Model

1. CÁC MÔ HÌNH PHỤC HỒI



Full Recovery Model:

- Là mô hình phục hồi đầy đủ. Cho phép phục hồi dữ liệu với ít rủi ro nhất.
- Tất cả các hoạt động của CSDL đều được ghi vào transaction log file.
- Khi có sự cố có thể phục hồi lại dữ liệu ngược trở lại đến một thời điểm trong quá khứ.

Khuyết điểm: Transaction log có thể rất lớn.

1. CÁC MÔ HÌNH PHỤC HỒI



Bulk-Logged Recovery Model:

- Các hoạt động mang tính hàng loạt như Bulk Insert, Bulk copy operations, Create Index, WriteText, UpdateText chỉ được ghi tối thiểu vào transaction log file, đủ để cho biết là các hoạt động này có diễn ra mà không ghi toàn bộ chi tiết như trong Full Recovery Mode.
- Các hoạt động khác như Insert, Update, Delete vẫn được ghi đầy đủ để dùng cho việc phục hồi sau này.

1. CÁC MÔ HÌNH PHỤC HỒI



Simple Recovery Model:

- File transaction log luôn được cập nhập nhưng lại không phục hồi.

1. CÁC MÔ HÌNH PHỤC HỒI

❖ **Thiết lập mô hình phục hồi cho CSDL:**

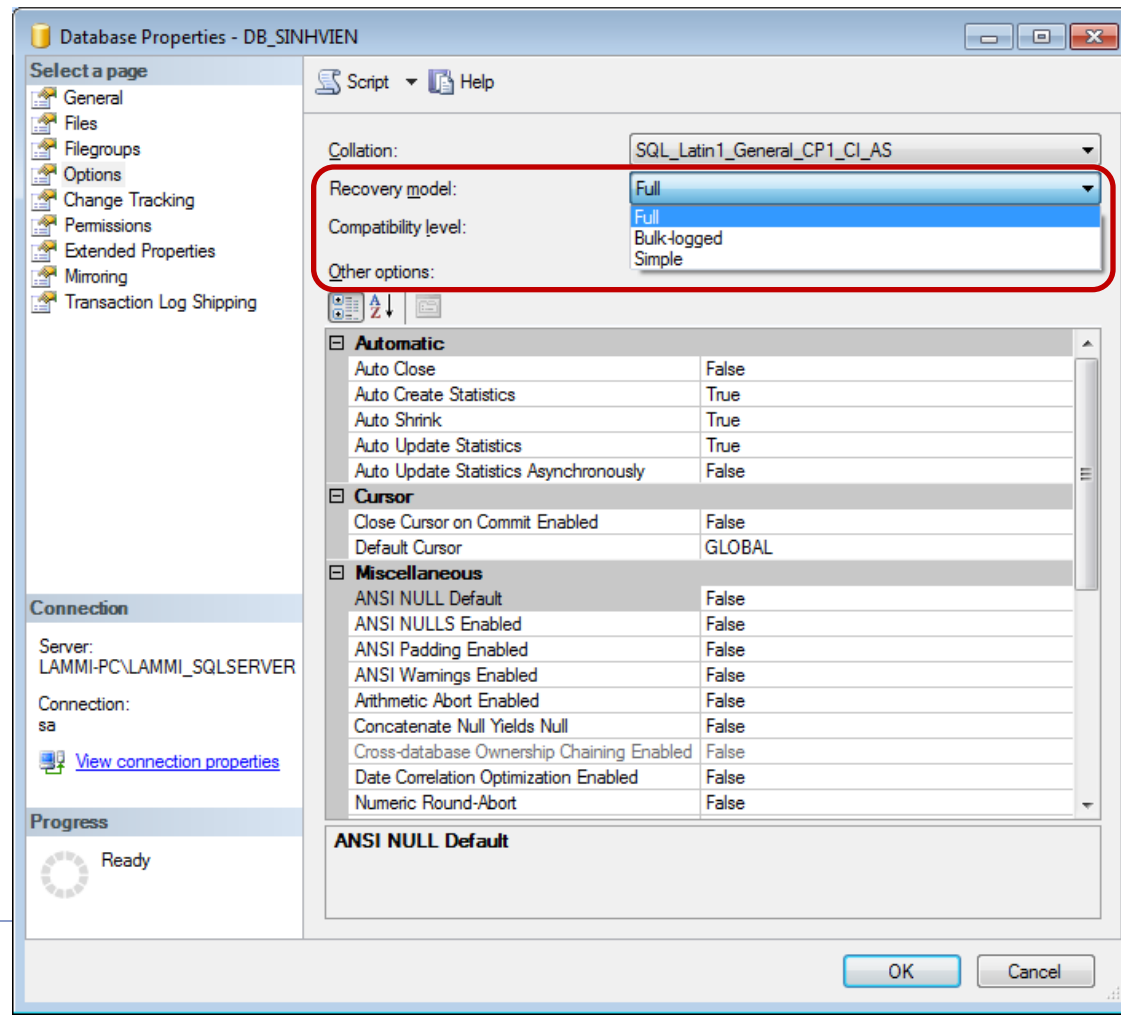
Cách 1: Nhấn chuột phải lên CSDL cần phục hồi trong SQL Server Management Studio → Properties → Options → Recovery model

1. CÁC MÔ HÌNH PHỤC HỒI



❖ Thiết lập mô hình phục hồi cho CSDL:

Cách 1:



1. CÁC MÔ HÌNH PHỤC HỒI

❖ Thiết lập mô hình phục hồi cho CSDL:

Cách 2: Sử dụng lệnh

Alter Database <tên_database>

Set Recovery <simple/full/bulk_logged>

Ví dụ:

Alter Database demo

Set Recovery full

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



❖ Phân loại sao lưu:

- ✓ **Sao lưu toàn phần:** Toàn bộ CSDL sẽ được ghi lại tại thời điểm sao lưu.
- ✓ **Sao lưu một phần:** Chỉ những phần thay đổi của CSDL so với thời điểm mà CSDL đã sao lưu toàn phần gần nhất.
- ✓ **Sao lưu bảng lưu vết của giao tác (transaction log):** Sao lưu vết của các giao tác đang được thực thi tại thời điểm sao lưu.

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



❖ Các loại hình sao lưu

✓ **Full Backup:** Sao lưu toàn bộ dữ liệu tại thời điểm thực hiện. Tập tin sao lưu có phần mở rộng **.bak**

➤ **Khuyết điểm:** Mất nhiều thời gian hơn các phương pháp khác nếu CSDL lớn.

✓ **Differential backup:** Sao lưu các dữ liệu mới kể từ lần **Full Backup** trước đó. Tập tin sao lưu có phần mở rộng **.bak**

➤ **Phải có một bản sao lưu Full Backup trước đó.**

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



❖ Các loại hình sao lưu

✓ **Transaction Log Backup:** Sao lưu các bản ghi Transaction Log (nghĩa là sao lưu các thao tác xảy ra trên CSDL mà không sao lưu dữ liệu).

Tập tin sao lưu có phần mở rộng là **.trn**

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



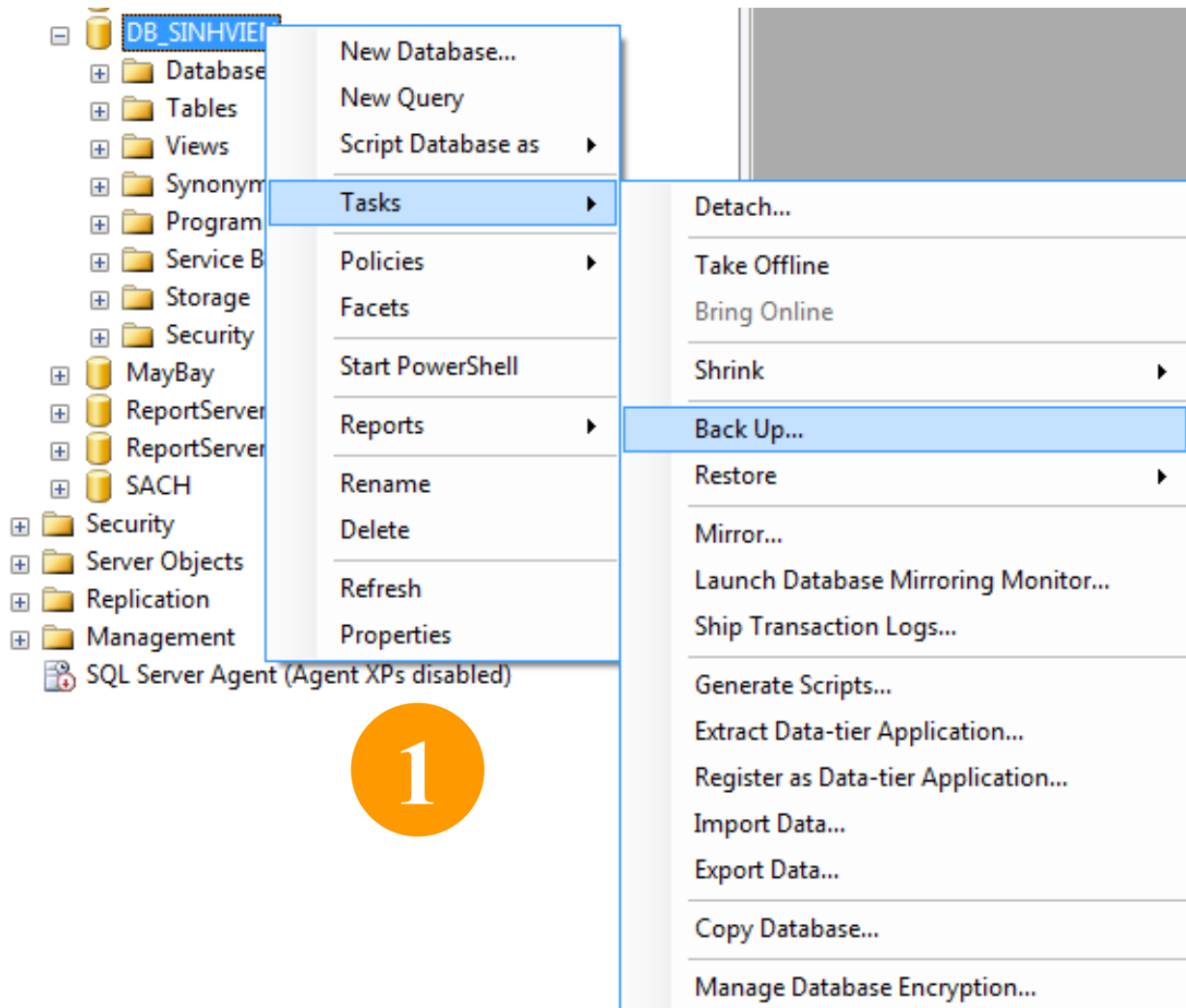
❖ Sao lưu Cơ sở dữ liệu:

Cách 1: Sử dụng SQL Server Management Studio

Nhấn chuột phải vào cơ sở dữ liệu cần sao lưu → chọn **Tasks → Back Up:**

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)

Cách 1:



2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



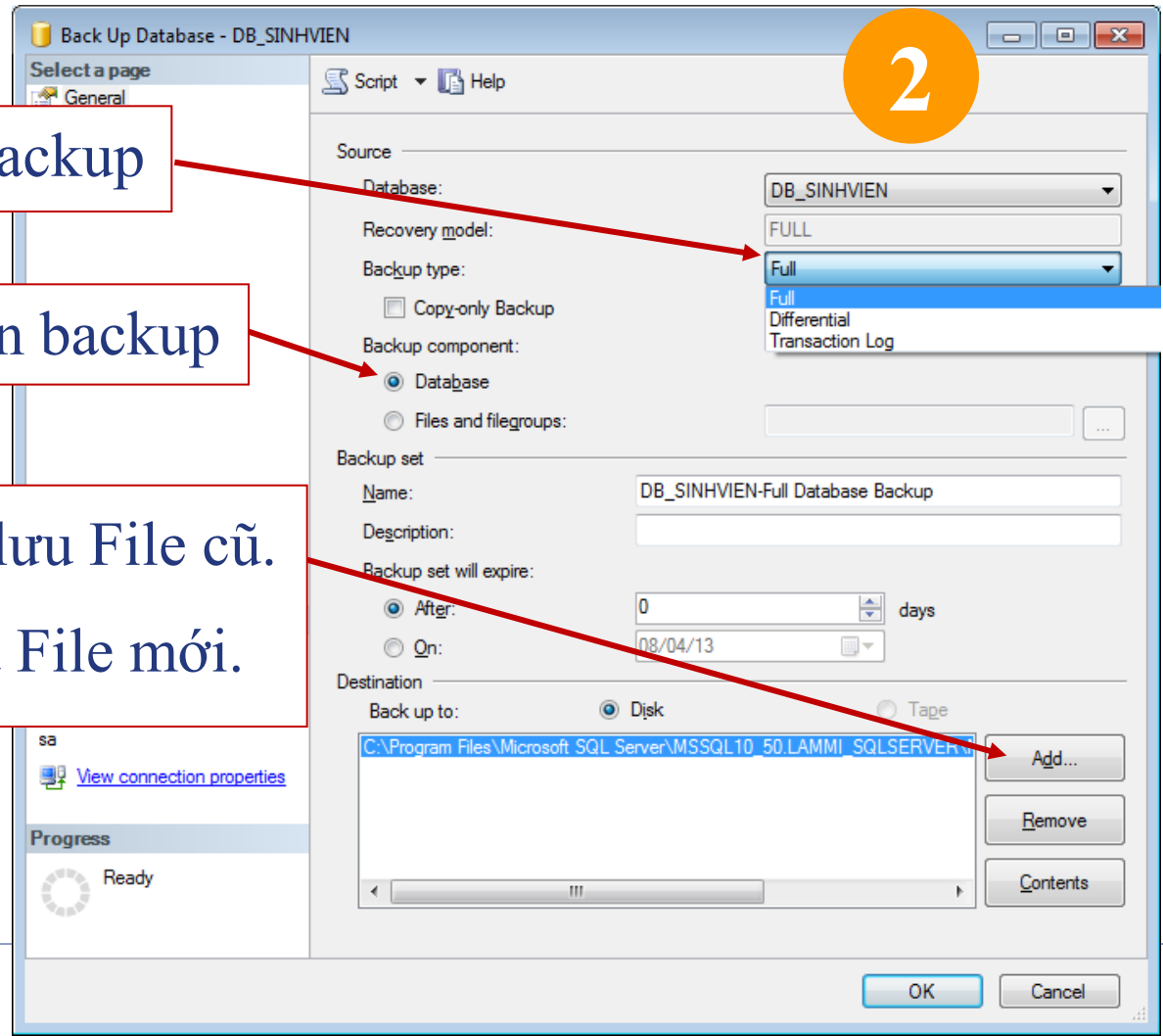
Cách 1:

Chọn loại hình backup

Chọn thành phần cần backup

Remove: Hủy vị trí lưu File cũ.

Add: Chọn vị trí lưu File mới.

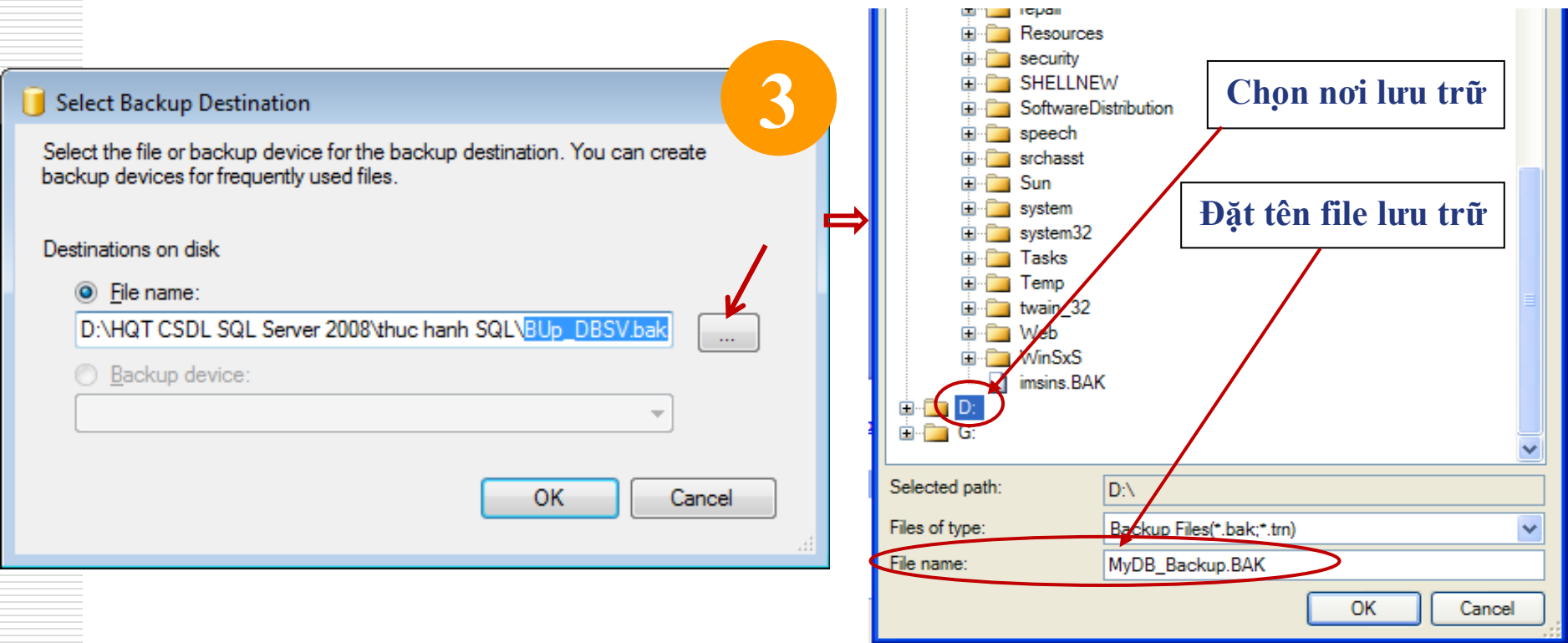


2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



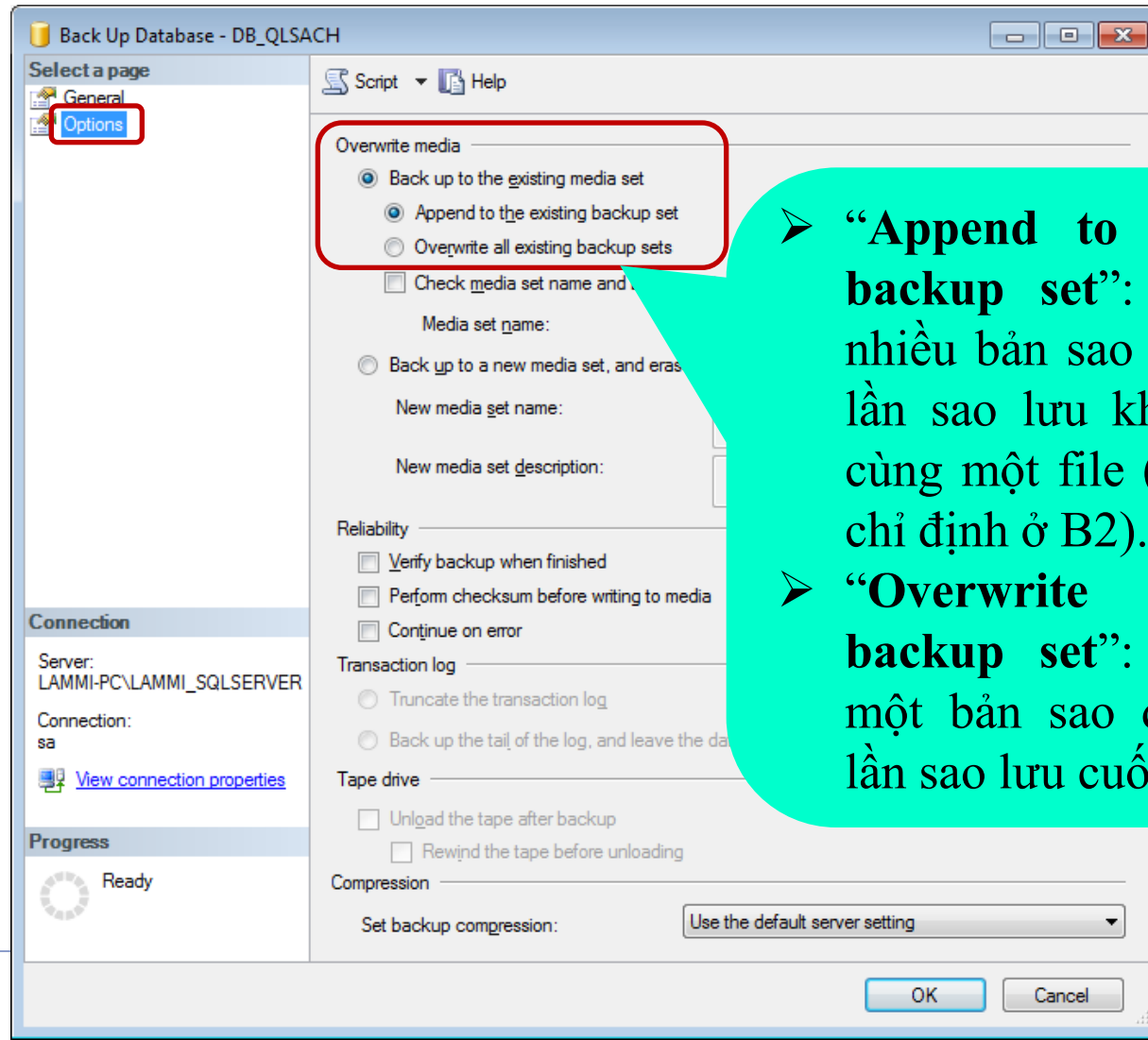
Cách 1:

Bảng Select Backup Destination xuất hiện



2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)

Cách 1:



- **“Append to the existing backup set”**: Có thể lưu nhiều bản sao CSDL từ các lần sao lưu khác nhau vào cùng một file (tên file được chỉ định ở B2).
- **“Overwrite all existing backup set”**: Chỉ giữ lại một bản sao duy nhất của lần sao lưu cuối cùng.

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



Cách 2: Sử dụng lệnh

Backup Database <tên_database>

To Disk = 'Đường dẫn đến nơi lưu trữ\
tên_file_backup.**bak**'

[option]

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



Cách 2:

Ví dụ:

-- Back up Full

BACKUP DATABASE DB_SINHVIEN

TO DISK = 'D:\Backup_DBSV.bak'

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



Cách 2:

Ví dụ:

-- Back up Full

BACKUP DATABASE DB_SINHVIEN

TO DISK = 'D:\Backup_DBSV.bak'

WITH DESCRIPTION = 'sao luu database

DB_SINHVIEN'

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



Cách 2:

Ví dụ:

-- Backup Difference

BACKUP DATABASE DB_SINHVIEN

TO DISK = 'D:\Backup_DBSV.bak'

WITH DIFFERENTIAL

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



Cách 2:

Ví dụ:

--Backup log

BACKUP LOG AdventureWorks

TO DISK = 'C:\Backup\AdventureWorks.trn'

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



Lưu ý:

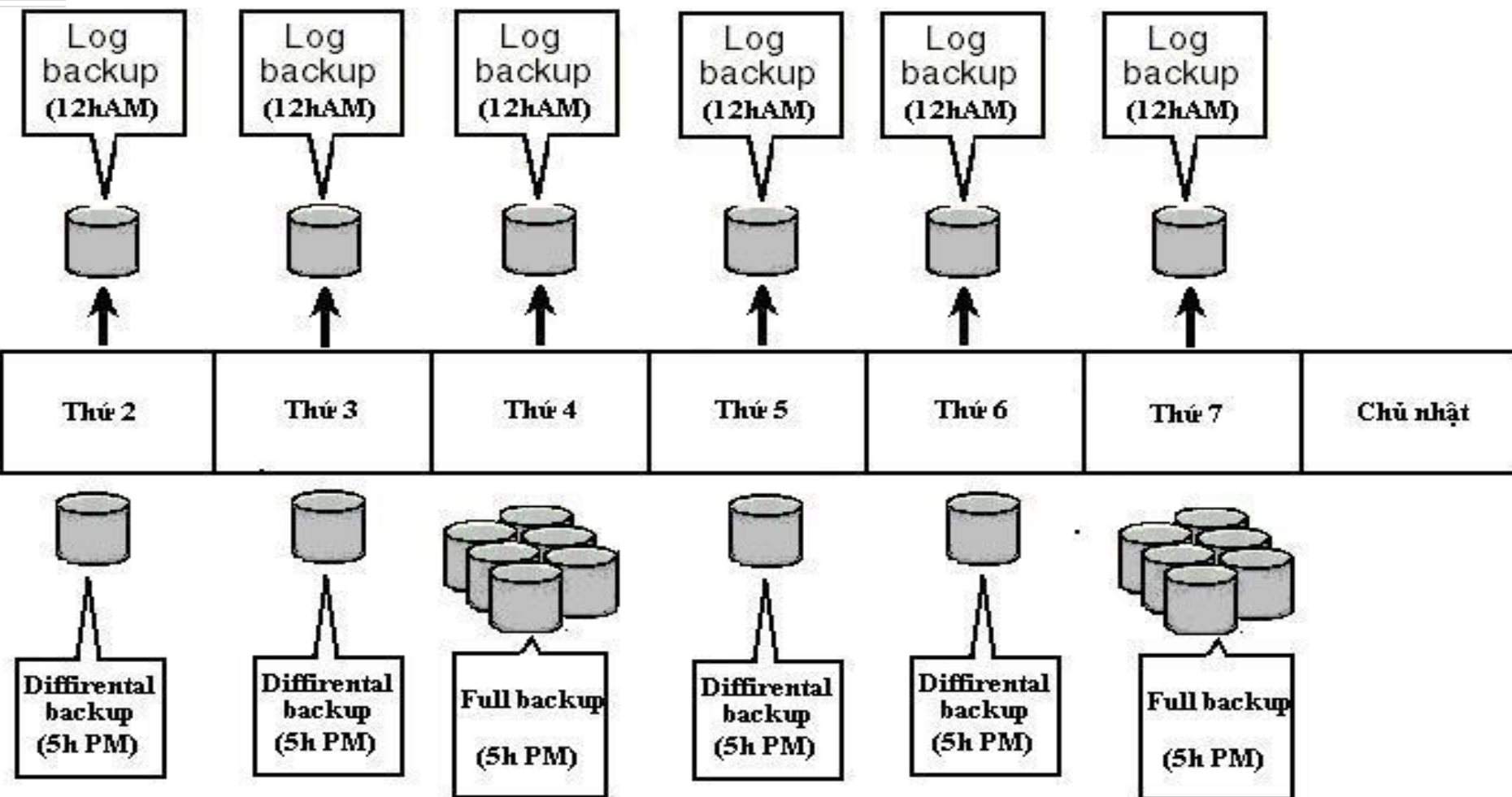
- **Differential Backup** luôn sao lưu các trang đã thay đổi kể từ lần Full Backup trước, không phải từ lần Differential Backup trước đó.
- **Transaction Log Backup** thì ngược lại, chỉ sao lưu các log record kể từ lần Transaction Log Backup trước đó.

2. SAO LƯU CƠ SỞ DỮ LIỆU (Backup Database)



Nguyên tắc chung để giảm thiểu mất mát dữ liệu khi có sự cố là tăng tần suất sao lưu. Tuy nhiên, với CSDL có dung lượng lớn và được cập nhật liên tục, thì việc thực hiện Full Backup với tần suất cao là không khả thi vì nó dùng rất nhiều tài nguyên (CPU, I/O). **Differential backup** và **Transaction log backup** có thể tạo lập các phương án sao lưu thích hợp, đảm bảo dữ liệu được sao lưu thường xuyên mà không chiếm nhiều tài nguyên của hệ thống.

LỊCH TRÌNH BACKUP DỮ LIỆU CỦA MỘT HỆ THỐNG



3. PHỤC HỒI CƠ SỞ DỮ LIỆU (Restore Database)

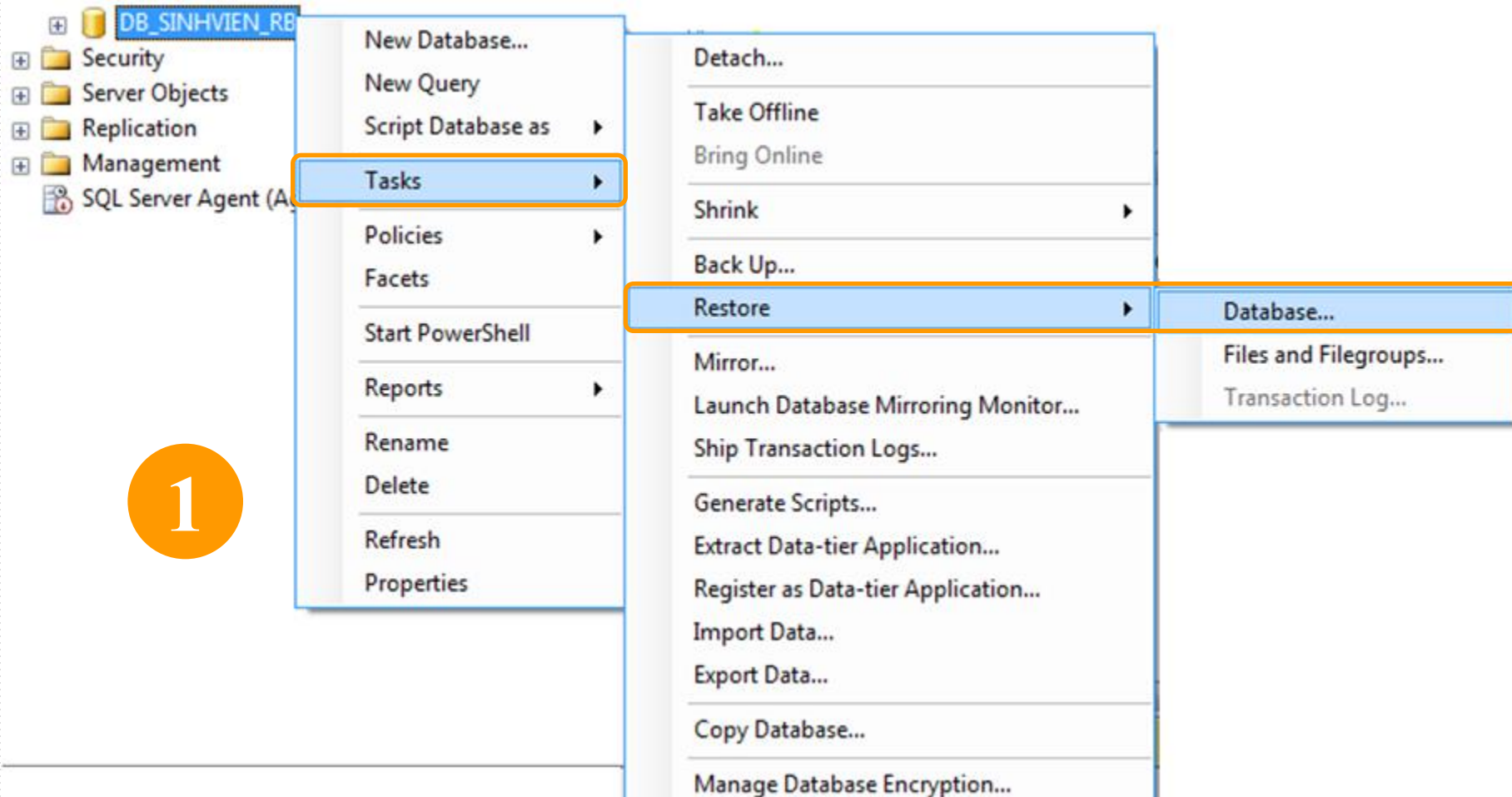


❖ **Phục hồi CSDL** là một công việc rất quan trọng nhằm trả về nguyên hiện trạng CSDL như lúc bắt đầu sao lưu. Mỗi phương thức sao lưu có một phương thức phục hồi tương ứng.

3. PHỤC HỒI CƠ SỞ DỮ LIỆU (Restore Database)



❖ Cách 1: Sử dụng SQL Server Management Studio



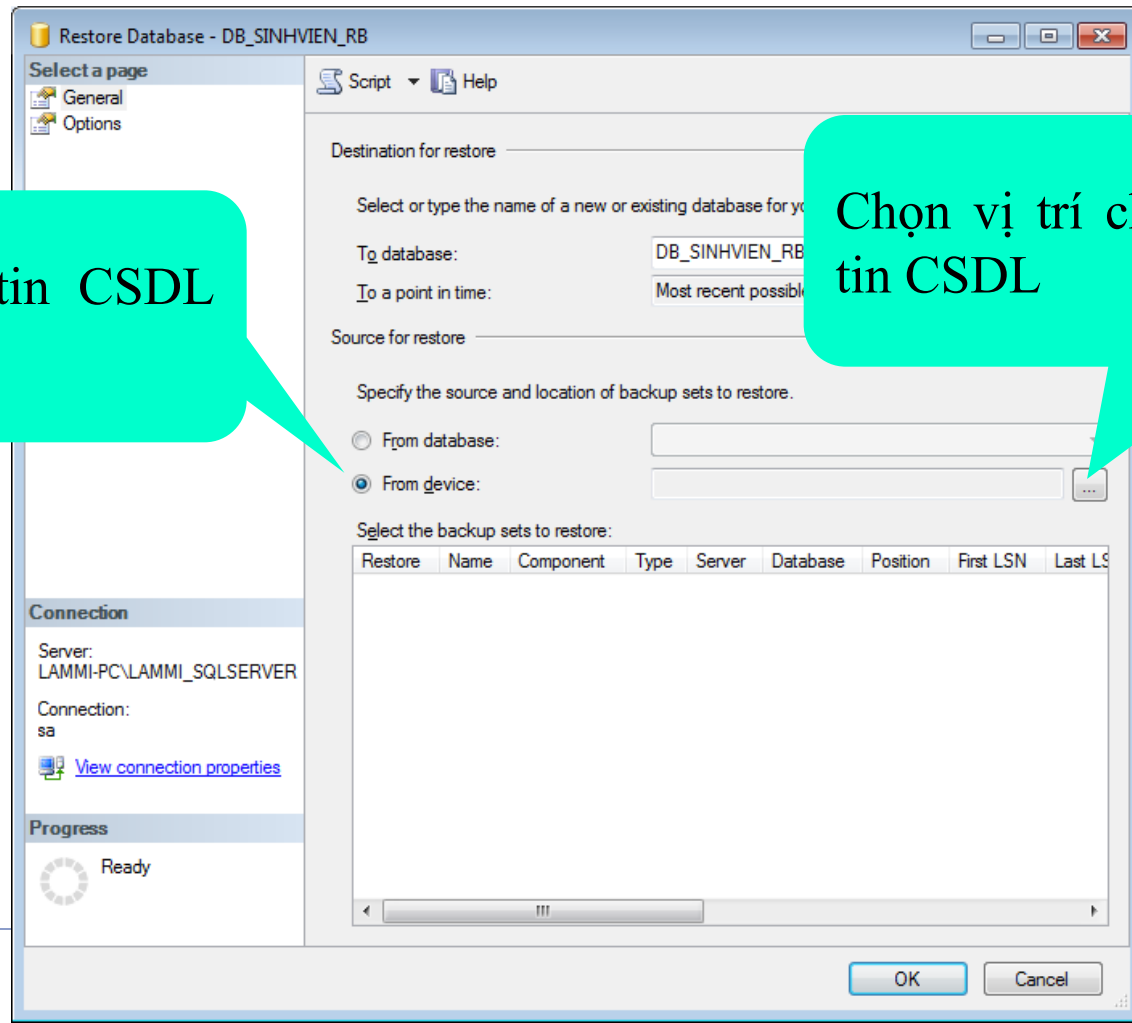
3. PHỤC HỒI CƠ SỞ DỮ LIỆU (Restore Database)



❖ Cách 1: Sử dụng SQL Server Management Studio

Chọn tập tin CSDL
trên ổ đĩa

2

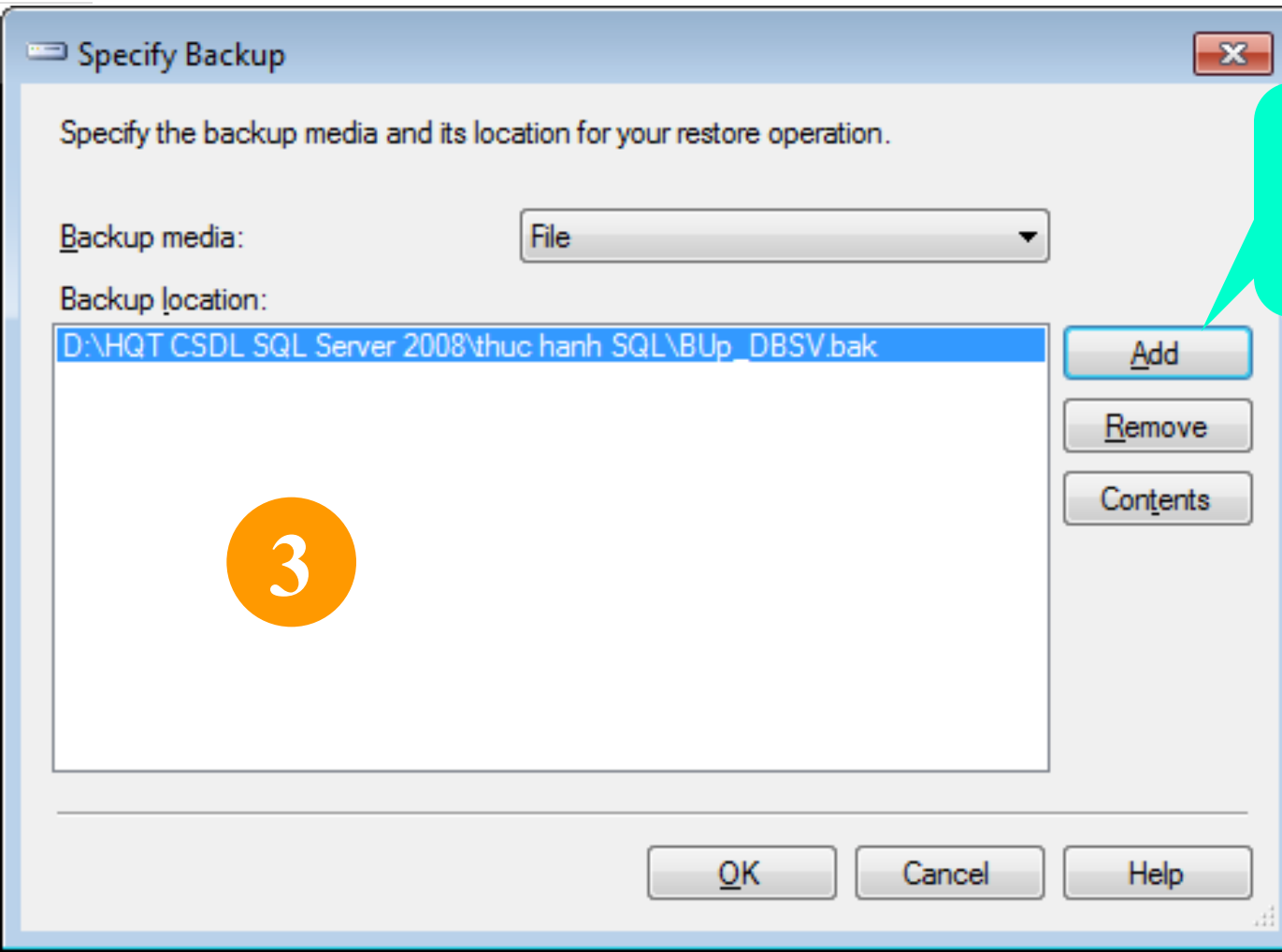


Chọn vị trí chứa tập
tin CSDL

3. PHỤC HỒI CƠ SỞ DỮ LIỆU (Restore Database)



❖ Cách 1: Sử dụng SQL Server Management Studio

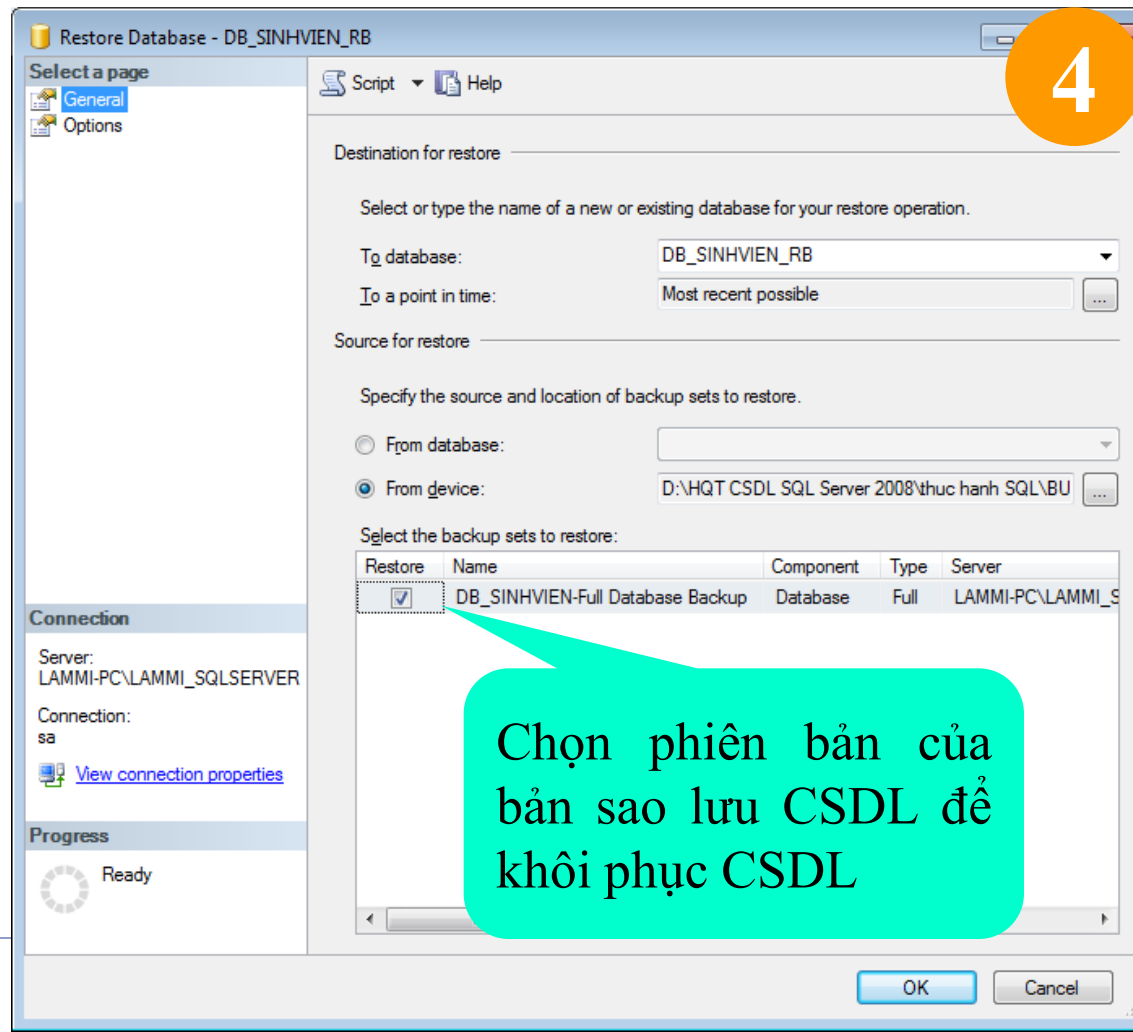


Lấy đường dẫn và file **.bak** đã sao lưu trước đó.

3. PHỤC HỒI CƠ SỞ DỮ LIỆU (Restore Database)



❖ Cách 1: SQL Server Management Studio



3. PHỤC HỒI CƠ SỞ DỮ LIỆU (Restore Database)



❖ Cách 2: Dùng lệnh

RESTORE DATABASE <Tên_database>

FROM DISK = 'Đường dẫn đến nơi lưu trữ\
tên_file_backup.bak'

[With Option]

👉 LƯU Ý: Khi sao lưu/ phục hồi CSDL

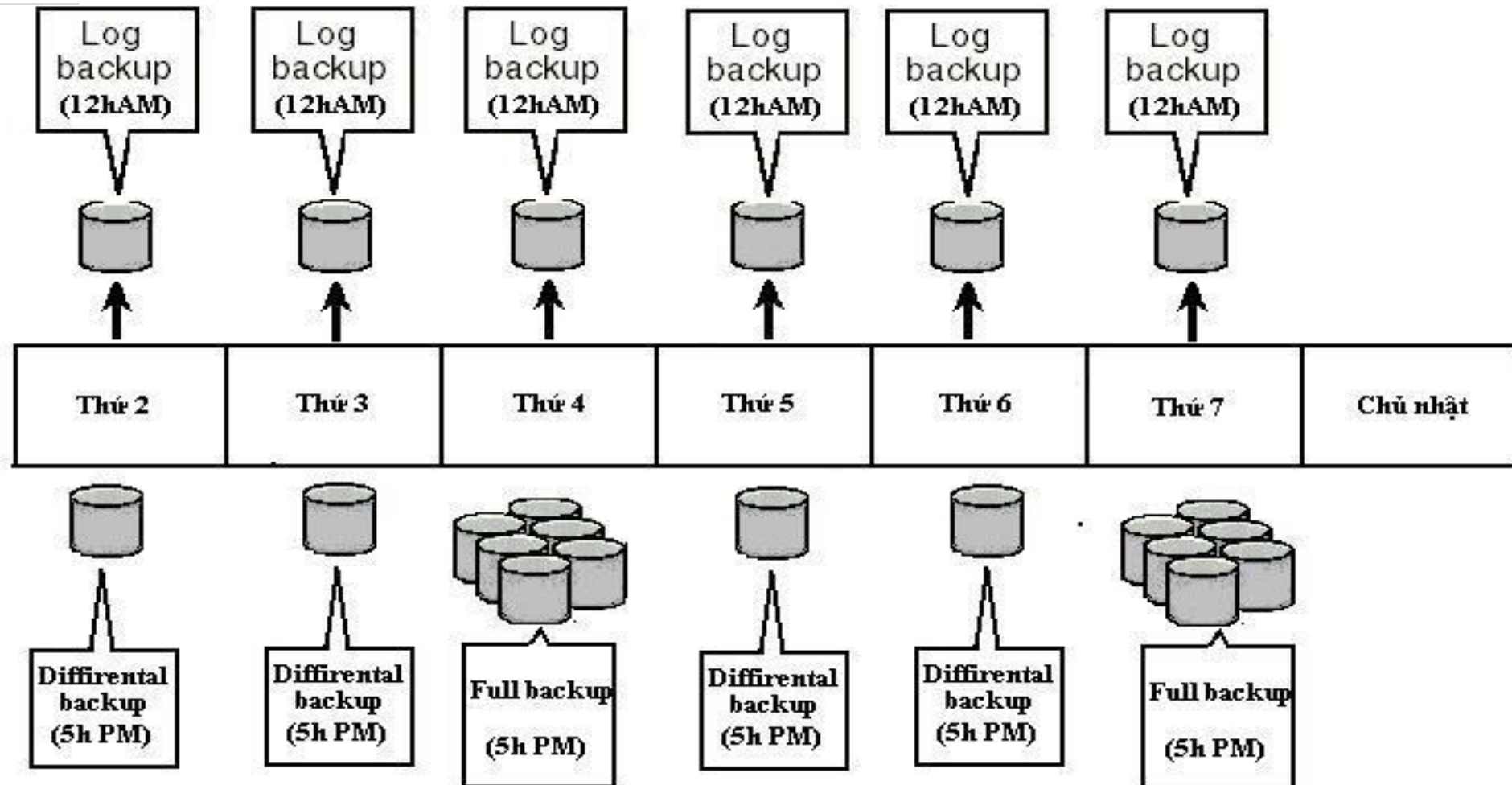
- Càng thực hiện sao lưu nhiều càng giảm rủi ro khi có sự cố.
- Với các CSDL quan trọng, thực hiện nhiều thay đổi trong ngày nên thực hiện nhiều lần sao lưu trong một ngày.
- Full Backup là phương pháp an toàn nhất, nhưng thực hiện Full Backup nhiều sẽ tốn dung lượng bộ nhớ.
- **Nên:**
 - Thực hiện sao lưu Full Backup 1 lần vào lúc bắt đầu một ngày (chuẩn bị làm việc với CSDL).
 - Thực hiện nhiều sao lưu Differential Backup trong ngày
 - Thực hiện nhiều sao lưu Transaction Log Backup trong ngày

👉 LƯU Ý: Khi sao lưu/ phục hồi CSDL

- **Khi có sự cố, tiến hành phục hồi như sau:**
 - Phục hồi CSDL sử dụng bản sao lưu Full Backup.
 - Phục hồi CSDL sử dụng bản sao lưu Differential Backup trước và gần thời điểm xảy ra sự cố.
 - Phục hồi tất cả các Transaction Log Backup kể từ sau lần Differential Backup gần nhất.

👉 VÍ DỤ 1:

Thời điểm xảy ra sự cố: 4 giờ chiều thứ 6



👉 Lịch trình khôi phục Cách 1 như sau:

Bước 1. Khôi phục từ bản Full Backup gần với thời điểm có sự cố nhất (bản Full Backup của ngày thứ 4).

Bước 2. Khôi phục từ bản Differential Backup gần với thời điểm có sự cố nhất (bản Differential Backup lúc 5h PM của ngày thứ 5).

Bước 3. Khôi phục tất cả các Transaction Log Backup kể từ sau lần Differential Backup gần nhất (bản Transaction Log Backup tại thời điểm 12hAM của chính ngày xảy ra sự cố)

Kết quả: Chỉ có thể khôi phục tới thời điểm 12gAM NGÀY THỨ 6, mất dữ liệu trong khoảng từ 12gAM – 4 giờ chiều.

👉 Lịch trình khôi phục Cách 2 như sau:

Điều Kiện: còn file .ldf

Bước 1. Sao lưu Transaction Log từ 12gAM thứ 6 đến thời điểm xảy ra sự cố. File backup trong trường hợp này được gọi là "**The tail of the log**" (phần đuôi của log). Khi backup “phần đuôi” này cần phải dùng option **NO_TRUNCATE**

Bước 2. Khôi phục từ bản Full Backup gần với thời điểm có sự cố nhất (bản Full Backup của ngày thứ 4).

Bước 3. Khôi phục bản Differential Backup gần với thời điểm có sự cố nhất (bản Differential Backup lúc 5h PM của ngày thứ 5).

Bước 4. Khôi phục Transaction Log tại thời điểm 12hAM,

Và cuối cùng bản Transaction Log lúc 4h PM (chính là phần đuôi đã thực hiện ở bước 1).

👉 **VÍ DỤ 2:**



Kịch bản sao lưu và khôi phục dữ liệu tham khảo:

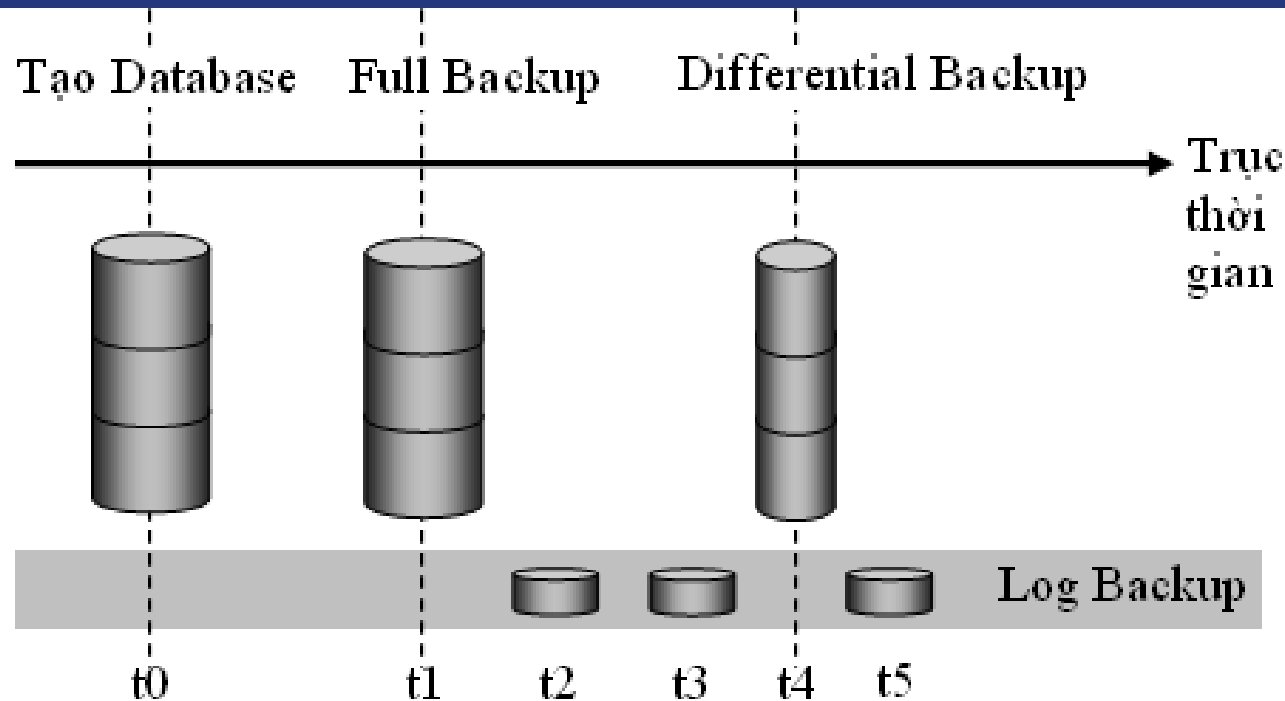
- **Full backup:** Một lần mỗi ngày vào 2h sáng.
- **Differential backup:** Vào các thời điểm 6h, 10h, 14h, 18h, 22h (5 lần/ngày).
- **Transaction Log Backup:** 15 phút một lần vào các thời điểm 5', 20', 35', và 50' của mỗi giờ (4 lần/giờ).
- ***Giả sử CSDL bị hỏng vào thời điểm 10h55', cần khôi phục lại CSDL như thế nào?***

VÍ DỤ 2:



*Giả sử CSDL bị hỏng vào thời
điểm 10h55', cần khôi phục
lại CSDL như thế nào?*

👉 BÀI TẬP 1:



👉 **Lưu ý:** Tại mỗi thời điểm t_i ($i \geq 1$) sinh viên tự thêm dữ liệu để đảm bảo có sự thay đổi dữ liệu trong CSDL. Giả sử sau đó xảy ra sự cố (Ví dụ CSDL bị xóa). Viết lệnh khôi phục lại CSDL từ các bản sao lưu.

BÀI TẬP 1:



- Tạo 1 database tên là **TEST**
- Tại thời điểm **t1**: Full Backup
- Tại thời điểm **t2, t3**: Log Backup
- Tại thời điểm **t4**: Differential Backup
- Tại thời điểm **t5**: Log Backup

BÀI TẬP 1:



-- Thời điểm t0: Tạo CSDL

```
CREATE DATABASE TEST  
USE TEST
```

```
CREATE TABLE Table1(c INT)
```

```
INSERT INTO Table1  
VALUES(1)
```

BÀI TẬP 1:



-- Thời điểm t1: Full Backup

BACKUP DATABASE TEST

TO DISK = 'D:\Backup\Test_FULL.bak'

WITH INIT

-- Thêm một bản ghi mới

INSERT Table1 VALUES(2)

BÀI TẬP 1:



-- Thời điểm t2: Log Backup

BACKUP LOG TEST

TO DISK = 'D:\Backup\Test_TRAN.trn'

WITH INIT

-- Thêm một bản ghi mới

INSERT Table1 VALUES(3)

BÀI TẬP 1:



-- Thời điểm t3: Log Backup

BACKUP LOG TEST

TO DISK = 'D:\Backup\Test_TRAN.trn'

-- Thêm một bản ghi mới

INSERT Table1 VALUES(4)

BÀI TẬP 1:



-- Thời điểm t4: Differential backup

BACKUP DATABASE TEST

TO DISK = 'D:\Backup\Test_DIFF.bak'

WITH INIT, DIFFERENTIAL

-- Thêm một bản ghi mới

INSERT Table1 VALUES(5)

BÀI TẬP 1:



-- Thời điểm t5: Log Backup

BACKUP LOG TEST

TO DISK = 'D:\Backup\Test_TRAN.trn'

-- Thêm một bản ghi mới

INSERT Table1 VALUES(6)

BÀI TẬP 1:



-- Giả sử sau đó xảy ra sự cố, ta mô phỏng sự việc này bằng cách xóa CSDL:

DROP DATABASE TEST

BÀI TẬP 1:



-- Bước 1: Khôi phục từ bản Full Backup

RESTORE DATABASE TEST

FROM DISK = 'D:\backup\Test_FULL.bak'

WITH NORECOVERY

-- Bước 2: Khôi phục từ bản Differential Backup

RESTORE DATABASE TEST

FROM DISK = 'D:\backup\Test_DIFF.bak'

WITH NORECOVERY

BÀI TẬP 1:



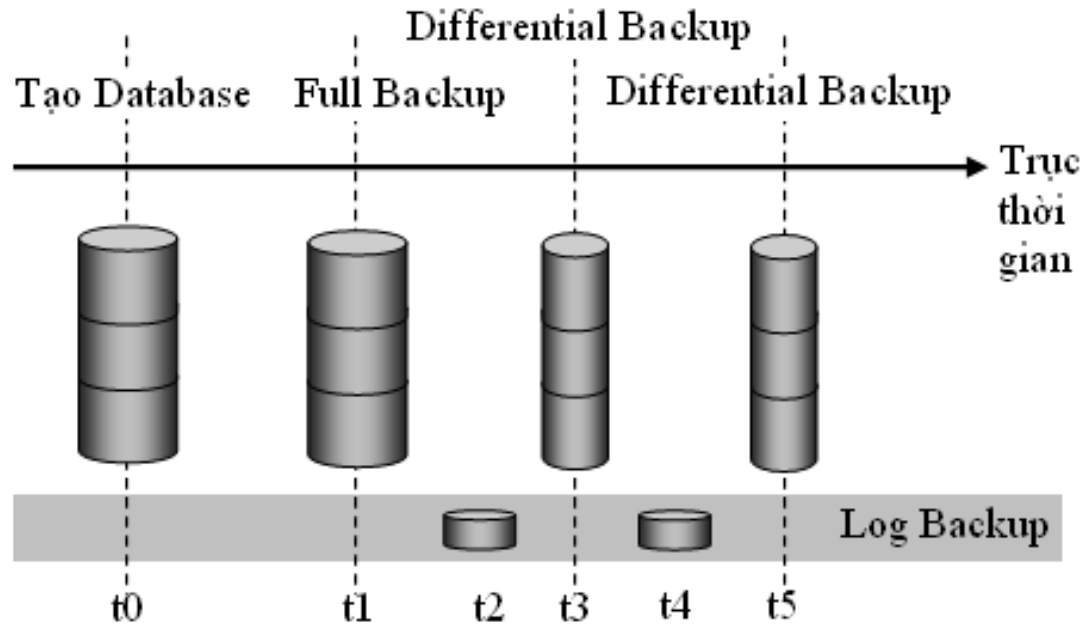
-- Bước 3: Khôi phục từ các bản Log Backup kể từ sau lần Diferential Backup gần nhất

RESTORE DATABASE TEST

FROM DISK = 'D:\backup\Test_TRAN.trn'

WITH FILE = 3

👉 BÀI TẬP 2:



👉 **Lưu ý:** Tại mỗi thời điểm t_i ($i \geq 1$) sinh viên tự thêm dữ liệu để đảm bảo có sự thay đổi dữ liệu trong CSDL. Giả sử sau đó xảy ra sự cố (Ví dụ CSDL bị xóa). Viết lệnh khôi phục lại CSDL từ các bản sao lưu.



QUẢN LÝ NGƯỜI DÙNG VÀ BẢO MẬT HỆ THỐNG

Bảo mật trong SQL gồm 3 lớp:

- + **Login security**: Kiểm soát ai có thể đăng nhập vào SQL Server.
- + **Database access security**: Kiểm soát ai có thể đăng nhập vào Database của SQL server .
- + **Permission security**: Kiểm soát một user có thể thực hiện thao tác gì trên Database.

NỘI DUNG



CÁC CHẾ ĐỘ XÁC THỰC

TÀI KHOẢN ĐĂNG NHẬP (LOGIN ACCOUNT)

TẠO TÀI KHOẢN NGƯỜI DÙNG (USER ACCOUNT)

QUẢN LÝ QUYỀN TRÊN CSDL

1. CÁC CHẾ ĐỘ XÁC THỰC



SQL Server có 2 chế độ xác thực:

+ Xác thực thông qua HDH: Windows Authentication Mode (Windows Authentication)

+ Xác thực hỗn hợp: Mixed Mode (Windows Authentication và SQL Server Authentication)



1. CÁC CHẾ ĐỘ XÁC THỰC



SQL Server có 2 chế độ xác thực:

- **Windows Authentication:** User truy nhập SQL Server phải là những user của Windows quản lý.
- **SQL Server Authentication:** User được quyền khai thác phải là những user do quản trị SQL Server tạo ra. User của Windows không được khai thác.

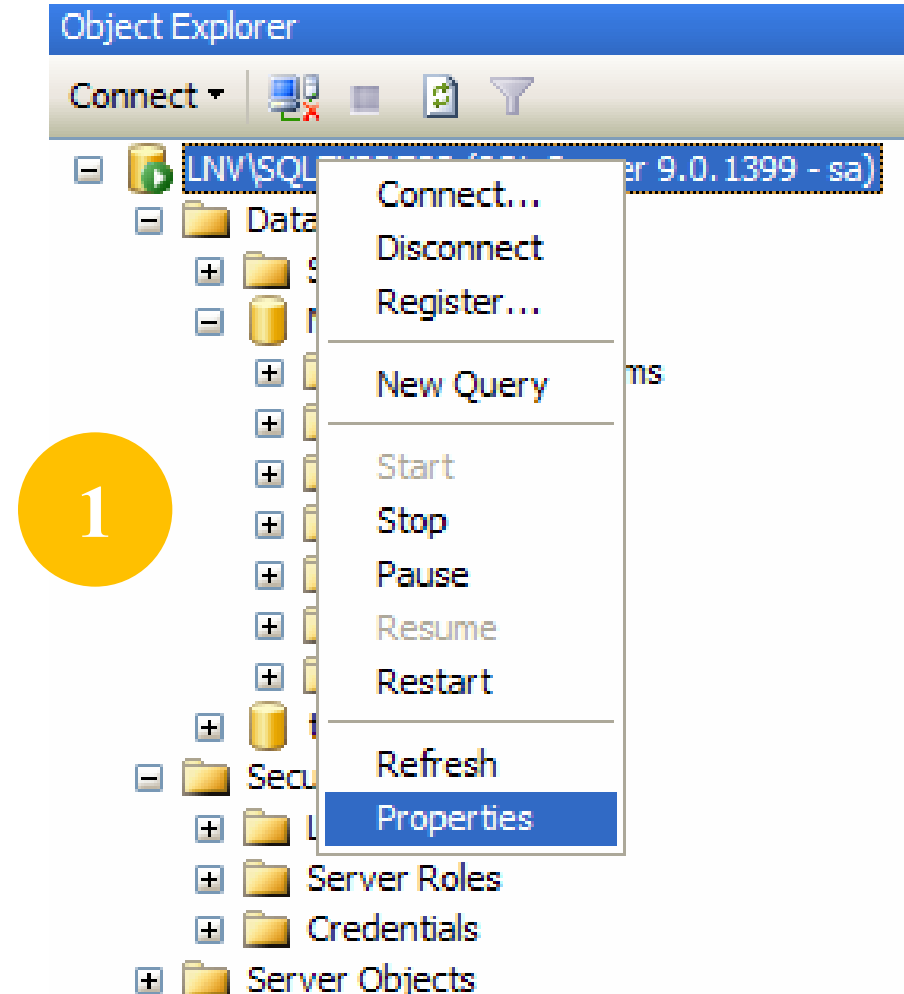
1. CÁC CHẾ ĐỘ XÁC THỰC



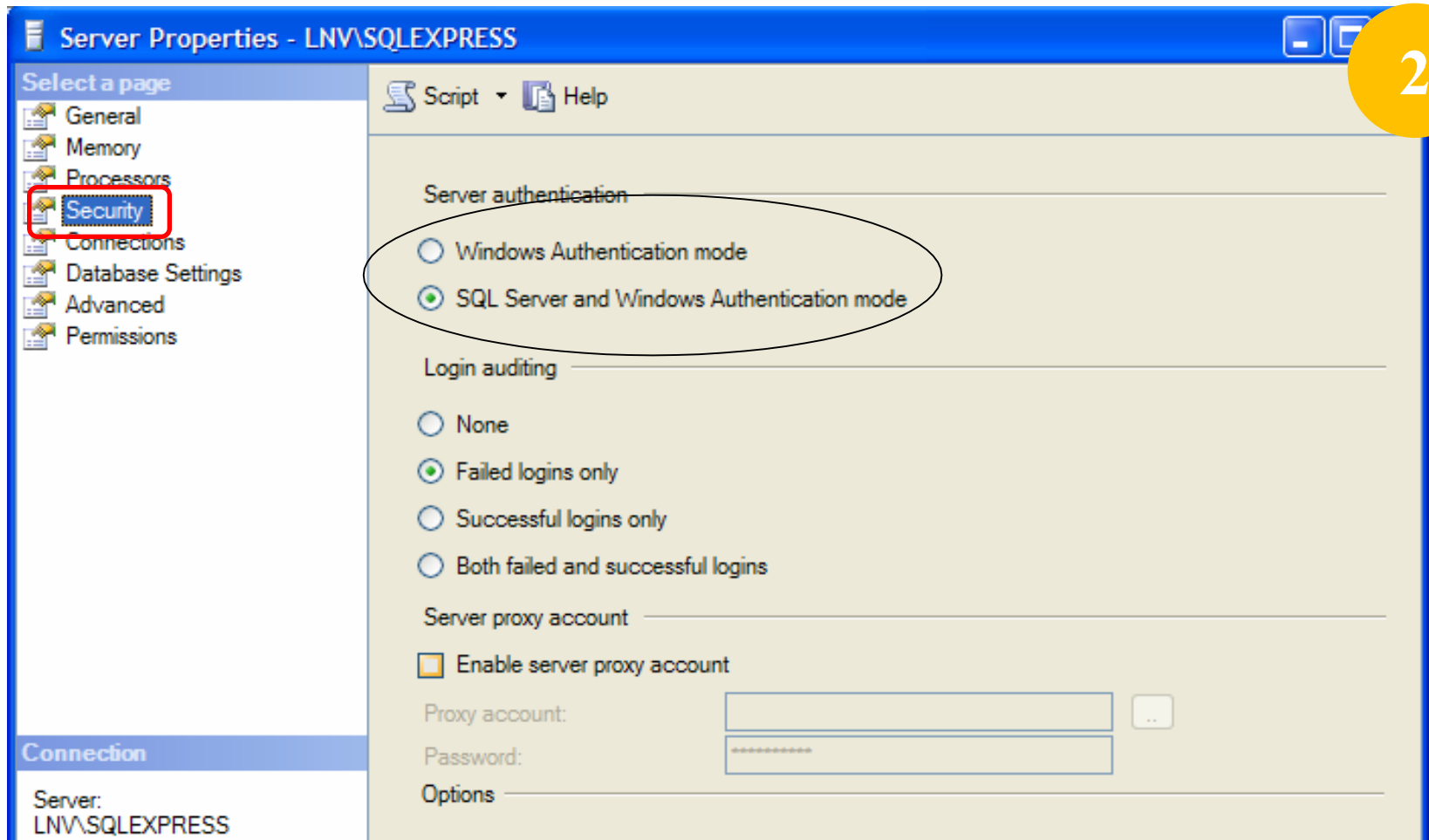
☞ **Lưu ý:** Chế độ xác thực được chọn trong quá trình cài đặt hệ quản trị SQL Server 2008. Muốn thay đổi chế độ xác thực thì thực hiện theo các bước như sau:

1. CÁC CHẾ ĐỘ XÁC THỰC

Trong cửa sổ Object Explorer nhấn chuột phải vào Server → chọn Properties



1. CÁC CHẾ ĐỘ XÁC THỰC



Chọn chế độ xác thực

Tài khoản người dùng:

Người dùng trong SQL Server được chia thành 2 mức:

- Người truy nhập vào SQL Server gọi là **Login account** (Tài khoản đăng nhập).
- Người khai thác CSDL gọi là **User account**.

2. TÀI KHOẢN ĐĂNG NHẬP (Login account)



Là tài khoản người dùng (user) sử dụng để đăng nhập vào hệ thống SQL Server.

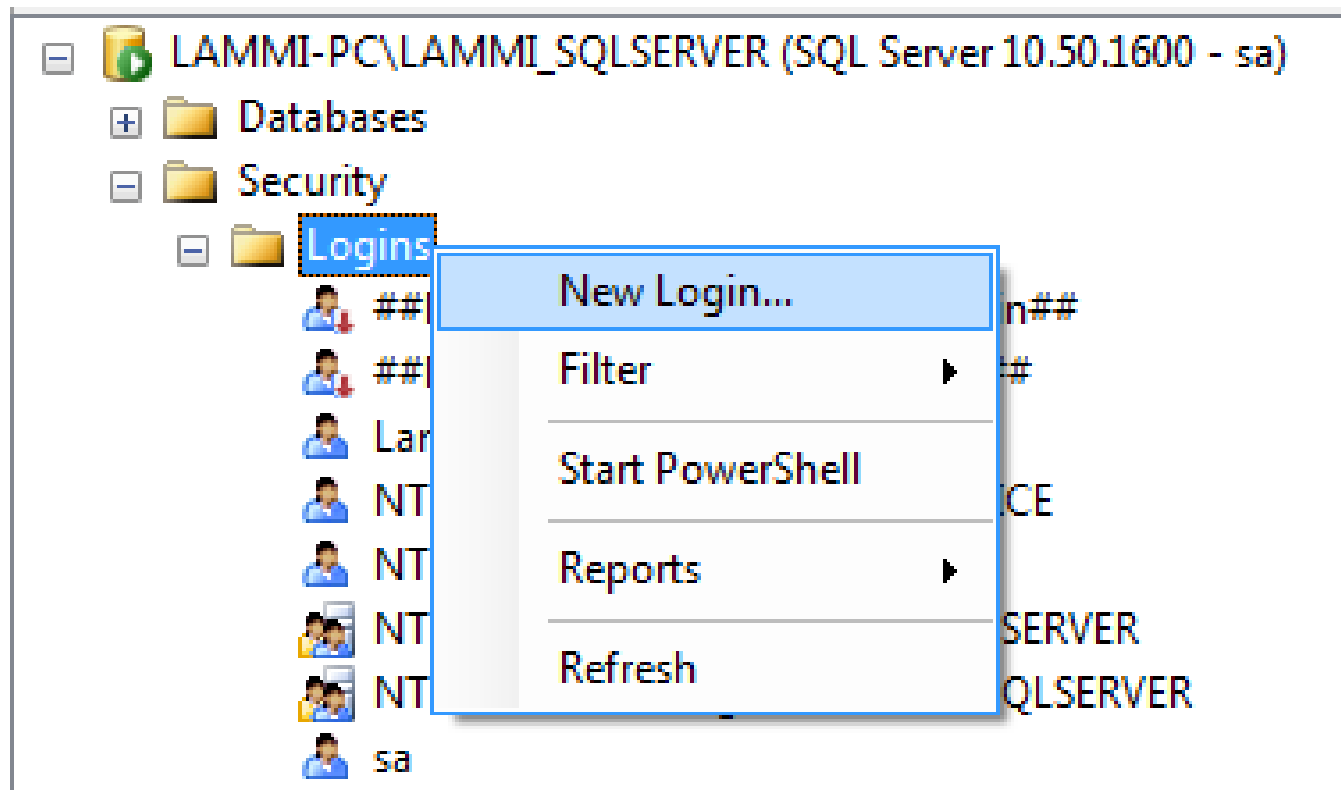
Tùy theo chế độ bảo mật của SQL Server, tài khoản đăng nhập là tài khoản của Windows hay của SQL Server.

2. TÀI KHOẢN ĐĂNG NHẬP (Login account)



Mở rộng danh mục **Security** trong server hiện hành → nhấn chuột phải vào **Logins** → chọn **New login...**

1



2. TÀI KHOẢN ĐĂNG NHẬP (Login account)



Đặt tên **Login name**, chọn quyền xác thực (là Windows authentication hay SQL Server Authentication)

2

Login - New

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Script Help

Login name: mimi Search...

☐ Windows authentication

☒ SQL Server authentication

Password:

Confirm password:

☐ Specify old password

Old password:

☒ Enforce password policy

☒ Enforce password expiration

☒ User must change password at next login

☐ Mapped to certificate

☐ Mapped to asymmetric key

☐ Map to Credential

Add

2. TÀI KHOẢN ĐĂNG NHẬP (Login account)



Chọn Tab **Server Roles**: Chỉ ra nhóm quyền cho đăng nhập mới.

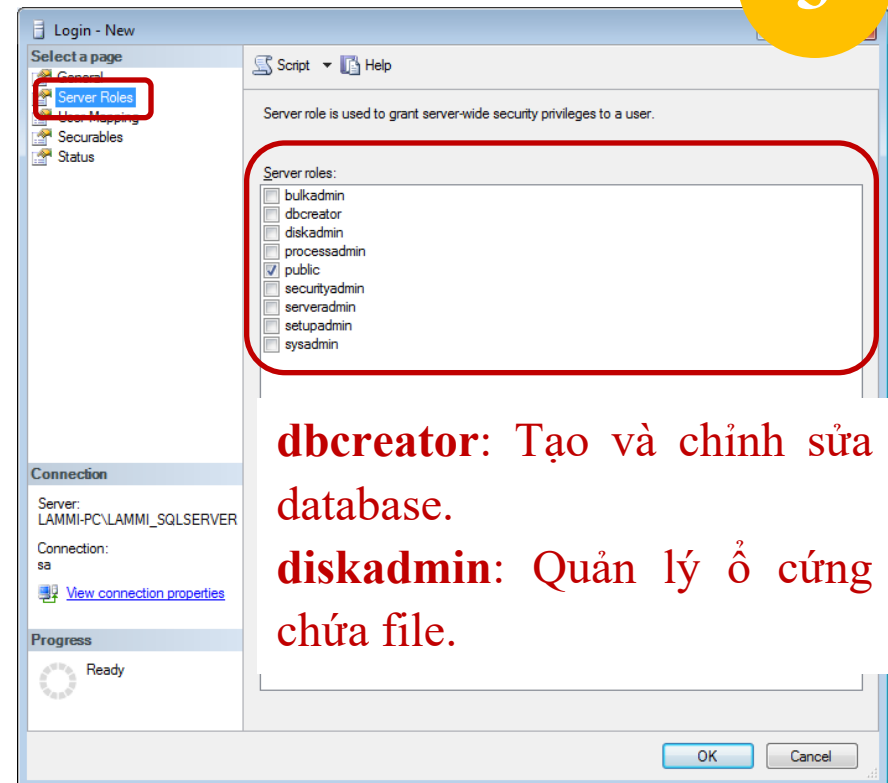
sysadmin: Có thể thực hiện mọi thao tác trên server.

serveradmin: Cấu hình các thiết lập của server.

setupadmin: Cài đặt ứng dụng và quản lý các chính sách.

securityadmin: Có thể quản lý ID và mật khẩu đăng nhập cho server; đồng thời có thể cấp, từ chối và thu hồi quyền trên cơ sở dữ liệu.

processadmin: Quản lý các xử lý của hệ thống.



dbcreator: Tạo và chỉnh sửa database.

diskadmin: Quản lý ổ cứng chứa file.

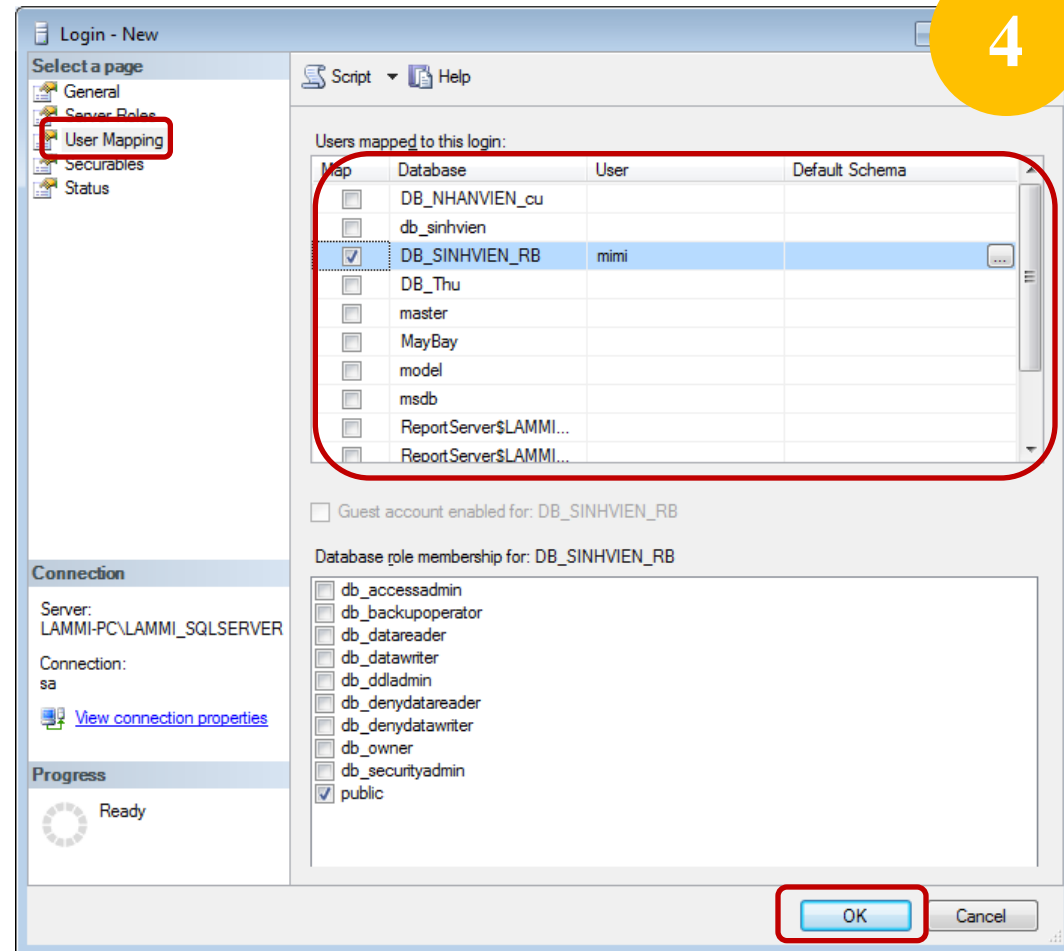
2. TÀI KHOẢN ĐĂNG NHẬP (Login account)



Tab User Mapping:

→ Chỉ định cơ sở dữ liệu mà người dùng được phép truy cập.

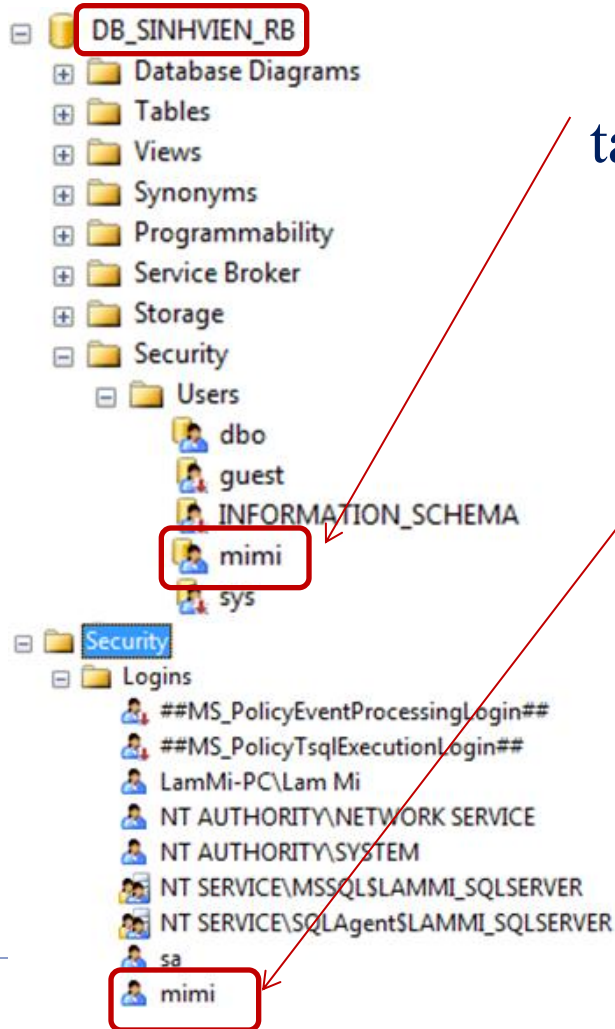
Click **OK** → Kết thúc.



2. TÀI KHOẢN ĐĂNG NHẬP (Login account)



Tạo tài khoản đăng nhập:



tài khoản user **mimi** được tạo ra

tài khoản login **mimi** được tạo ra

2. TÀI KHOẢN ĐĂNG NHẬP (Login account)



❖ Tạo tài khoản đăng nhập: Sử dụng lệnh

Xác thực bằng quyền Windows:

sp_grantlogin <tên_tài_khoản>

(Ví dụ: **sp_grantlogin** 'ADMIN-PC\newT')

Xác thực bằng quyền SQL Server:

sp_addlogin <tên_tài_khoản>, <mật_khẩu>

(Ví dụ: **sp_addlogin** 'tk1','123')

❖ Xóa tài khoản: **sp_droplogin**

<tên_tài_khoản>

3. TÀI KHOẢN NGƯỜI DÙNG (User account)



❖ **Tài khoản người dùng:** Là đối tượng khai thác CSDL, nếu login account chỉ xác định đăng nhập vào SQL Server thì **User account** là đối tượng tham gia khai thác CSDL. **Tài khoản người dùng do CSDL mà nó khai thác quản lý trực tiếp.**

Tài khoản người dùng được tạo từ trong danh mục **Security** của một CSDL cụ thể.

3. TÀI KHOẢN NGƯỜI DÙNG (User account)

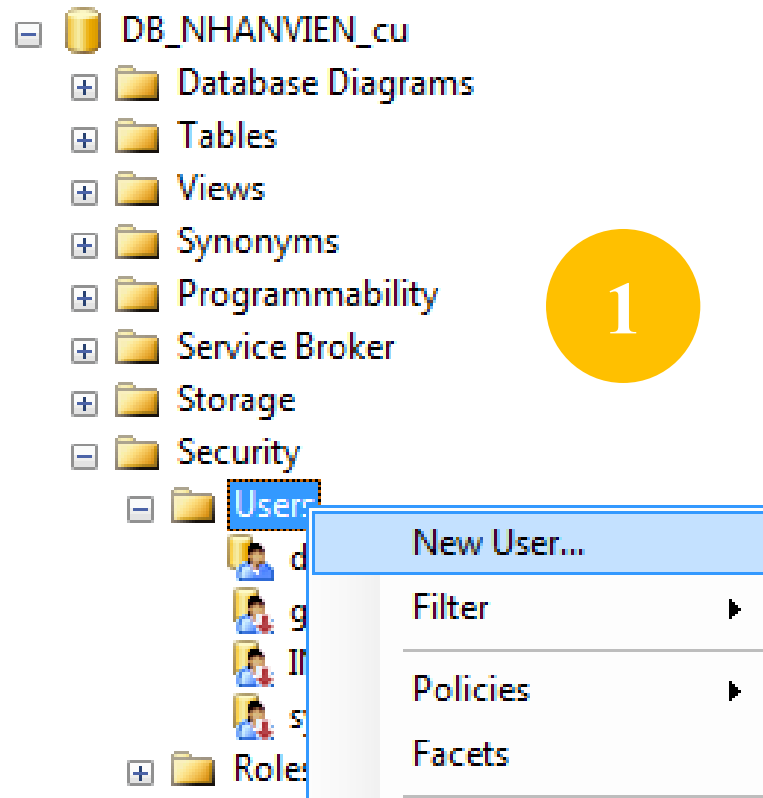


- Sau khi cấp **Login account** để truy cập vào SQL Server, cần cấp cho Login account này quyền là một **user truy cập một/ nhiều Database**.
- Một **Login account** có thể trở thành **user** của nhiều **Database** với những quyền hạn và mang nhiều **user name** khác nhau. Mặc định **user name** trùng tên với **Login account**.

3. TÀI KHOẢN NGƯỜI DÙNG (User account)



Mở rộng mục **Security** trong cơ sở dữ liệu hiện hành → nhấn chuột phải vào đối tượng **Users** → chọn **New User**.






3. TÀI KHOẢN NGƯỜI DÙNG (User account)



Tạo tài khoản người dùng:

Trên cửa sổ Database User gõ tên **user** và chọn một tài khoản đăng nhập tương ứng.

Select a page

-  General
-  Securables
-  Extended Properties

 Script  Help

User name:

user1

☒ Login name:

mimi

☐ Certificate name:☐ Key name:☐ Without login

Default schema:

Schemas owned by this user:

Owned Schemas

- ☒ db_accessadmin
- ☐ db_backupoperator
- ☐ db_datareader
- ☐ db_datawriter
- ☐ db_ddladmin
- ☐ db_denydatareader
- ☐ db_denydatawriter

Database role membership:

Role Members

- ☒ db_accessadmin
- ☐ db_backupoperator
- ☐ db_datareader
- ☐ db_datawriter
- ☐ db_ddladmin

2

Connection

Server:
LAMMI-PC\LAMMI_SQLSERVERConnection:
sa [View connection properties](#)

Progress

3. TÀI KHOẢN NGƯỜI DÙNG (User account)



Tạo tài khoản người dùng: Chọn CSDL → Dùng lệnh

sp_adduser <tên_đăng_nhập>, <tên_người_dùng>

Ví dụ: **sp_adduser** 'tk1','user1'

- Tên_đăng_nhập: Là tên tài khoản đăng nhập đã được tạo trước đó.
- Tên_người_dùng: Là tên người dùng cần tạo. Mặc định cùng tên với tên đăng nhập.

Hủy tài khoản người dùng:

sp_dropuser <tên_tài_khoản_người_dùng>

4. QUẢN LÝ QUYỀN TRÊN CSDL



Quyền trên cơ sở dữ liệu cho phép người dùng thực hiện các hành động trong Database nhằm mục đích tránh sự sửa đổi dữ liệu của một số người dùng không được phép.

Có 2 loại quyền:

- Quyền thao tác trên các đối tượng cơ sở dữ liệu
- Quyền định nghĩa các đối tượng trong cơ sở dữ liệu.

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng (Object Permission)

Kiểm soát một **User** có thể thực thi hành động gì trên một đối tượng cụ thể trong CSDL.

DELETE	table , view
SELECT	table, view, và column
INSERT	table , view
EXECUTE	stored procedure
UPDATE	table, view, và column

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Có 3 hình thức:

- **Grant:** Cấp quyền
- **With Grant:** Cho phép người dùng hoặc nhóm cấp quyền cho người dùng hoặc nhóm khác
- **Revoke:** Thu hồi quyền

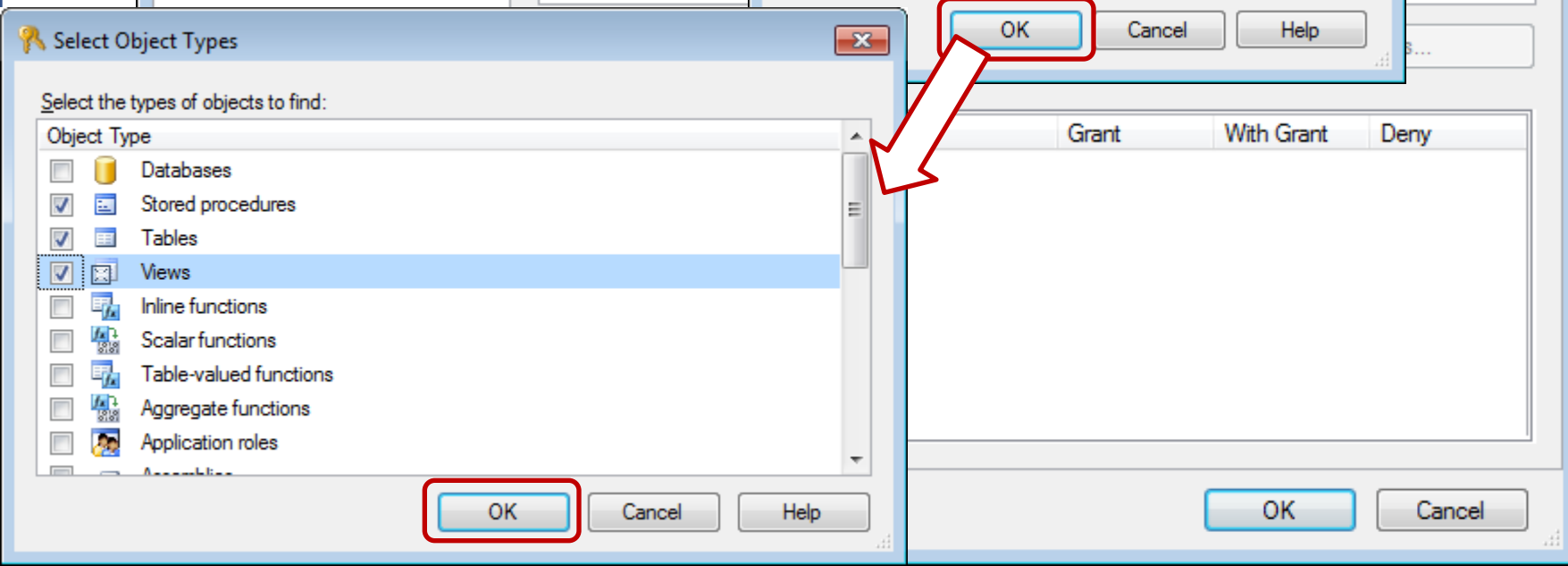
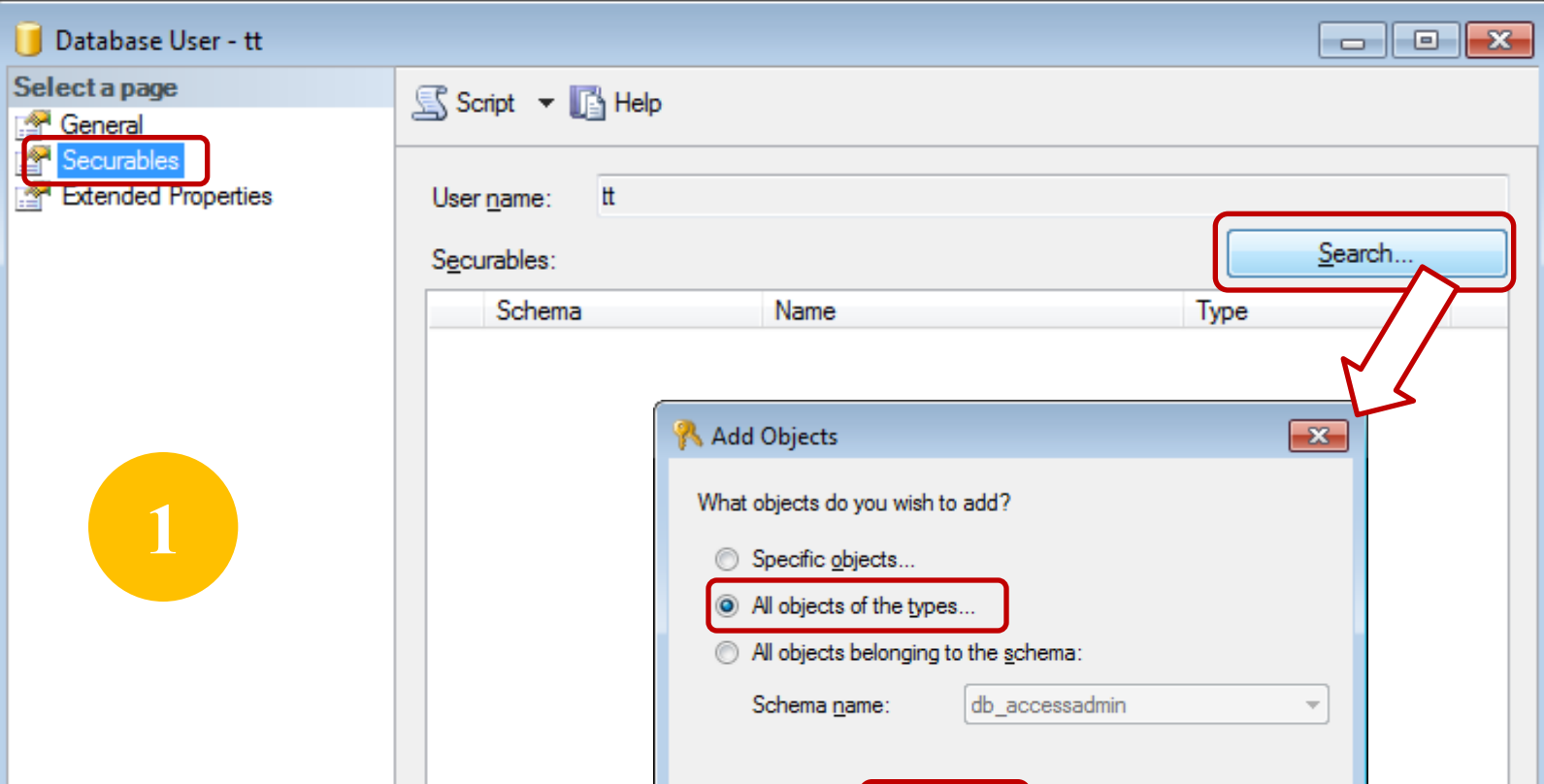
4. QUẢN LÝ QUYỀN TRÊN CSDL



☐ Quyền thao tác trên đối tượng

Để thực hiện cấp quyền, nhấn chuột phải vào tên **user**

> chọn **Properties** xuất hiện cửa sổ như sau:





Select a page

- General
- Securables
- Extended Properties

Connection

Server:
ACER\LAMMI_SQLSERVER

Connection:
sa

 [View connection properties](#)

Progress









Ready

Database User - moi

 Script  Help

User name: moi

Securables:

	Schema	Name	Type
	dbo	KETQUA	Table
	dbo	LOP	Table
	dbo	MONHOC	Table
	dbo	SINHVIEN	Table
	dbo	sysdiagrams	Table
	dbo	Table_1	Table

Permissions for dbo.KETQUA:

Explicit Effective

Permission	Grantor	Grant	With Grant	Deny
Alter	dbo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Control	dbo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Delete	dbo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Insert	dbo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
References	dbo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Select	dbo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

OK

Cancel

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Sử dụng lệnh “**Grant...**” để cấp quyền thao tác cho user:

```
GRANT các_quyền_cấp_phát [(danh_sách_cột)] | <ALL>  
ON <tên_đối_tượng>  
TO <danh_sách_người_dùng>  
[WITH GRANT OPTION ]
```

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Trong đó:

- **các_quyền_cấp_phát:** SELECT, INSERT, UPDATE, DELETE, lệnh thực thi thủ tục là EXEC.
- **tên_đối_tượng:** có thể là table, view hay procedure.

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Ví dụ 1: Cấp phát cho người dùng có tên **user1** quyền thực thi các câu lệnh SELECT, INSERT và UPDATE trên bảng LOP.

GRANT SELECT, INSERT, UPDATE

ON LOP

TO user1

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Ví dụ 2: Cho phép người dùng **user1** quyền xem họ tên và ngày sinh của các sinh viên (cột HOTEN và NGAYSINH của bảng SINHVIEN)

GRANT SELECT (hoten, ngaysinh)

ON sinhvien

TO user1

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Ví dụ 2: Cho phép người dùng **user1** quyền xem họ tên và ngày sinh của các sinh viên (cột HOTEN và NGAYSINH của bảng SINHVIEN)

hoặc

GRANT SELECT

ON sinhvien (hoten, ngaysinh)

TO user1

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Ví dụ 3: Cấp phát cho người dùng **user1** tất cả các quyền trên bảng SINHVIEN

GRANT ALL

ON SINHVIEN

TO user1

4. QUẢN LÝ QUYỀN TRÊN CSDL



☐ Quyền thao tác trên đối tượng

Tùy chọn **WITH GRANT OPTION** trong câu lệnh **GRANT** cho phép người đó chuyển tiếp quyền được cấp cho người dùng khác.

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Ví dụ 4: Cho phép người dùng **user1** quyền xem dữ liệu trên bảng SINHVIEN đồng thời có thể chuyển tiếp quyền này cho người dùng khác.

```
GRANT SELECT  
ON SINHVIEN  
TO user1  
WITH GRANT OPTION
```


4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Để thu hồi quyền đã cấp, sử dụng lệnh REVOKE.

Cú pháp:

REVOKE <các_quyền_cấp_phát>[(danh_sách_cột)] | <ALL>

ON <tên_đối_tượng>

FROM <danh_sách_người_dùng>

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Ví dụ 5: Thu hồi quyền thực thi lệnh INSERT trên bảng LOP đối với người dùng **user1**.

REVOKE INSERT

ON LOP

FROM user1

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền thao tác trên đối tượng

Ví dụ 6: **REVOKE** **SELECT**

ON sinhvien (ngaysinh)

FROM user1

→ Chỉ thu hồi quyền **SELECT** trên cột ngày sinh của bảng **SINHVIEN**.

→ Những quyền mà người dùng được cấp bởi những người dùng khác thì vẫn còn có hiệu lực.

Ví dụ: A thực hiện lệnh sau để cấp phát quyền xem dữ liệu trên bảng SINHVIEN cho C:

```
GRANT SELECT  
ON SINHVIEN  
TO C
```

B cấp phát quyền xem và thêm dữ liệu trên bảng SINHVIEN cho C bằng câu lệnh:

```
GRANT SELECT, INSERT  
ON SINHVIEN  
TO C
```

B thu hồi quyền đã cấp:

```
REVOKE SELECT, INSERT  
ON SINHVIEN  
FROM C
```

Kết quả: **C vẫn còn quyền xem dữ liệu trên bảng SINHVIEN do A cấp.**

Nếu cấp phát quyền cho người dùng nào đó bằng câu lệnh GRANT với tùy chọn **WITH GRANT OPTION** thì khi thu hồi quyền bằng câu lệnh REVOKE phải chỉ định tùy chọn **CASCADE**.

Ví dụ : Cấp phát cho người dùng A trên bảng SINHVIEN

```
GRANT SELECT  
ON SINHVIEN  
TO A  
WITH GRANT OPTION
```

Sau đó người dùng A lại cấp phát cho người dùng B quyền xem dữ liệu trên SINHVIEN với câu lệnh:

```
GRANT SELECT  
ON SINHVIEN  
TO B
```

Nếu muốn thu hồi quyền đã cấp phát cho người dùng A, ta sử dụng câu lệnh REVOKE như sau:

```
REVOKE SELECT  
ON SINHVIEN  
FROM A  
CASCADE
```

Kết quả:

- + Câu lệnh trên sẽ thu hồi quyền SELECT của A
- + **Đồng thời cũng thu hồi luôn quyền chuyển tiếp mà người dùng A đã cấp cho người dùng B.**

Thu hồi quyền chuyển tiếp mà không thu hồi quyền **select** trên bảng SINHVIEN của người dùng A

```
REVOKE GRANT OPTION FOR SELECT
```

```
ON SINHVIEN
```

```
FROM A
```

```
CASCADE
```

Kết quả:

+Người dùng B sẽ không còn quyền xem dữ liệu trên bảng SINHVIEN

+Người dùng A không thể chuyển tiếp quyền đã được cấp phát cho những người dùng khác (tuy nhiên A vẫn còn quyền xem dữ liệu trên bảng SINHVIEN).

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Quyền định nghĩa đối tượng

Quyền định nghĩa các đối tượng cho phép người dùng tạo ra các đối tượng cơ sở dữ liệu.

Các quyền này thường là: **CREATE DATABASE, Create Table, Create View, Create Proc, Create Function, Backup Database, Backup Log.**

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Cấp quyền định nghĩa đối tượng

GRANT ALL | *< danh_sách_câu_lệnh >*

TO *< danh_sách_người_dùng >*

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Cấp quyền định nghĩa đối tượng

Ví dụ: Để cấp phát quyền tạo bảng và khung nhìn cho người dùng có tên là **user1**, ta sử dụng câu lệnh như sau:

GRANT CREATE TABLE, CREATE VIEW
TO user1

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Cấp quyền định nghĩa đối tượng

Chú ý:

- Đối tượng cơ sở dữ liệu do người dùng nào tạo ra sẽ do người đó sở hữu và do đó người này có quyền cho người dùng khác sử dụng đối tượng và cũng có thể xóa bỏ (DROP) đối tượng do mình tạo ra.
- Không thể sử dụng tùy chọn WITH GRANT OPTION, tức là người dùng không thể chuyển tiếp được các quyền thực thi các câu lệnh đã được cấp phát.

4. QUẢN LÝ QUYỀN TRÊN CSDL



❑ Thu hồi quyền định nghĩa đối tượng

REVOKE <các_câu_lệnh_cần_thu_hồi> | All
FROM <danh_sách_người_dùng>

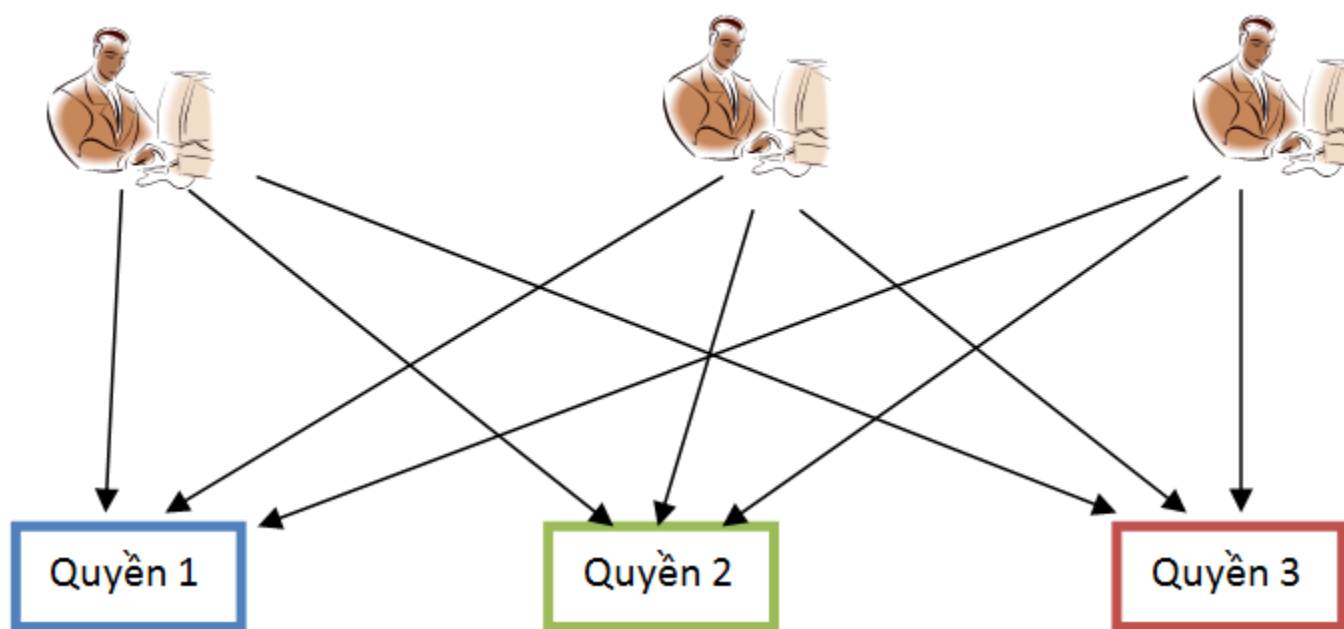
Ví dụ:

REVOKE CREATE TABLE
FROM user1

QUẢN LÝ THÔNG QUA NHÓM QUYỀN



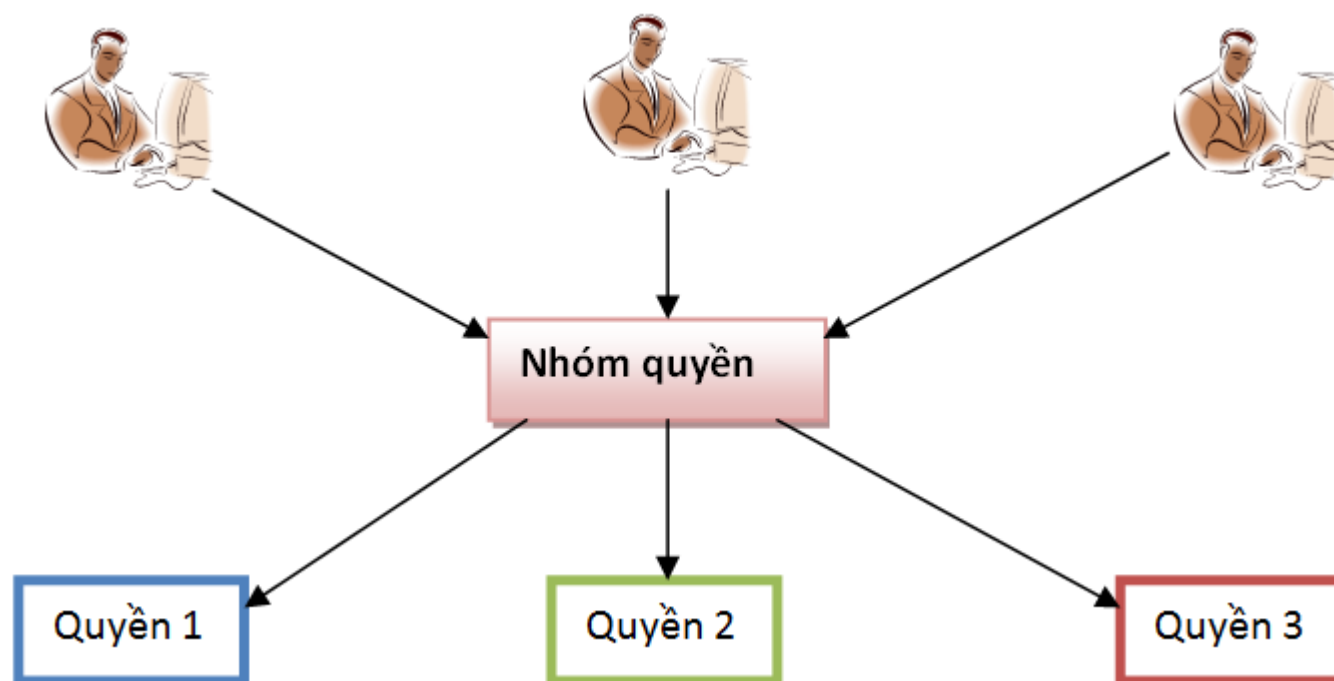
Mô hình quyền thông thường:



QUẢN LÝ THÔNG QUA NHÓM QUYỀN (tt)



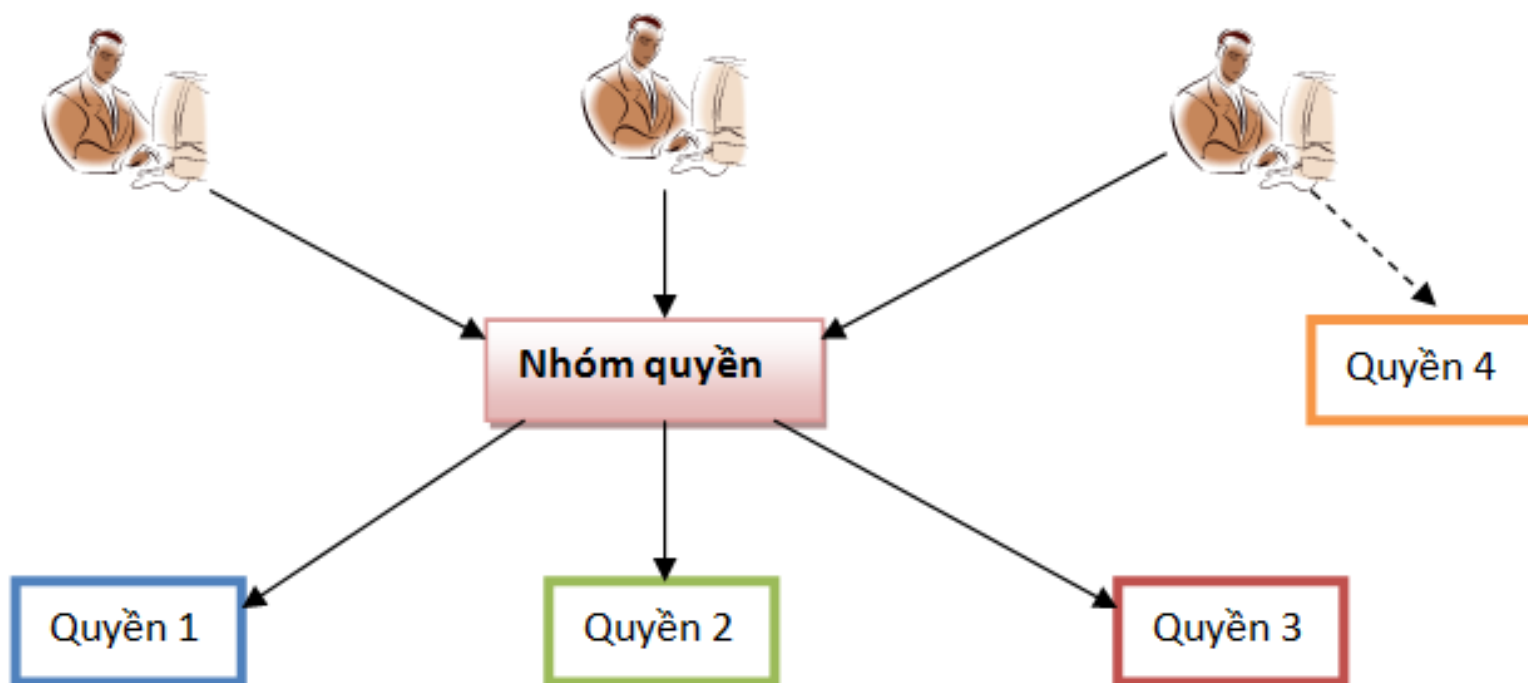
Mô hình cấp quyền dựa trên nhóm:



QUẢN LÝ THÔNG QUA NHÓM QUYỀN (tt)



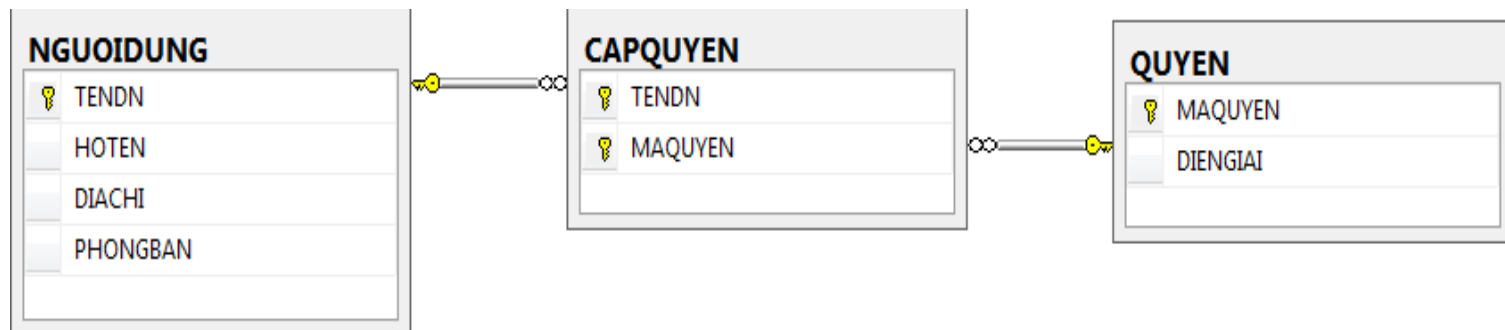
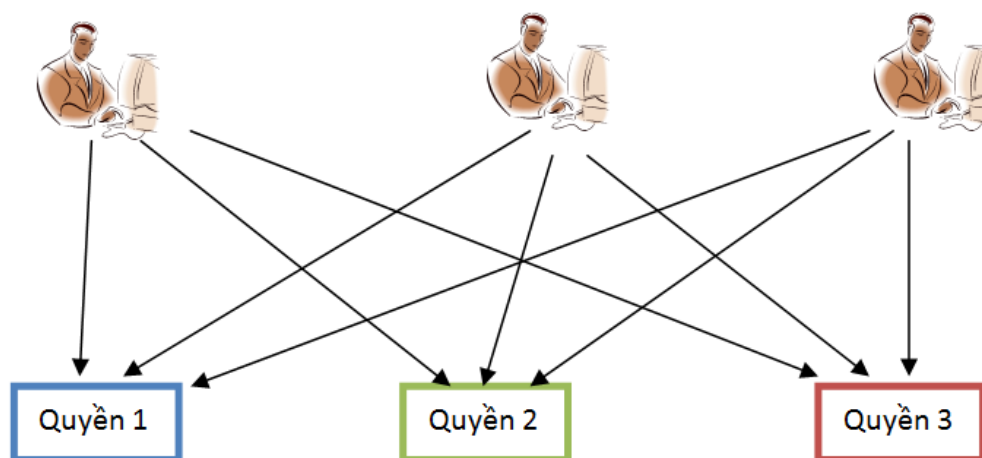
Mô hình cấp quyền kết hợp:



Mô hình cấp quyền – Thiết kế database

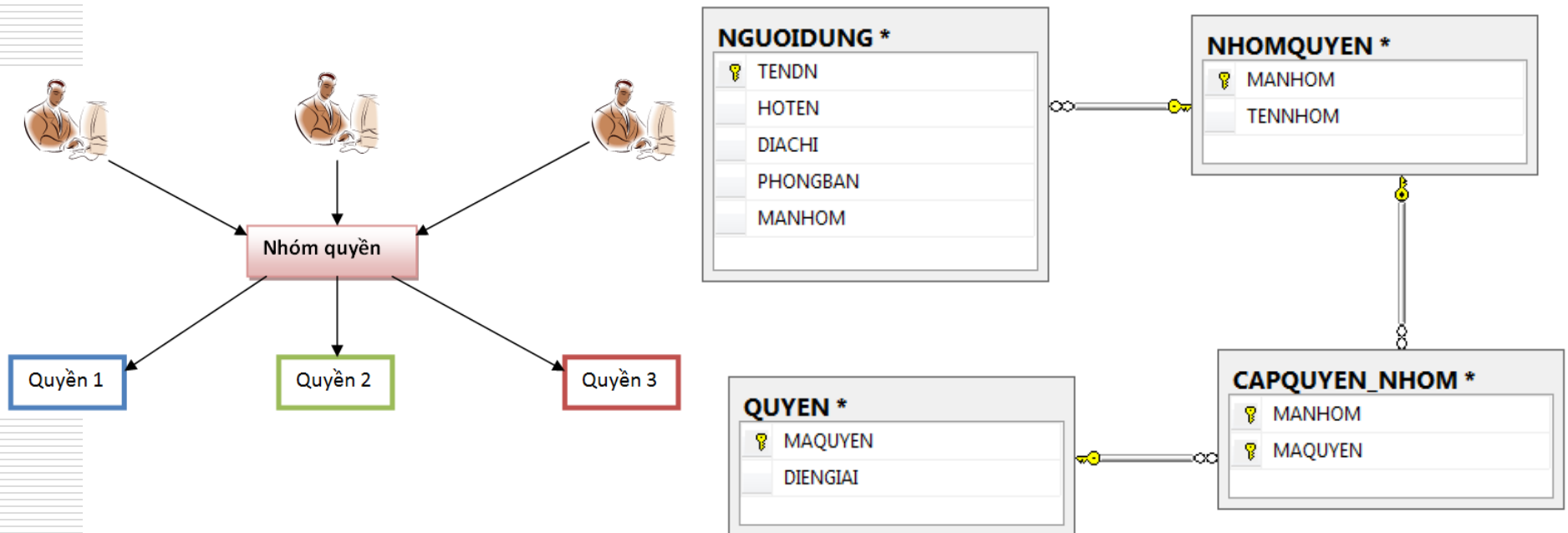


❖ Cấp quyền trực tiếp cho người dùng



Mô hình cấp quyền – Thiết kế database

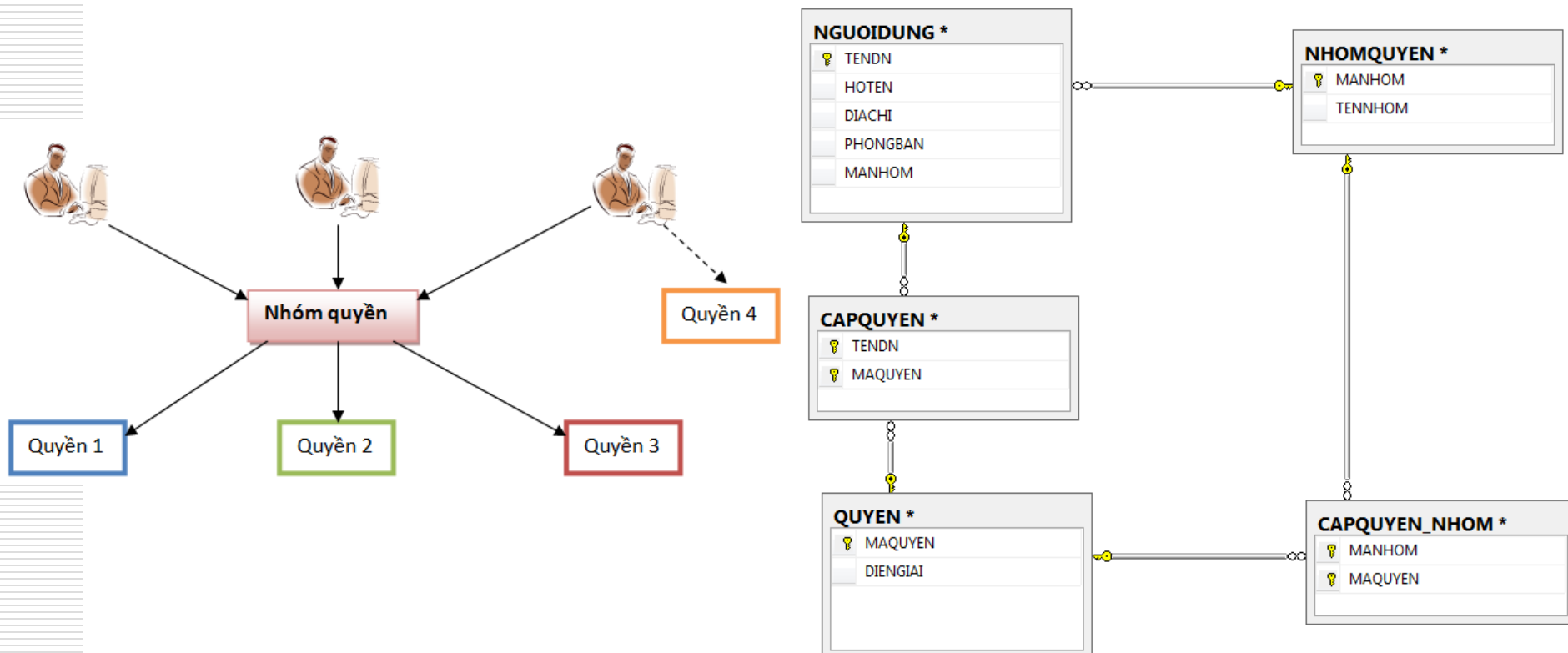
❖ Cấp quyền thông qua nhóm quyền



Mô hình cấp quyền – Thiết kế database



❖ Cấp quyền kết hợp



4. QUẢN LÝ QUYỀN TRÊN CSDL



QUẢN LÝ THÔNG QUA NHÓM QUYỀN

Quản lý nhóm quyền bằng lệnh T-SQL như sau:

sp_addrole <tên_nhóm_quyền>

Ví dụ: Tạo nhóm quyền xem_dulieu

sp_addrole 'Xem_dulieu'

4. QUẢN LÝ QUYỀN TRÊN CSDL



QUẢN LÝ THÔNG QUA NHÓM QUYỀN

Thêm quyền vào nhóm quyền:

GRANT <lệnh cập nhật hay thực thi>| <ALL>
ON <tên đối tượng>
TO <tên nhóm quyền>

Ví dụ: Gán các quyền SELECT, INSERT, UPDATE trên bảng SINHVIEN cho nhóm quyền Xem_dulieu.

GRANT SELECT, INSERT, UPDATE
ON SINHVIEN
TO Xem_dulieu

4. QUẢN LÝ QUYỀN TRÊN CSDL



QUẢN LÝ THÔNG QUA NHÓM QUYỀN

Thu hồi quyền của nhóm quyền:

REVOKE <lệnh cập nhật hay thực thi thủ tục>| <ALL>
ON <tên đối tượng>
FROM <tên nhóm quyền>

Ví dụ: Thu hồi các quyền SELECT trên bảng SINHVIEN từ nhóm quyền Xem_dulieu.

REVOKE SELECT
ON SINHVIEN
FROM Xem_dulieu

Thêm người dùng vào nhóm quyền:

sp_addrolemember <tên nhóm quyền>, <tên người dùng>

Ví dụ: Thêm người dùng **user1** vào nhóm quyền Xem_dulieu.

sp_addrolemember 'Xem_dulieu', 'user1'

Xoá người dùng khỏi nhóm quyền:

sp_droprolemember <tên nhóm quyền>, <tên người dùng>

Ví dụ: Xoá người dùng **user1** khỏi nhóm quyền Xem_dulieu

sp_droprolemember 'xem_dulieu','user1'

Huỷ nhóm quyền:

Cú pháp: **sp_droprole** 'tên_nhóm_quyền'

Ví dụ: Huỷ bỏ nhóm quyền có tên Xem_dulieu

sp_droprole Xem_dulieu

BÀI TẬP



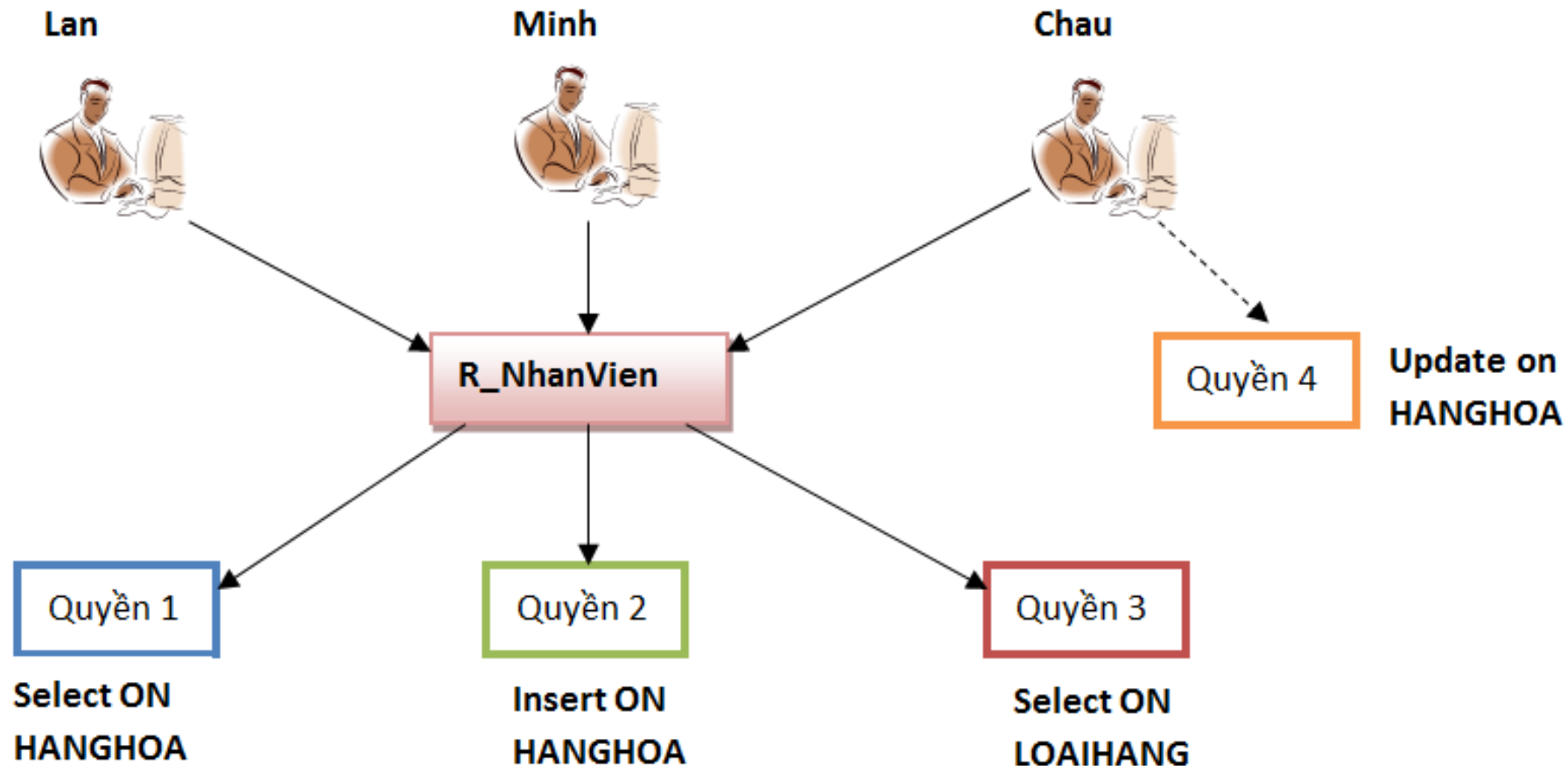
❖ Làm nhóm:

Chọn một hệ thống quen thuộc, thực hiện các yêu cầu sau:

- Thiết kế các chức năng(quyền) của hệ thống
- Thiết kế các nhóm người dung
- Mỗi chức năng thiết kế các roles trong SQL Server
- Mỗi role có những quyền gì? Trên những đối tượng nào (vd: bảng, view, store procedure,..)

Ví dụ sử dụng nhóm quyền

Tạo người dùng, nhóm quyền và cấp quyền như hình, đăng nhập và kiểm thử quyền.



Ví dụ sử dụng nhóm quyền

❖ Cho cơ sở dữ liệu gồm các bảng sau:

NHANVIEN, KHACH, HOADON, CHITIETHD,
HANGHOA, NHACC, PHIEUNHAP, CHITIETPN

- a/ Tạo 2 nhóm quyền có tên là nhaphang và banhang.
- b/ Gán quyền thao tác phù hợp cho các nhóm này.
- c/ Tạo 2 người dùng, mỗi người dùng gán vào một nhóm. Đăng nhập bằng người dùng vừa tạo và kiểm tra quyền hạn được cấp.



Thank You !