

# THỰC HÀNH CƠ SỞ DỮ LIỆU

*Tài liệu tham khảo*





















# Phần 1

## NGÔN NGỮ TRUY VẤN SQL

# GIỚI THIỆU

- SQL (Structured Query Language) là ngôn ngữ truy vấn mang tính cấu trúc dùng để thực thi các thao tác tạo, sửa và lấy dữ liệu từ một hệ CSDL quan hệ (Relational Database Management System – **RDBMS**).
- Các hệ quản trị CSDL quan hệ phổ biến:
  - + Oracle
  - + MySQL
  - + SQL Server
  - + .....

# Bảng xếp hạng CSDL DB-Engine

Rank			DBMS	Database Model	Score		
Nov 2019	Oct 2019	Nov 2018			Nov 2019	Oct 2019	Nov 2018
1.	1.	1.	Oracle 	Relational, Multi-model 	1336.07	-19.81	+34.96
2.	2.	2.	MySQL 	Relational, Multi-model 	1266.28	-16.78	+106.39
3.	3.	3.	Microsoft SQL Server 	Relational, Multi-model 	1081.91	-12.81	+30.36
4.	4.	4.	PostgreSQL 	Relational, Multi-model 	491.07	+7.16	+50.83
5.	5.	5.	MongoDB 	Document, Multi-model 	413.18	+1.09	+43.70
6.	6.	6.	IBM Db2 	Relational, Multi-model 	172.60	+1.83	-7.27
7.	7.	 8.	Elasticsearch 	Search engine, Multi-model 	148.40	-1.77	+4.94
8.	8.	 7.	Redis 	Key-value, Multi-model 	145.24	+2.32	+1.06
9.	9.	9.	Microsoft Access	Relational	130.07	-1.10	-8.36
10.	10.	 11.	Cassandra 	Wide column	123.23	+0.01	+1.48

Nguồn: <https://db-engines.com/en/ranking>

# CÁC THAO TÁC VỚI CSDL QUAN HỆ

## — Định nghĩa dữ liệu:

1. Tạo bảng (*CREATE TABLE*).
2. Sửa bảng (*ALTER TABLE*).
3. Xoá bảng (*DROP TABLE*).
4. Ràng buộc (*CONSTRAINT*).
5. Trigger.

## — Thao tác dữ liệu:

1. Thêm (*INSERT*).
2. Xoá (*DELETE*).
3. Sửa (*UPDATE*).
4. Lấy dữ liệu (*SELECT*)

# NGÔN NGỮ ĐỊNH NGHĨA DỮ LIỆU

# 1. TẠO BẢNG

— Cú pháp:

```
CREATE TABLE <tên bảng> (  
    <tên cột 1> <kiểu dữ liệu cột 1>,  
    <tên cột 2> <kiểu dữ liệu cột 2>,  
    ...  
)
```

# 1. TẠO BẢNG

```
VD: CREATE TABLE KHACHHANG (  
    MAKH CHAR(4) ,  
    HOTEN VARCHAR(100) ,  
    SODT CHAR(9) ,  
    NGSINH SMALLDATETIME ,  
    DOANHISO MONEY  
)
```



# 1. TẠO BẢNG

— Các kiểu dữ liệu:

- + Kiểu chuỗi (string): CHAR(n), VARCHAR(n).
- + Kiểu luận lý (boolean): TRUE/FALSE.
- + Kiểu ngày tháng (date): DATE, TIMESTAMP.
- + Kiểu số (numeric): INT, FLOAT, NUMERIC(i,j)

## 2. SỬA BẢNG

— Thêm một cột (trường) mới vào bảng:

```
ALTER TABLE <tên bảng>
```

```
ADD <tên cột mới> <kiểu dữ liệu>
```

VD:

```
ALTER TABLE KHACHHANG
```

```
ADD DCHI VARCHAR(50)
```

## 2. SỬA BẢNG

— Sửa dữ liệu của một trường:

ALTER TABLE <tên bảng>

ALTER COLUMN <tên cột> <kiểu dữ liệu mới>

VD:

ALTER TABLE KHACHHANG

ALTER COLUMN HOTEN VARCHAR(200)

## 2. SỬA BẢNG

— Xoá một trường (cột) trong bảng:

ALTER TABLE *<tên bảng>*

DROP COLUMN *<tên cột>*

VD:

ALTER TABLE KHACHHANG

DROP COLUMN DCHI

# 3. XOÁ BẢNG

— Cú pháp:

DROP TABLE < *tên bảng* >

VD: DROP TABLE KHACHHANG

— Lưu ý:

- + Lệnh xoá bảng sẽ xoá tất cả dữ liệu hiện có trong bảng.
- + Đối với các bảng có nhiều khoá ngoại liên kết đến thì lệnh xoá bảng sẽ gây lỗi trong quá trình thực thi.

## 4. RÀNG BUỘC

— Thêm ràng buộc:

```
ALTER TABLE <tên bảng>
```

```
ADD CONSTRAINT <tên ràng buộc> <ràng buộc>
```

VD: Ràng buộc kiểm tra HOTEN không được bỏ trống:

```
ALTER TABLE KHACHHANG
```

```
ADD CONSTRAINT chk_hoten CHECK (HOTEN IS NOT NULL)
```

— Các ràng buộc thường gặp: *CHECK, PRIMARY KEY, FOREIGN KEY.*

## 4. RÀNG BUỘC

— Xoá ràng buộc:

```
ALTER TABLE <tên bảng>
```

```
DROP CONSTRAINT <tên ràng buộc>
```

VD:

```
ALTER TABLE KHACHHANG
```

```
DROP CONSTRAINT chk_hoten
```

# 4. RÀNG BUỘC

— Ràng buộc khoá chính (primary key):

```
ALTER TABLE <tên bảng>
```

```
ADD CONSTRAINT <tên khoá chính> PRIMARY KEY (<tên  
cột>)
```

VD:

```
ALTER TABLE KHACHHANG
```

```
ADD CONSTRAINT pk_kh PRIMARY KEY (MAKH)
```



## 4. RÀNG BUỘC

— Ràng buộc khoá ngoại (foreign key):

```
ALTER TABLE <tên bảng>
```

```
ADD CONSTRAINT <tên khoá ngoại> FOREIGN KEY (<tên cột>)
```

```
REFERENCES <tên bảng liên kết>(<tên cột khoá chính>)
```

VD: Bảng KHACHHANG có MAHD liên kết khoá ngoại tới  
HOADON

```
ALTER TABLE KHACHHANG
```

```
ADD CONSTRAINT fk_kh_hd FOREIGN KEY (MAHD) REFERENCES  
HOADON (MAHD)
```

# NGÔN NGỮ THAO TÁC DỮ LIỆU

# 1. THÊM DỮ LIỆU

— Cú pháp:

```
INSERT INTO <tên bảng> (<tên cột 1>, <tên cột 2>,  
..., <tên cột n>)  
VALUES (<giá trị 1>, <giá trị 2>, ..., <giá trị n>)
```

VD:

```
INSERT INTO KHACHHANG (MAKH, HOTEN) VALUES ('KH01',  
'Nguyen Van A')
```

## 2. CẬP NHẬT DỮ LIỆU

— Cú pháp:

UPDATE <tên bảng>

SET <tên cột> = <giá trị mới>

WHERE <điều kiện>

VD: Cập nhật lại họ tên Bui Thi C cho khách hàng có mã là KH03

```
UPDATE KHACHHANG
```

```
SET HOTEN = 'Bui Thi C'
```

```
WHERE MAKH='KH03'
```

# 3. XOÁ DỮ LIỆU

— Cú pháp:

DELETE FROM <tên bảng>

[WHERE <điều kiện>]

Lưu ý: Nếu như không có mệnh đề WHERE thì sẽ xoá dữ liệu toàn bộ bảng.

VD: Xoá khách hàng có thu nhập dưới 5000

DELETE FROM KHACHHANG WHERE DOANHSON < 5000

VD: Xoá toàn bộ dữ liệu trong bảng KHACHHANG

DELETE FROM KHACHHANG hoặc DELETE FROM KHACHHANG  
WHERE 1

## 4. LẤY DỮ LIỆU

— Cú pháp câu truy vấn **SELECT**:

**SELECT** <cột 1>, <cột 2>, ....

**FROM** <tên bảng>

**WHERE** <điều kiện>

**ORDER BY** <tên cột> **ASC** | **DESC**

**GROUP BY** <tên cột 1>, <tên cột 2>, ....

**HAVING** <điều kiện>

— Lưu ý:

+ Mệnh đề **HAVING** sử dụng cho các **hàm gom nhóm**.

+ **ASC** – sắp xếp tăng dần; **DESC** – sắp xếp giảm dần.

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— **Dạng 1:** Truy vấn lấy dữ liệu tất cả.

```
SELECT * FROM <tên bảng>
```

hoặc

```
SELECT <danh sách cột> FROM <tên bảng>
```

**VD:**

```
SELECT * FROM KHACHHANG
```

```
SELECT MAKH, HOTEN, DCHI FROM KHACHHANG
```

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— **Dạng 2:** Truy vấn dữ liệu có điều kiện:

```
SELECT <danh sách cột> FROM <tên bảng>  
WHERE <điều kiện>
```

VD: Lấy thông tin MAKH, HOTEN DCHI của khách hàng có doanh số trên 1000

```
SELECT MAKH, HOTEN, DCHI FROM KHACHHANG  
WHERE DOANHSHO > 10000
```



# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— **Dạng 3:** Truy vấn dữ liệu có kết bảng.

SELECT <danh sách cột> FROM <tên bảng 1>

INNER JOIN <tên bảng 2> ON <tên bảng 1>.<mã khoá ngoại> = <tên bảng 2>.<mã khoá chính>

[WHERE <điều kiện>]

— Các phép kết:

+ **INNER JOIN:** kết bằng.

+ **LEFT OUTER JOIN:** kết mở rộng về bên trái.

+ **RIGHT OUTER JOIN:** kết mở rộng về bên phải.

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

VD:

```
SELECT MAKH, HOTEN  
FROM HOADON INNER JOIN KHACHHANG ON KHACHHANG.MAKH  
= HOADON.MAKH  
WHERE NGHD = '16/7/2019'
```

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— **Dạng 4:** Truy vấn dữ liệu có sắp xếp

SELECT <danh sách tên cột> FROM <tên bảng>

[WHERE <điều kiện>]

ORDER BY <danh sách cột cần sắp xếp> ASC hoặc DESC

VD: Sắp xếp khách hàng theo ngày sinh giảm dần

SELECT MAKH, HOTEN, NGSINH FROM KHACHHANG

ORDER BY NGSINH DESC

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— **Dạng 5:** Truy vấn sử dụng các hàm gom nhóm

SELECT <các hàm gom nhóm> FROM <tên bảng>

[WHERE <điều kiện>]

GROUP BY <tên cột 1>, <tên cột 2>, ...

— Các hàm gom nhóm: *COUNT()*, *AVG()*, *MAX()*, *MIN()*, *SUM()*.

— **Lưu ý:** Các thuộc tính trong mệnh đề SELECT (trừ các hàm kết hợp), phải xuất hiện trong mệnh đề **GROUP BY**.

## 5. CÁC DẠNG CÂU TRUY VẤN SELECT

VD: Tính giá trị trung bình doanh số theo từng MAKH đối với khách hàng có doanh số trên 10000

```
SELECT MAKH, AVG(DOANHSO) FROM KHACHHANG  
WHERE DOANHSO > 10000  
GROUP BY MAKH
```

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— **Dạng 6:** Truy vấn sử dụng hội - giao - trừ

```
SELECT <danhsách cột 1> FROM <tên bảng>  
[WHERE <điều kiện 1>]
```

UNION (hội) | INTERSECT (giao) | EXCEPT (trừ)

```
SELECT <danhsách cột 2> FROM <tên bảng>  
[WHERE <điều kiện 2>]
```

**Lưu ý:** Để sử dụng các phép hội giao trừ thì 2 quan hệ phải **khả hợp**,  
tức *<danhsách cột 1> = <danhsách cột 2>*

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

VD1: Tìm khách hàng mua hoá đơn HD01 hoặc HD02

```
SELECT MAKH, HOTEN FROM KHACHHANG INNER JOIN  
HOADON ON KHACHHANG.MAKH = HOADON.MAKH
```

```
WHERE MAHD = 'HD01'
```

UNION

```
SELECT MAKH, HOTEN FROM KHACHHANG INNER JOIN  
HOADON ON KHACHHANG.MAKH = HOADON.MAKH
```

```
WHERE MAHD = 'HD02'
```

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

VD2: Tìm khách hàng mua cùng lúc hoá đơn HD01 và HD02

```
SELECT MAKH, HOTEN FROM KHACHHANG INNER JOIN  
HOADON ON KHACHHANG.MAKH = HOADON.MAKH
```

```
WHERE MAHD = 'HD01'
```

INTERSECT

```
SELECT MAKH, HOTEN FROM KHACHHANG INNER JOIN  
HOADON ON KHACHHANG.MAKH = HOADON.MAKH
```

```
WHERE MAHD = 'HD02'
```



# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

VD3: Tìm khách hàng không mua hoá đơn nào

```
SELECT MAKH, HOTEN FROM KHACHHANG
```

```
EXCEPT
```

```
SELECT MAKH, HOTEN FROM KHACHHANG INNER JOIN  
HOADON ON KHACHHANG.MAKH = HOADON.MAKH
```

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— **Dạng 7:** Truy vấn lồng:

```
SELECT <danh sách cột> FROM <tên bảng>  
WHERE <so sánh tập hợp> (  
    SELECT <danh sách cột> FROM <tên bảng>  
    WHERE <điều kiện>  
)
```

<so sánh tập hợp>: *ALL, IN, NOT IN, ALL, ANY, EXISTS, NOT EXISTS.*

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

VD: Tìm thông tin mã hoá đơn có trị giá cao nhất

```
SELECT SOHD FROM HOADON
```

```
WHERE TRIGIA = (
```

```
    SELECT MAX(TRIGIA) FROM HOADON
```

```
    GROUP BY SOHD
```

```
)
```

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

## — Dạng 8: Truy vấn lồng tương quan

```
SELECT <danh sách cột> FROM <tên bảng> AS OB1  
WHERE <so sánh tập hợp> (  
    SELECT <danh sách cột> FROM <tên bảng> AS OB2  
    WHERE OB1.<tên cột> = OB2.<tên cột>  
)
```

<so sánh tập hợp>: *ALL, IN, NOT IN, ALL, ANY, EXISTS, NOT EXISTS.*

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

VD: Tìm sản phẩm có giá cao nhất theo từng nước sản xuất.

```
SELECT NUOCSX FROM SANPHAM AS SP1  
WHERE GIA = (  
    SELECT MAX(GIA) FROM SANPHAM AS SP2  
    WHERE SP1.MASP = SP2.MASP  
)
```

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— Dạng 9: Truy vấn dùng bảng “con” (inner aggregate).

```
SELECT <danh sách cột 1> FROM (  
    SELECT <danh sách cột 2> FROM <tên bảng>  
    WHERE <điều kiện>  
) AS <tên bảng con>
```

Lưu ý: <danh sách cột 1> phụ thuộc vào <danh sách cột 2> trả về từ câu truy vấn con (subquery).

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

VD: Tìm khách hàng có số lần mua hàng nhiều nhất.

```
SELECT MAKH FROM HOADON
```

```
GROUP BY MAKH
```

```
HAVING COUNT(SOHD) = (
```

```
    SELECT MAX(SL_HD) FROM (
```

```
        SELECT MAKH, COUNT(SOHD) AS SL_HD
```

```
        FROM HOADON
```

```
        GROUP BY MAKH) AS T
```

```
)
```

Lưu ý: Không thể sử dụng dạng *COUNT (MAX (SOHD) )* trong SQL Server

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— Dạng 10: Phép chia

Tìm <đối tượng 1> đã ... tất cả <đối tượng 2>

Cần xác định:

Đối tượng 1 (MaDT1, ....).

Đối tượng 2 (MaDT2, ....).

Quan hệ Đối tượng 1 và đối tượng 2 (*MaDT1*, *MaDT2*, ....).



# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

```
SELECT <danh sách cột> FROM <tên bảng đối tượng 1>  
AS OB1 WHERE NOT EXISTS (  
    SELECT <danh sách cột> FROM <tên bảng đối  
tượng 2> AS OB2 WHERE AND NOT EXISTS (  
    SELECT * FROM <tên bảng quan hệ đối tượng  
1 và 2> as OB3 WHERE OB2.<khoá chính> = OB3.<khoá  
ngoại> and OB3.<khoá ngoại> = OB1.<khoá chính>  
    )  
    )  
)
```

# 5. CÁC DẠNG CÂU TRUY VẤN SELECT

— VD: Tìm hoá đơn đã mua tất cả sản phẩm xuất xứ Thái Lan:

*Đối tượng 1:* HOADON(SOHD, NGHD, ...)

*Đối tượng 2:* SANPHAM(MASP, TENSP, XUATXU)

*Quan hệ giữa 2 đối tượng:* CTHD(MASP, TENSP).

```
SELECT SOHD FROM HOADON AS T1 WHERE NOT EXISTS (  
    SELECT MASP FROM SANPHAM AS T2 WHERE XUATXU =  
    "Thái Lan" AND NOT EXISTS (  
        SELECT MASP, TENSP FROM CTHD AS T3 WHERE  
        T2.MASP = T3.MASP AND T1.SOHD = T3.SOHD)  
    )  
)
```

## 6. TRIGGER

— Mục tiêu: Thực hiện một tập các lệnh một cách tự động khi có sự biến động trong CSDL nhằm đảm bảo sự RBTV của dữ liệu.

— Cú pháp:

```
CREATE TRIGGER <tên trigger> ON <tên bảng>  
FOR INSERT | UPDATE  
AS  
BEGIN  
    <các câu lệnh SQL>  
END
```

## 6. TRIGGER

- Khi một trigger thực thi, nó sẽ tạo ra 2 bảng đặc biệt: **inserted** và **deleted**. Hai bảng này chỉ tồn tại trong thời gian thực thi trigger. Chức năng của 2 bảng này như sau:

	<b>inserted</b>	<b>deleted</b>
<b>INSERT</b>	dữ liệu được insert	trống
<b>UPDATE</b>	dữ liệu đã được thay đổi (modified).	dữ liệu hiện tại.
<b>DELETE</b>	<trống>	dữ liệu sẽ bị xoá.

## 6. TRIGGER

— VD: Ngày mua hàng phải lớn hơn ngày đăng ký.

Bối cảnh: DONHANG, KHACHHANG

Nội dung:  $\forall d \in DONHANG, \forall k \in KHACHHANG, k.NGDK < d.NGHD$ .

Bảng tầm ảnh hưởng:

	Thêm	Sửa	Xoá
DONHANG	+ (NGHD)	+ (NGHD)	-
KHACHHANG	-	+ (NGDK)	-

## 6. TRIGGER

- Tạo ra 2 trigger trên 2 bảng:
  - + **HOADON**: Trigger kiểm tra thao tác **thêm** vào 1 hoá đơn mới, và **cập nhật** cho 1 hoá đơn (cụ thể là cập nhật NGHD).
  - + **KHACHHANG**: Trigger kiểm tra việc **cập nhật** ngày đăng ký (NGDK) của một khách hàng.

# 6. TRIGGER

```
CREATE TRIGGER TRG_KH_DH ON HOADON    // Xử lý trên bảng HOADON
FOR INSERT
AS
BEGIN
    DECLARE @NGHD SMALLDATETIME, @NGDK SMALLDATETIME, @MAKH CHAR(4)
    SELECT @NGHD = NGHD, @MAKH = MAKH FROM inserted
    SELECT @NGDK = NGDK FROM KHACHHANG WHERE MAKH=@MAKH
    IF (@NGDK < @NGHD)
        BEGIN
            PRINT 'LOI. NGAY DANG KY PHAI LON HON NGAY MUA'
            ROLLBACK TRANSACTION
        END
    ELSE
        PRINT 'THANH CONG'
END
```

# 6. TRIGGER (tt)

```
CREATE TRIGGER TRG_KH_DH ON KHACHHANG          // Xử lý trên bảng KHACHHANG
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @NGHD SMALLDATETIME, @NGDK SMALLDATETIME, @MAKH CHAR(4)
    SELECT @MAKH = MAKH, @NGDK = NGDK FROM inserted
    SELECT TOP 1 @NGHD = NGDK FROM HOADON WHERE MAKH=@MAKH ORDER BY NGHD DESC

    IF (@NGDK < @NGHD)
        BEGIN
            PRINT 'LOI. NGAY DANG KY PHAI LON HON NGAY MUA'
            ROLLBACK TRANSACTION
        END
    ELSE
        PRINT 'THANH CONG'
END
```

*Lấy ra ngày hoá đơn **lớn nhất** của 1 khách hàng để xét*



## Phần 2

# MỘT SỐ LỖI THƯỜNG GẶP KHI DÙNG SQL SERVER

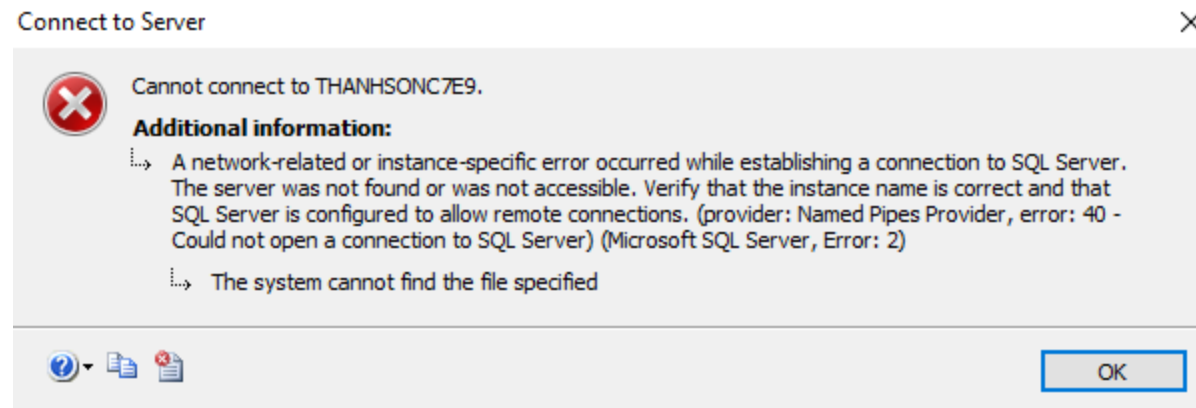
# CÁC LỖI THƯỜNG GẶP

1. Lỗi không thể connect SQL Server.
2. Lỗi liên quan đến câu truy vấn SELECT.
3. Lỗi liên quan đến câu truy vấn INSERT.
4. Lỗi liên quan đến DELETE, DROP.
5. Lỗi liên quan đến ALTER TABLE.
6. Lỗi liên quan đến CREATE TABLE.

# LỖI KHÔNG THỂ CONNECT SQL SERVER

# Lỗi không thể connect SQL Server

— Nguyên nhân: Chưa khởi động SQL Service.



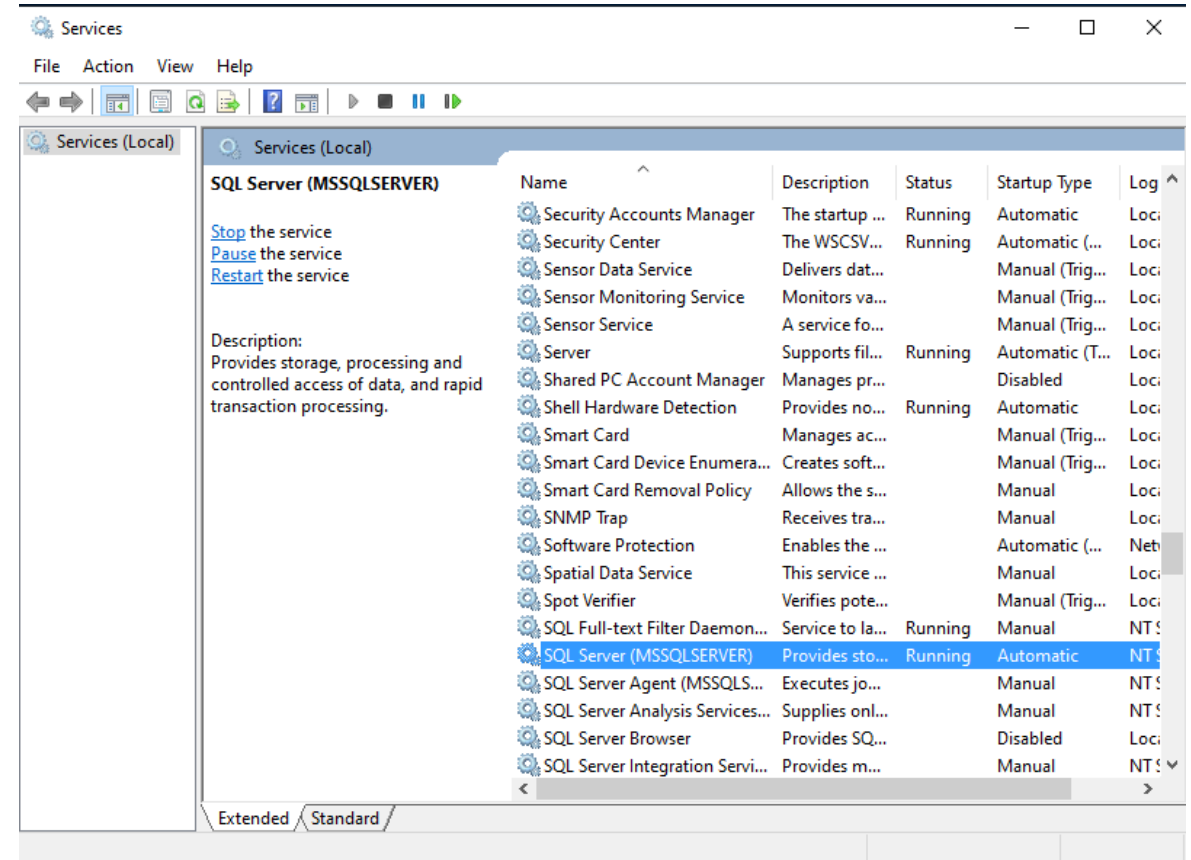
— Cách khắc phục:

# Lỗi không thể connect SQL Server

- Bước 1: Vào Start, ở khung tìm kiếm, gõ “*Services*”. Hoặc nhấn tổ hợp phím Windows + R, gõ “*services.msc*”.

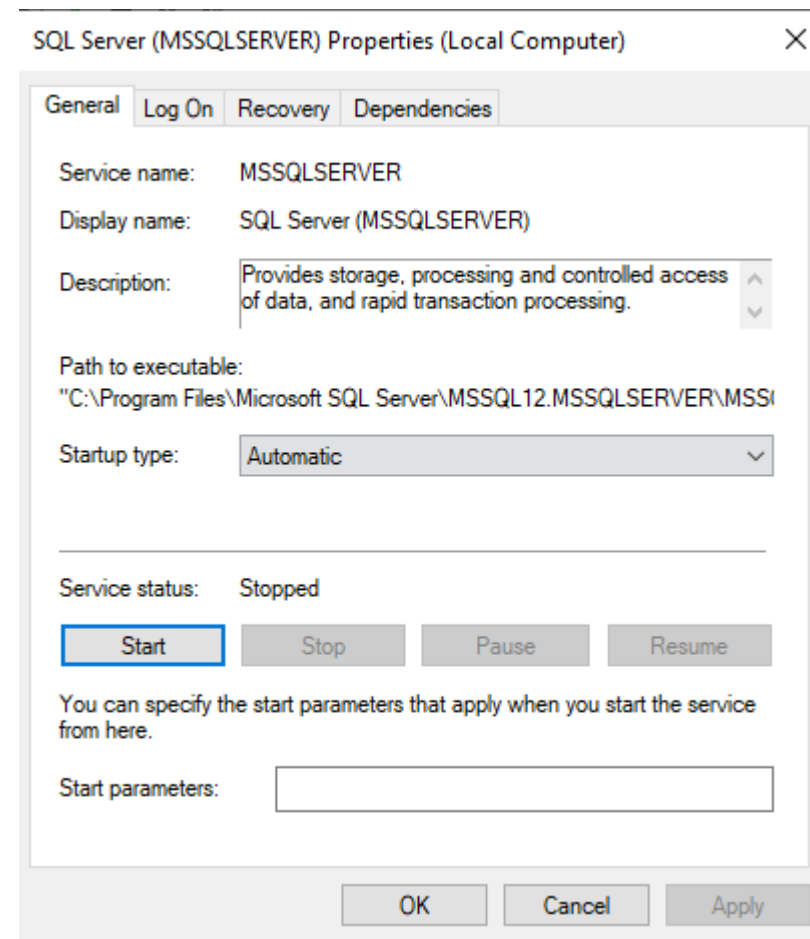
# Lỗi không thể connect SQL Server

- Bước 1: Vào Start, ở khung tìm kiếm, gõ “*Services*”. Hoặc nhấn tổ hợp phím Windows + R, gõ “*services.msc*”.
- Bước 2: Ở hộp thoại Services hiện ra, chọn tên dịch vụ: **SQL Server (MSSSQL)**.



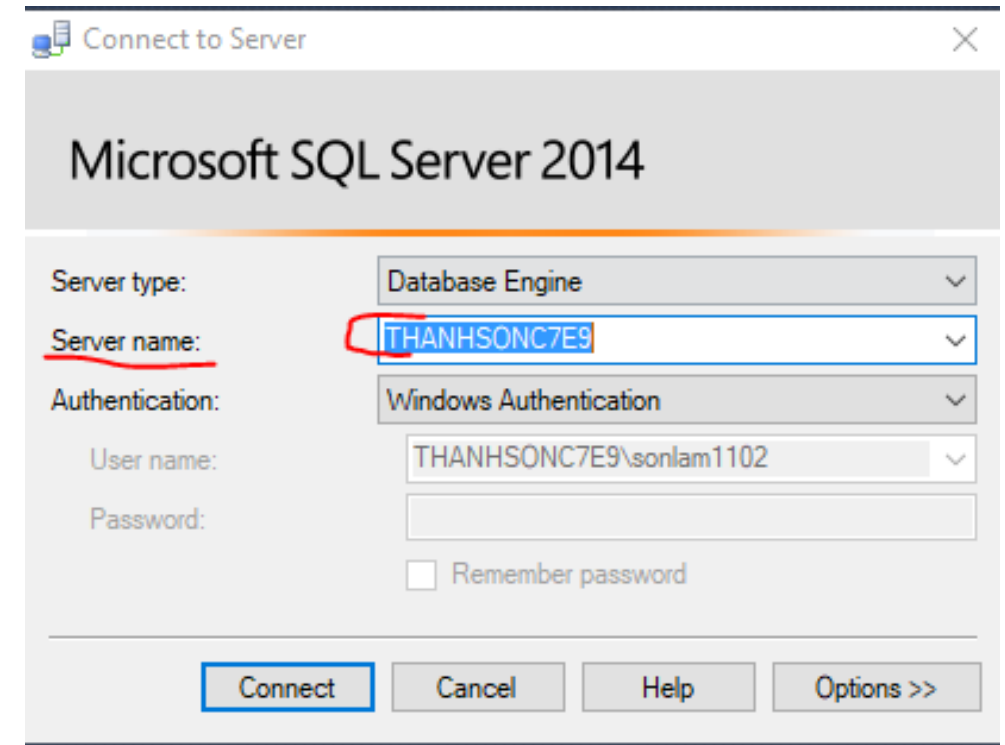
# Lỗi không thể connect SQL Server

- **Bước 3:** Click chuột phải vào tên dịch vụ **SQL Server (MSSQL)**, chọn **Properties**. Màn hình như bên phải sẽ hiện ra.
- **Bước 4:** Nếu **Services status** là **Stopped** thì click nút **Start** và đợi cho đến khi dịch vụ khởi động xong. (Nếu **Status type** là **Disable** thì chọn mũi tên bên phải và đổi lại thành **Automatic**).



# Lỗi không thể connect SQL Server

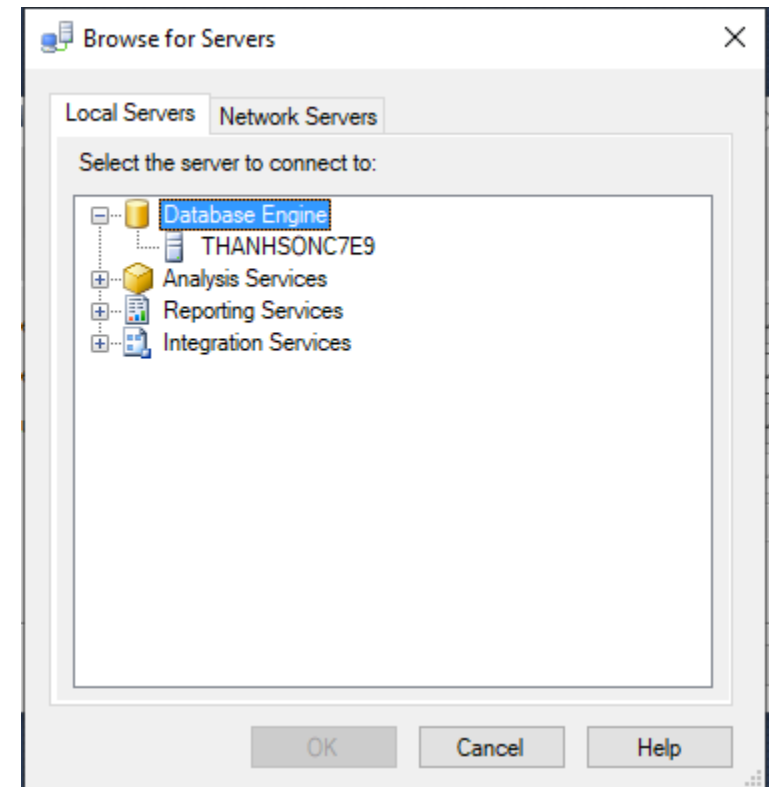
- Bước 5: Khởi động SQL Server.
- Bước 6: Ở hộp thoại Connect như hình bên phải, Mục **Server name**, click dấu mũi tên bên phải và chọn <Browse for more...>





# Lỗi không thể connect SQL Server

- Bước 7: Ở hộp thoại như hình bên phải, Chọn Database Engine và chọn tên máy chủ cần kết nối (theo hình bên là THANSONC79E).
- Bước 8: Nhấn OK. Sau khi về hộp thoại kết nối, nhấn Connect.



# **LỖI LIÊN QUAN ĐẾN CƠ SỞ DỮ LIỆU**

# LỖI LIÊN QUAN ĐẾN CƠ SỞ DỮ LIỆU

- Mô tả lỗi: *Invalid object name 'SANPHAM'*.
- Nguyên nhân: Chưa chọn CSDL tương ứng.
- Các khắc phục: Sử dụng lệnh `USE <tên DB tương ứng>`.

QLBH.sql - THANHSONC7E9.master (THANHSONC7E9\sonlam1102 (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Debug Tools Window Help

master

Object Explorer

Connect

THANHSONC7E9 (SQL Server 12.0.4100)

- Databases
- Security
- Server Objects
- Replication
- AlwaysOn High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)

QLBH.sql - THANHSONC7E9\sonlam1102 (51))

```
/* CAU 27*/
SELECT TOP 3 MAKH, HOTEN FROM KHACHHANG ORDER BY DOANHSO DESC

/*CAU 28*/
SELECT MASP, TENS FROM SANPHAM WHERE GIA IN (SELECT DISTINCT TOP 3 GIA FROM SANPHAM ORDER BY GIA DESC)

/* CAU 29*/
SELECT MASP, TENS FROM SANPHAM WHERE GIA IN (SELECT DISTINCT TOP 3 GIA FROM SANPHAM ORDER BY GIA DESC) AND NUOCSX = 'Thai Lan'

/*CAU 30*/
SELECT MASP, TENS FROM SANPHAM WHERE GIA IN (SELECT DISTINCT TOP 3 GIA FROM SANPHAM ORDER BY GIA DESC) AND NUOCSX = 'Trung Quoc'

/*CAU 31*/
SELECT TOP 3 *, RANK() OVER (ORDER BY DOANHSO DESC) AS DS_RANK FROM KHACHHANG

/*CAU 32*/
SELECT COUNT(MASP) AS SPCHINA FROM SANPHAM WHERE NUOCSX = 'Trung Quoc'

/*CAU 33 */
SELECT NUOCSX, COUNT(MASP) AS SL_NUOCSX FROM SANPHAM GROUP BY NUOCSX
```

Messages

Msg 208, Level 16, State 1, Line 553  
Invalid object name 'SANPHAM'.

Query completed with errors.

THANHSONC7E9 (12.0 SP1) | THANHSONC7E9\sonlam110... | master | 00:00:00 | 0 rows

Ready

Ln 2 Col 31 Ch 31 INS

**master** là CSDL  
mặc định của SQL  
Server.  
Nếu muốn sử  
dụng đúng CSDL  
của mình phải sử  
dụng lệnh  
**USE <tên CSDL>**

# **LỖI LIÊN QUAN ĐẾN CÂU TRUY VẤN SELECT**

# Lỗi 1

- Câu truy vấn: `SELECT * FROM KHACHHANG.`
  - Lỗi: *Invalid object name 'KHACHHANG'.*
  - Nguyên nhân: Trong CSDL chưa có bảng KHACHHANG.
  - Cách khắc phục:
    1. Tạo lại bảng KHACHHANG.
    2. Kiểm tra xem CSDL mình đang thao tác có đúng hay không? Nếu không thì dùng lệnh: *USE <tên CSDL>* để sử dụng đúng CSDL của mình.
- VD: Nếu trước đó đã có CSDL tên là: QLBNH thì sử dụng: **USE QLBNH**

## Lỗi 2

- Câu truy vấn: `SELECT MASP, TENS P FROM CTHD INNER JOIN SANPHAM ON SANPHAM.MASP = CTHD.MASP.`
- Lỗi: *Ambiguous column name MASP.*
- Nguyên nhân: Cột **MASP** tồn tại trên 2 quan hệ là **SANPHAM** và **CTHD**.
- Cách khắc phục: Khai báo thêm tên quan hệ chứa thuộc tính **MASP**.

```
SELECT SANPHAM.MASP, TENS P FROM CTHD INNER JOIN  
SANPHAM ON SANPHAM.MASP = CTHD.MASP
```

# Lỗi 3

- Câu truy vấn: `SELECT COUNT(MASP) FROM CTHD.`
- Lỗi: *Column CTHD.MASP is invalid in the select list because it is not contained in an aggregate function and there is no GROUP BY clause.*
- Nguyên nhân: Cần sử dụng mệnh đề gom nhóm `GROUP BY` vì câu truy vấn select muốn đếm số sản phẩm theo từng hoá đơn (phép count là phép toán gom nhóm).
- Cách khắc phục:  
`SELECT COUNT(MASP) FROM CTHD GROUP BY SOHD`



# Lỗi 4

- Câu truy vấn: `SELECT MASP FROM SP WHERE MASP IN (SELECT * FROM CTHD).`
- Lỗi: *Only one expression can be specified in the select list when the subquery is not introduced with EXIST.*
- Nguyên nhân: Dùng toán tử `IN` để so sánh 2 biểu thức không tương đương.
- Cách khắc phục:  
`SELECT MASP FROM SP WHERE MASP IN (SELECT MASP FROM CTHD)`

# Lỗi 5

— Câu truy vấn: `SELECT SOHD, MANV, MAKH FROM HOADON WHERE 20000 < TRIGIA < 10000000.`

— Lỗi: *Incorrect syntax near '<'.*

— Nguyên nhân: Sai cú pháp.

— Cách khắc phục:

`SELECT SOHD, MANV, MAKH FROM HOADON WHERE TRIGIA > 2000000 AND TRIGIA < 10000000.`

Hoặc

`SELECT SOHD, MANV, MAKH FROM HOADON WHERE TRIGIA BETWEEN 2000000 AND 10000000`

## Lỗi 6

- Câu truy vấn: `SELECT MASP FROM SP WHERE MASP = (SELECT MASP FROM CTHD)`.
- Lỗi: *Subquery return more than 1 value. This is not permitted when the subquery follows =, !=, <, <=, >, >= or when the subquery is used as an expression.*
- Nguyên nhân: Không thể dùng các toán tử so sánh như: =, !=, <, <=, >, >= khi câu truy vấn phụ trả về NHIỀU HƠN 1 GIÁ TRỊ .
- Cách khắc phục: Đối với trường hợp trả về nhiều hơn một giá trị, ta sử dụng toán tử IN.

`SELECT MASP FROM SP WHERE MASP IN (SELECT MASP FROM CTHD)`

# **LỖI LIÊN QUAN ĐẾN CÂU TRUY VẤN INSERT**

# Lỗi 1

- Câu truy vấn: INSERT INTO NHANVIEN (MANV, HOTEN, SODT, NGVL) VALUES ('NV01', 'Nguyen Nhu Nhut', '0927345678', '13/4/2006').
- Lỗi: *Violation of PRIMARY KEY constrain 'PK\_NV\_xxxx', Cannot insert duplicate key in object 'NHANVIEN'. The statement has been terminated.*
- Nguyên nhân: Đã có dữ liệu trong bảng NHANVIEN với mã số NV01 trước đó => vi phạm ràng buộc khoá chính.
- Cách khắc phục: Kiểm tra xem dữ liệu nhân viên có tồn tại trong CSDL không. Hoặc là đổi lại mã số nhân viên (MANV) khác.

## Lỗi 2

- Câu truy vấn: INSERT INTO NHANVIEN (MANV, HOTEN, SODT, NGVL) VALUES ('NV01', 'Nguyen Nhu Nhut', '13/4/2006')
- Lỗi: *Insert Error: Column name or number of supplied values does not match table definition.*
- Nguyên nhân: Đưa dữ liệu vào bảng NHANVIEN không đủ số cột (theo ví dụ trên là thiếu dữ liệu cho cột SODT).
- Cách khắc phục: Kiểm tra lại kỹ dữ liệu đã nhập vào. LƯU Ý dữ liệu nhập vào phải theo THỨ TỰ các cột khai báo phía trước. VD:

MANV	HOTEN	SODT	NGVL
'NV01'	'Nguyen Nhu Nhut'	'0927345678'	'13/4/2006'

# Lỗi 3

- Câu truy vấn: INSERT INTO NHANVIEN (MANV, HOTEN, SODT, NGVL) **VALUE** ('NV01', 'Nguyen Nhu Nhut', '0927345678', '13/4/2006')
  - Lỗi: *Incurrect syntax near 'value'.*
  - Nguyên nhân: Sai cú pháp từ khoá value.
  - Cách khắc phục: Sửa lại thành Values
- ```
INSERT INTO NHANVIEN (MANV, HOTEN, SODT, NGVL)  
VALUES ('NV01', 'Nguyen Nhu Nhut', '0927345678', '13/4/2006')
```

# Lỗi 4

- Câu truy vấn: INSERT INTO NHANVIEN (MANV, HOTEN, SODT, NGVL) VALUES ('NV001', 'Nguyen Nhu Nhat', '13/4/2006')
- Lỗi: *String or binary data would be truncated. The statement has been terminated.*
- Nguyên nhân: Quá chiều dài cho phép. Chiều dài cho phép của cột MANV là 4 (CHAR(4)). 'NV001' có 5 ký tự.
- Cách khắc phục: Kiểm tra lại độ dài của chuỗi khi insert vào bảng theo cột tương ứng.



# Lỗi 5

- Câu truy vấn: INSERT INTO CTHD (SOHD, MASP, SL) VALUES (1001, 'TV02', 12)
- Lỗi: *INSERT statement conflicted with COLUMN FOREIGN KEY constrain FK\_CTHD\_SP. The conflict occurred in database QLBH, table 'SANPHAM', column 'MASP'.*
- Nguyên nhân: Vi phạm ràng buộc khoá ngoại và khoá chính: sản phẩm có mã sản phẩm TV02 chưa có trong bảng SANPHAM.
- Cách khắc phục: Thêm dữ liệu sản phẩm mới có mã sản phẩm là TV02 cho bản sản phẩm.

# Lỗi 6

- Câu truy vấn: `INSERT INTO NHANVIEN (MANV, HOTEN, SODT, NGVL) VALUES ('NV01', 'Nguyen Nhu Nhat', '13/4/2006')`
- Lỗi: *The conversion of char data type to smalldatetime data type resulted in an out-of-range smalldatetime value.*
- Nguyên nhân: Kiểu định dạng ngày tháng không phù hợp trong SQL Server. (SQL Server có kiểu định dạng ngày tháng là: YYYY-MM-DD H:m:s).
- Cách khắc phục: Sử dụng lệnh `SET DATEFORMAT DMY.`

# LỖI LIÊN QUAN ĐẾN DELETE, DROP

# Lỗi 1

- Câu truy vấn: DELETE FROM SANPHAM
- Lỗi: *DELETE statement conflicted with COLUMN REFERENCE constraint PK\_CTHD\_SP. The conflict occurred in database 'QLBH', table 'CTHD', column 'MASP'.*
- Nguyên nhân: Vi phạm ràng buộc khoá ngoại. Các sản phẩm trong bảng SANPHAM có ràng buộc khoá ngoại với bảng CTHD.
- Cách khắc phục: Xóa dữ liệu trong bảng CTHD trước, sau đó mới xóa bảng SANPHAM.

## Lỗi 2

- Câu truy vấn: DROP TABLE SANPHAM
- Lỗi: *Could not drop object 'SANPHAM' because it is referenced by a FOREIGN KEY constraint..*
- Nguyên nhân: Bảng SANPHAM có liên kết khoá ngoại với bảng CTHD.
- Cách khắc phục: Xóa bảng CTHD trước, sau đó mới xóa bảng SANPHAM.

# LỖI LIÊN QUAN ĐẾN ALTER TABLE

# Lỗi 1

- Câu truy vấn: ALTER TABLE NHANVIEN ALTER COLUMN DCHI VARCHAR(30)
- Lỗi: *String or binary data would be truncated. The statement has been terminated.*
- Nguyên nhân: Thay đổi chiều dài của một cột mà dữ liệu tồn tại trên cột đó bị vi phạm. Như ví dụ trên, dữ liệu DCHI trong bảng NHANVIEN hiện tại là 50, khi thay đổi xuống 30 sẽ bị lỗi.
- Cách khắc phục: Xóa dữ liệu đi hoặc sửa lại dữ liệu với độ dài thích hợp nếu muốn thay đổi cột.

## Lỗi 2

- Câu truy vấn: ALTER TABLE KHACHHANG ADD GHICHU VARCHAR(50) NOT NULL
- Lỗi: *ALTER TABLE only allows columns to be added that can contain nulls or have a DEFAULT definition specified. Column 'GHICHU' cannot be added to table 'KHACHHANG' because it does not allow nulls and does not specify a DEFAULT definition.*
- Nguyên nhân: Không thể thêm một cột với ràng buộc NOT NULL vào một bảng đã có dữ liệu trước đó.
- Cách khắc phục: Bỏ đi NOT NULL  
ALTER TABLE KHACHHANG ADD GHICHU VARCHAR(50)



## Lỗi 3

- Câu truy vấn: ALTER TABLE CTHD ADD CONSTRAINT FK\_CTHD\_SP FOREIGN KEY (MASP) REFERENCES SANPHAM(MASP).
- Lỗi: Column 'CTHD.MASP' is not the same length as referencing column 'SANPHAM.MASP' in foreign key FK\_CTHD\_SP .
- Nguyên nhân: Khoá chính MASP trong bảng SANPHAM và khoá ngoại MASP trong CTHD phải cùng kiểu dữ liệu và chiều dài.
- Cách khắc phục: Sửa lại kiểu dữ liệu trên 2 cột giống nhau.

# TÀI LIỆU THAM KHẢO

1. Đồng Thị Bích Thủy, Nguyễn Trần Minh Thư, Phạm Thị Bạch Huệ, *Giáo trình Cơ sở dữ liệu*, NXB Khoa học và Kỹ thuật, 2010.
2. Đỗ Phúc, Nguyễn Đăng Ty, *Giáo trình Cơ sở dữ liệu*, NXB ĐHQG.TPHCM, 2010.
3. Nguyễn Gia Tuấn Anh, Trương Châu Long, *Bài tập và bài giải SQL Server*, NXB Thanh niên, 2005.
4. W3schools: <https://www.w3schools.com/sql/default.asp>