**ĐẠI HỌC UEH**

**TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ**

**KHOA CÔNG NGHỆ THÔNG TIN KINH DOANH**



**ĐỒ ÁN CUỐI KỲ**

**CƠ SỞ DỮ LIỆU**

**ĐỀ BÀI  
XÂY DỰNG MÔ HÌNH CƠ SỞ DỮ LIỆU VỀ ĐẶT  
VÀ CHO THUÊ PHÒNG TP ĐÀ LẠT**

Giảng viên: Thạc sĩ Viên Thanh Nhã

Mã lớp học phần: 22C1IFN50900603

Họ và tên: Nguyễn Trịnh Hiếu Kiên - 31211027199

Lưu Quang Huy - 31211027642

Khóa – Lớp: DS002 K47

TP Hồ Chí Minh, ngày 5 tháng 11 năm 2022

MỤC LỤC

[**I.** **KHẢO SÁT HIỆN TRẠNG** 4](#_Toc119070511)

[1.Tống quan 4](#_Toc119070512)

[2.Mục tiêu quản lý 4](#_Toc119070513)

[**II. PHÂN TÍCH & THIẾT KẾ CSDL:** 4](#_Toc119070514)

[*1. Xác định thực thể & thuộc tính để hình thành thực thể.* 4](#_Toc119070515)

[*2. Xác định mối quan hệ giữa các thực thể* 5](#_Toc119070516)

[*3. Xây dựng mô hình thực thể kết hợp (ERD).* 8](#_Toc119070517)

[*4. Chuẩn hóa mô hình* 8](#_Toc119070518)

[*5. Xây dựng mô hình quan hệ.* 9](#_Toc119070519)

[*6. Xây dựng các phép toán đại số quan hệ* 9](#_Toc119070520)

[a. Phép chọn 9](#_Toc119070521)

[b. Phép chiếu 12](#_Toc119070522)

[c. Phép giao 14](#_Toc119070523)

[d. Phép hội 16](#_Toc119070524)

[e. Phép trừ 18](#_Toc119070525)

[f. Phép Tích Decac 20](#_Toc119070526)

[g. Phép kết nối 22](#_Toc119070527)

[h. Phép gom nhóm 24](#_Toc119070528)

[i. Phép gom nhóm có điều kiện 25](#_Toc119070529)

[7. Các lệnh mô tả dữ liệu (Data Definition Language ) 26](#_Toc119070530)

[a. Tạo database 26](#_Toc119070531)

[b. Tạo các table 26](#_Toc119070532)

[c. Thay đổi cấu trúc bảng 29](#_Toc119070533)

[8. Các lệnh thao tác dữ liệu 30](#_Toc119070534)

[a. Thêm dữ liệu vào các bảng bằng lệnh. 30](#_Toc119070535)

[b. Update dữ liệu vào các table bằng lệnh SQL 32](#_Toc119070536)

[c. Xóa dữ liệu ở các bảng table bằng lệnh SQL 32](#_Toc119070537)

[9. Lệnh truy vấn dữ liệu 33](#_Toc119070538)

[a. Truy vấn 1 bảng 33](#_Toc119070539)

[b. Truy vấn nhiều bảng (Phép kết) 35](#_Toc119070540)

[c. Truy vấn có điều kiện 37](#_Toc119070541)

[d. Truy vấn tính toán 38](#_Toc119070542)

[e. Truy vấn có gom nhóm 40](#_Toc119070543)

[f. Truy vấn gom nhóm có điều kiện (Having) 41](#_Toc119070544)

[g. Truy vấn có sử dụng phép giao, hội, trừ 41](#_Toc119070545)

[h. Truy vấn con (Subquery) 43](#_Toc119070546)

[i. Truy vấn chéo (Pivot table) 45](#_Toc119070547)

[10.Viết store procedure và function 46](#_Toc119070548)

[11.Viết trigger ràng buộc dữ liệu cho các bảng 50](#_Toc119070549)

[12.Phân quyền 55](#_Toc119070550)

[13.Sao lưu dữ liệu 61](#_Toc119070551)

1. **KHẢO SÁT HIỆN TRẠNG**

## 1.Tống quan

Ngành du lịch là một ngành có từ rất lâu trên thế giới. Tại các nước phát triển, ngành du lịch phát triển hết sức mạnh mẽ và có lợi nhuận cao với sự hỗ trợ của công nghệ thông tin trong tổ chức và hoạt động. Sự đóng góp của công nghệ là một công cụ đắc lực để hỗ trợ cho tiến trình của lĩnh vực du lịch. Chính vì thế, chúng em chọn chủ đề:

**Xây dựng mô hình cơ sở dữ liệu về đặt và thuê phòng TP Đà Lạt**

## 2.Mục tiêu quản lý

Xây dựng cơ sở dữ liệu nhằm quản lý việc đặt và thuê phòng.Cơ sở dữ liệu bao gồm tất cả các thông tin về các khách sạn và khách đến thuê phòng.

Ý nghĩa : Cơ sở dữ liệu được xây dựng nhằm phục vụ nhu cầu thông tin cho người quản lý của khách sạn. Nhằm để quản lý rõ ràng, hướng đến việc quản lý và phục vụ cho việc phân tích một cách thông minh nhờ áp dụng công nghệ.

# **II. PHÂN TÍCH & THIẾT KẾ CSDL:**

## *1. Xác định thực thể & thuộc tính để hình thành thực thể.*

1.1 Hotel

* Bao gồm các thông tin: ID , name(tên khách sạn), address (địa chỉ), manager\_id ( mã người quản lý)
* **Hotel (ID, name, address, manager\_id)**

1.2 Room (Phòng)

* Bao gồm các thông tin: ID(id phòng), hotel id(mã khách sạn), number( số phòng), status (trạng thái ), smoke, room type of ( loại phòng)
* **Room ( id, hotel\_id, number, name, status, smoke, room\_type\_of)**

1.3 Room\_type ( Loại phòng)

* Bao gồm các thông tin: ID( id phòng), description ( miêu tả, ghi chú cho phòng) , max\_capacity ( số người ở tối đa)
* **Room\_type (id, description, max\_capacity)**

1.4 Occupied room ( phòng có khách lưu trú)

* Bao gồm các thông tin: ID(id phòng), check in ( ngày nhận), check out(ngày trả), reservation id ( mã đặt phòng)
* **Occupied\_room( id, check\_in, check\_out, reservation\_id)**

1.5 Hosted\_at ( Thông tin lưu trú)

* Bao gồm các thông tin: ID ( mã lưu trú ), guest id ( mã khách), occupied room id( mã phòng lưu trú)
* **Hosted\_at ( ID, guest\_id, occupied\_room\_id)**

1.6 Guest(Khách hàng).

* Bao gồm các thông tin: ID ( mã KH ) , first name( tên ) , last name ( họ ), phone number ( số điện thoại )
* **Guest(ID, first\_name, last\_name,phone\_number)**

1.7 Reservation ( Đặt phòng )

* Bao gồm các thông tin: ID(mã đặt), date\_in ( ngày vào ) , date\_out(ngày ra), guest id ( mã khách ), deposit( đặt cọc)
* **Reservation(ID, date\_in, date\_out, guest\_id, deposit)**

1.8 Staff ( Nhân viên )

* Bao gồm các thông tin: ID(mã nhân viên),hotel id(mã khách sạn),first name (tên),middle name(tên lót), last name(họ), birthday(ngày sinh nhật), manger\_id(mã người quản lý), salary(lương).
* **Staff(ID, hotel\_id, level, first\_name, middle\_name,last\_name, birthday, manager\_id, salary)**

1.9 Reserved room ( Phòng đã được đặt trước )

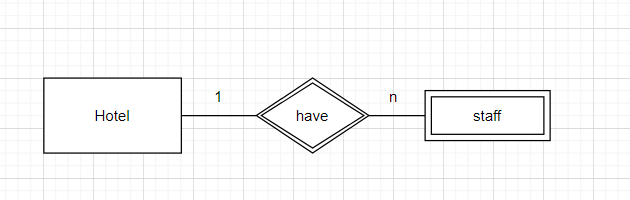
- Bao gồm các thông tin : ID( mã phòng đặt trước ), number of rooms ( số lượng phòng), room type id ( mã của loại phòng ), reservation id ( mã đặt trước), status ( tình trạng)

* **Reserved\_room ( ID, number\_of\_rooms, room\_type\_id, reservation\_id, status)**

## *2. Xác định mối quan hệ giữa các thực thể*

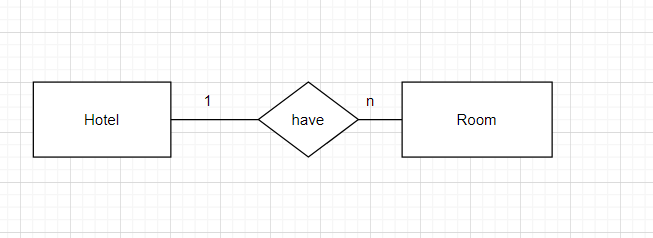
2.1 Mối quan hệ giữa Hotel và Staff

Mỗi Hotel có nhiều Staff, và mỗi Staff chỉ thuộc về 1 Hotel



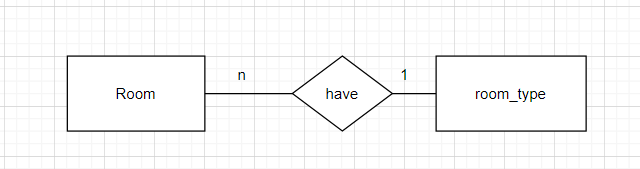
2.2 Mối quan hệ giữa Hotel và Room

Mỗi Hotel có nhiều Room, và mỗi Room chỉ thuộc về 1 Hotel



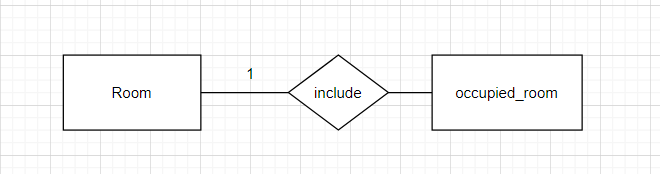
2.3 Mối quan hệ giữa Room và room type

Mỗi room chỉ có từng loại room type, 1 loại room type có thể có nhiều room



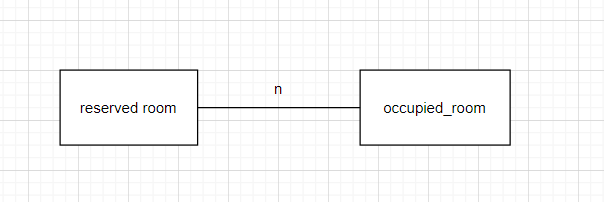
2.4 Mối quan hệ giữa Room và occupied\_room

Room bao gồm nhiều occupied room



2.5 Mối quan hệ giữa room type và reserved room

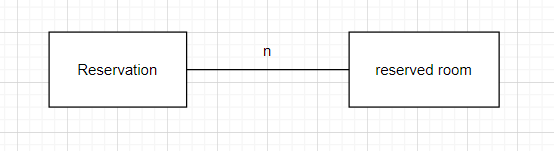
Mỗi reserved room có thể bao gồm nhiều room type



2.6 Mối quan hệ giữa reserved room và reservation

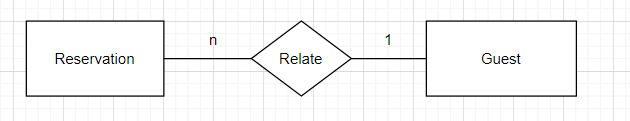
Khách có thể đặt trước nhiều phòng

Một reservation có nhiều reserved room



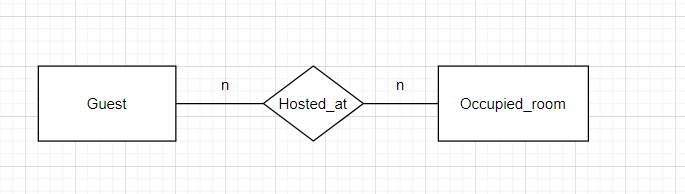
2.7 Mối quan hệ giữa guest và reservation

Một Guest có thể đặt nhiều reservation, mỗi reservation chỉ liên quan tới 1 Guest

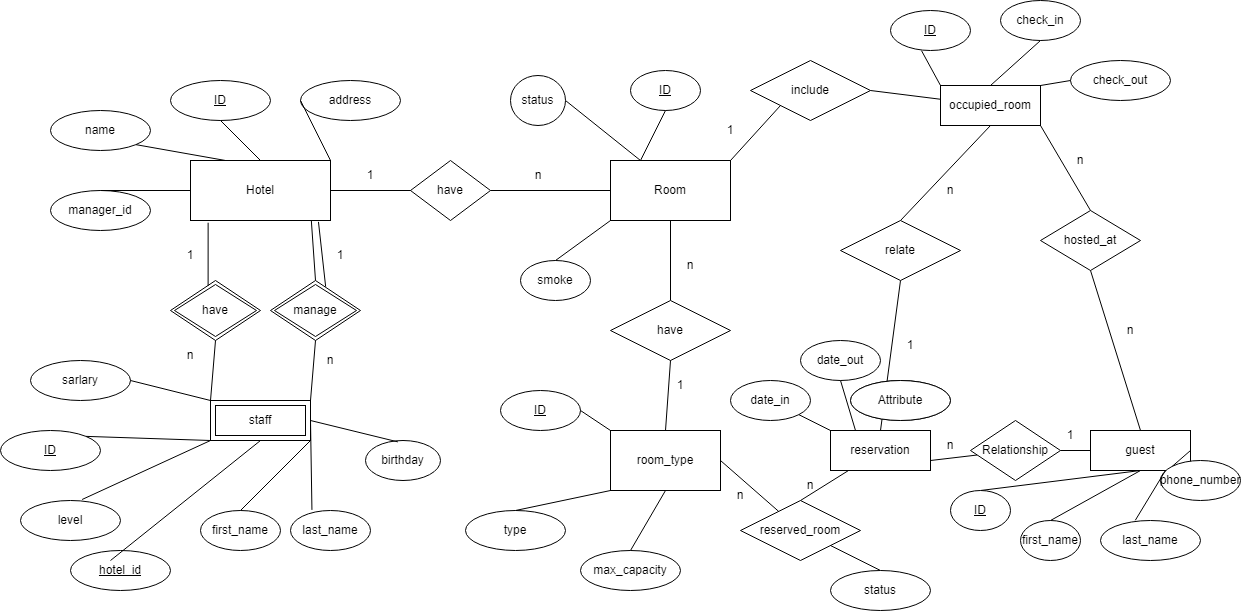


2.8 Mối quan hệ giữa Occupied\_room và Guest

Guest có thể ở nhiều phòng, một phòng có thể có nhiều guest



## *3. Xây dựng mô hình thực thể kết hợp (ERD).*



## *4. Chuẩn hóa mô hình*

Lược đồ các thuộc tính:

**Hotel**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | name | address | manager\_id |

**Room\_type**

|  |  |  |
| --- | --- | --- |
| **id** | description | max\_capacity |

**Room**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **id** | **hotel\_id** | number | name | status | smoke | room\_type\_of |

**Occupied\_room**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | check\_in | check\_out | room\_id | reservation\_id |

**Hosted\_at**

|  |  |  |
| --- | --- | --- |
| **id** | guest\_id | occupied\_room\_id |

**Guest**

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | first\_name | last\_name | phone\_number |

**Reservation**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | date\_in | date\_out | guest\_id | deposit |

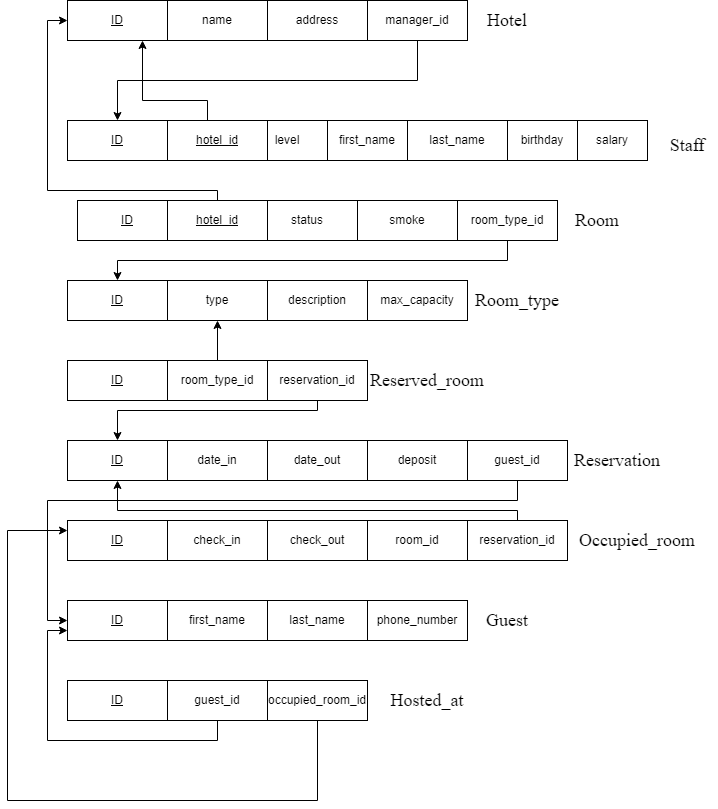
**Staff**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **hotel\_id** | level | first\_name | middle\_name | last\_name | birthday | manager\_id | salary |

**Reserved\_room**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | number\_of\_rooms | room\_type\_id | reservation\_id | status |

## *5. Xây dựng mô hình quan hệ.*



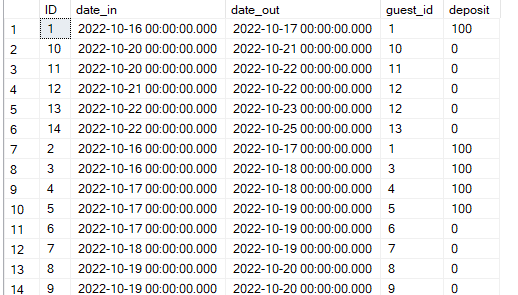
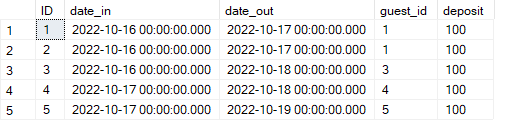
## *6. Xây dựng các phép toán đại số quan hệ*

### a. Phép chọn

a ) Ví dụ Chọn ra những reservation có deposit khác 0

Quan hệ Reservation (ID, date\_in, date\_out, guest\_id, deposit) với tập p: deposit=100

SV (p)

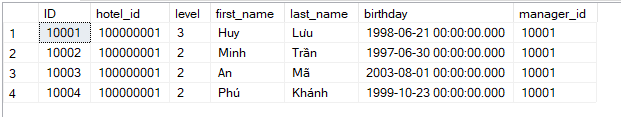
 

deposit = 100(reservation)

b) Chọn ra những staff có mã manager\_id là 10001

Ta có quan hệ Staff ( ID, hotel\_id, level, first\_name, last\_name, birthday, manager\_id ) và tập P manager\_id = 10001

SV (p)

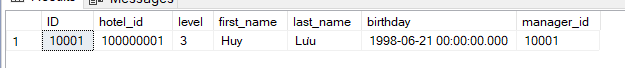


manager\_id = 10001(staff)

c) Chọn ra những staff có level = 3

Ta có quan hệ Staff ( ID, hotel\_id, level, first\_name, last\_name, birthday, manager\_id ) và tập P level =3

SV (p)

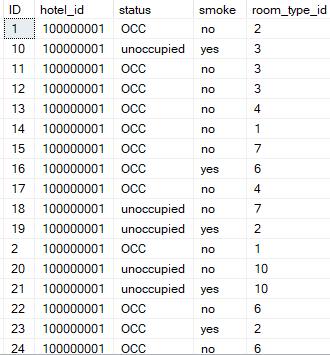
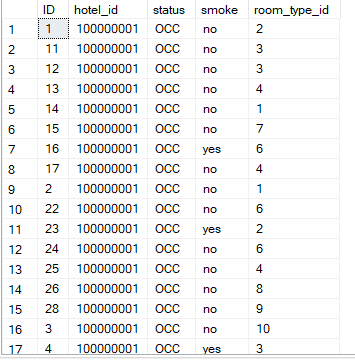


level = 3(staff)

d) Chọn ra những room có status = OCC

Ta có quan hệ Room ( ID, hotel\_id, status, smoke, room\_type\_id)

SV (p)

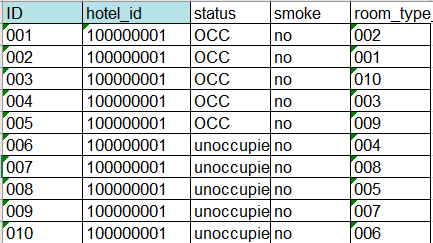
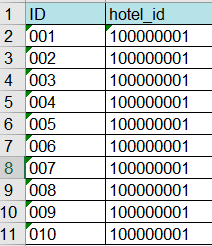
status = ‘OCC’(room)

### b. Phép chiếu

a) Ta có quan hệ Room(ID, hotel\_id, status, smoke ) và tập X {ID, status}

Ta có chiếu SV trên X như sau :

SV π[X]

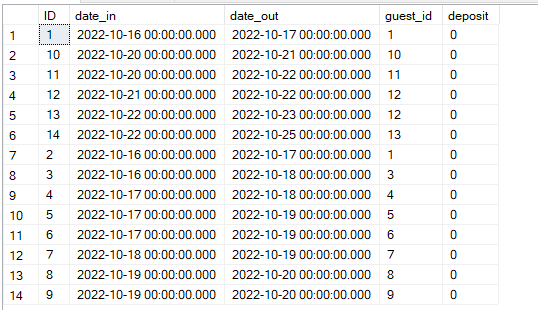
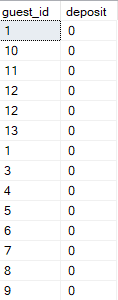
 

ID, status(room)

b) Ta có quan hệ Reservation (ID, date\_in, date\_out, guest\_id , deposit ) và tập X {guest\_id, deposit)

Ta có chiếu SV trên X như sau:

SV π[X]

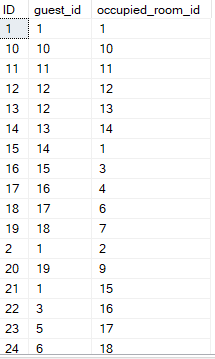
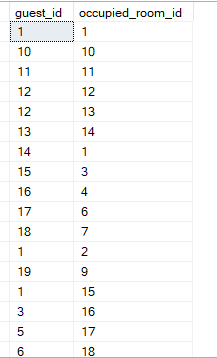
 

guest\_id, deposit (reservation)

c) Ta có quan hệ Hosted at ( ID, guest\_id, occupied\_room\_id) và tập X { guest\_id, occupied\_room\_id}

Ta có chiếu SV trên X như sau :

SV π[X]

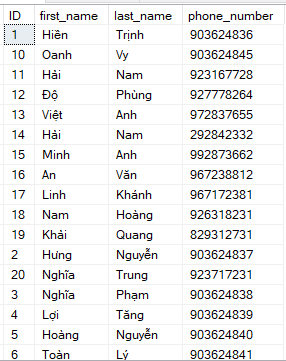
 

guest\_id, occupied\_room\_id  (hosted\_at)

d) Ta có quan hệ Guest ( ID, first\_name, last\_name, phone\_number) và tập X {first\_name, last\_name }

Ta có phép chiếu SV trên X

SV π[X]

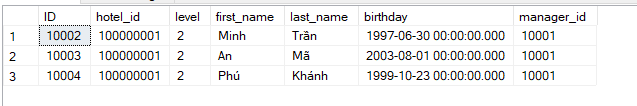
 

first\_name,last\_name (guest)

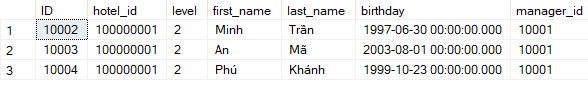
### c. Phép giao

a) Chọn ra những nhân viên ở level 2 có manager\_id là 10001

R S

R S



level=’2’ (staff) manager\_id=10001 (staff)

b) Chọn ra những nhân viên có mã manager\_id = Null và ID có dạng ‘1..’

R S



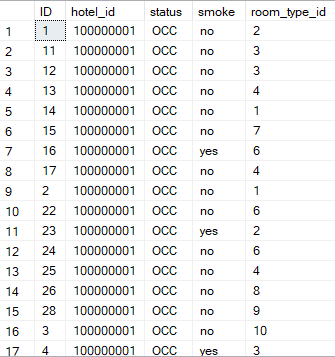
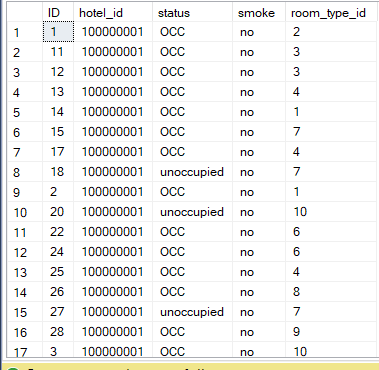
R S



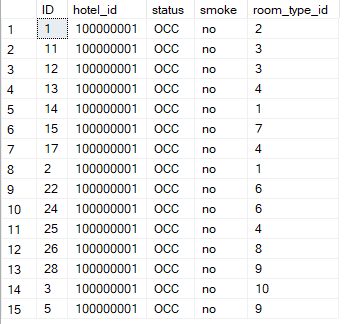
ID > 10000 and ID < 20000 (staff) manager\_id=null (staff)

c) Chọn ra những Room có status = OCC và smoke = no

R S

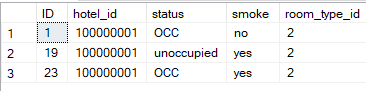
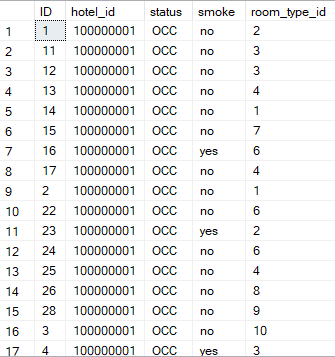
R S



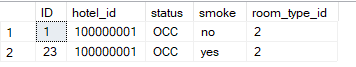
status=’OCC’ (room) smoke=’no’ (staff)

d) Chọn ra những Room có room\_type\_id = 2 và status = OCC

R S

R S

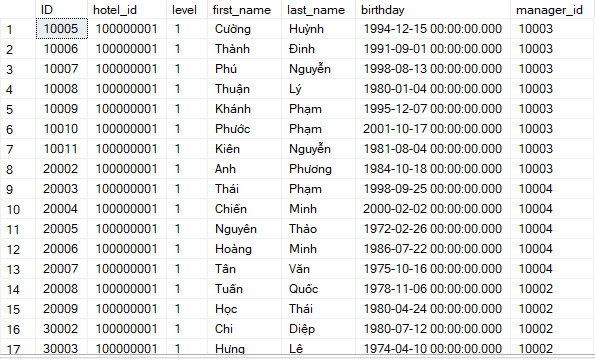
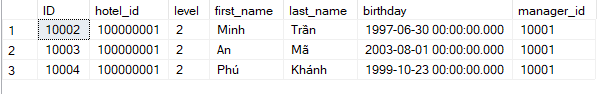


status=’OCC’ (room) smoke=’no’ (staff)

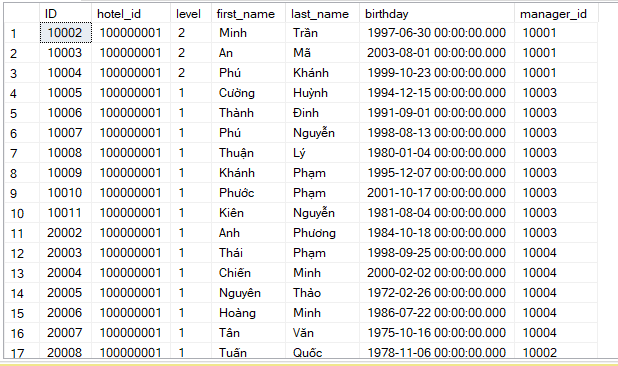
### d. Phép hội

a) Chọn ra những nhân viên ở level 1 hoặc có mã manager\_id = 10001

R S

R S



level=1 () manager\_id=10001 (staff)

b) Chọn ra những staff ở level 2 hoặc manager\_id = Null

R S

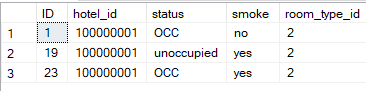
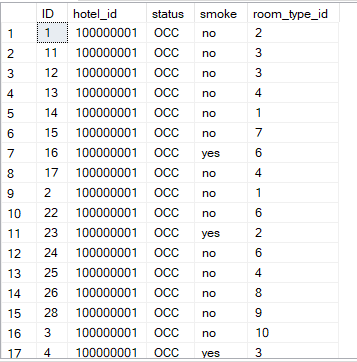
  R S



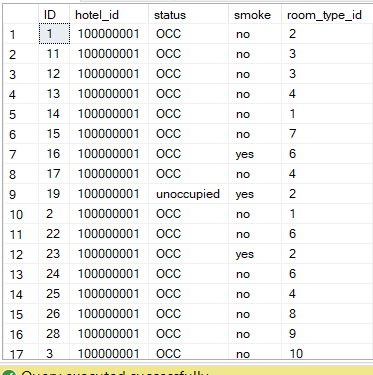
level=2() manager\_id=null (staff)

c) Chọn ra những Room có room\_type\_id = 2 hoặc status = OCC

R S

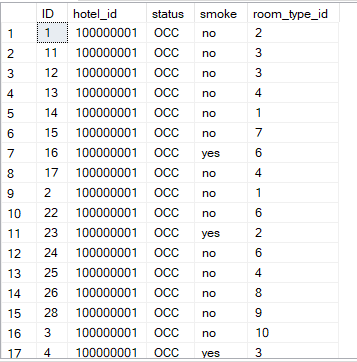
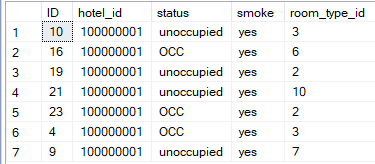
R S



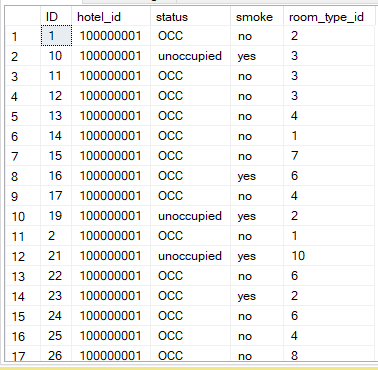
room\_type\_id=2 () status=’OCC' (room)

d) Chọn ra những Room có status = OCC hoặc smoke = yes

R S

R S

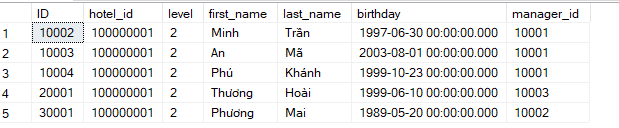


status=’OCC’(room)) smoke=’yes’ (room)

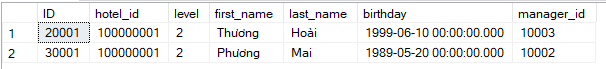
### e. Phép trừ

a) Chọn ra những Staff có level = 2 nhưng không có manager\_id = 10001

R S

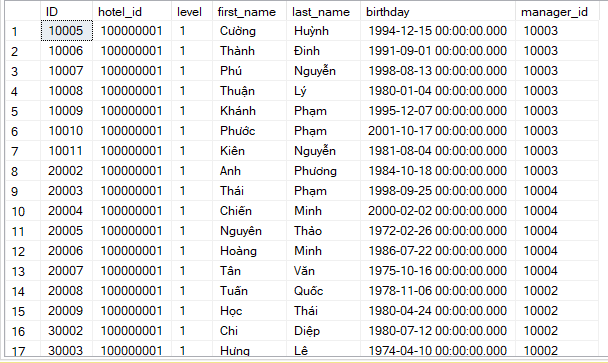


R-S

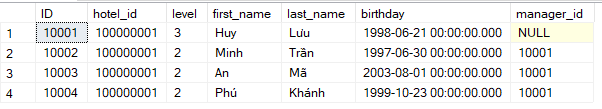


ID > 10000 and ID < 20000 (staff) - manager\_id=10001 (staff)

b) Chọn ra những Staff có ID bắt đầu bằng 1 nhưng level khác 1

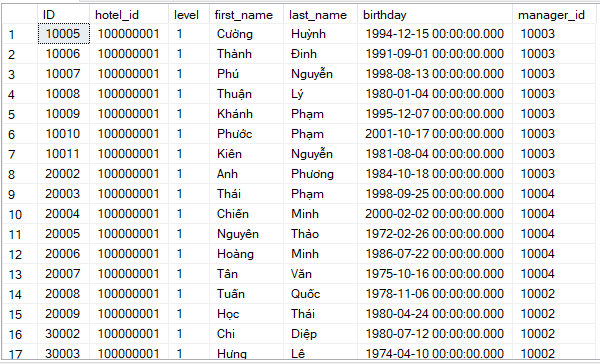
R – S



ID > 10000 and ID < 20000 (staff) – level=1 (staff)

c)Chọn ra những Staff có manager\_id = 10003 nhưng level khác 1

R S

R - S

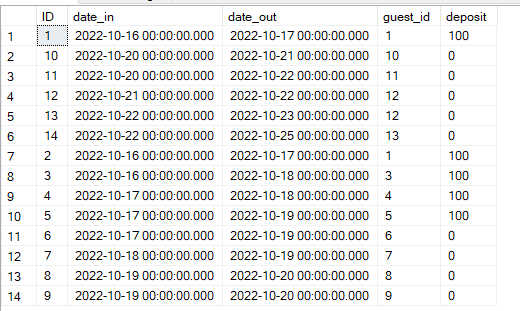
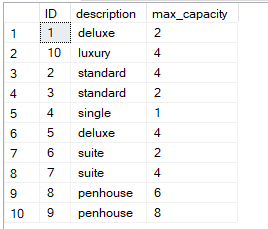


manager\_id=10003 (staff) – level=1 (staff)

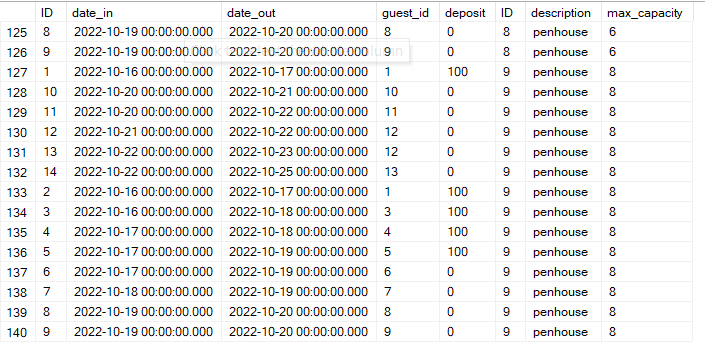
### f. Phép Tích Decac

a) Thực hiện tích hai bảng reservation và room\_type

R ( 14 dòng 5 cột ) S(10 dòng 3 cột )

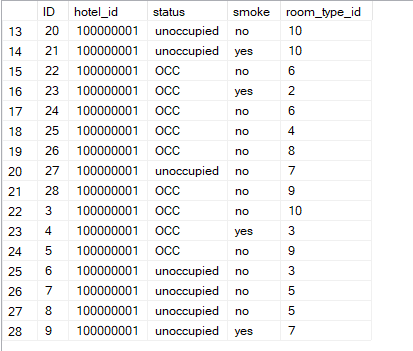
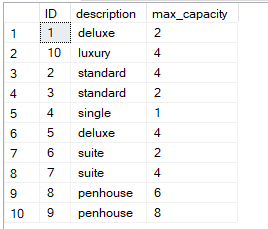
 

(reservation)) (room\_type))

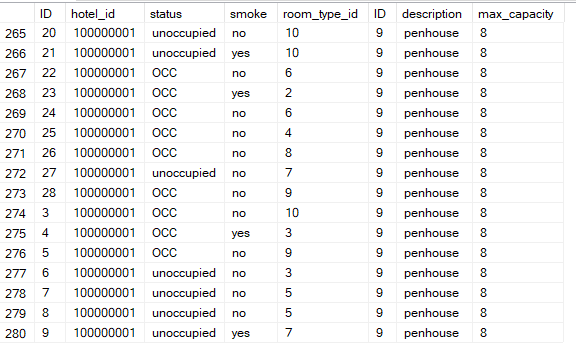


b) Phép tích bảng room và room\_type

R ( 28 dòng và 5 cột ) S ( 10 dòng và 3 cột)

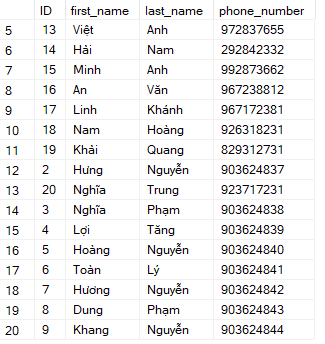
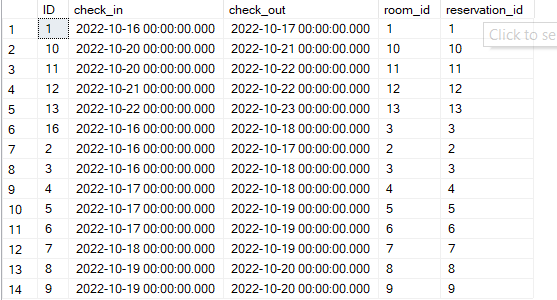
 

(room) (room\_type)

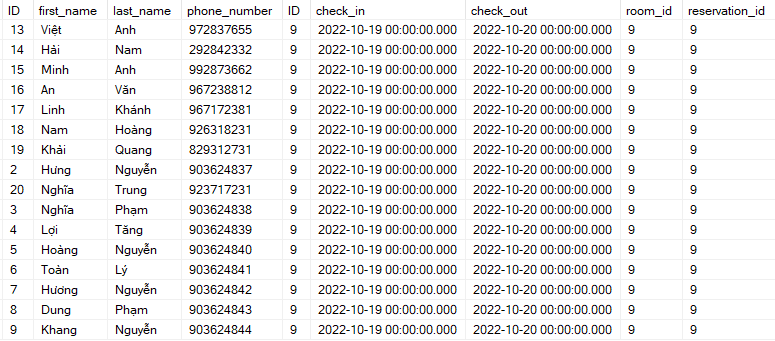


c) Phép tích hai bảng guest và occupied\_room ( thêm điều kiện room\_id = reservation\_id )

R ( 20 dòng 4 cột ) S ( 14 dòng 5 cột)

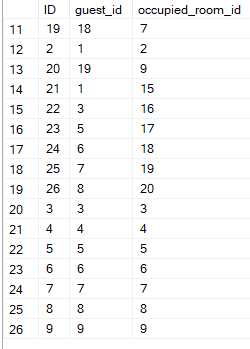
 

(guest)) (occupied\_room))

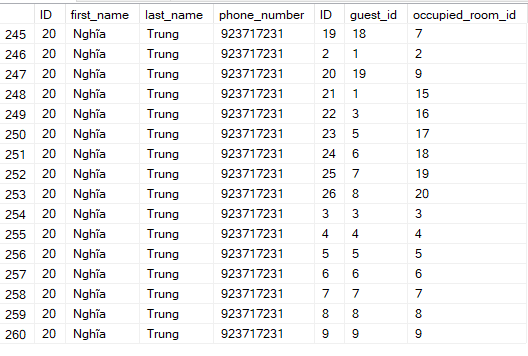


d) Phép tích bảng Guest và bảng hosted\_at ( có thêm điều kiện Guest.ID >10 )

R ( 10 dòng 4 cột) S ( 26 dòng 3 cột)

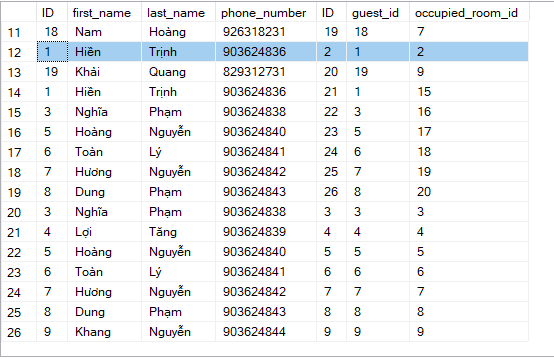
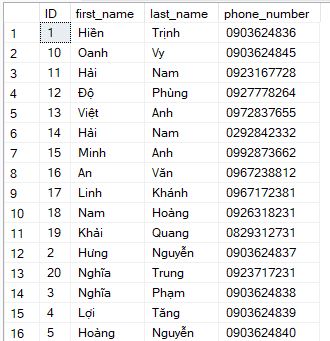
`(guest)) (hosted\_at))

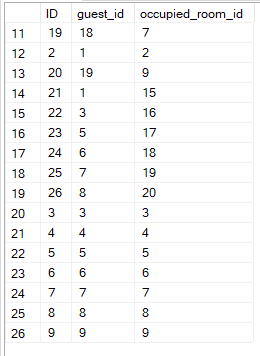


### g. Phép kết nối

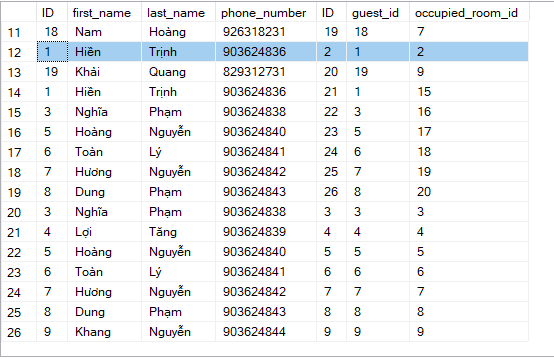
a) Phép kết nối ( Bảng guest và hosted\_at sử dụng inner join )

R S



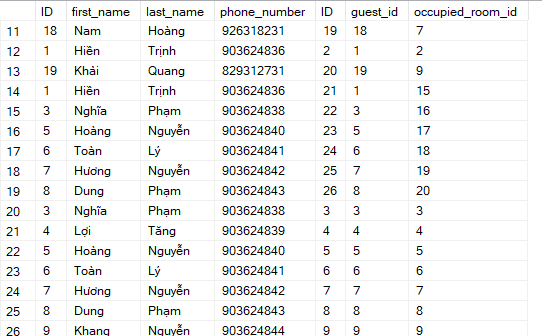


(guest)) ⋈ (guest).ID = (hosted\_at).guest\_id(hosted\_at))



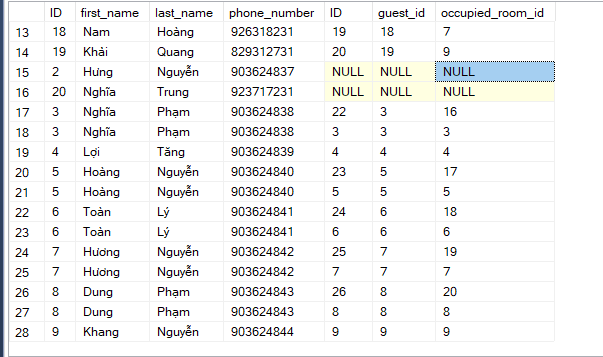
b) Phép kết nối ( Bảng Guest và hosted \_at sử dụng right join)

((guest)) ⋊(guest).ID = (hosted\_at).guest\_id  (hosted\_at))



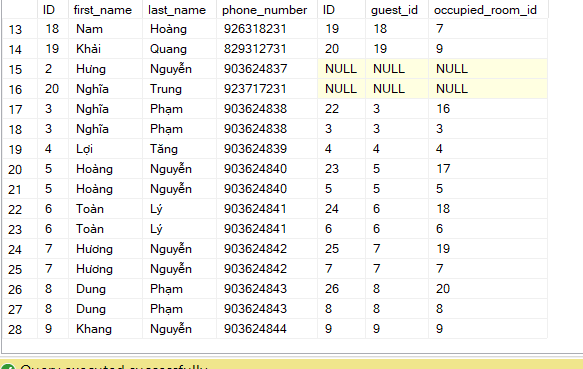
c) Phép kết nối ( Bảng Guest và hosted\_at sử dụng left join)

((guest)) ⋉(guest).ID = (hosted\_at).guest\_id  (hosted\_at))



d) Phép kết nối ( Bảng Guest và hosted\_at sử dụng full outer join)

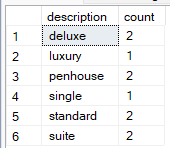
(guest)) ⋈FULL(guest).ID = (hosted\_at).guest\_id(hosted\_at))



### h. Phép gom nhóm

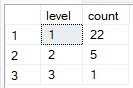
a) Gom nhóm bảng room\_type dựa trên description

<description,count>(desctiption ℑ COUNT(desciption) room\_type)



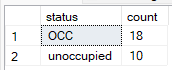
b) Gom nhóm bảng Staff dựa trên level

<level,count>(level ℑ COUNT(level) Staff)



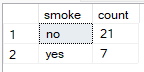
c) Gom nhóm bảng Room dựa trên status

<status,count>(status ℑ COUNT(status) Room)



d) Gom nhóm bảng Room dựa trên smoke

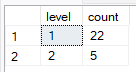
<smoke,count>(smoke ℑ COUNT(smoke) Room)



### i. Phép gom nhóm có điều kiện

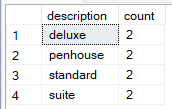
a) Gom nhóm bảng Staff dựa trên level có số lượng lớn hơn 3

<level,count>[<COUNT\_level > 3>(level ℑ COUNT(level) Staff)]



b) Gom nhóm bảng Room dựa trên description có số lượng từ 2

<description,count>[<COUNT\_desciptio > 2>(description ℑ COUNT(desciption) Room)]



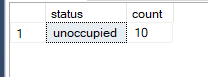
c) Gom nhóm bảng Room dựa trên smoke có điều kiện smoke = no

<smoke,count>[<COUNT\_smoe =no>(smoke ℑ COUNT(smoke) Room)]



d) Gom nhóm bảng Room dựa trên status = unoccupied

<status,count>[<COUNT\_status=unoccupied>(status ℑ COUNT(desciption) Room)]



## 7. Các lệnh mô tả dữ liệu (Data Definition Language )

Mã nguồn

|  |
| --- |
| [SQL-Project-Code/HotelManageFinal.sql at main · ghbihuy/SQL-Project-Code (github.com)](https://github.com/ghbihuy/SQL-Project-Code/blob/main/HotelManageFinal.sql) |

### a. Tạo database

- Khởi tạo cơ sở dữ liệu có tên Dalat.

|  |
| --- |
| CREATE DATABASE Dalat; |

### b. Tạo các table

* Tạo table Hotel

|  |
| --- |
| create table Hotel ( ID nvarchar(11), name nvarchar(40), address nvarchar(100), primary key (ID) ) |

* Tạo table room\_type

|  |
| --- |
| create table room\_type ( ID nvarchar(10), description nvarchar(80), max\_capacity int, primary key(ID) ) |

* Tạo table room

|  |
| --- |
| create table room ( ID nvarchar(10), hotel\_id nvarchar(11), status nvarchar(40), smoke nvarchar(5), room\_type\_id nvarchar(10), primary key (ID) ) |

* Tạo table occupied\_room

|  |
| --- |
| create table occupied\_room ( ID nvarchar(10), check\_in datetime, check\_out datetime, room\_id nvarchar(10) not null, reservation\_id nvarchar(10) not null, primary key (ID) ) |

* Tạo table hosted\_at

|  |
| --- |
| create table hosted\_at ( ID nvarchar(10), guest\_id nvarchar(10), occupied\_room\_id nvarchar(10), primary key (ID) ) |

* Tạo table guest

|  |
| --- |
| create table guest ( ID nvarchar(10), first\_name nvarchar(80), last\_name nvarchar(80), phone\_number nvarchar(10), primary key (ID) ) |

* Tạo table reservation

|  |
| --- |
| create table reservation ( ID nvarchar(10), date\_in datetime, date\_out datetime, guest\_id nvarchar(10), deposit int, primary key (ID) ) |

* Tạo table staff

|  |
| --- |
| create table staff ( ID nvarchar(10), hotel\_id nvarchar(11), level int, first\_name nvarchar(10), last\_name nvarchar(20), birthday datetime, manager\_id nvarchar(10), salary int, primary key (ID, hotel\_id) ) |

* Tạo table reserved\_room

|  |
| --- |
| create table reserved\_room ( ID nvarchar(10), number\_of\_rooms int, room\_type\_id nvarchar(10), reservation\_id nvarchar(10) not null, primary key (ID) ) |

### c. Thay đổi cấu trúc bảng

|  |
| --- |
| ALTER TABLE reserved\_room  ADD CONSTRAINT fk\_1 FOREINGN KEY(reservation\_id)  REFERENCES reservation(ID); |

|  |
| --- |
| ALTER TABLE reserved\_room  ADD CONSTRAINT fk\_2 FOREIGN KEY(room\_type\_id)  REFERENCES room\_type(ID); |

|  |
| --- |
| ALTER TABLE room  ADD CONSTRAINT fk\_3 FOREIGN KEY(room\_type\_id)  REFERENCES room\_type(ID); |

|  |
| --- |
| ALTER TABLE occupied\_room  ADD CONSTRAINT fk\_4 FOREIGN KEY(room\_id)  references room(ID); |

|  |
| --- |
| ALTER TABLE hosted\_at  ADD CONSTRAINT fk\_5 FOREIGN KEY(occupied\_room\_id)  references occupied\_room(id); |

|  |
| --- |
| ALTER TABLE hosted\_at  ADD CONSTRAINT fk\_6 FOREIGN KEY(guest\_id)  references guest(id); |

|  |
| --- |
| ALTER TABLE reservation  ADD CONSTRAINT fk\_7 FOREIGN KEY(guest\_id)  REFERENCES guest(ID); |

|  |
| --- |
| ALTER TABLE occupied\_room  ADD CONSTRAINT fk\_8 FOREIGN KEY(reservation\_id)  REFERENCES reservation(ID); |

|  |
| --- |
| ALTER TABLE room  ADD CONSTRAINT fk\_9 FOREIGN KEY(hotel\_id)  REFERENCES hotel(ID); |

|  |
| --- |
| ALTER TABLE staff  ADD CONSTRAINT fk\_10 FOREIGN KEY(hotel\_id)  REFERENCES hotel(ID); |

|  |
| --- |
| ALTER TABLE staff  ADD CONSTRAINT fk\_11 FOREIGN KEY(manager\_id, hotel\_id)  REFERENCES staff(ID,hotel\_id); |

## 8. Các lệnh thao tác dữ liệu

### a. Thêm dữ liệu vào các bảng bằng lệnh.

* Insert dữ liệu vào bảng Hotel

|  |
| --- |
| INSERT INTO Hotel (ID,name,address) values(100000001,N'Cú trên Cây',N'37 Đường Đặng Thái Thân, Phường 3, Thành phố Đà Lạt, Lâm Đồng ') |

* Insert dữ liệu vào bảng Staff

|  |
| --- |
| set dateformat DMY;  INSERT INTO staff (ID,hotel\_id, [level], first\_name, last\_name, birthday, manager\_id) values(10001,100000001,3,N'Huy',N'Lưu','21/6/1998',10001)  INSERT INTO staff (ID,hotel\_id, [level], first\_name, last\_name, birthday, manager\_id) values(10002,100000001,2,N'Minh',N'Trần','30/6/1997',10001) |

* Insert dữ liệu vào bảng room\_type

|  |
| --- |
| INSERT INTO room\_type (ID, description, max\_capacity) values(001,'deluxe','2') |

* Insert dữ liệu vào bảng room

|  |
| --- |
| INSERT INTO room (ID,hotel\_id, smoke, status, room\_type\_id) values(002,100000001,'no','OCC',001) |

* Insert dữ liệu vào bảng guest

|  |
| --- |
| INSERT INTO guest (ID, first\_name, last\_name, phone\_number) values(001,N'Hiền',N'Trịnh','0903624836') |

* Insert dữ liệu vào bảng reservation

|  |
| --- |
| set dateformat YMD;  INSERT INTO reservation (ID, date\_in, date\_out, guest\_id, deposit) values(001,'2022-10-16 00:00:00','2022-10-17 00:00:00',001,'0') |

* Insert dữ liệu vào bảng reserved\_room

|  |
| --- |
| INSERT INTO reserved\_room (ID, number\_of\_rooms, room\_type\_id, reservation\_id) values(001,1,001,001) |

* Insert dữ liệu vào bảng occupied\_room

|  |
| --- |
| INSERT INTO occupied\_room (ID, check\_in, check\_out, room\_id, reservation\_id) values(001,'2022-10-16 00:00:00','2022-10-17 00:00:00',001,001) |

* Insert dữ liệu vào bảng hosted\_at

|  |
| --- |
| INSERT INTO hosted\_at (ID, guest\_id, occupied\_room\_id) values(001,001,001) |

* Mã nguồn Insert Data

|  |
| --- |
| * Insert data vào Database bằng SQL file [SQL-Project-Code/Insert\_Dalat.sql at main · ghbihuy/SQL-Project-Code (github.com)](https://github.com/ghbihuy/SQL-Project-Code/blob/main/Insert_Dalat.sql) * Insert data vào Database bằng Python (Pandas Library and Apache Airflow)   [SQL-Project-Code/insert\_operation\_dalat.py at main · ghbihuy/SQL-Project-Code (github.com)](https://github.com/ghbihuy/SQL-Project-Code/blob/main/insert_operation_dalat.py)  Data để insert vào bằng Python  [SQL-Project-Code/Data Dalat.xlsx at main · ghbihuy/SQL-Project-Code (github.com)](https://github.com/ghbihuy/SQL-Project-Code/blob/main/Data%20Dalat.xlsx) |

1. Update dữ liệu vào các table bằng lệnh SQL

* Update dữ liệu bảng staff thay đổi manager\_id

|  |
| --- |
| set dateformat DMY;  UPDATE staff SET manager\_id= NULL, WHERE ID='10001'; |

1. Xóa dữ liệu ở các bảng table bằng lệnh SQL

* Xóa dữ liệu trong bảng staff

|  |
| --- |
| DELETE FROM staff |

* Xóa dữ liệu trong bảng hosted\_at

|  |
| --- |
| DELETE FROM hosted\_at |

* Xóa dữ liệu trong bảng occupied\_room

|  |
| --- |
| DELETE FROM occupied\_room |

* Xóa dữ liệu trong bảng reserved\_room

|  |
| --- |
| DELETE FROM reserved\_room |

* Xóa dữ liệu trong bảng reservation

|  |
| --- |
| DELETE FROM reservation |

* Xóa dữ liệu trong bảng guest

|  |
| --- |
| DELETE FROM guest |

* Xóa dữ liệu trong bảng room

|  |
| --- |
| DELETE FROM room |

* Xóa dữ liệu trong bảng room\_type

|  |
| --- |
| DELETE FROM room\_type |

* Xóa dữ liệu trong bảng hotel

|  |
| --- |
| DELETE FROM hotel |

## 9. Lệnh truy vấn dữ liệu

### a. Truy vấn 1 bảng

* Truy vấn bảng staff

|  |  |
| --- | --- |
| SELECT   \* FROM   staff |  |

* Truy vấn bảng room\_type

|  |  |
| --- | --- |
| SELECT   \* FROM   room\_type |  |

* Truy vấn bảng reservation

|  |  |
| --- | --- |
| SELECT   \* FROM   reservation |  |

* Truy vấn bảng occupied\_room

|  |  |
| --- | --- |
| SELECT   \* FROM   occupied\_room |  |

1. Truy vấn nhiều bảng (Phép kết)

|  |  |
| --- | --- |
| SELECT   room.ID,  status,  room\_type\_id,  description FROM   room INNER JOIN room\_type  ON room.room\_type\_id = room\_type.ID |  |

* Cho biết kiểu phòng của phòng trong bảng room
* Cho biết kiểu phòng (room\_type) của đơn đặt trước (reservation)

|  |
| --- |
| SELECT  reservation.ID,  date\_in,  date\_out,  description,  max\_capacity FROM  (reservation  INNER JOIN reserved\_room  ON reservation.ID = reserved\_room.reservation\_id)  INNER JOIN room\_type  ON reserved\_room.room\_type\_id = room\_type.ID |
|  |

* Cho biết thông tin của phòng đã sử dụng (occupied\_room) và thông tin của khách hàng có ID = 1

|  |
| --- |
| SELECT  guest\_id,   last\_name,  first\_name,  check\_in,  check\_out,  occupied\_room\_id FROM  (occupied\_room  INNER JOIN hosted\_at  ON occupied\_room.ID = hosted\_at.occupied\_room\_id)  INNER JOIN guest  ON hosted\_at.guest\_id = guest.ID WHERE  guest\_id = 1 |
|  |

1. Truy vấn có điều kiện

* Truy vấn phòng không cho hút thuốc

|  |  |
| --- | --- |
| SELECT   \* FROM   room WHERE   smoke = ‘no’ |  |

* Truy vấn nhân viên có level 2

|  |  |
| --- | --- |
| SELECT   \* FROM   staff WHERE   level = ‘2’ |  |

* Truy vấn đặt trước có guest\_id là 1

|  |  |
| --- | --- |
| SELECT   \* FROM   reservation WHERE   guest\_id = 1 |  |

* Truy vấn kiểu phòng có sức chứa lớn hơn 2

|  |  |
| --- | --- |
| SELECT   \* FROM   room\_type WHERE   max\_capacity > 2 |  |

### Truy vấn tính toán

* Tính tuổi của từng nhân viên

|  |  |
| --- | --- |
| SELECT  ID,  first\_name,  last\_name,  DATEDIFF(YEAR,birthday,GETDATE()) as age  FROM staff |  |

* Tính sức chứa của khách sạn 100000001 ‘Cú trên cây’

|  |  |
| --- | --- |
| SELECT  hotel\_id,  SUM(max\_capacity) as capacity  FROM (room INNER JOIN room\_type  ON room.room\_type\_id = room\_type.ID)  GROUP BY  hotel\_id |  |

* Tính tổng số phòng đã được đặt từ trước đến nay trong bảng hosted\_at

|  |  |
| --- | --- |
| SELECT  SUM(number\_of\_rooms) as sum  FROM  reserved\_room |  |

1. Truy vấn có gom nhóm

* Đếm số lượng room\_type gom nhóm bởi description

|  |  |
| --- | --- |
| SELECT   description,   COUNT(\*) AS count  FROM   room\_type  GROUP BY   description |  |

* Đếm số lượng nhân viên của mỗi level trong bảng staff

|  |  |
| --- | --- |
| SELECT   level,   COUNT(\*) AS count FROM   staff GROUP BY   level |  |

* Đếm số lượng phòng gom nhóm bởi status trong bảng room

|  |  |
| --- | --- |
| SELECT   status,  COUNT(\*) as count FROM   room GROUP BY   status |  |

* Đếm số lượng phòng cho hút thuốc và không cho hút thuốc trong bảng room

|  |  |
| --- | --- |
| SELECT   smoke,  COUNT(\*) as count FROM   room GROUP BY   smoke |  |

### Truy vấn gom nhóm có điều kiện (Having)

* Đếm số lượng nhân viên của mỗi level trong bảng staff số lượng phải lớn hơn 3

|  |  |
| --- | --- |
| SELECT   level,   COUNT(\*) AS count FROM   staff GROUP BY   level HAVING  COUNT(\*) > 3 |  |

* Đếm số room type gom nhóm bởi description mà lớn hoặc bằng 2

|  |  |
| --- | --- |
| SELECT   description,   COUNT(\*) AS count  FROM   room\_type  GROUP BY   description HAVING  COUNT(\*) > = 2 |  |

1. Truy vấn có sử dụng phép giao, hội, trừ

* Phép trừ

|  |  |
| --- | --- |
| (SELECT   \* FROM   room) UNION (SELECT   \* FROM   room WHERE   smoke = ‘no’) |  |

* Phép hội

|  |  |
| --- | --- |
| (SELECT   \* FROM   room WHERE   smoke = ‘yes’) UNION (SELECT   \* FROM   room WHERE   smoke = ‘yes’) |  |

* Phép giao

|  |  |
| --- | --- |
| (SELECT   \* FROM   room WHERE   status = ‘OCC’) INTERSECT (SELECT  \* FROM   room WHERE   smoke = ‘NO’) |  |

### Truy vấn con (Subquery)

* Truy vấn nhân viên có level lớn hơn hoặc bằng 2

|  |
| --- |
| SELECT \* FROM staff WHERE ID IN  (SELECT ID  FROM staff  WHERE level = 2) |
|  |

* Truy vấn phòng cho hút thuốc

|  |  |
| --- | --- |
| SELECT \* FROM room  WHERE ID IN  (SELECT ID  FROM room  WHERE smoke = 'no') |  |

* Truy vấn khách đã từng đặt phòng

|  |  |
| --- | --- |
| SELECT \* FROM guest  WHERE ID IN  (SELECT guest\_id  FROM reservation) |  |

* Chọn những đơn đặt trước (reservation) của tất cả các phòng đã từng ở (occupied\_room)

|  |  |
| --- | --- |
| SELECT \* FROM reservation  WHERE ID IN  (SELECT guest\_id  FROM occupied\_room) |  |

### Truy vấn chéo (Pivot table)

* Đếm số phòng theo room\_type\_id và status

|  |  |
| --- | --- |
| SELECT room\_type\_id, [OCC], [unoccupied]  FROM  (SELECT  room\_type\_id,  status,ID  FROM room  ) r  PIVOT  (count(ID)  FOR status IN ([OCC], [unoccupied])  ) AS pivottable  ORDER BY room\_type\_id DESC |  |

* Đếm số nhân viên theo khách sạn và level

|  |  |
| --- | --- |
| SELECT hotel\_id, [1], [2], [3]  FROM  (SELECT  hotel\_id,  level,ID  FROM staff  ) r  PIVOT  (count(ID)  FOR level IN ([1], [2], [3])  ) AS pivottable |  |

* Đếm số phòng cho hút thuốc hay không theo status

|  |  |
| --- | --- |
| SELECT status, [no], [yes]  FROM  (SELECT  status,  smoke,ID  FROM room  ) r  pivot  (count(ID)  FOR smoke in ([no], [yes])  ) AS pivottable |  |

## 10.Viết store procedure và function

* Tạo procedure chèn thông tin vào bảng

|  |
| --- |
| -- Create procedure to insert data  GO  create procedure insert\_hotel(  @ID nvarchar(11),  @name nvarchar(40),  @address nvarchar(100)  )  as  insert into Hotel  values(@ID,@name,@address)  GO  create procedure insert\_staff  (  @ID nvarchar(10),  @hotel\_id nvarchar(11),  @level int,  @first\_name nvarchar(10),  @last\_name nvarchar(20),  @birthday datetime,  @manager\_id nvarchar(10),  @salary int  )  as  insert into staff  values(@ID, @hotel\_id, @level, @first\_name, @last\_name, @birthday, @manager\_id, @salary)  GO  create procedure insert\_roomtype  (  @ID nvarchar(10),  @description nvarchar(80),  @max\_capacity int  )  as  insert into room\_type  values(@ID, @description, @max\_capacity)  GO  create procedure insert\_room  (  @ID nvarchar(10),  @hotel\_id nvarchar(11),  @status nvarchar(40),  @smoke nvarchar(5),  @room\_type\_id nvarchar(10)  )  as  insert into room  values(@ID, @hotel\_id, @status, @smoke, @room\_type\_id)  GO  create procedure insert\_guest  (  @ID nvarchar(10),  @first\_name nvarchar(80),  @last\_name nvarchar(80),  @phone\_number int  )  as  insert into guest  values(@ID, @first\_name, @last\_name, @phone\_number)  GO  create procedure insert\_reservation  (  @ID nvarchar(10),  @date\_in datetime,  @date\_out datetime,  @guest\_id nvarchar(10),  @deposit int  )  as  insert into reservation  values(@ID, @date\_in, @date\_out, @guest\_id, @deposit)  GO  create procedure insert\_reserved\_room  (  @ID nvarchar(10),  @number\_of\_rooms int,  @room\_type\_id nvarchar(10),  @reservation\_id nvarchar(10)  )  as  insert into reserved\_room  values(@ID, @number\_of\_rooms, @room\_type\_id, @reservation\_id)  GO  create procedure insert\_occupied\_room  (  @ID nvarchar(10),  @check\_in datetime,  @check\_out datetime,  @room\_id nvarchar(10),  @reservation\_id nvarchar(10)  )  as  insert into occupied\_room  values(@ID, @check\_in, @check\_out, @room\_id, @reservation\_id)  GO  create procedure insert\_hosted\_at  (  @ID nvarchar(10),  @guest\_id nvarchar(10),  @occupied\_room\_id nvarchar(10)  )  as  insert into hosted\_at  values(@ID, @guest\_id, @occupied\_room\_id);  GO |

* Tạo function

|  |  |
| --- | --- |
| CREATE FUNCTION fullname  (  @firstname nvarchar(max),  @lastname nvarchar(max)  )  RETURNS NVARCHAR(MAX)  AS  BEGIN  RETURN @lastname +' '+ @firstname;  END; | SELECT  dbo.fullname(first\_name, last\_name) as fullname  FROM staff |

* Tạo table function

|  |  |
| --- | --- |
| CREATE FUNCTION roomtype\_of\_room  (  @room\_type\_id nvarchar(max)  )  RETURNS TABLE  AS  RETURN  SELECT  \*  FROM  room  WHERE  room\_type\_id = @room\_type\_id | SELECT  \*  FROM  roomtype\_of\_room(2) |

* Table function đa câu lệnh

|  |  |
| --- | --- |
| CREATE FUNCTION Manager\_Staff()  returns @contacts table(  ID varchar(10),  first\_name nvarchar(max),  last\_name nvarchar(max),  level int,  title nvarchar(max)  )  AS  BEGIN  INSERT INTO @contacts  SELECT ID, first\_name, last\_name,level,'staff'  FROM staff  WHERE level = 1;  INSERT INTO @contacts  SELECT ID, first\_name, last\_name,level,'manager'  FROM staff  WHERE level = 2;  INSERT INTO @contacts  SELECT ID, first\_name, last\_name,level,'chairman'  FROM staff  WHERE level = 3;  RETURN;  END; | SELECT \*  FROM  Manager\_Staff() |

## 11.Viết trigger ràng buộc dữ liệu cho các bảng

* Status trong bảng room phải nhập vào là ‘OCC’ hoặc là unoccupied

ALTER TABLE room

ADD CONSTRAINT check\_status\_room check (PHAI =N'OCC' or PHAI=N'unoccupied');

* Smoke trong bảng room nhập vào phải là yes hoặc no

ALTER TABLE room

ADD CONSTRAINT check\_smoke\_room check (PHAI =N'yes' or PHAI=N'no');

* Staff ID nhập vào phải là 5 số

|  |
| --- |
| Create Trigger staff\_id\_5number On staff For Insert As If (Select a.ID  From staff a, INSERTED b  where a.ID = b.ID) >99999  or (Select a.ID  From staff a, INSERTED b  where a.ID = b.ID) < 10000 Begin  Print 'STAFF ID MUST BE 5 NUMBER'  RollBack Tran End |
| * Nhập thử ID có nhiều hơn 5 số, sẽ bị báo lỗi   INSERT INTO staff values(1000111,100000001,3,N'Huy',N'Lưu','21/6/1998',10001) |

* Hotel ID nhập vào phải là 9 số

|  |
| --- |
| Create Trigger hotel\_id\_9number On Hotel For Insert As If (Select a.ID  From Hotel a, INSERTED b where a.ID = b.ID) >999999999  or (Select a.ID  From Hotel a, INSERTED b where a.ID = b.ID) < 100000000 Begin  Print 'HOTEL ID MUST BE 9 NUMBER'  RollBack Tran End |
| * Nhập thử ID có nhiều hơn hoặc ít hơn 9 số, sẽ bị báo lỗi   INSERT INTO Hotel values(1000000021,N'Cú trên Cây',N'37 Đường Đặng Thái Thân, Phường 3, Thành phố Đà Lạt, Lâm Đồng ') |

* Check\_in phải bé hơn check\_out trong table occupied\_room

|  |
| --- |
| Create Trigger check\_in\_check\_out\_occupied On occupied\_room For Insert As If (SELECT   DATEDIFF(HOUR,a.check\_in,a.check\_out)   FROM  occupied\_room a, INSERTED b  WHERE   a.check\_in = b.check\_in and  a.check\_out = b.check\_out  ) <0 Begin  Print 'WRONG CHECK\_IN AND CHECK\_OUT'  RollBack Tran End |
| * Insert thử check\_in lớn hơn check\_out   set dateformat YMD; INSERT INTO occupied\_room values(088,'2022-10-19 00:00:00','2022-10-17 00:00:00',001,001) |

* Date\_in phải bé hơn date\_out trong bảng reservation

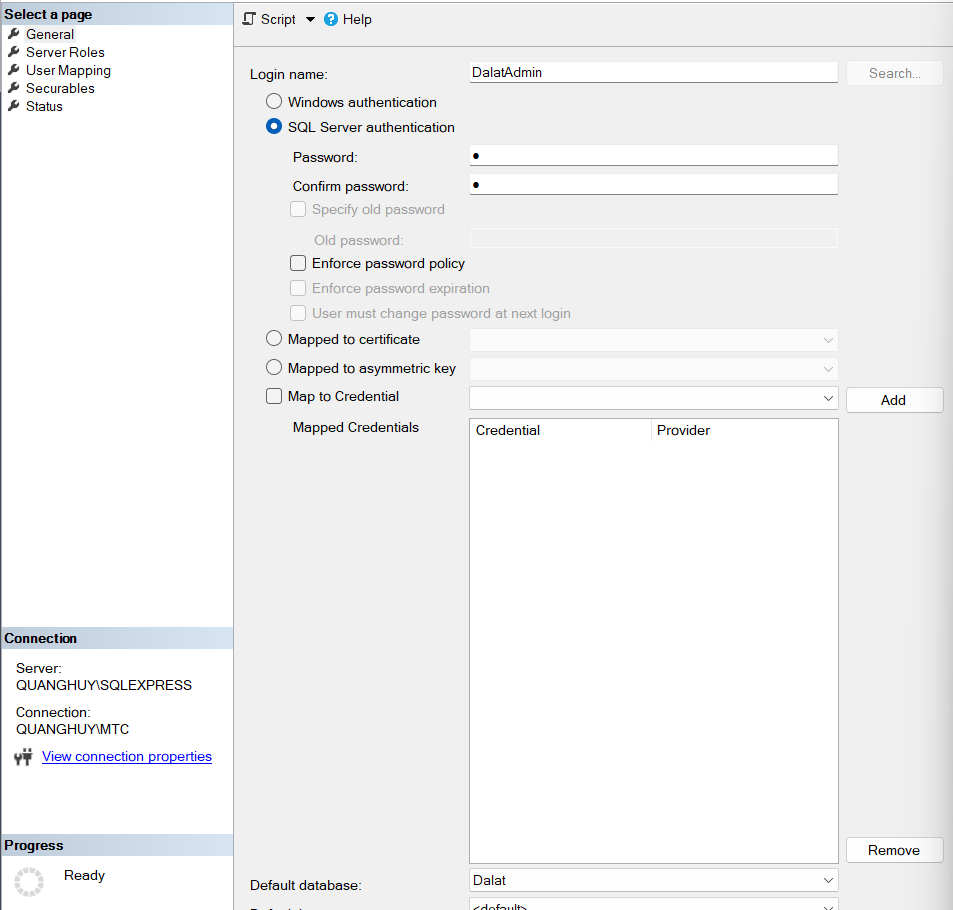
|  |
| --- |
| Create Trigger date\_in\_date\_out\_occupied On reservation For Insert As If (SELECT   DATEDIFF(HOUR,a.date\_in,a.date\_out)   FROM  reservation a, INSERTED b  WHERE   a.date\_in = b.date\_in and  a.date\_out = b.date\_out  ) <0 Begin  Print 'WRONG DATE\_IN AND DATE\_OUT'  RollBack Tran End |
| * Insert thử date\_in lớn hơn date\_out   INSERT INTO reservation values(081,'2022-10-18 00:00:00','2022-10-17 00:00:00',001,'0') |

* Nhập vào reserved\_room, nếu number\_of\_rooms lớn hơn phòng đang sẵn sàng (Trong trạng thái ‘OCC’ và không nằm trong thời gian có khách đang thuê (occupied\_room)) thì sẽ báo lỗi không đủ phòng

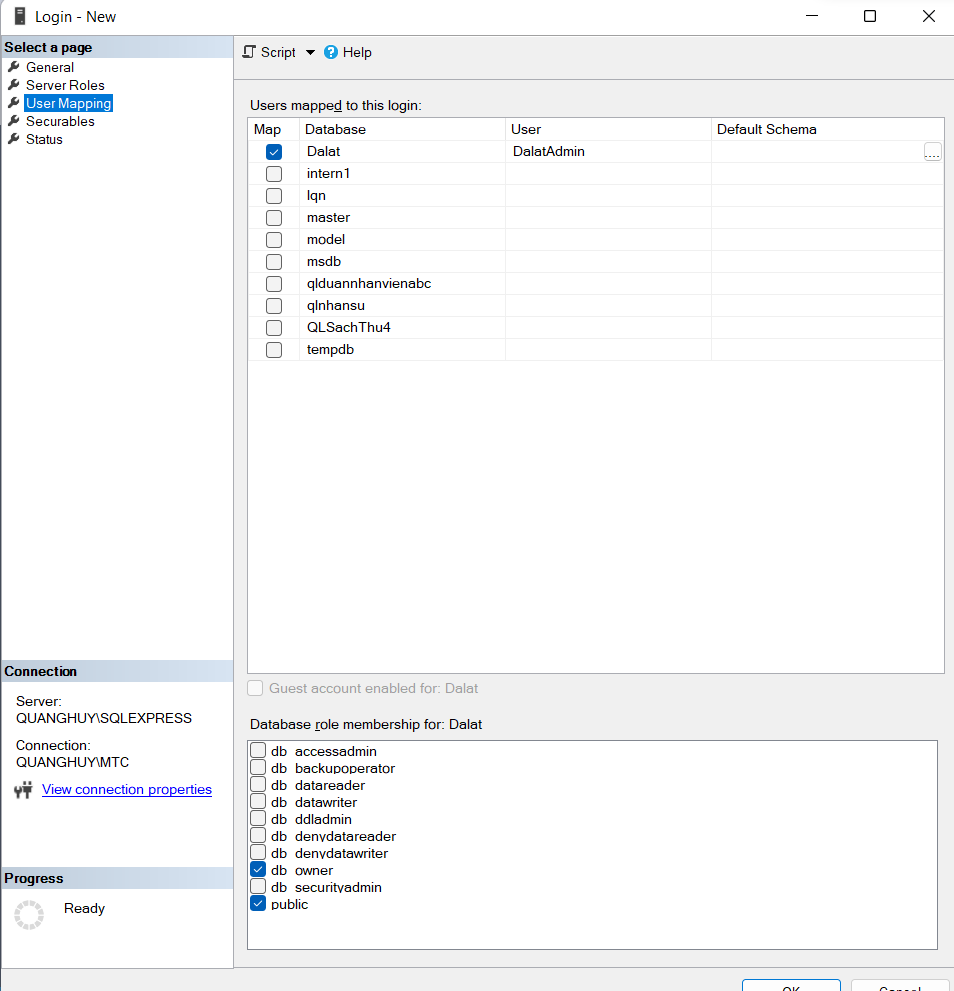
|  |
| --- |
| Create Trigger checking\_room On reserved\_room For Insert  As  IF  (  SELECT  reserving - ready as checking  FROM  (  SELECT  room\_type\_id,  SUM(number\_of\_rooms) as reserving  FROM  reserved\_room r INNER JOIN  (  SELECT  ID  FROM  reservation  WHERE  DATEDIFF(HOUR, date\_in, GETDATE()) > 0  AND DATEDIFF(HOUR, date\_out, GETDATE()) < 0  ) temp3  ON r.reservation\_id = temp3.ID  GROUP BY room\_type\_id  ) reserving  INNER JOIN  (  SELECT  room.room\_type\_id,  COUNT(room.ID) as ready  FROM room LEFT JOIN  (  SELECT  \*  FROM  occupied\_room  WHERE  DATEDIFF(HOUR, check\_out, GETDATE()) < 0  )temp1 ON room\_id = temp1.room\_id  WHERE room.status = ‘OCC’  GROUP BY  room.room\_type\_id  ) ready  ON reserving.room\_type\_id = ready.room\_type\_id  ) > 0  Begin  Print 'THE RESERVATION IS OUT OF THE READY ROOM'  RollBack Tran  End |

## 12.Phân quyền

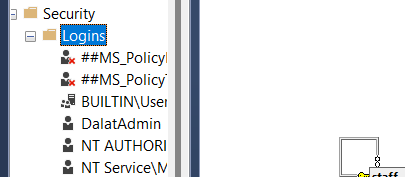
* Tạo user sở hữu database, đặt tên là DalatAdmin, chọn Default database là Dalat



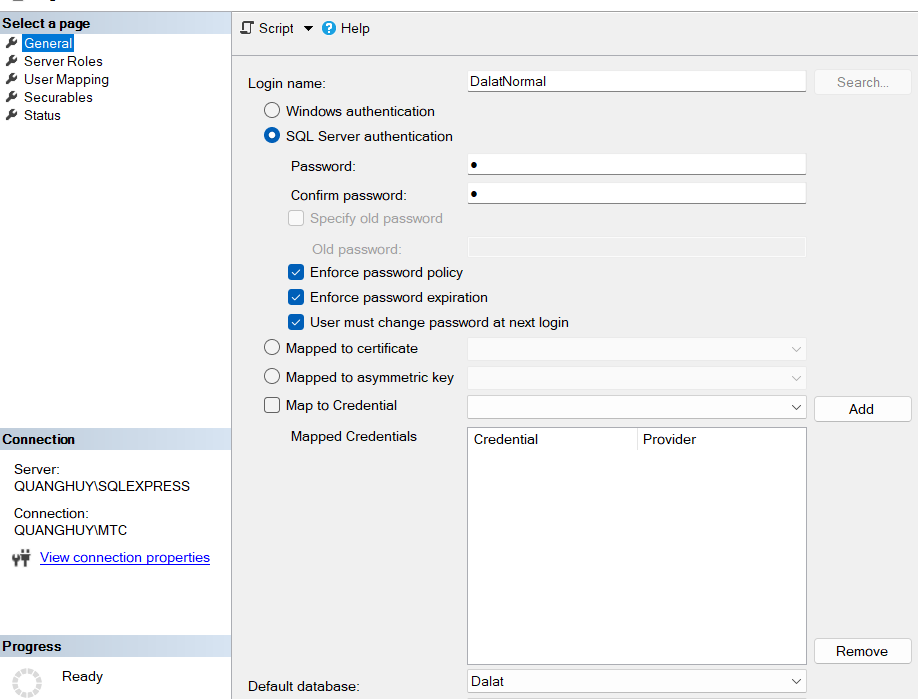
* Ở phần User Mapping, chọn Dalat, chọn db owner để đặt user này làm chủ sở hữu, có nghĩa là có toàn quyền trong database Dalat



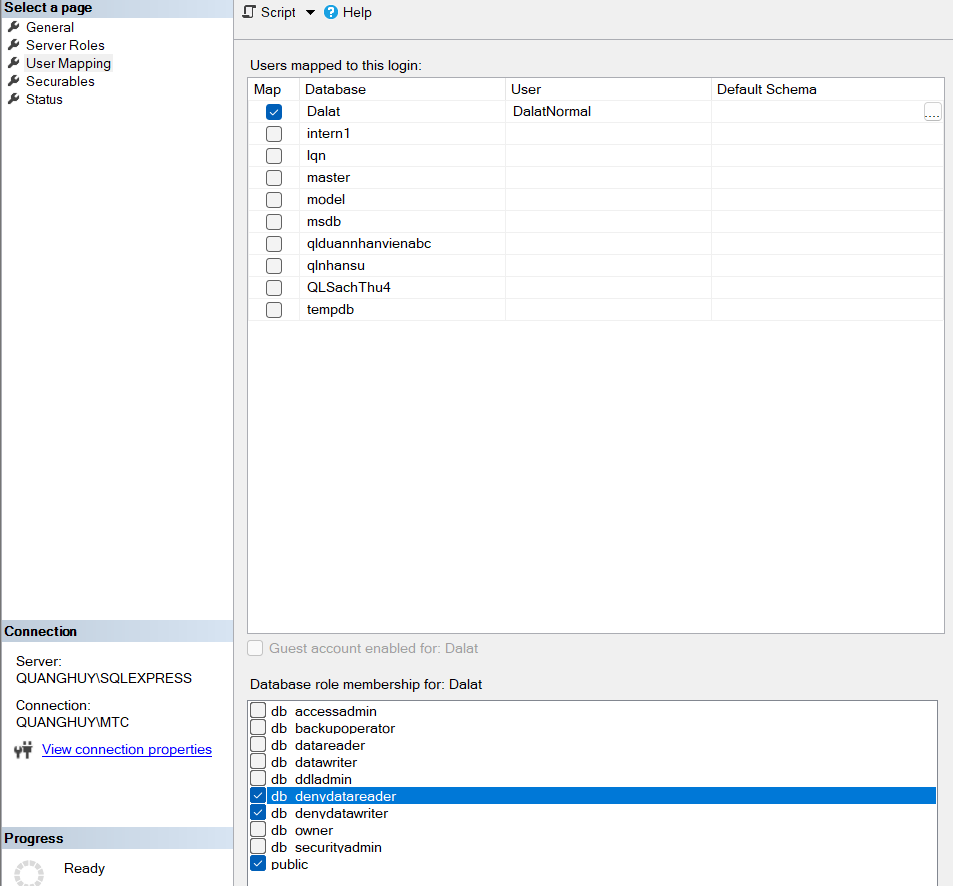
* Nhấn OK



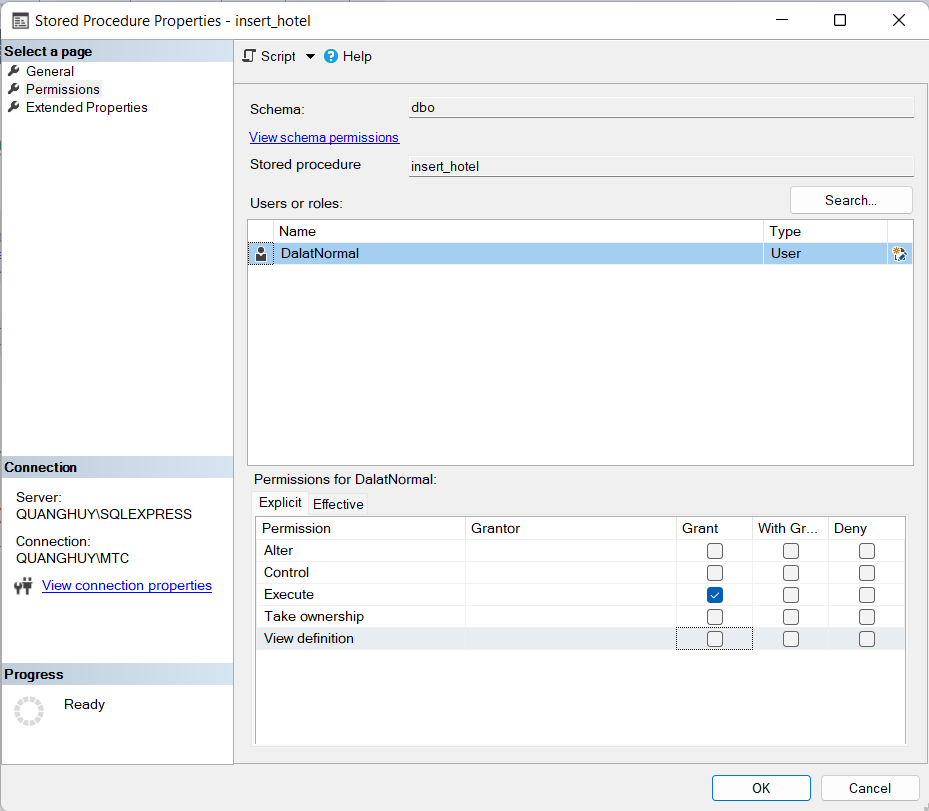
* Tạo User chỉ được quyền đọc và ghi dữ liệu:



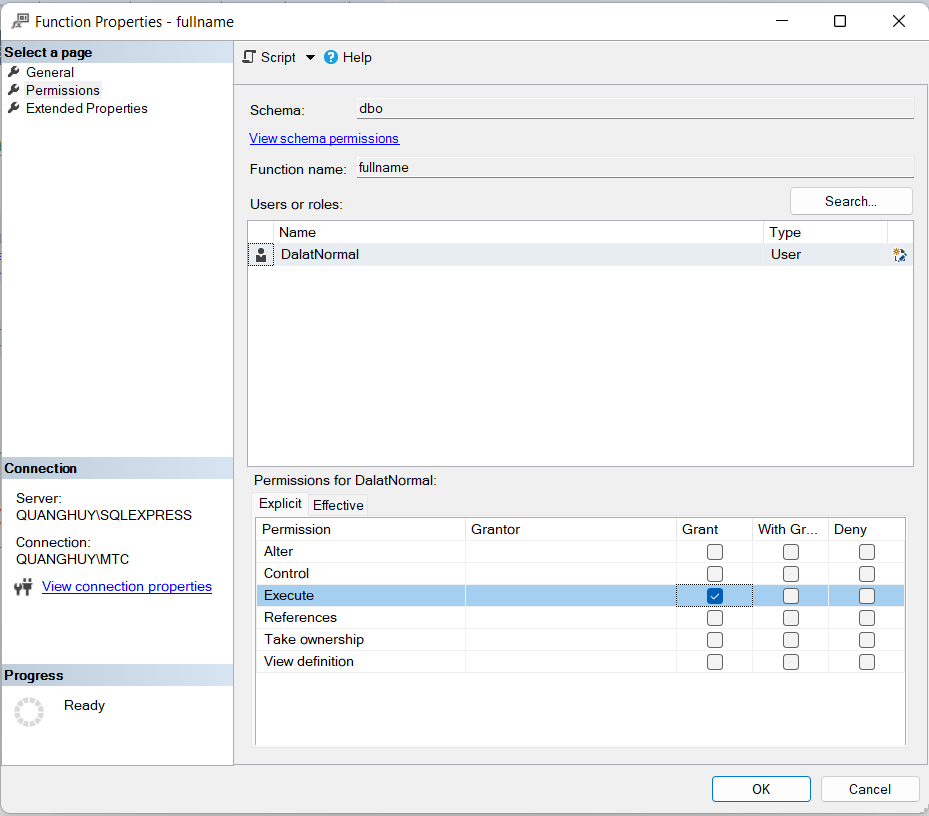
* Ở phần User Mapping, tích vào Dalat và cấp quyền như ở dưới



* Cấp quyền dùng chỉ được execute proceduce insert\_hotel



* Cấp quyền dùng chỉ được execute function dbo.fullname



## 13.Sao lưu dữ liệu

DECLARE @name VARCHAR(50)  
DECLARE @path VARCHAR(256)  
DECLARE @filename VARCHAR(256)  
DECLARE @fileDate VARCHAR(20)

set @path = ‘C:\Backup\’

SELECT @fileDate = CONVERT(VARCHAR(20), GETDATE(), 112)

DECLARE db\_cusor CURSOR FOR  
SELECT name  
FROM master.dbo.sysdatabases

OPEN db\_cursor  
FETCH NEXT FROM db\_cursor INTO @name

WHILE @@FETCH\_STATUS = 0  
BEGIN  
SET @fileName = @path + @name + ‘\_’ + @fileDate + ‘.BAK’  
BACKUP DATABASE @name TO DISK = @fileName

FETCH NEXT FROM db\_cursor INTO @name   
END

CLOSE db\_cursor  
DEALLOCATE db\_cursor