



# Giới Thiệu Sơ Lược Về Transact SQL (T-SQL)

Bởi:

Khoa CNTT ĐHSP KT Hưng Yên

Transact-SQL là ngôn ngữ SQL mở rộng dựa trên SQL chuẩn của ISO (International Organization for Standardization) và ANSI (American National Standards Institute) được sử dụng trong SQL Server khác với P-SQL (Procedural-SQL) dùng trong Oracle.

Trong bài này chúng ta sẽ tìm hiểu sơ qua về T-SQL. Chúng được chia làm 3 nhóm:

## **Data Definition Language (DDL):**

Đây là những lệnh dùng để quản lý các thuộc tính của một database như định nghĩa các hàng hoặc cột của một table, hay vị trí data file của một database...thường có dạng

- Create *object\_Name*
- Alter *object\_Name*
- Drop *object\_Name*

Trong đó *object\_Name* có thể là một table, view, stored procedure, indexes...

Ví dụ:

Lệnh Create sau sẽ tạo ra một table tên Importers với 3 cột CompanyID,CompanyName,Contact

USE Northwind

CREATE TABLE Importers(

CompanyID int NOT NULL,

CompanyName varchar(40) NOT NULL,

Contact varchar(40) NOT NULL

)

Lệnh Alter sau đây cho phép ta thay đổi định nghĩa của một table như thêm(hay bớt) một cột hay một Constraint...Trong ví dụ này ta sẽ thêm cột ContactTitle vào table Importers

USE Northwind

ALTER TABLE Importers

ADD ContactTitle varchar(20) NULL

Lệnh Drop sau đây sẽ hoàn toàn xóa table khỏi database **nghĩa là cả định nghĩa của table và data bên trong table đều biến mất** (khác với lệnh Delete chỉ xóa data nhưng table vẫn tồn tại).

USE Northwind

DROP TABLE Importers

### **Data Control Language (DCL):**

Đây là những lệnh quản lý các quyền truy cập lên từng object (table, view, stored procedure...). Thường có dạng sau:

- Grant
- Revoke
- Deny

Ví dụ:

Lệnh sau sẽ cho phép user trong Public Role được quyền Select đối với table Customer trong database Northwind (**Role** là một khái niệm giống như Windows Group sẽ được bàn kỹ trong phần Security)

USE Northwind

GRANT SELECT

ON Customers

TO PUBLIC

Lệnh sau sẽ từ chối quyền Select đối với table Customer trong database Northwind của các user trong Public Role

## Giới Thiệu Sơ Lược Về Transact SQL (T-SQL)

USE Northwind

DENY SELECT

ON Customers

TO PUBLIC

Lệnh sau sẽ xóa bỏ tác dụng của các quyền được cho phép hay từ chối trước đó

USE Northwind

REVOKE SELECT

ON Customers

TO PUBLIC

### **Manipulation Language (DML):**

Đây là những lệnh phổ biến dùng để xử lý data như Select, Update, Insert, Delete

Ví dụ:

Select

USE Northwind

SELECT CustomerID, CompanyName, ContactName

FROM Customers

WHERE (CustomerID = 'alfki' OR CustomerID = 'anatr')

ORDER BY ContactName

Insert

USE Northwind

INSERT INTO Territories

VALUES (98101, 'Seattle', 2)

## Giới Thiệu Sơ Lược Về Transact SQL (T-SQL)

### Update

USE Northwind

UPDATE Territories

SET TerritoryDescription = 'Downtown Seattle'

WHERE TerritoryID = 98101

### Delete

USE Northwind

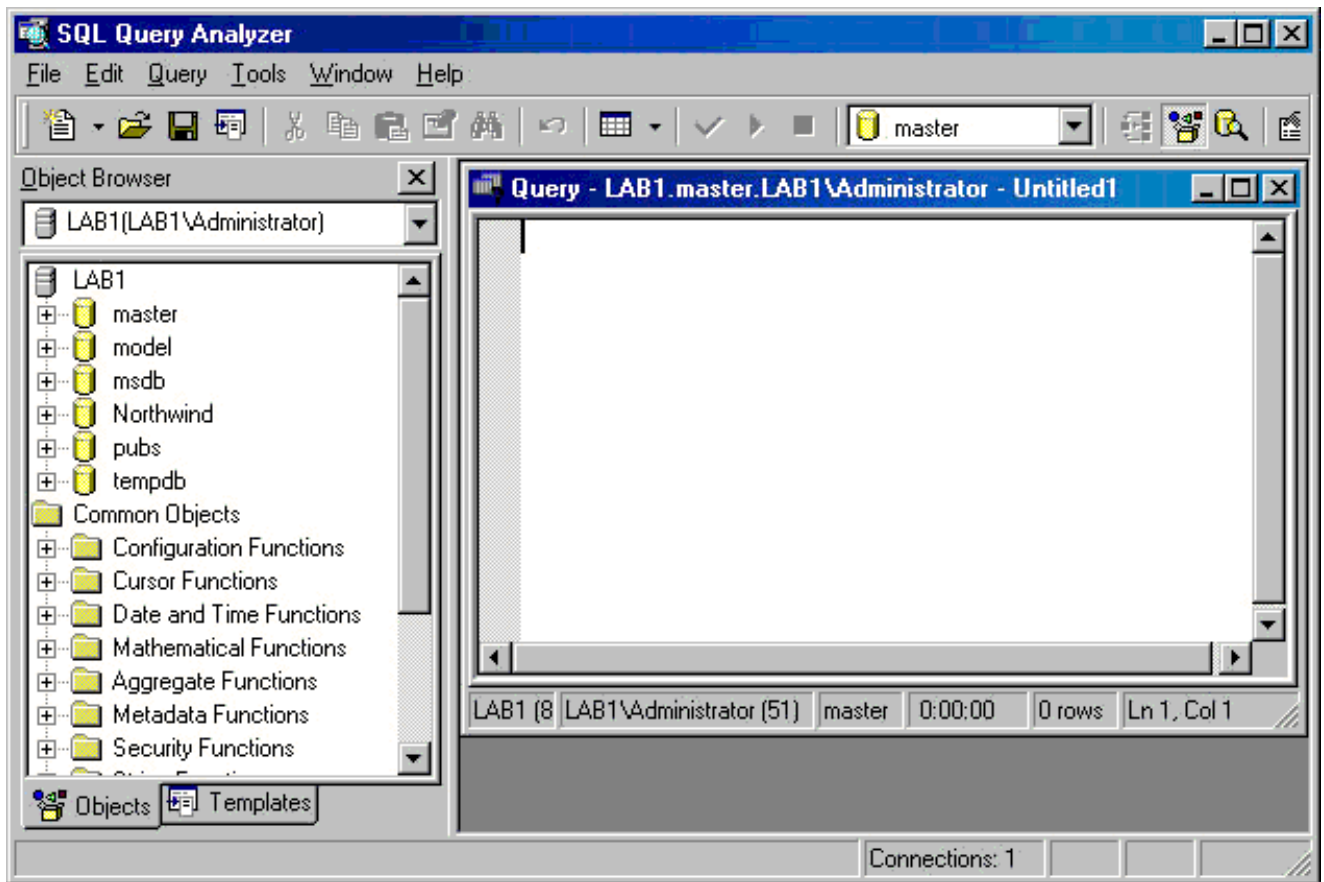
DELETE FROM Territories

WHERE TerritoryID = 98101

Vì phần này khá căn bản nên chúng tôi thiết nghĩ không cần giải thích nhiều. Chú ý trong lệnh Delete bạn có thể có chữ From hay không đều được.

Nhưng mà chúng ta sẽ chạy thử các ví dụ trên ở đâu?

Để chạy các câu lệnh thí dụ ở trên bạn cần sử dụng và làm quen với **Query Analyser**



## Cú Pháp Của T-SQL:

Phần này chúng ta sẽ bàn về các thành phần tạo nên cú pháp của T-SQL

### Identifiers

Đây chính là tên của các database object. Nó dùng để xác định một object. (Chú ý khi nói đến Object trong SQL Server là chúng ta muốn đề cập đến table, view, stored procedure, index..... Vì hầu như mọi thứ trong SQL Server đều được thiết kế theo kiểu hướng đối tượng (object-oriented)). Trong ví dụ sau TableX, KeyCol, Description là những identifiers

```
CREATE TABLE TableX
```

```
(KeyCol INT PRIMARY KEY, Description NVARCHAR(80))
```

Có hai loại Identifiers một loại thông thường (**Regular Identifier**) và một loại gọi là **Delimited Identifier**, loại này cần có dấu "'" hay dấu [] để ngăn cách. Loại Delimited được dùng đối với các chữ trùng với từ khóa của SQL Server (reserved keyword) hay các chữ có khoảng trống.

Ví dụ:

```
SELECT * FROM [My Table]
```

```
WHERE [Order] = 10
```

Trong ví dụ trên chữ Order trùng với keyword Order nên cần đặt trong dấu ngoặc vuông [].

### Variables (Biến)

Biến trong T-SQL cũng có chức năng tương tự như trong các ngôn ngữ lập trình khác nghĩa là cần khai báo trước loại dữ liệu trước khi sử dụng. Biến được bắt đầu bằng dấu @ (Đối với các global variable thì có hai dấu @@)

Ví dụ:

```
USE Northwind
```

```
DECLARE @EmpIDVar INT
```

```
SET @EmpIDVar = 3
```

```
SELECT * FROM Employees
```

```
WHERE EmployeeID = @EmpIDVar + 1
```

### Functions (Hàm)

Có 2 loại hàm một loại là built-in và một loại user-defined

Các hàm **Built-In** được chia làm 3 nhóm:

- **Rowset Functions** : Loại này thường trả về một object và được đối xử như một table. Ví dụ như hàm OPENQUERY sẽ trả về một recordset và có thể đứng vị trí của một table trong câu lệnh Select.
- **AggregateFunctions** : Loại này làm việc trên một số giá trị và trả về một giá trị đơn hay là các giá trị tổng. Ví dụ như hàm AVG sẽ trả về giá trị trung bình của một cột.
- **Scalar Functions** : Loại này làm việc trên một giá trị đơn và trả về một giá trị đơn. Trong loại này lại chia làm nhiều loại nhỏ như các hàm về toán học, về thời gian, xử lý kiểu dữ liệu String.... Ví dụ như hàm MONTH('2002-09-30') sẽ trả về tháng 9.

Các hàm **User-Defined** (được tạo ra bởi câu lệnh CREATE FUNCTION và phần body thường được gói trong cặp lệnh BEGIN...END) cũng được chia làm các nhóm như sau:

- **Scalar Functions** : Loại này cũng trả về một giá trị đơn bằng câu lệnh RETURNS.
- **Table Functions** : Loại này trả về một table

### Data Type (Loại Dữ Liệu)

Các loại dữ liệu trong SQL Server sẽ được bàn kỹ trong các bài sau

### Expressions

Các Expressions có dạng **Identifier + Operators** (như +, -, \*, /, =...) + **Value**

### Các thành phần Control-Of Flow

Như BEGIN...END, BREAK, CONTINUE, GOTO, IF...ELSE, RETURN, WHILE....  
Xin xem thêm Books Online để biết thêm về các thành phần này.

### Comments (Chú Thích)

T-SQL dùng dấu -- để đánh dấu phần chú thích cho câu lệnh đơn và dùng /\*...\*/ để chú thích cho một nhóm

### Thực Thi Các Câu Lệnh SQL

Thực thi một câu lệnh đơn:

Một câu lệnh SQL được phân ra thành các thành phần cú pháp như trên bởi một parser, sau đó SQL Optimizer (một bộ phận quan trọng của SQL Server) sẽ phân tích và tìm cách thực thi (Execute Plan) tối ưu nhất ví dụ như cách nào nhanh và tốn ít tài nguyên của máy nhất... và sau đó SQL Server Engine sẽ thực thi và trả về kết quả.

### Thực Thi một nhóm lệnh (Batches)

Khi thực thi một nhóm lệnh SQL Server sẽ phân tích và tìm biện pháp tối ưu cho các câu lệnh như một câu lệnh đơn và chứa execution plan đã được biên dịch (compiled) trong bộ nhớ sau đó nếu nhóm lệnh trên được gọi lại lần nữa thì SQL Server không cần biên dịch mà có thể thực thi ngay điều này giúp cho một batch chạy nhanh hơn.

### Lệnh GO

Lệnh này chỉ dùng để gửi một tín hiệu cho SQL Server biết đã kết thúc một batch job và yêu cầu thực thi. Nó vốn không phải là một lệnh trong T-SQL.